



UNIVERSIDAD DE BUENOS AIRES
Facultad de Ciencias Exactas y Naturales
Departamento de Computación

Análisis bayesiano del aprendizaje en comunidades de video juegos

Tesis presentada para optar al título de Doctor de la
Universidad de Buenos Aires en el área Ciencias de la Computación

Lic. Gustavo Andrés Landfried

Director de tesis: Dr. Esteban Mocskos
Director de tesis: Dr. Diego Fernández Slezak
Consejero de estudios: Dr. Hernán Melgratti

Lugar de trabajo: Departamento de Computación, Facultad de Ciencias Exactas y Naturales

Buenos Aires, diciembre 2023

Fecha de defensa: 9 de abril de 2024

Firma

El conocimiento emerge como la vida

Análisis bayesiano del aprendizaje en comunidades de video juegos

Resumen

La especie humana tiene una singular capacidad de aprendizaje social, que le permite acumular adaptaciones sucesivas en el tiempo que van produciendo el inimaginablemente complejo sistema de información cultural. Si bien el aprendizaje humano ocurre en los cuerpos biológicos, el aprendizaje cultural es un fenómeno poblacional que emerge de la interacción prolongada entre individuos y entre generaciones. Entender las propiedades de los sistemas de información cultural es uno de los problemas fundamentales de la antropología, relevante para las ciencias de la computación y la inteligencia artificial multi-agente. En particular, comprender las relaciones que vinculan el nivel poblacional (como la dinámica y estructura de la red de intercambios de información cultural) con el nivel individual (el aprendizaje efectivo que se produce en los cuerpos), ha sido la pregunta que motivó esta tesis de doctorado. Para ello, decidimos estudiar las comunidades de video juegos en línea. Además de que ofrecen la oportunidad de estudiar poblaciones grandes sin perder un alto grado de detalle, ellas son un lugar privilegiado para estudiar cómo cambian las estrategias en el tiempo. Algunas de las preguntas que nos propusimos responder durante el doctorado fueron las siguientes. ¿Cuál es la mejor forma de medir el aprendizaje de un individuo en el tiempo? ¿Cuál es la relación entre la formación de equipos y el aprendizaje individual a largo plazo? ¿Cuál es el efecto que la posición topológica de un individuo en la red información cultural tiene sobre el aprendizaje individual? ¿Cuáles son algunas de las propiedades generales del aprendizaje?

Durante el doctorado en ciencias de la computación el enfoque bayesiano de la probabilidad respondió las preguntas metodológicas y epistemológicas más profundas que traía de mi formación previa en ciencias antropológicas: cómo alcanzar acuerdos respecto a la correcta evaluación de argumentos causales alternativos. Las reglas de la probabilidad se conocen desde finales del siglo 18 y desde entonces se las ha adoptado como sistema de razonamiento en todas las ciencias empíricas. Si bien en todo este tiempo no se ha propuesto nada mejor en términos prácticos, el costo computacional asociado a la evaluación de todo el espacio de hipótesis ha limitado históricamente la aplicación estricta de las reglas de la probabilidad (enfoque bayesiano). El enfoque bayesiano se fue incorporando lentamente durante el transcurso de la tesis, hasta transformarse en un protagonista.

En el primer trabajo usamos el modelo bayesiano de habilidad más utilizado en la industria del video juego para estudiar una comunidad en el que las personas podían jugar individualmente o en equipos. Mediante test de hipótesis estadísticos detectamos que jugar en equipo está asociado a mayor aprendizaje a largo plazo, y que mantener un equipo estable está asociado a mayor velocidad de aprendizaje. Los estimadores de habilidad disponibles, considerados estado del arte al inicio del doctorado, no permiten sin embargo obtener estimaciones iniciales fiables ni garantizar la comparabilidad entre estimaciones distantes en el tiempo y el espacio. En el transcurso del doctorado descubrimos la especificación matemática de un modelo conocido como TrueSkill Through Time (TTT) que resolvía estos problemas realizando la inferencia en única red bayesiana, propagando la información histórica correctamente. Debido a que el modelo no se encontraba disponible en ningún lenguaje de programación en el segundo trabajo resolvimos la matemática de este modelo, la documentamos, la implementamos y la publicamos en los lenguajes de programación **Julia**, **Python** y **R**. En un tercer trabajo, estudiamos la evolución de una red de partidas en el juego de Go durante un periodo de ocho años y encontramos, con el nuevo estimador, que la posición de los individuos en la red tiene un efecto de segundo orden sobre el aprendizaje en las personas que están en el medio del proceso de aprendizaje, ausente entre novatas y expertas.

Palabras claves: Ciencias Sociales Computacionales, Inferencia bayesiana, Cultura, Habilidad, Aprendizaje, Comunidades virtuales, Videojuegos

Bayesian analysis of learning in video game communities

Abstract

The human species has a unique capacity for social learning, accumulating successive solutions over time and producing an unimaginably complex cultural information system. While human learning occurs in biological bodies, cultural learning is a population phenomenon that emerges in the prolonged interaction between individuals and between generations. Understanding the properties of cultural information systems is one of the fundamental problems of anthropology, relevant to computer science and multi-agent artificial intelligence.

Understanding the relationships that link the population level (such as the dynamics and structure of the cultural information exchange network) with the individual level (the effective learning that occurs within individuals' bodies) has been the central question driving this doctoral thesis. To address this, we chose to investigate online gaming communities. In addition to providing the opportunity to study large populations without sacrificing a high degree of detail, these communities are a privileged setting for examining how strategies change over time. Some of the questions we aimed to answer during the course of the doctoral research were as follows: What is the most effective way to measure an individual's learning over time? What is the relationship between team formation and long-term individual learning? What is the effect of an individual's topological position in the dynamic exchange network on individual learning? What are some of the general properties of learning?

During my doctoral studies in computer science, the Bayesian approach to probability addressed the deeper methodological and epistemological questions that emerged from my prior education in anthropological sciences, specifically, how to reach agreements concerning the proper assessment of alternative causal arguments. Probability rules have been known since the late 18th century and have since been adopted as a reasoning system in all empirical sciences. Although, in practical terms, nothing better has been proposed over this period, the computational cost associated with the evaluation of the entire hypothesis space has historically limited the strict application of probability rules (Bayesian approach). The Bayesian approach was gradually integrated throughout the course of the thesis, eventually becoming a central player.

In the first study, we employed the most widely used Bayesian skill model in the video game industry to examine a community where individuals could play either individually or in teams. Through statistical hypothesis tests, we identified that playing in a team is associated with greater long-term learning, and maintaining a stable team is linked to a higher learning speed. The skill estimators available, considered state-of-the-art at the beginning of the doctoral research, however, did not allow for obtaining reliable initial estimates or ensuring comparability between estimates across distant time and space. During the course of the doctoral studies, we discovered the mathematical specification of a model known as TrueSkill Through Time (TTT), which addressed these issues by conducting inference in a single Bayesian network and accurately propagating historical information. As this model was not available in any programming language in the second study, we tackled the mathematics of the model, documented it, implemented it, and published it in the **Julia**, **Python**, and **R** programming languages. In a third paper, we studied the evolution of a network of games in the game of Go over a period of eight years and found with the new estimator that the position of individuals in the network has a second-order effect on learning in individuals who are in the middle of the learning process, absent between novices and experts.

Keywords: Computational social science, Bayesian inference, Culture, Skill, Learning, Virtual communities, Video games

Índice general

1. Introducción	3
1.1. Motivación	3
1.1.1. Aprendizaje humano	4
1.1.2. La cultura	5
1.1.3. Objetivos: aprendizaje en comunidades de video juegos	7
1.2. Teoría elemental de la probabilidad	8
1.2.1. Sistema axiomático	9
1.2.2. Terminología basada en la teoría de conjuntos	9
1.2.3. Las reglas de la probabilidad	10
1.3. Aplicación estricta de las reglas de la probabilidad	12
1.3.1. Acuerdos intersubjetivos en contextos de incertidumbre	12
1.3.2. Evaluación de modelos	19
1.3.3. Consecuencias de la ruptura de las reglas de la probabilidad	21
1.4. Acerca de la <i>habilidad</i>	25
1.4.1. Modelo de habilidad Elo	26
1.4.2. Evidencia, verosimilitud y posterior exacto	28
1.4.3. Posterior aproximado y extensión del modelo	30
2. Resultado 1. Efecto de la formación de equipos sobre el aprendizaje	33
2.1. Introducción	33
2.2. Materiales y métodos	35
2.3. La ley de la práctica	37
2.4. Aprendizaje social	39
2.4.1. Faithfulness-boost effect	39
2.5. Discusión	43
3. Resultado 2. Implementación del estimador de habilidad estado del arte	45
3.1. Introducción	45
3.1.1. Modelo TrueSkill Through Time	47
3.1.2. Evaluación de modelos	48
3.2. Software: implementación del modelo	50
3.2.1. Sum-product algorithm	50
3.2.2. Propiedades matemáticas y notación	52
3.2.3. La clase gaussiana	52
3.2.4. Solución exacta para eventos con dos equipos	53
3.2.5. Modelo básico de empates	56

3.2.6. Aproximación óptima del posterior exacto	57
3.2.7. Varios equipos	59
3.2.8. Propagación de la información en la clase History	61
3.3. Interfaz y aplicaciones	63
3.3.1. Único evento	63
3.3.2. Secuencia de eventos	66
3.3.3. Evolución de habilidad	67
3.3.4. Historia de la Asociación de Tenistas Profesionales (ATP)	68
3.3.5. Habilidades multidimensionales	70
3.4. Detalles computacionales	71
3.5. Conclusiones y discusión	73
4. Resultado 3. Efecto en el aprendizaje de la posición en la red de intercambios	77
4.1. Introducción	77
4.2. Materiales y métodos	80
4.2.1. Base de datos	80
4.2.2. Modelo de estimación de habilidad	81
4.2.3. Una metodología clásica de análisis de grafos dinámicos	82
4.2.4. La posición de los individuos en la red	83
4.3. Análisis	85
4.3.1. La habilidad inicial de las curvas de aprendizaje individuales	85
4.3.2. La posición topológica de los individuos en la red	86
4.3.3. El impacto que la posición en la red tiene sobre el aprendizaje individual.	87
4.4. Discusiones	89
5. Conclusiones	91
5.1. Enfoque bayesiano de la probabilidad	91
5.2. Efecto de la formación de equipos sobre el aprendizaje.	93
5.3. Una hipótesis que fue rechazada	94
5.4. La primera implementación del modelo de habilidad estado del arte	95
5.5. Efecto en el aprendizaje individual de la posición en el red cultural	96
6. Trabajo futuro	99
6.1. Comunidad	99
6.2. Enseñaje	102
6.3. Praxis	105
7. Anexo	117
7.1. TrueSkill Through Time	117
7.1.1. Evolución de habilidad	117
7.1.2. Historia de la Asociación de Tenistas Profesionales (ATP)	118
7.2. TrueSynergy	119
7.2.1. TrueSkill: sobrestimación de habilidad en contextos de sinergia	119
7.2.2. Modelo TrueSynergy	121
7.2.3. Experimentos sintéticos con TrueSynergy	122
7.3. Efectos sociales sobre el aprendizaje	124

7.3.1. Formación de grupos	124
7.4. Bayes Plurinacional	129
7.5. Propiedad de las distribución gaussiana	131
7.5.1. Producto de gaussianas	131
7.5.2. Suma de gaussianas	133
7.5.3. gaussiana por acumulada de gaussiana.	134
7.5.4. Division de gaussianas	135

Agradecimientos: soy porque son.

El primer derecho es formar parte de una comunidad. Hay que saber dar y recibir con dicha y agradecimiento porque la vida es la unión de muchos seres y fuerzas, dice el último de los 13 principios aymaras del buen vivir (*Suma Churaña, Suma Katukaña*). Nuestra propia vida depende al menos de 4 niveles de cooperación para sobrevivir: la célula con la mitocondria, el organismo multicelular, la sociedad y el ecosistema. No hay mérito individual, el mérito siempre es colectivo. De quienes ya no están. De quienes no recordamos. De quienes nunca conocimos. Hacia ellas quedará mi deuda, que espero compensar con quienes tuve y tengo cerca día a día. A mi compañera Laura, que me nutrió en todos los aspectos de la vida, día tras día, desde el mero sueño hasta la realidad concreta, todo el mérito es tuyo. Siempre merecés más de lo que puedo dar. A mi madre Adriana, que me dio la vida y me hizo ser quien soy, quien no puedo dejar de ser aunque quisiera, todo el mérito es tuyo. Siempre merecés más de lo que puedo dar. A mi director Esteban, que me abrió todas las puertas y me dejó ser y hacer a pesar de mis locuras y errores, todo el mérito es tuyo. Espero por lo menos no quedar en deuda con vos. A mi hermana Carolina y mi hermano Hernán, con quién compartí intensamente los primeros años de mi vida, gracias, sin su estímulo no me imagino como sería mi presente. A mis sobrinos Delfina e Ignacio, a quienes les deseo que se les abra el mundo y vivan en compañía, salud y bienestar. A mi padre Alberto que me cuidó con el amor de un abuelo. A la madre de mi madre, la abuela Pichi, por darme toda una familia que le dio vida a mi corazón. A la abuela de mi abuela, y a la mitad de la población argentina, a quienes le impidieron hablar su lengua materna. A mis referentes intelectuales Rita, Enrique, Juan, Shara, Edwin de quienes obtuve las principales respuestas. A todo el Grupo Antropocaos, por señalarme el camino. A mi codirector Diego nuevamente, que estuvo presente para darme el primer impulso. A mis compañeros doctorales de oficina antes de la pandemia, Maxi, David, Emmanuel y Rodolfo, además de ser un gran apoyo, disfruté mucho compartir con ustedes. A los primeros profesores del Laboratorio de Inteligencia Artificial Aplicada, Diego, Agustín, Luciana y Juanka, por integrarme en un hermoso grupo humano. A mis contemporáneos Edgard, Facundo, Juanma, Riera, Bruno, March, Lara, Jazmín, Gastón, Marcos, Daniela, Ramiro, Brusco, Lao, con quienes pude compartir. A mis estudiantes Macarena, Tobías, Martín, Agustín, Gustavo, gracias por ayudarme a ser mejor. A todo el Departamento de Computación y el Instituto de Ciencias de la Computación, sus autoridades, sus profesores, sus docentes, administrativos, por haberme acogido en su comunidad. A las y los consejeros de la Facultad de Ciencias Exactas y Naturales, por asumir la responsabilidad de construir día a día una institución pública, de excelencia, dinámica, popular, inclusiva y comprometida con la realidad nacional. A Cristina Fernandez por haber fortalecido el financiamiento en ciencia que permitió hacer realidad este proyecto.

Capítulo 1

Introducción

1.1. Motivación

El aprendizaje es una forma de plasticidad conductual que permite a los individuos ajustar su comportamiento a las condiciones ambientales locales. Cuando el aprendizaje puede ser transmitido socialmente (a partir de la observación de otros o de la interacción con ellos), aparece un sistema de información no-genético denominado cultura que se transmite de un individuo a otro, entre poblaciones y a lo largo de las generaciones. Muchos animales son capaces de transmitir información no-genética y producir tradiciones simples persistentes en el tiempo. En algunos casos, como las ballenas, delfines, primates y aves, las tradiciones pueden ser incluso bastante complejas. Por ejemplo, los cantos de las aves, sus estrategias de búsqueda de alimento, e incluso las rutas migratorias son comportamientos que se transmiten por vía social, produciendo la formación de culturas locales [1]. Sin embargo, todas ellas son manifiestamente más simples que la cultura humana [2].

Lo singular de la especie humana es la capacidad de acumular adaptaciones culturales en el tiempo, modificaciones sucesivas que van produciendo un sistema de información cultural inimaginablemente complejo [3, 4]. Si bien el aprendizaje humano ocurre en los cuerpos biológicos, es parte de un fenómeno poblacional que emerge como consecuencia del intercambio de información entre individuos. La cultura humana es la información transmitida por vía comportamental, y la sociedad humana el sistema distribuido de almacenamiento y procesamiento [5].

Comprender algunas de las relaciones que vinculan el nivel poblacional (como la dinámica y estructura de la red de intercambios de información cultural) con el nivel individual (el aprendizaje efectivo que se produce en cada uno), es la pregunta general que intentamos explorar en esta tesis. Surgen rápidamente muchos conceptos y muchas definiciones a distinguir. Por ejemplo, ¿hay una forma óptima de aprender en contextos de incertidumbre? Este problema lo tratamos en el capítulo 1, donde discutimos: las reglas de razonamiento de las ciencias empíricas, el método de evaluación de modelos causales alternativos, y una aplicación de interés para la tesis (los modelos de estimación de habilidad en el tiempo).

El concepto de aprendizaje tiene diferentes acepciones. Los modelos de estimación de habilidad que se utilizan para medir el desempeño de los individuos en la industria del videojuego difieren del concepto de aprendizaje que se establece desde un punto de vista biológico, cognitivo o social. En pocas palabras, los modelos utilizados en la industria del videojuego (así como gran parte de las evaluaciones internacionales de los sistemas educativos) son instru-

mentos que predicen la probabilidad de éxito o fracaso. En definitiva, el modelo causal sobre el que se basan sus estimaciones (ver detalles en capítulo 1 y 3) es una simplificación extrema de la realidad causal que gobierna los procesos de aprendizaje humano.

Esta tesis se destaca, en particular, por haber realizado la primera implementación del modelo de estimación de habilidad estado del arte en la industria del video juego, que si bien ya estaba especificado matemáticamente, no se encontraba todavía disponible como paquete para ser utilizado en algún lenguaje de programación. La novedad no es tanto el modelo causal de base (que se ve en detalle en el capítulo 1 y 3), el cual es tan simple como en los anteriores. La mejora que introduce proviene de integrar todos los modelos causales de base (los eventos donde ocurren los éxitos o los fracasos) en una única red causal histórica (la habilidad del pasado es una causa de la habilidad del presente). Esto permite propagar la información por toda la red, obteniendo así estimaciones con menor incertidumbre en todas las series temporales.

Este modesto avance da cuenta de la dificultad que existe en el campo de los estudios del aprendizaje para proponer y evaluar modelos causales capaces de integrar los diversos elementos de interés que forman parte de nuestros objetos reales de estudio. Por ese motivo, cuando en los capítulos 2 y 4 exploramos en bases de datos reales el efecto que algunos factores sociales tienen sobre el aprendizaje de los individuos, nos vimos obligados a recurrir a diversas heurísticas estadísticas que si bien arrojan luz, no garantizan conclusiones definitivas. Solo recientemente, en el transcurso del doctorado apareció quizás el primer modelo causal (en el sentido probabilístico del término) capaz de integrar diversos elementos que son específicos al problema de estudio [6]. Este modelo es una extensión de nuestra implementación, y todavía no se encuentra disponible como paquete en ningún lenguaje de programación.

Durante el transcurso del doctorado, investigadores que han sido protagonistas del desarrollo de la teoría de la evolución moderna han comenzado a señalar el isomorfismo que existe entre la teoría de la evolución y la inferencia bayesiana [7, 8] (el modelo estándar de evolución conocido como *replicator dynamic* [9] es estructuralmente equivalente al teorema de Bayes [10, 11]).

1.1.1. Aprendizaje humano

El aprendizaje se define como un cambio persistente en las representaciones o asociaciones mentales como resultado de la experiencia individual o social, y que es observable a través del comportamiento. Los comportamientos observables pueden ser: la aparición de una nueva habilidad (por ejemplo, aprender a contar); cambios en la frecuencia de un comportamiento existente (por ejemplo, cooperar más regularmente con otros); modificar la velocidad o eficiencia de un comportamiento existente (en ajedrez, el tiempo de mover la pieza); cambios en la intensidad de un comportamiento existente (en deporte, obtener victorias cada vez más holgadas); modificar la complejidad de un comportamiento existente (por ejemplo, debatir un tema con mayor profundidad conceptual); responder de forma diferente a un estímulo concreto (en psicología, la respuesta al miedo) [12].

Aunque los investigadores pueden diferir en sus opiniones sobre la mejor manera de definir el aprendizaje y determinar cuándo se ha producido, existe una coincidencia respecto de que para estudiar el aprendizaje es necesario observar las modificaciones en el comportamiento de modo tal de poder inferir aquellos cambios persistentes que van desarrollando los individuos a través de la experiencia. A finales del siglo 19, las perspectivas dominantes en psicología eran el *estructuralismo* y el *funcionalismo*. Aunque ellas diferían considerablemente entre sí,

compartían una debilidad común: el principal medio para investigar el aprendizaje y otros fenómenos psicológicos era un método llamado “*introspección*” basado en la auto-percepción del paciente [12].

A comienzos del siglo 20, una corriente comienza a proponer métodos de estudio basado en las conductas de las personas (respuestas) y los cambios ambientales (estímulos) que preceden y siguen a esas respuestas. Desde entonces, muchos psicólogos han intentado describir y comprender el aprendizaje y la conducta principalmente a través del análisis de las relaciones estímulo-respuesta. Esta escuela fue denominada *conductista* y, en su primera etapa, se enfocaba exclusivamente en el aprendizaje que se producía por experiencia individual. Alrededor de 1940, algunos psicólogos propusieron una perspectiva alternativa, la *teoría del aprendizaje social*, que se centraba en la transmisión de información entre las personas [12]. Ambas corrientes se desarrollaron principalmente en Estados Unidos.

Con el tiempo, los teóricos del aprendizaje social fueron incorporando el análisis de los procesos mentales (o cognitivos) a los estudios del aprendizaje. Sin embargo, ni la cognición ni el comportamiento permitían determinar las características distintivas del aprendizaje humano. ¿Cuáles son, entonces, las propiedades particulares del aprendizaje humano? La increíble flexibilidad social de los humanos permite una acumulación de conocimientos a través de las generaciones mucho más compleja que la que logran otras especies.

En las últimas décadas, la psicología ha comenzado a incorporar los análisis de las interacciones sociales y la transmisión cultural en sus estudios sobre los procesos del aprendizaje humano. La etiqueta más utilizada es la de *teoría sociocultural*, pero en términos más generales podemos pensar en ellas como teorías contextuales [12]. Los seres humanos crecen y se desarrollan en entornos culturales, por lo que la dinámica de los sistemas de intercambio de información cultural afectan directamente nuestros procesos de aprendizaje individual.

En la historia reciente, la intensificación de la interconexión global a partir del uso de energías fósiles en el siglo 19 y el desarrollo de las comunicaciones electrónicas durante el siglo 20 han tenido profundas consecuencias para el desarrollo de la ciencia moderna. En la actualidad, la adopción de los *smartphones* como dispositivos de comunicación instantánea, está teniendo consecuencias concretas sobre los procesos de aprendizaje colectivo e individual que aún desconocemos.

1.1.2. La cultura

En la historia del ser humano, la transición cultural tuvo un efecto positivo radical para la especie. Hubo un período en el que se enfrentó un grave peligro de extinción, lo que se evidencia en la baja diversidad del genoma humano respecto, incluso, de los homínidos más cercanos [13]. Pero cuando el conocimiento, que antes debía ser redescubierto individualmente, pasó a ser un recurso común transmitido de generación en generación, la especie humana comenzó a ser capaz de ocupar todos los nichos ecológicos de la tierra como ningún otro vertebrado terrestre lo había logrado antes [4].

Antes del surgimiento de los humanos anatómicamente modernos (masa cerebral actual) y de los humanos conductualmente modernos (lenguaje), surgió en África una linaje emocionalmente moderno, con capacidades para el entendimiento mutuo [14]. A diferencia de las especies más cercanas, nuestros ancestros comenzaron a desarrollar un tipo de crianza cooperativa que favoreció la selección de jóvenes con capacidades para comunicar y comprender las intenciones de los demás [14]. A través de rondas repetidas de co-evolución genético-cultural, se fueron desarrollando las características biológicas, cognitivas y sociales que permiten hoy

a los humanos aprender y transmitir rasgos culturales complejos a través de las generaciones [15].

Esta capacidad biológica para la comprensión mutua y la imitación (*aprendizaje social*) dio inicio a este proceso inter-generacional de acumulación de innovaciones, que ha sido tan exitoso para la adaptación humana. Lo que antes debía ser redescubierto una y otra vez mediante la costosa experiencia individual, ahora podía ser transmitido a la siguiente generación. Como consecuencia, nadie individualmente es capaz de redescubrir en un período de vida ni siquiera los repertorios culturales de las sociedades cazadoras recolectoras actuales consideradas más simples. Debido a que la cultura es un sistema de información, a largo de la evolución humana fueron emergiendo naturalmente diversos mecanismos de comunicación, cada uno de los cuales impactaron directamente sobre la tasa de aprendizaje cultural de la especie y sobre nuestras tasas individuales de aprendizaje.

Una importante modificación en el sistema de intercambio de información cultural comenzó hace 10 mil años con el surgimiento independiente de la agricultura en seis regiones distintas del mundo (África subsahariana, Oriente medio, China, Oceanía, América del Norte y América del Sur). El aumento progresivo de la población y el surgimiento de la escritura intensificaron los procesos de acumulación cultural, haciendo de estas regiones los principales centros tecnológicos de la humanidad. Para el año 1400 en el mundo florecían sociedades prósperas: China llevaba 2 milenios siendo el principal centro productivo y tecnológico del mundo; en el mundo Árabe se comerciaban estos productos desde el océano Atlántico en España, hasta el océano Pacífico en las Filipinas; el océano Pacífico estaba totalmente ocupado, y ya se habían producido intercambios entre Oceanía y América del Sur; y en el jardín de la diversidad genética y cultural humana, África subsahariana, se desarrollaba entre otras, la sociedad Bantú.

Así como los sistemas de comunicación favorecen los procesos de aprendizaje social y acumulación cultural, el aislamiento está asociado a pérdidas masivas de información cultural. El ejemplo paradigmático es el aislamiento total que ocurrió con la separación de Tasmania del continente Australiano por la subida del nivel del mar a comienzos del Holoceno [16]. La evidencia indica que las sociedades que permanecieron en la isla de Tasmania perdieron gran parte de su cultura tecnológica. De forma similar, el aislamiento de Europa occidental del sistema mundo luego de la masiva destrucción de la diversidad cultural al interior del imperio Romano condujo a un largo proceso de involución cultural y de violencia interna conocido como Edad Media [17].

Si bien se han propuesto muchas causas internas para explicar la prosperidad actual de Europa occidental¹, ninguna de ellas puede responder la paradoja de cómo una sociedad como la feudal, sumida en un proceso de involución cultural único, pudo generar de repente el extraordinario desarrollo científico y técnico de la modernidad. Nada de esto resulta extraño cuando se consideran los eventos globales: las enfermedades transmitidas por los exploradores feudales que eliminaron al menos a 2/3 de la población americana; y el giro geopolítico que se produce a partir de que en 1546 los exploradores feudales descubren la montaña de plata de Potosí, metal que China había incorporado recientemente como una de sus monedas oficiales [21].

En la actualidad existe un amplio consenso respecto que el aprendizaje humano es una propiedad poblacional [5]. Sin embargo, estudios recientes han cuestionado la hipótesis de que

¹La “ética protestante” de Max Weber [18], la “mentalidad burguesa” de José Luis Romero [19], o más recientemente el “sistema de parentesco” de Joseph Henrich [20]

todo lo que maximice el flujo de información cultural debería repercutir positivamente [5]. El aumento en la interconexión global suele ser percibido como positivo para el desarrollo de la cultura, como causa del extraordinario crecimiento poblacional de la sociedad moderna. Sin embargo la biomasa de nuestra especie representa actualmente tan solo el 0,01 % de la biomasa total y el 2,5 % de la biomasa animal [22], y las proyecciones indican que estamos a menos de un siglo del estancamiento. Lo cierto es que este límite a la biomasa humana no está impuesto por la capacidad de la biosfera, sino por nuestra incapacidad actual para integrarnos de forma orgánica en los sistemas ecológicos que la componen [23, 24].

En efecto, no todo aumento en la interconexión es positivo. Los estudios comparados muestran que las instituciones capaces de administrar exitosamente los bienes comunes y establecer relaciones de largo plazo con los sistemas ecológicos emergen en comunidades autónomas con fuerte arraigo local [23, 24]. El remplazo repentino de estos sistemas culturales por instituciones externas, estatales o de mercado, ha producido devastadoras consecuencias ecológicas [25]. A pesar de todos los avances, la ciencia metropolitana no es capaz de compensar la pérdida de los conocimientos milenarios, y la crisis ecológica actual no deja de profundizarse.

Actualmente, una de las hipótesis más influyentes afirma que las redes totalmente conectadas producen una homogeneización cultural que reduce las posibilidades de innovación, mientras que las redes fragmentadas tienden a perder variantes culturales complejas con el tiempo [26–28]. Sin embargo son pocos los estudios que ponen a prueba estas hipótesis en datos del mundo real [5, 29].

1.1.3. Objetivos: aprendizaje en comunidades de video juegos

El objetivo del plan de tesis ha sido explorar los efectos que diversos factores sociales tienen sobre el aprendizaje de los individuos. El aprendizaje individual se ve especialmente afectado por la dinámica de las “redes culturales”. Entender las relaciones que vinculan el nivel poblacional (como la dinámica y estructura de la red de intercambios de información cultural) con el nivel individual (el aprendizaje efectivo que se produce en los cuerpos), ha sido la pregunta que motivó esta tesis de doctorado.

Una de las razones por las que es difícil evaluar el efecto que las interacciones sociales (su contexto, dinámica y estructura) tienen sobre el aprendizaje individual es la necesidad de registrar los contactos a nivel individual, lo que suele ser difícil en estudios antropológicos clásicos. Para evaluar algunas de estas hipótesis, decidimos estudiar la evolución de la habilidad al interior de comunidades de video juegos en línea. Las comunidades virtuales son un lugar privilegiado para estudiar cómo cambian las estrategias en el tiempo, al mismo tiempo que ofrecen la posibilidad de estudiar poblaciones suficientemente grandes con un alto grado de detalle, lo que no se logra con ninguna de las metodologías clásicas de las ciencias sociales (censo, encuesta, etnografía). Sin dejar de tener en cuenta las particularidades que supone el estudio de comunidades virtuales en oposición a las comunidades reales, es necesario reconocer que ellas son manifestaciones propias de las sociedades y como tales, son un medio para entender el comportamiento social emergente de los seres humanos.

En la última década se han desarrollado investigaciones sobre comportamientos humanos basadas en datos obtenidos de comunidades virtuales. Durante el transcurso del doctorado emergió con fuerza una área conocida como *Ciencias Sociales Computacionales* [30, 31], que tiene en la actualidad reconocimiento tanto en institutos de primer nivel de ciencias de la computación como en los gigantes tecnológicos relacionados a la industria del software. En antropología se han llevado adelante estudios sobre la dinámica de copia y modificación de

código en competencias de programación en línea [32]. En computación se ha estudiado el efecto del miedo sobre el desempeño en una base de datos de ajedrez rápido [33].

Una de las ventajas del estudio de las comunidades virtuales es el intercambio de conocimientos entre los participantes. Esto crea oportunidades para estudiar no solo el aprendizaje individual, sino también cómo se construye el conocimiento de manera colaborativa en entornos en línea. Además, las comunidades virtuales tienen la ventaja de que las interacciones se producen de forma espontánea y natural, ya que los participantes se involucran en el aprendizaje de manera voluntaria. Esto contrasta con entornos de laboratorio, lo que hace que los datos recopilados reflejen con mayor fidelidad el comportamiento real de las personas en contextos de aprendizaje.

Entender cómo los agentes aprenden rasgos culturales es también una de las preguntas que deberán resolver las Ciencias Sociales Computacionales en los próximos años. Empresas como Deep Mind y Meta abrieron nuevas líneas de investigación en lo que se ha dado a llamar *cooperative AI* [34, 35]. Solo recién durante la escritura de esta tesis, han comenzado a surgir los primeros resultados de nivel humano en un juego de estrategia “*Diplomacy*”, que involucra negociación entre varios jugadores [36, 37]. Sin embargo, mediante discusiones personales con investigadores formados en el campo, podemos ratificar que todavía no ha madurado lo suficiente este tipo de inteligencia artificial multi-agente.

Conocer cómo las personas aprendemos es un problema importante, además, para diversos ámbitos de la vida, como la educación, el trabajo y el deporte. Entre las preguntas surgidas en este contexto, podemos destacar:

- ¿Cuál es la relación entre la formación de equipos y el aprendizaje individual a largo plazo?
- ¿Cuál es la mejor forma de medir el aprendizaje de un individuo en el tiempo?
- ¿Cuál es el efecto que la posición topológica de un individuo en la red de intercambios dinámica tiene sobre el aprendizaje individual?
- ¿Cuáles son las propiedades más generales del aprendizaje?

Estas son algunas de las preguntas particulares que desarrollamos en esta tesis.

1.2. Teoría elemental de la probabilidad

Desde el descubrimiento de las reglas de la probabilidad a finales del siglo 18 hasta el presente, la teoría de la probabilidad es el enfoque más utilizado para razonar en contextos de incertidumbre. Las reglas de la probabilidad, conocidas como “regla de la suma” y “regla del producto”, han sido derivadas formalmente a partir de varios sistemas axiomáticos conceptualmente distintos e independientes entre sí [38]. El sistema axiomático de Ramsey [39] las deriva haciendo una analogía con los pagos que aceptaríamos en una apuesta. El sistema axiomático de Cox [40] las deriva a partir de axiomas del “sentido común”. El sistema axiomático de Kolmogorov [41] en cambio, incluye una caso particular de la regla de la suma como axioma e incorpora la regla del producto como una definición complementaria que cumple con los axiomas. Por su elegancia matemática el sistemas axiomático de Kolmogorov es el más utilizado. En esta sección presentamos la teoría elemental de la probabilidad siguiendo los axiomas de Kolmogorov, en su versión simple.

1.2.1. Sistema axiomático

En la siguiente cita se reproduce el sistema axiomático, definido sobre elementos discretos [41],

Let \mathbf{U} be a collection of elements u_1, u_2, u_3, \dots , which we shall call *elementary events*, and \mathbb{F} a set of subsets of \mathbf{U} ;

1. \mathbb{F} is a field of sets.
2. \mathbb{F} contains the set \mathbf{U}
3. To each set $R_i \in \mathbb{F}$ is assigned a non-negative real number, $P(R_i)$.
4. $P(\mathbf{U}) = 1$
5. If $R_1 \in \mathbb{F}$ and $R_2 \in \mathbb{F}$ have no elements in common, then

$$P(R_1 \cup R_2) = P(R_1) + P(R_2). \quad (1.1)$$

A system of sets, \mathbb{F} , together with a definite assignment of numbers $P(R_i)$, satisfying Axioms 1-5, is called *field of probabilities*.

Field (campo en español) hace referencia a que el conjunto de conjuntos \mathbb{F} es cerrado respecto a complemento, unión e intersección: “*A system of sets is called a field if the sum, product and difference of two sets of the system also belongs to the same system*” [41]. Kolmogorov denomina al campo \mathbb{F} bajo la etiqueta *random events* (rangos de eventos elementales). La forma más simple de crear un campo de probabilidad (*field of probability*) es tomar un conjunto arbitrario de eventos elementales $\mathbf{U} = \{u_1, u_2, u_3, \dots\}$, y asignarle una probabilidad arbitraria a cada evento elemental $\{p_1, p_2, p_3, \dots\}$ tal que la suma $p_1 + p_2 + p_3 + \dots = 1$.

Los eventos elementales $u_i \in \mathbf{U}$ representan las hipótesis conjuntas elementales (o universos paralelos). Para construir el conjunto \mathbf{U} de hipótesis conjuntas elementales, Kolmogorov supone la existencia de un *complex of conditions* $\mathcal{H} = \{H_1, \dots, H_N\}$. En esta tesis denominamos a \mathcal{H} problema de investigación y a sus elementos H_i hipótesis de investigación. Las hipótesis de investigación están compuestas por hipótesis elementales $H_i = \{h_1^{(i)}, h_2^{(i)}, \dots\}$ mutuamente excluyentes, y el producto cartesiano de todas las hipótesis de investigación produce el conjunto de eventos elementales,

$$\mathbf{U} = H_1 \times H_2 \times H_3 \times \dots = \underbrace{\{h_1^{(1)} h_1^{(2)} h_1^{(3)} \dots, \dots\}}_{u_1}, \underbrace{\{h_2^{(1)} h_1^{(2)} h_1^{(3)} \dots, \dots\}}_{u_2}, \dots.$$

Por ejemplo, si tiramos dos veces la moneda, hay 4 posibles eventos elementales $\mathbf{U} = \{CC, CS, SC, SS\} = H_1 \times H_2$, con $H_1 = H_2 = \{C, S\}$, donde C representa Cara y S Seca.

1.2.2. Terminología basada en la teoría de conjuntos

Kolmogorov aclara algunos aspectos terminológicos, que creemos conveniente reproducir, ya que relacionan la teoría de conjuntos con la probabilidad.

We have defined the object of our future study \mathbb{H} . However, in the theory of probability many set-theoretic concepts are designed by other terms. We shall give here a brief list of such concepts

Theory of Sets

1. R_1 and R_2 do not intersect, i.e. $R_1 R_2 = \emptyset$
2. $R_1 R_2 \dots R_n = \emptyset$
3. $R_1 R_2 \dots R_n = X$
4. $R_1 + R_2 + \dots + R_n = Y$
5. The complementary set $\overline{R_1}$
6. $R_0 = \emptyset$
7. $R_* = \mathbf{U}$
8. The system \mathbb{E} of sets R_1, R_2, \dots that do not intersect forms a decomposition of the set if \mathbf{U} if $R_1 + R_2 + \dots = \mathbf{U}$.
9. B is a subset of R_1 : $B \subset R_1$

Random Events

1. Events R_1 and R_2 are incompatible
2. Events R_1, R_2, \dots, R_n are incompatible
3. Event X is defined as the simultaneous occurrence of events R_1, R_2, \dots, R_n
4. Event Y is defined as the occurrence of at least one of the events R_1, R_2, \dots, R_n
5. The opposite event $\overline{R_1}$ consisting of the non-occurrence of event R_1 .
6. Event R_0 is impossible
7. Event R_* must occur
8. The experiment \mathbb{E} consists of determining which of the events R_1, R_2, \dots occurs. We therefore call R_1, R_2, \dots the possible results of experiment \mathbb{E} .
9. From the occurrence of event B follows the inevitable occurrence of R_1

1.2.3. Las reglas de la probabilidad

En el sistema axiomático de Kolmogorov, un caso particular de la regla de la suma de la teoría de la probabilidad aparece incluido directamente como axioma 5. La regla del producto (o probabilidad condicional) en cambio se incluye mediante una definición complementaria que cumple con los axiomas. En la sección 4 “*Immediate Corollaries of the Axioms; Conditional Probabilities; Theorem of Bayes*” dice,

From $R + \overline{R} = \mathbf{U}$ and the Axioms 4 and 5 it follows that

$$P(R) + P(\overline{R}) = 1 \quad (1.2)$$

$$P(\overline{R}) = 1 - P(R). \quad (1.3)$$

Since $\overline{\mathbf{U}} = \emptyset$, then, in particular

$$P(\emptyset) = 0. \quad (1.4)$$

If R_1, R_2, \dots, R_n are incompatible, then from Axiom 5 follows the formula (the Addition Theorem)

$$P(R_1 + R_2 + \dots + R_n) = P(R_1) + P(R_2) + \dots + P(R_n). \quad (1.5)$$

If $P(R_1) > 0$, then the quotient

$$P(R_2|R_1) = \frac{P(R_1R_2)}{P(R_1)} \quad (1.6)$$

is defined to be the *conditional probability* of the event R_2 under the condition R_1 .

From (1.6) it follows immediately that

$$P(R_1R_2) = P(R_1)P(R_2|R_1). \quad (1.7)$$

And by induction we obtain the general formula (the Multiplication Theorem)

$$P(R_1R_2 \dots R_n) = P(R_1)P(R_2|R_1) \dots P(R_n|R_1R_2 \dots R_{n-1}). \quad (1.8)$$

The following theorem follows easily:

$$P(R_2|R_1) \geq 0 \quad (1.9)$$

$$P(\mathbf{U}|R_1) = 1 \quad (1.10)$$

$$P(R_2 + R_3|R_1) = P(R_2|R_1) + P(R_3|R_1) \quad (1.11)$$

Comparing formulae (1.9)-(1.11) with axioms 3 to 5, we find that the system \mathbb{F} of sets together with the set function $P(R_2|R_1)$ (provided R_1 is a fixed set), form a field of probability and therefore, *all the above general theorems concerning $P(R_2)$ hold true for the conditional probability $P(R_2|R_1)$* (provided the event R_1 is fixed). It is also easy to see that

$$P(R_1|R_1) = 1. \quad (1.12)$$

From (1.7) and the analogous formula

$$P(R_1R_2) = P(R_2)P(R_1|R_2) \quad (1.13)$$

we obtain the important formula:

$$P(R_2|R_1) = \frac{P(R_1|R_2)P(R_2)}{P(R_1)} \quad (1.14)$$

which contains, in essence, the Theorem of Bayes. (. . .)

Los rangos R_i se suelen expresar mediante proposiciones basadas en los valores de las diferentes hipótesis de investigación. En el ejemplo del lanzamiento de dos monedas tenemos el problema de investigación $\mathcal{H} = \{H_1, H_2\}$, con $H_1 = H_2 = \{C, S\}$. En este contexto, para expresar que en el primer lanzamiento de la moneda se observó Cara (o que el verdadero valor de la hipótesis de investigación H_1 es C) se suele utilizar la siguiente notación $H_1 = \{C\}$ o simplemente $H_1 = C$. El rango que se corresponde con esta proposición es $D = \{CC, CS\}$, cualquiera de todos los eventos elementales en los que aparece la hipótesis elemental $H_1 = \{C\}$. Al conjunto D se lo suele llamar Dato. Por el axioma 5, su probabilidad se descompone naturalmente como una suma.

$$P(H_1 = C) = P(\{CC\}) + P(\{CS\})$$

Además, para actualizar la probabilidad de la hipótesis de investigación H_2 expresamos el teorema de Bayes de la siguiente forma,

$$P(H_2 = h|H_1 = d) = \frac{P(H_1 = d|H_2 = h)P(H_2 = h)}{P(H_1 = d)} \stackrel{*}{=} \frac{P(d|h)P(h)}{P(d)} = P(h|d), \quad \forall h \in H_2$$

con $d = C$. En el lado derecho de la igualdad $\stackrel{*}{=}$ se presenta una notación aún más simplificada, en la que solo se escribe el valor de la hipótesis elemental sin hacer referencia a las hipótesis de investigación que pertenecen a \mathcal{H} . Utilizamos las variables h y d para sugerir la idea de que h es una hipótesis y d es un dato. En palabras de Kolmogorov

The formula (1.2.3) is considered as the probability $P(h|d)$ of the hypothesis h after the occurrence of event d ($P(h)$ then denotes the *a priori* probability of h).

En línea con esta interpretación, volvemos a expresar el teorema de Bayes señalando los nombres que se les ha asignado históricamente a las probabilidades involucradas.

$$\underbrace{P(\text{Hipótesis} \mid \text{Datos})}_{\text{Posterior}} = \frac{\underbrace{P(\text{Datos} \mid \text{Hipótesis})}_{\text{Verosimilitud}} \underbrace{P(\text{Hipótesis})}_{\text{Prior}}}{\underbrace{P(\text{Datos})}_{\text{Evidencia}}}. \quad (1.15)$$

1.3. Aplicación estricta de las reglas de la probabilidad

La teoría de la probabilidad es el sistema de razonamiento para contextos de incertidumbre. La aplicación estricta de sus reglas (enfoque bayesiano de la probabilidad) nos obliga a actualizar las probabilidades de las hipótesis a través del teorema de Bayes $P(H_i|D)$ y, por lo tanto, a definir una probabilidad *a priori* de la hipótesis $P(H_i)$. Sin embargo, la única restricción que impone el sistema axiomático para asignar probabilidades a las hipótesis elementales es que la suma sea $h_1 + h_2 + \dots + h_n = 1$. Bajo este criterio, existen infinitas formas de asignar probabilidades *a priori*. Es decir, la teoría de la probabilidad nos obliga a definir una probabilidad *a priori*, pero no nos dice cómo. En esta sección revisaremos algunas cuestiones relacionadas con la aplicación estricta de las reglas de la probabilidad.

1.3.1. Acuerdos intersubjetivos en contextos de incertidumbre

Debido a que las ciencias empíricas (desde la física hasta las ciencias sociales) deben validar sus proposiciones en sistemas naturales abiertos, el conocimiento contiene siempre algún grado de incertidumbre. La teoría de la probabilidad es usada desde el siglo 18 para expresar conocimiento que es parcial. Por ejemplo, supongamos que tenemos tres cajas *exactamente* iguales y una persona esconde un regalo dentro de una de ellas. Una vez escondido, el regalo permanece en la misma caja, su posición no es aleatoria. En este escenario tenemos tres hipótesis elementales r_i , donde cada una se interpreta como *el regalo se encuentra en la caja i* , siendo el conjunto de hipótesis elementales $H_R = \{r_1, r_2, r_3\}$. En la figura 1.1 representamos las cajas con cuadrados negros con la probabilidad $P(r_i)$ de cada una de las hipótesis elementales. La pregunta es, ¿cuál de todas las infinitas distribuciones de probabilidades expresan más fielmente nuestro conocimiento parcial del mundo?

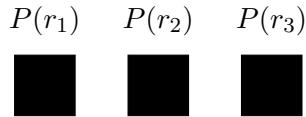


Figura 1.1: Los cuadrados representan las cajas y $P(r_i)$ la probabilidad que le asignamos a la hipótesis elemental r_i .

Por ejemplo, si por algún motivo supiéramos que el regalo está en la caja del medio, podríamos expresar una certeza total asignando toda la probabilidad a la caja del medio. Esta situación se representa en la figura 1.2(a) en la que las hipótesis r_1 y r_2 tienen asignada probabilidad nula y la hipótesis r_2 tiene asignada probabilidad 1. Sin certeza, en cambio, nos vemos obligados a dividir la probabilidad de alguna forma. Cuando tenemos una creencia a favor de que el regalo se encuentra en alguna de las cajas r_i , podemos expresarlo asignando mayor probabilidad a esa hipótesis que al resto. Esta situación se representa en la figura 1.2(b), en la que asignamos mayor probabilidad a la hipótesis r_2 que al resto. Cuando no tenemos información alguna que nos haga preferir alguna de las hipótesis, podemos expresarlo dividiendo la probabilidad en partes iguales $P(r_1) = P(r_2) = P(r_3)$. De aquí en adelante, diremos

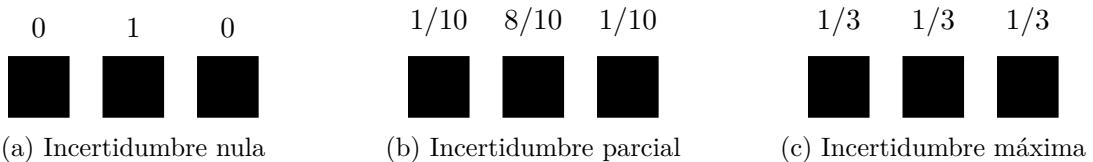


Figura 1.2: Tres distribuciones de probabilidades que expresan: (a) un conocimiento completo (incertidumbre nula), (b) conocimiento parcial (incertidumbre parcial) y (c) ignorancia total (incertidumbre máxima).

que *distribución de probabilidad* y *distribución de creencias* se utilizarán indistintamente.

Lo que queremos en ciencia empírica no es expresar nuestras subjetividades individuales sino alcanzar acuerdos entre subjetividades. La ciencia es una institución que tiene pretensión de descubrir verdades, esto es, afirmaciones que valgan para todas las personas. Siguiendo al epistemólogo Juan Samaja, llamamos a esto *acuerdos intersubjetivos* pues, a diferencia del concepto “objetividad”, incluye una dimensión constructiva entre las subjetividades [42].

Pero no en una “intersubjetividad” que podría resultar de un acuerdo ocasional, contingente, entre los sujetos, sino como la forma de la equilibración en que pueden coexistir los sujetos en una comunidad que “distribuye” entre ellos las cosas de “un único y mismo modo”.

Las ciencias formales (matemática, lógica) alcanzan estos acuerdos derivando teoremas dentro de sistemas axiomáticos cerrados. Las ciencias empíricas, desde la física hasta las ciencias sociales, deben alcanzar estos acuerdos en sistemas naturales abiertos que contienen siempre algún grado de incertidumbre. ¿Es posible alcanzar acuerdos intersubjetivos en contextos de incertidumbre (i.e., *verdades empíricas*)?

Esta pregunta está formulada desde el inicio de los debates del desarrollo de la teoría de la probabilidad en 1648, cuando Pascal y Fermat discuten cuál es el *precio justo* en contextos de incertidumbre [43]. La primer definición matemática de justicia, verdad u honestidad ya existía previamente a estos debates bajo el viejo principio filosófico de *indiferencia*, que es incorporado

al inicio del siglo 18 por quienes serán considerados luego los fundadores de la teoría de la probabilidad [44]. Tres siglos después, luego que se deriva la teoría de la información a partir de la aplicación estricta de las reglas de la probabilidad y se define finalmente el concepto de *entropía* como medida de la tasa de información, se generaliza finalmente el concepto de *honestidad* como máxima entropía dada la información disponible (alias MaxEnt) [45].

The essence of “honesty” or “objectivity” demands that we take into account all the evidence we have, not just some arbitrarily chosen subset of it. Any such choice would amount either to ignoring evidence that we have, or presuming evidence that we do not have.

En este sentido, la distribución de probabilidad honesta será la que no afirma más de lo que se sabe (maximizando incertidumbre o entropía) sin dejar de decir todo lo que sí se sabe (incorporando toda la información disponible).

We interpret Shannon’s theorem as indicating that, out of all distributions p_i that agree with the constraints, the one that maximizes the Shannon entropy, represents the “most honest” description of our state of knowledge, in the following sense: it expresses the enumeration of the possibilities and the evidence E; but is careful to assume nothing beyond that. If we subsequently acquire more information B that can be interpreted as an event in H_0 , then we can update this distribution by Bayes’ theorem. [44]

Todas las distribuciones de probabilidad de la familia exponencial maximizan entropía dada la información de los parámetros (Uniforme($x|a, b$), Bernoulli($x|p$), Beta($x|\alpha, \beta$), gaussiana($x|\mu, \sigma^2$), ...). Se ha demostrado también que las soluciones que encuentra la física estadística en el siglo 19 cumplen con este principio [46, 47]. Lo interesante es que este principio ya está codificado en la propia regla del producto de la teoría de la probabilidad. Lo que esperamos de una probabilidad *a priori* es justamente que incorpore toda la información previa M que tenemos antes de comenzar a ver los datos D [45]. En palabras de Edwin Jaynes,

To solve this problem does not require any new principles beyond the product rule (...). Let us now use the notation:

$$\begin{aligned} M &= \text{prior information,} \\ H &= \text{some hypothesis to be tested,} \\ D &= \text{the data,} \end{aligned}$$

and write the product rule in the form

$$P(DH|M) = P(D|HM)P(H|M) = P(H|DM)P(D|M). \quad (1.16)$$

Luego, para determinar la probabilidad *a priori* $P(H|M)$ necesitamos definir con precisión cuál es la información previa M . Generalmente M representa el modelo causal, pero en principio M incluye el sistema completo de restricciones que utilizamos sobre el cual se condiciona toda la inferencia. La sección actual, *Acuerdos intersubjetivos en contextos de incertidumbre*, terminará con la evaluación de los modelos causales alternativos. Pero antes de ello, veamos un ejemplo concreto.

Retomemos el ejemplo de las tres cajas y el regalo escondido. Supongamos que solo sabemos que el regalo está escondido detrás de alguna de las tres cajas y no tenemos información

adicional que nos haga preferir alguna caja por sobre otra. En este contexto, **no** es difícil estar de acuerdo en que debemos dividir la probabilidad en partes iguales para expresar de forma honesta el estado actual de nuestro conocimiento. Esta distribución de probabilidad se corresponde con la figura 1.2(c) y no casualmente corresponderá con la distribución que maximiza la incertidumbre (o entropía) entre todas las opciones que incorporan la única información disponible: el regalo está en alguna de las tres cajas. Para derivar las distribuciones de probabilidad que maximizan la entropía dadas las restricciones disponibles, se suele usar los multiplicadores de Lagrange. Aquí nos bastó con utilizar el viejo principio filosófico de indiferencia, que suele corresponderse con la intuición común.

Continuemos con el ejemplo. Supongamos que se nos pide que elijamos una caja y alguien (un oráculo), quien conoce la ubicación del regalo, muestra el interior de una de las cajas que **no** contiene el regalo. Se incorporan, entonces, las hipótesis elementales c_i que corresponde con “elegimos la caja i ” y s_i con “muestran el interior de la caja i ”. Luego, nuestro problema de investigación \mathcal{H} consiste de tres hipótesis, $\mathcal{H} = \{H_R, H_C, H_S\}$, cada una con tres hipótesis elementales, $H_R = \{r_1, r_2, r_3\}$, $H_C = \{c_1, c_2, c_3\}$ y $H_S = \{s_1, s_2, s_3\}$. En la descripción del juego se explicita la siguiente restricción: si el regalo se encuentra en la caja i , no puede ocurrir que se muestre el interior de la caja i . Debido a que no tenemos ninguna otra restricción adicional, si el regalo se encontrara en la caja 1, es decir que es cierta r_1 , no hay preferencia por ninguna de las hipótesis elementales s_2 o s_3 (de forma equivalente se puede hacer el razonamiento para los casos en los que fueran ciertas r_2 o r_3).

A esta información inicial la llamaremos *Modelo A*, o simplemente M_A . Necesitamos entonces la distribución de probabilidad que sigue la visión honesta del sistema $P(H_R, H_C, H_S | M_A)$. Para simplificar el problema, y sin perder generalidad, supondremos además que siempre elegimos la caja 1, es decir, que es cierta c_1 . De esta manera, reducimos el problema a solo dos hipótesis: $P(H_R, H_S | c_1, M_A)$. El sistema de restricciones de los modelos se suele representar mediante lo que se conoce como *redes bayesianas*. En la figura 1.3(a) se especifica el sistema de restricciones M_A usando la notación de redes bayesianas. Los nodos de las redes bayesianas representan las hipótesis del problema de investigación $\mathcal{H} = \{H_R, H_S, \{c_1\}\}$ y las flechas entre nodos representan la dependencia entre hipótesis. En el modelo A hay una única dependencia entre la hipótesis H_R y la hipótesis H_S , $\forall i \in \{1, 2, 3\} P(s_i | r_i, M_A) = 0$. Debido a que en M_A no existe ninguna dependencia que involucre a H_C , por simplicidad decidimos no incluir esa hipótesis en la red bayesiana.

Nada nos obliga a suponer que el M_A es correcto. Por ejemplo, podemos sospechar que s_i , la caja que se muestra, tampoco puede coincidir con la caja elegida c_1 , $P(s_1 | c_1, M_A) = 0$. Esto plantea una restricción (o dependencia) adicional. El sistema de restricciones que incluye esta restricción y todas las restricciones del modelo A , lo llamamos *Modelo B* (M_B). Este modelo tiene asociado la distribución de probabilidad $P(H_R, H_S | c_1, M_B)$. Bajo el sistema de restricciones M_B , si el regalo se encontrara en la caja 1, es decir si r_1 fuera cierta, no tenemos información que nos haga preferir ninguna de las hipótesis elementales s_2 o s_3 , $P(s_2 | r_1, c_1, M_B) = P(s_3 | r_1, c_1, M_B) = 1/2$. En cambio, si el regalo se encontrara en una caja distinta a la que elegimos (i.e., $r_i \neq r_1$), entonces el oráculo que muestra el interior de la caja tiene una única opción s_j con las condiciones $j \neq i$ y $j \neq 1$, $P(s_3 | r_2, c_1, M_B) = 1$ y $P(s_2 | r_3, c_1, M_B) = 1$.

En la figura 1.3(b) se especifica el modelo B de forma gráfica a través de una red bayesiana. En la notación de redes bayesianas, las hipótesis que finalmente son observadas (los datos) se agregan como nodos pintados de gris. Es el caso de la hipótesis c_1 , que aparece en la representación gráfica del modelo M_B . En M_A podríamos haber agregado ese nodo, pero por

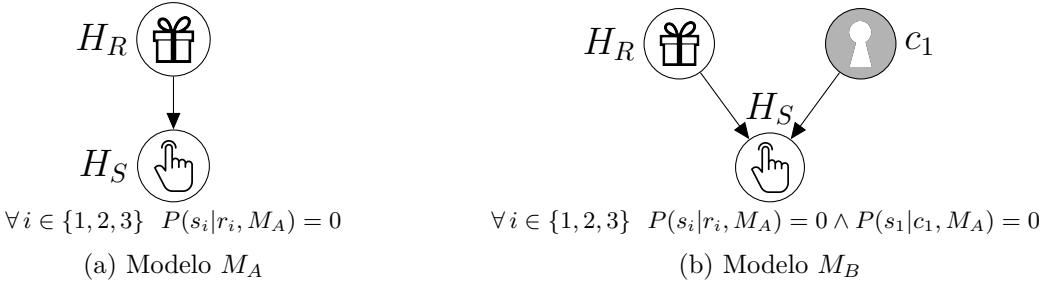


Figura 1.3: Modelos alternativos especificados como redes bayesianas. Los nodos blancos representan las hipótesis, la flecha representa relaciones de dependencia entre hipótesis, y los nodos pintados de gris representan hipótesis ya conocidas (datos). En el Modelo A (a) la pista sólo depende de la posición del regalo, no puede mostrarse el interior de la caja que contiene el regalo $P(s_i|r_i, M_A) = 0$. En el Modelo B (b) la pista depende de la posición del regalo y la caja elegida previamente, no puede mostrarse el interior de la caja que contiene el regalo, ni la caja elegida $P(s_i|r_i, M_B) = 0, P(s_1|c_1, M_B) = 0$.

simplicidad no lo hicimos debido a que, según M_A , la hipótesis H_C no interactúa con el resto de las hipótesis.

Para calcular las distribuciones de probabilidad $P(H_R, H_S|c_1, M_A)$ y $P(H_R, H_S|c_1, M_B)$, analicemos en detalle las restricciones de los modelos. En particular, cada modelo tiene asociado un conjunto de *universos posibles*. En ambos modelos, el regalo puede estar indistintamente en cualquier caja. Sin embargo, las cajas que se pueden mostrar dependen del modelo y la posición del regalo. En M_A , si el regalo se encontrara en la caja 1 (i.e., r_1 fuera cierta), la caja que se nos muestra puede ser cualquiera de las dos opciones disponibles, es decir s_2 o s_3 pueden ser ciertas. De forma equivalente, para cada posible posición del regalo r_i , en M_A nos pueden mostrar indistintamente cualquiera de las otras dos cajas. En el modelo M_B , solo cuando la posición de regalo **no coincide** con la de la caja que elegimos, se nos puede mostrar indistintamente cualquiera de las dos otras cajas (es decir, s_2 o s_3 pueden ser ciertas). En el resto de los casos, cuando la elección de la caja no coincide con la posición del regalo, solo es posible que nos muestren una única caja.

Aplicando el principio de indiferencia, la distribución surge de dividir la probabilidad en partes iguales en cada una de las bifurcaciones de los universos paralelos que producen los modelos alternativos. En la figura 1.4 se representan el árbol de los posibles universos paralelos bajo cada uno de los modelos. Cada camino que va de la raíz a una de las hojas representa un universo posible. Los nodos de ese camino representan las hipótesis elementales que son ciertas en ese universo (o el estado del universo). Y las bifurcaciones contienen una fracción que representa cómo se divide la probabilidad entre los universos paralelos (en este caso aplicando el principio de indiferencia en todos los niveles, es decir, se dividen en partes iguales).

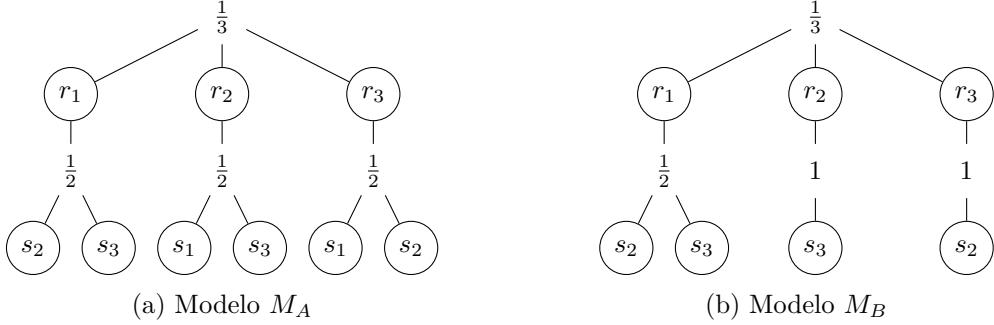


Figura 1.4: Los posibles universos dados los sistemas de restricciones alternativos (o modelos). Debido a que no hay información previa que nos haga preferir alguna de las bifurcaciones, siguiendo el principio de indiferencia dividimos la creencia en partes iguales en cada una de ellas. Por simplicidad suponemos c_1 como válida.

Luego, para calcular la probabilidad conjunta de que el regalo se encuentre en la caja i y se nos muestre la caja j , $P(r_i, s_j | c_1, M_k)$, debemos multiplicar todas las fracciones presentes en cada uno de los caminos que van de la raíz a las hojas. A los caminos no presentes en el árbol de universos paralelos se les debe asignar probabilidad 0 debido a que son caminos imposibles dadas las restricciones del modelo. La probabilidad asignada a cada uno de los universos paralelos (o combinaciones) se resume en la tabla 1.1 en donde sí se muestran las combinaciones imposibles con una probabilidad de 0.

$P(r_i, s_j c_1, M_A)$				$P(r_i, s_j c_1, M_B)$			
	r_1	r_2	r_3		r_1	r_2	r_3
s_1	0	$1/6$	$1/6$		0	0	0
s_2	$1/6$	0	$1/6$		$1/6$	0	$1/3$
s_3	$1/6$	$1/6$	0		$1/6$	$1/3$	0

Tabla 1.1: Distribución de probabilidad honesta para cada uno de los modelos. Las celdas representan universos paralelos y el valor la probabilidad asignada a cada uno de ellos. Las combinaciones que tienen 0 como valor corresponden con aquellas combinaciones imposibles de acuerdo a las restricciones del modelo.

Una vez calculada la distribución de probabilidad conjunta, podemos responder cualquier pregunta relacionada al problema de investigación mediante la aplicación estricta de las reglas de la probabilidad. La regla de la suma establece que cualquier creencia marginal puede ser obtenida integrando la creencia conjunta. Por ejemplo, para obtener la probabilidad de la hipótesis s_j debemos recuperar las probabilidades que asignamos a todos los universos paralelos en los que esa hipótesis elemental está presente,

$$\begin{aligned}
 P(s_j | c_1, M_A) &= \sum_{r_i \in H_R} P(r_i, s_j | c_1, M_A) = 1/3 \quad \forall j \in \{1, 2, 3\}, \\
 P(s_j | c_1, M_B) &= \sum_{r_i \in H_R} P(r_i, s_j | c_1, M_B) = 1/2 \quad \forall j \in \{2, 3\}.
 \end{aligned} \tag{1.17}$$

Esto es equivalente a sumar todos los elementos de una de las filas de la tabla. Del mismo modo podemos calcular la probabilidad marginal de las hipótesis $r_i \in H_R$. En ese caso deberíamos

sumar todos los elementos de una columna. En ambos modelos, la probabilidad marginal sobre la posición del regalo es $1/3$.



Figura 1.5: En la figura 1.5(a) mostramos la distribución de probabilidad marginal de la hipótesis R a priori. La cerradura representa la restricción c_1 . En ambos modelos el prior sobre el regalo es el mismo. En la figura 1.5(b) nos preguntamos cuál es la distribución de probabilidad honesta sobre la posición del regalo luego de observar que la caja 2 está vacía.

Para calcular la distribución de probabilidad luego de observar que la caja dos está vacía, debemos aplicar la regla del producto. La regla del producto simplemente codifica el principio de coherencia pues para calcularla nos debemos quedar únicamente con la creencia previa que sigue siendo compatible con el observable (i.e., $P(r_i, s_2)$). Esto es la fila de la distribución de probabilidad conjunta a priori correspondiente a la hipótesis elemental s_2 .

$P(r_i, s_2 c_1, M_A)$				$P(r_i, s_2 c_1, M_B)$					
	r_1	r_2	r_3		r_1	r_2	r_3		
s_2	1/6	0	1/6	1/3	s_2	1/6	0	1/3	1/2

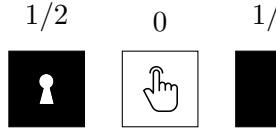
Tabla 1.2: La creencia conjunta y marginal que sobrevive luego de ver el dato.

La probabilidad que sobrevive, $P(r_i, s_2|c_1, M_k)$, no representa todavía nuestra nueva creencia total pues no suma 1, $\sum_i P(r_i, s_2|c_1, M_k) = P(s_2|c_1, M_k) < 1$. Para que la probabilidad que sobrevive sea considerada nuestra nueva creencia total necesitamos normalizarla para que en conjunto vuelva a sumar 1.

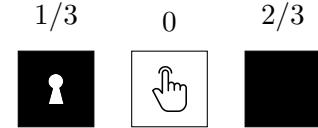
$$\underbrace{P(r_i|s_2, c_1, \text{Modelo})}_{\text{Nueva creencia}} = \frac{\overbrace{P(r_i, s_2|c_1, \text{Modelo})}^{\text{Creencia que sobrevive}}}{\underbrace{P(s_2|c_1, \text{Modelo})}_{\text{Creencia total que sobrevive}}}.$$

Si multiplicamos a ambos lados por el denominador, obtenemos la expresión de la regla del producto de la teoría de la probabilidad. La nueva distribución de probabilidad a la que llegamos depende del modelo elegido. Si bien ambas respuestas son diferentes, ellas comparten la propiedad de ser la distribución de probabilidad que cumple con las restricciones. Lo que hemos hecho no es otra cosa que aplicar el Teorema de Bayes,

$$\underbrace{P(r_i|s_2, c_1, M_k)}_{\text{Posterior}} = \frac{P(r_i, s_2|c_1, M_k)}{P(s_2|c_1, M_k)} = \frac{\overbrace{P(s_2|r_i, c_1, M_k)}^{\text{Verosimilitud}} \overbrace{P(r_i|c_1, M_k)}^{\text{Prior}}}{\underbrace{P(s_2|c_1, M_k)}_{\text{Evidencia}}}. \quad (1.18)$$



(a) $P(r_i|s_2, c_1, M_A)$



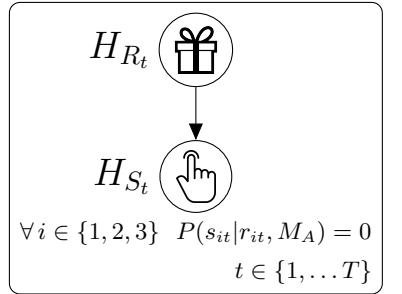
(b) $P(r_i|s_2, c_1, M_A)$

Figura 1.6: Distribución de probabilidad a posterior dado el dato s_2 y el modelo. El modelo A asigna la misma probabilidad a la caja 1 y 3. El modelo B asigna $1/3$ a la hipótesis r_1 y $2/3$ a la hipótesis r_3 .

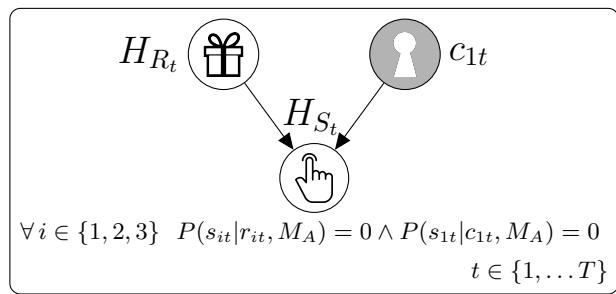
1.3.2. Evaluación de modelos

Computar las distribuciones de probabilidad dados los modelos no tendría ningún valor si la elección del modelo quedara librada al arbitrio de las subjetividades individuales. Y, justamente cuando tenemos más de un sistema de restricciones (o modelos), estamos en presencia de una hipótesis de investigación adicional $H_M = \{M_A, M_B\}$ que también puede ser evaluada mediante la aplicación estricta de las reglas de la probabilidad. Es decir, en el fondo nuestro problema de investigación \mathcal{H} debe incluir H_M , $\mathcal{H} = \{H_R, H_S, \{c_1\}, H_M\}$.

Para evaluar los modelos, necesitamos datos que provengan de la realidad subyacente. Con esos datos luego podremos calcular el posterior de los modelos, $P(M_k|\text{Datos})$. En particular, en este ejemplo vamos a generar los datos utilizando la distribución de probabilidad del modelo M_B , $P(H_R, H_S|c_1, M_B)$, eligiendo primero la posición del regalo r_i a partir de la distribución $P(r_i|c_1, M_B)$ y eligiendo luego la pista s_j a partir de la distribución $P(s_j|r_i, c_1, M_B)$. Repetiremos el procedimiento T veces, $t \in \{1, \dots, T\}$. Luego, nuestro problema de investigación contiene $\mathcal{H} = \{H_{R_1}, H_{S_1}, \dots, H_{R_T}, H_{S_T}, M_A, M_B\}$, con $H_{R_t} = \{r_{1t}, r_{2t}, r_{3t}\}$ y $H_{S_t} = \{s_{1t}, s_{2t}, s_{3t}\}$. Vamos a considerar las mismas restricciones entre hipótesis al interior de cada repetición y ninguna nueva restricción adicional entre repeticiones. Las repeticiones se especifican en redes bayesianas mediante el uso de placas (ver figura 1.7). Las placas son como naipes repetidos,



(a) Modelo A



(b) Modelo B

Figura 1.7: Extensión de los modelos A y B mediante la repeticiones de las restricciones entre hipótesis. La notación de redes bayesianas especifica las repeticiones mediante placas. Las placas son como pilas de naipes, que difieren entre sí solo por el número indicado en el subíndice, $t \in \{1, \dots, T\}$.

uno arriba del otro, que difieren solamente en el número indicado por el subíndice. Si se despliegan las placas de la figura 1.7 veríamos T redes bayesianas, una al lado de la otra, sin dependencias entre hipótesis que las vinculen entre sí. El hecho de que no estén conectadas

hace que sean independientes entre sí, las hipótesis observadas en una placa no afectan las probabilidades en el resto de las placas.

Para computar la distribución de probabilidades respecto de los modelos alternativos debemos aplicar el Teorema de Bayes luego de observar los datos. Sea $\text{Datos} \in (H_{R_1}, H_{S_1}, \dots, H_{R_T}, H_{S_T})$ el vector de hipótesis elementales observadas $(r_{i1}, s_{j1}, \dots, r_{iT}, s_{jT})$, y sea la restricción restante $R = c_1$ es, el posterior del modelo M_k es

$$\underbrace{P(M_k | \underbrace{r_{i1}, s_{j1}, \dots,}_{\text{Datos}}, \underbrace{c_1}_{R})}_{P(\text{Modelo} | \text{Datos}, \text{Restricciones})} = \frac{\overbrace{P(\text{Datos} | \text{Modelo}, \text{Restricciones})}^{P(r_{i1}, s_{j1}, \dots | M_k, c_1)} P(M_k | c_1)}{P(\text{Datos} | c_1)}, \quad (1.19)$$

donde la probabilidad de los datos dado el modelo y la restricción restante es

$$P(\text{Datos} | \text{Modelo}, \text{Restricciones}) = \prod_{t=1}^T P(s_{jt} | M, c_{1t}) P(r_{it} | s_{jt}, M, c_{1t}). \quad (1.20)$$

En la figura 1.8 mostramos cómo se actualizan las probabilidades de los modelos a medida que observamos datos nuevos. Para producir esta figura primero generamos la variable `regaloEn1`, que indica si el regalo se encuentra en la caja 1. Luego calculamos las predicciones de los modelos (o evidencia) usando el lenguaje de programación `Julia`.

```
regaloEn1 = [ rand(1:3) == 1 ? 1 : 0 for _ in 1:32]
evidencia_A = cumprod([(1/3)*(1/2) for _ in regaloEn1])
evidencia_B = cumprod([(1/2)*( (1/3)^(r1)*(2/3)^(1-r1) ) for r1 in regaloEn1])
```

Considerando priors iguales, la probabilidad de los modelos es,

```
prior = 1/2
p_datos = (evidencia_A .* prior) .+ (evidencia_B .* prior)
p_modelo_A = evidencia_A .* prior ./ p_datos
p_modelo_B = evidencia_B .* prior ./ p_datos
```

Debido a que los datos fueron generados usando la distribución de probabilidad de M_B , los datos también respetan la restricción entre hipótesis del modelo M_A . Esto significa que ninguno de los modelos observarán eventos imposibles, es decir, la productoria (1.20) no contiene ningún cero. En particular, las predicciones que hace el modelo A son siempre las mismas debido que no importa cual sea la pista s_{jt} y la posición del regalo r_{it} , siempre vale $P(s_{jt} | M_A, c_{1t}) = 1/3$ y $P(r_{it} | s_{jt}, M_A, c_{1t}) = 1/2$. El modelo B también predice la pista siempre con la misma probabilidad $P(s_{jt} | M_A, c_{1t}) = 1/2$, pues no habrá en los datos generados s_{1t} . Sin embargo, cuando el regalo se encuentra dentro de la caja elegida (1/3 de las veces), el modelo M_B predice $P(r_{1t} | s_{jt}, M_A, c_{1t}) = 1/3$, y cuando el regalo se encuentra en la caja no elegida uno abierta (2/3 de las veces), el modelo M_B predice $P(r_{it} | s_{jt}, M_A, c_{1t}) = 2/3$, con $i \neq 1$ y $i \neq j$.

Rápidamente, con unas decenas de observaciones podemos tener una preferencia fuerte por el modelo M_B , que es justamente el modelo que se utilizó para generar los datos.

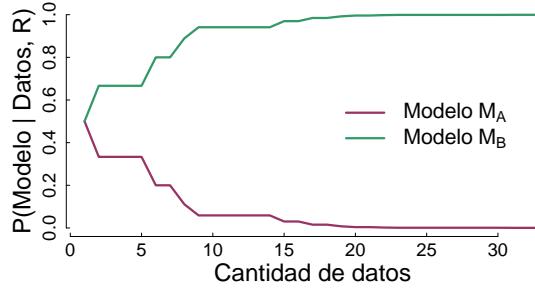


Figura 1.8: Distribución de probabilidad a posteriori sobre los sistemas de restricciones (o modelos) alternativos M_A y M_B dado los datos y la restricción restante R .

Este mismo método será usado para evaluar modelos a lo largo de la tesis.

1.3.3. Consecuencias de la ruptura de las reglas de la probabilidad

El obstáculo histórico de la aplicación estricta de las reglas de la probabilidad ha sido el costo computacional asociado a la evaluación completa de las hipótesis de investigación $\mathcal{H} = \{H_1, H_2, \dots, H_N\}$. En el caso más simple, con hipótesis de investigación binarias $|H_i| = 2$, la cantidad de combinaciones (o universos paralelos) a evaluar, $P(h_1, h_2, \dots, h_N)$, crece de forma exponencial como 2^N . No solo hay que asignar probabilidades iniciales a cada una de esas combinaciones de hipótesis elementales. Cuando observamos alguna de las hipótesis elementales, se debe incorporar la información que transmite a través del grado de sorpresa (o predicción a priori) asociada a cada una de la 2^{N-1} combinación de hipótesis elementales no observadas. En casos más complejos, que involucran además preguntas (o hipótesis) de investigación continuas ni siquiera es posible aplicar la regla de la suma cuando sus integrales no tienen solución analítica. El famoso modelo lineal es un caso paradigmático en la historia de la probabilidad, pues sus integrales tienen solución analítica pero que no pudo ser evaluado correctamente hasta el desarrollo de la computación moderna debido al costo computacional asociado a la inversión de matrices.

Revisemos el modelo lineal. Comenzaremos a introducir lentamente algunos conceptos de causalidad. Un modelo causal lineal de grado D (o modelo polinomial) supone que el mundo tiene $D + 1$ causas (W_0 a W_D con $D \geq 0$) que actúan sobre $N + 1$ consecuencias (Y_0 a Y_N con $N \geq 1$) y que entre ellas existe las siguientes dependencias causales, representadas mediante una flecha. Esta notación, adoptada en el área de Inferencia Bayesiana Causal [48], involucra la idea de que cambios en los valores de las causas producen cambios en los valores de las consecuencias, pero no a la inversa.

$$y_i \leftarrow \sum_{d=0}^D w_d x_d^d \quad \text{con } i \in \{0, \dots, N\}; y_i \in Y_i; w_d \in W_d .$$

Los valores $x_i \in \mathbb{R}$ no son hipótesis sino constantes que conocemos con exactitud al momento de recibir la observación de la hipótesis Y_i . Las hipótesis de investigación \mathcal{H} de un modelo causal lineal son de Y_0 a Y_N y de W_0 a W_D , donde cada uno corresponde con un subconjunto de los números reales. A través de la observación de las consecuencias Y_0 a Y_{N-1} tenemos que inferir el valor de las causas no observables W_0 a W_D para predecir el valor de la consecuencia todavía no observada Y_N para cualquier valor x_N .

Al modelo lineal, se le agrega un elemento más. Los valores observados y_i el modelo supone que se generan con un ruido aleatorio que tiene media 0 y desvío β .

$$p(y_i|x_i, w_0, \dots, w_D, \beta, M_D) = \mathcal{N}(y_i | \sum_{d=0}^D w_d x_d^d, \beta^2) \quad i \in \{0, \dots, N\}. \quad (1.21)$$

Esta es la distribución de probabilidad condicional de las hipótesis Y_i dado el modelo de grado D , M_D . Las distribuciones de probabilidad sobre hipótesis continuas se notan con p minúscula. La distribución de probabilidad a priori de las causas W_i es

$$p(w_d|\sigma, M_D) = \mathcal{N}(w_d|0, \sigma^2) \quad d \in \{0, \dots, D\}, \quad (1.22)$$

donde σ representa la incertidumbre a priori de las hipótesis W_0 a W_D . En la figura 1.9 se especifican las dependencias del modelo causal lineal, M_D , mediante la notación de redes bayesianas que representan a las hipótesis ocultas con círculos blancos, las hipótesis observadas con círculos grises, y las dependencias causales mediante flechas. En esta red bayesiana se incluye además los parámetros del modelo, que al ser constantes al momento de la inferencia se representan con puntos negros.

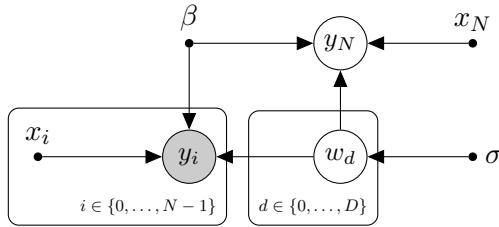


Figura 1.9: Modelo causal lineal. Continuamos con la notación de redes bayesianas presentada en la figura 1.3. Los puntos negros representan constantes.

Sea $\mathbf{w} = (w_0, \dots, w_D)$ un vector que contiene las hipótesis elementales donde cada $w_i \in W_i$ con $0 \leq i < D$, $\mathbf{y} = (y_0, \dots, y_{N-1})$ el vector que contiene las hipótesis elementales observadas con $y_i \in Y_i$ con $0 \leq i < N$, y $\mathbf{x} = (x_0, \dots, x_N)$ el vector que contiene los valores asociados a las hipótesis observadas. La distribución de probabilidad a posteriori de las causas, $p(\mathbf{w}|\mathbf{y}, \mathbf{x}, \sigma, M_D)$, y la predicción de la próxima consecuencia $p(y_N|x_N, \mathbf{w}, \beta, M_D)$, ambas son distribuciones gaussianas conocidas (capítulo 2 [49]). Sin embargo, ante la dificultad para resolver la inferencia del modelo lineal aplicando estrictamente las reglas de la probabilidad, durante el siglo 20 se adoptó el hábito de seleccionar una única hipótesis elemental para cada hipótesis de investigación no observable en base a algún criterio ad hoc, como por ejemplo *máxima verosimilitud*.

$$\max_{\mathbf{w}} p(\mathbf{y}|\mathbf{x}, \mathbf{w}, \beta, M_D).$$

A la acción de seleccionar hipótesis elementales se la conoce en la jerga de Inteligencia Artificial como “ajustar el modelo”, a los vectores \mathbf{y} y \mathbf{x} se los conoce como “datos de entrenamiento”, y a las hipótesis observables todavía no observadas, y_N , se las conoce como “datos de testeо”. Esto permitió durante el siglo 20 utilizar los modelos lineales para predecir nuevos datos.

$$\begin{aligned} p(y_N|x_N, \beta, M_D) &= \int \dots \int_{\mathbf{w} \in (W_0, \dots, W_D)} p(y_N|x_N, \mathbf{w}, \beta, M_D) p(\mathbf{w}|\mathbf{y}, \mathbf{x}, \sigma, M_D) d\mathbf{w} \\ &\approx p(y_N|x_N, \underbrace{\max_{\mathbf{w}} p(\mathbf{y}|\mathbf{x}, \mathbf{w}, \beta, M_D)}_{\text{Hipótesis elementales } \mathbf{w} \text{ que maximizan la función de costo ad hoc}}, \beta, M_D). \end{aligned} \quad (1.23)$$

En vez de predecir el valor de la hipótesis aún no observada y_N con la contribución de todas las hipótesis no observables \mathbf{w} mediante la aplicación estricta de la regla de la suma, estos enfoques aproximan la predicción de y_N mediante la selección del vector de hipótesis elementales \mathbf{w} que maximiza la función de costo ad hoc sobre los datos de entrenamiento. A diferencia de la predicción realizada mediante la regla de la suma, las predicciones realizadas con un único vector \mathbf{w} de hipótesis elementales son más inestables, lo que significa que tiene mayor tendencia a asignar a priori probabilidad cercana a cero a la hipótesis elemental y_N que finalmente se hace observable. Además, la ruptura de la aplicación estricta de las reglas de la probabilidad impide evaluar correctamente los modelos alternativos.

Por ejemplo, supongamos que tenemos un conjunto de datos generado a partir de una función sinusoidal con un ruido aleatorio que tiene desvío estándar β y media nula.

$$p(y_i|x_i, \beta) = \mathcal{N}(y_i | \sin(x_i), \beta^2) \quad i \in \{0, \dots, N-1\}.$$

En la figura 1.10(a) se grafica la sinusoidal mediante una línea punteada, y las observaciones y_0 a y_{N-1} mediante puntos negros. Supongamos además que tenemos 10 modelos lineales alternativos M_D , con $D \in \{0, \dots, 9\}$. Una pregunta básica que quisiéramos responder es, ¿cuál de todos los modelos lineales “ajusta” mejor los datos? Los modelos lineales de grado 0 a 2 no tienen la flexibilidad suficiente para “ajustar” (o acercarse) a la sinusoidal en el rango $x_i \in [-\pi, \pi]$, mientras que todos los modelos de grado mayor o igual a grado 3 sí tienen la flexibilidad suficiente para “ajustar” (o acercarse) a la sinusoidal en el rango $x_i \in [-\pi, \pi]$. Sin la posibilidad de aplicar estrictamente las reglas de la probabilidad, ¿qué significa que un modelo sea mejor que otro? El siguiente ejemplo está presente en el capítulo 3 del libro de Bishop [49].

En la figura 1.10(a) graficamos el ajuste de todos los modelos, destacando los modelos de grado 3 (línea roja) y de grado 9 (línea celeste). En la figura 1.10(b) podemos ver que a medida que aumentamos la complejidad de los modelos (aumentamos el grado) tienen efectivamente flexibilidad y, por lo tanto, el valor de máxima verosimilitud aumenta. El problema es que el modelo de grado 9, a pesar de ser el que más cerca pasa por los todos puntos observados, no es el que más cerca pasa por la función objetivo oculta (la sinusoidal), y por lo tanto no será el modelo que mejor desempeño tenga sobre datos nuevos. A este problema se lo conoce en la jerga de Inteligencia Artificial con el nombre de *overfitting* (o sobreajuste).

Hemos visto en la sección anterior que para evaluar modelos alternativos necesitamos la predicción a priori de los datos que hace cada uno de ellos, $P(\text{Datos}|\text{Modelo}, R)$, la que se descompone como una secuencia de predicciones individuales de cada condicionada en la información de todos los datos anteriores (revisar ecuación 1.20). Para aproximar el posterior de los modelos bajo el enfoque ad hoc, cada una de las predicciones individuales involucradas en la productoria (1.20) se computan aplicando la predicción ad hoc 1.23 mediante un procedimiento de “entrenamiento en línea”, en el que en cada paso se selecciona un nuevo vector \mathbf{w} en base al conjunto de datos de entrenamiento disponible hasta el momento. En la figura 1.10(c) se puede observar que esta aproximación del posterior de los modelos asigna probabilidad 1 al modelo de grado 3. Podría pensarse que este procedimiento de entrenamiento en línea resuelve el problema del sobreajuste en tanto penaliza los modelos que tienen mayor complejidad de la necesaria. Sin embargo, hemos “sobreajustado” nuevamente los datos debido a que hemos seleccionado un único modelo que no será capaz de predecir correctamente datos que se encuentren fuera del rango $x_i \in [-\pi, \pi]$.

¿Será que la aplicación estricta de las reglas de la probabilidad produce sobreajuste? Los modelos lineales que estamos implementando tienen solución analítica, por lo que hoy

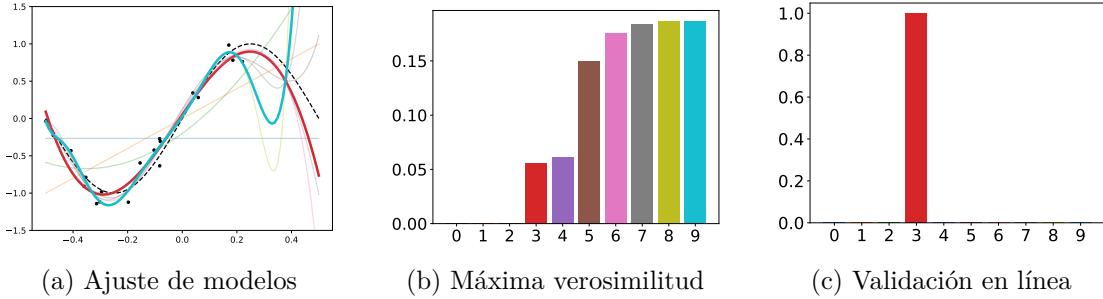


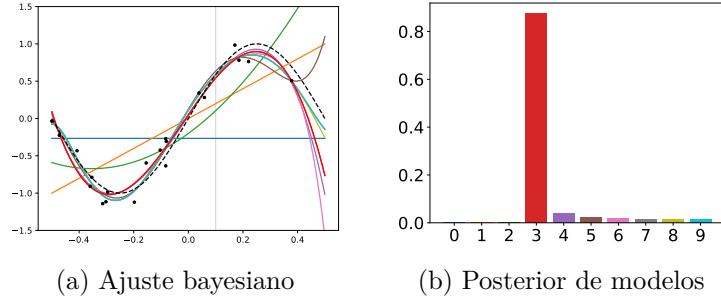
Figura 1.10: Modelos lineal de grado $D = 0$ a $D = 9$ ajustados por máxima verosimilitud a datos generados con una sinodal, $\mathcal{N}(y_i | \sin(x_i), \beta^2)$ en el rango $[-\pi, \pi]$. En (a) se grafica la sinodal con línea punteada, y los ajustes obtenidos por máxima verosimilitud de los modelos, destacando el modelo de grado 3 (línea roja) y el modelo grado 9 (línea celeste). En (b) se muestra el valor de máxima verosimilitud obtenido por cada uno de los modelos. En (c) se muestra la probabilidad de los modelo aproximando cada una de las predicciones individuales a priori mediante la ecuación ad hoc (1.23), usando como dato de entrenamiento en cada paso solo los datos observados previamente.

en día podemos calcular la predicción exacta que hacen los modelos $P(\mathbf{y}|\mathbf{x}, \mathbf{w}, \beta, \sigma, M_D)$, y calcular entonces el posterior exacto de los modelos $P(M_D|\mathbf{y}, \mathbf{x}, \beta, \sigma)$. En la figura 1.11(a) se grafica el máximo valor de la distribución posterior (o predicción) de la hipótesis aún no observada y_N para cada uno de los modelos (sin su incertidumbre asociada), y se grafica nuevamente la sinodal (línea punteada) y los datos (puntos negros). En la figura 1.11(b) se reporta la probabilidad a posteriori sobre el conjunto de modelos alternativos en base a los datos observados.

La aplicación estricta de las reglas de la probabilidad no produce ninguno de los sobreajustes que emergen cuando se aplican enfoques con criterios de selección de hipótesis ad hoc. Por un lado, el modelo que tiene la complejidad necesaria y suficiente (el de grado 3), es el que recibe mayor probabilidad a posteriori. Pero por otro lado, los modelos con complejidad mayor a la necesaria, si bien tienen menor probabilidad que el modelo de grado 3, no son rechazados, todos ellos reciben una probabilidad visiblemente mayor a 0. Los únicos modelos rechazados son los que tienen menor complejidad de la necesaria, con grado 0 a 2.

Veamos, concretamente, cómo son las predicciones que los modelos hacen del siguiente dato y_N cuando se nos informa un valor de $x_n = 0,1$ (línea vertical de la figura 1.11(a)). En la figura 1.12 se grafica la predicción a priori de $P(y_N|x_N = 0,1, M_D)$ que hace un modelo rígido $D = 1$, un modelo simple $D = 3$, y un modelo complejo $D = 9$. Allí se puede ver que el modelo rígido (grado $D = 1$) concentra toda la probabilidad en una región lejana al verdadero valor, por lo que siempre tienen un mal desempeño predictivo. El modelo simple (grado $D = 3$), que tiene la complejidad mínima necesaria, concentra la mayor parte de su probabilidad alrededor del verdadero valor. El modelo complejo (grado $D = 9$) también concentra la mayor parte de su probabilidad sobre el verdadero valor, pero con mayor dispersión que el modelo simple.

El balance natural de las reglas de la probabilidad está relacionado con el hecho de que las distribuciones de probabilidad deben sumar o integrar 1. Esto hace que los modelos más complejos, al ser más flexibles, distribuyan la predicción en un espacio más amplio que los modelos más simples. Y, dependiendo de dónde caiga el siguiente dato, hay una región en la que gana el modelo más simple, y hay otra región donde gana el modelo más complejo.



(a) Ajuste bayesiano

(b) Posterior de modelos

Figura 1.11: Modelos lineal de grado $D = 0$ a $D = 9$ ajustados mediante la aplicación estricta de las reglas de la probabilidad a datos generados con una sinusoidal, $\mathcal{N}(y_i | \sin(x_i), \beta^2)$ en el dominio $[-\pi, \pi]$. En (a) se grafica la sinusoidal con línea punteada, y los ajustes obtenidos por máxima verosimilitud de los modelos, destacando el modelo de grado 3 (línea roja) y el modelo grado 9 (línea celeste). En (b) se muestra la probabilidad de los modelos calculada mediante la aplicación estricta de las reglas de la probabilidad.

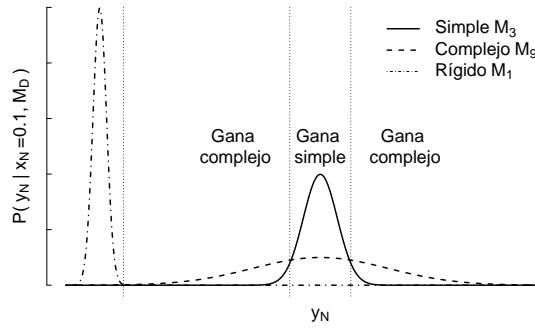


Figura 1.12: Representa la predicción que los diversos tipos de modelos polinomiales (clasificados por su complejidad) hacen de la variable y para $x = 0.2$.

A diferencia de los enfoques basados en criterios ad hoc, la aplicación estricta de las reglas de la probabilidad realiza las predicciones con la contribución de todas las hipótesis \mathbf{w} . Si empezamos a ver datos por fuera del dominio $[-\pi, \pi]$, los modelos más complejos están aún disponibles para predecirlos.

$$\underbrace{p(\mathbf{y}|\mathbf{x}, \beta)}_{p(\text{Datos}|R)} = \sum_{d=1}^D \left(\underbrace{\int \cdots \int_{\mathbf{w} \in (W_0, \dots, W_d)} \underbrace{p(\mathbf{y}|\mathbf{x}, \mathbf{w}, \beta, M_d)}_{p(\text{Datos}|\text{Hipótesis, Modelo, } R)} d\mathbf{w}}_{p(\text{Datos}|\text{Modelo, } R)} \right) P(M_d).$$

Con que una única hipótesis asigne a los datos probabilidad mayor a 0, la predicción del dato será mayor a 0, evitando los problemas de sobreajuste asociados a los enfoque que seleccionan una única hipótesis del espacio.

1.4. Acerca de la *habilidad*

Conocer cómo aprenden las personas a lo largo del tiempo es importante en muchos aspectos de la vida, especialmente en el sistema educativo, deportivo y laboral. Dado que la

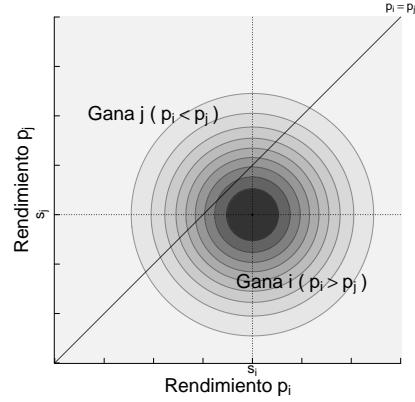
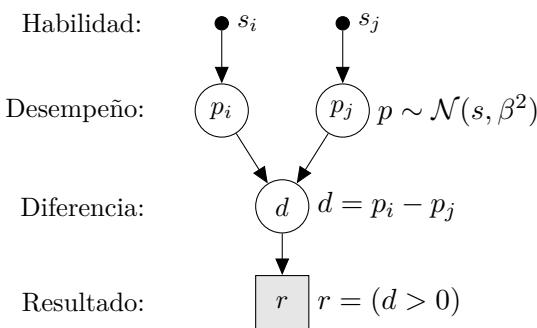
habilidad es una hipótesis oculta, debemos estimarla a partir de sus consecuencias observables directas: el éxito o fracaso en la resolución de problemas y competencias. Considerar sólo la frecuencia de éxitos como indicador de la habilidad de los individuos puede conducir a aproximaciones erróneas, fundamentalmente porque el valor de un éxito o un fracaso depende de la dificultad de los desafíos. Por esta razón, todos los modelos de habilidad ampliamente utilizados consideran que el resultado de la resolución de problemas se produce en relación a otra entidad oculta, que puede ser la habilidad de la persona contra la que compiten o la dificultad intrínseca del problema. Los primeros modelos causales, propuestos hace casi un siglo por Thurstone [50] y Zermelo [51], suponían que la probabilidad de un resultado observado r depende del rendimiento p (*performance*) del agente i y de su oponente j , $P(r|p_i, p_j)$. El campo siguió progresando con los trabajos de Bradley y Terry [52] y Mosteller [53–55], que permitió a Arpad Elo [56] desarrollar una metodología de estimación de habilidad para la Federación de Ajedrez de los Estados Unidos (USCF), adoptada hasta el día de hoy por la Federación Internacional de Ajedrez (FIDE).

Hasta las vísperas del siglo 21, la estimación de habilidad se resolvía utilizando funciones de costo ad hoc. Con el desarrollo de la computación moderna, estos modelos comienzan a ser resueltos mediante la aplicación estricta de las reglas de la probabilidad. Actualmente, los modelos bayesianos de habilidad Glicko [57] y TrueSkill [58] son los más utilizados por la industria del video juego y, al inicio de esta tesis, eran considerados como estándares para la estimación de habilidad en el tiempo. En esta sección entonces, revisaremos en detalle los modelos de habilidad mediante los cuales analizaremos los posibles factores de aprendizaje social al interior de las comunidades virtuales estudiadas en esta tesis.

1.4.1. Modelo de habilidad Elo

Arpad Elo [56] desarrolla en la década del 60' una metodología de estimación de habilidad para la Federación de Ajedrez de los Estados Unidos (USCF), adoptada hasta el día de hoy por la Federación Internacional de Ajedrez (FIDE). El modelo causal de una partida de ajedrez supone que las habilidades (*skills*) ocultas de dos agentes i y j , $s_i, s_j \in \mathbb{R}$ (donde \mathbb{R} representa el conjunto de los números reales), generan el resultado observable $r \in \{\text{Ganar, Perder}\}$ a través de la diferencia $d \in \mathbb{R}$ de los desempeños aleatorios $p_i, p_j \in \mathbb{R}$. Luego, las hipótesis de investigación son, $\mathcal{H} = \{H_{S_i}, H_{S_j}, H_{P_i}, H_{P_j}, H_D, H_R\}$, donde cada H_* representa el conjunto de hipótesis elementales.

En la figura 1.13(a) especificamos la versión “gaussiana” del modelo Elo (descrita en el trabajo de Herbrich et. al [58]) utilizando la notación de redes bayesianas. Debido a que durante el siglo 20 se seleccionaba una única hipótesis elemental de habilidad $H_{S_i} = \{s_i\}$, $H_{S_j} = \{s_j\}$, (en base a una función de costo ad hoc que revisaremos más adelante), especificamos las hipótesis de habilidad mediante puntos negros, que representan constantes. En cualquier caso, el modelo supone que las personas exhiben desempeños aleatorios, que varían alrededor de su verdadera habilidad con un desvío estándar β , $\mathcal{N}(p|s, \beta^2)$. Una vez que se generan los desempeños, el resultado queda determinado pues, según el modelo, gana el agente que obtuvo mayor desempeño, $r = (p_i > p_j)$. El parámetro β^2 , es el mismo para todos los agentes, y actúa como la escala de las estimaciones: habilidades a un β de distancia, $s_i - s_j = \beta$, implica 76 % de probabilidad de ganar, $P(p_i > p_j|s_i, s_j) \approx 0,76$, independientemente del valor absoluto de las habilidades s_i y s_j . Debido a que la probabilidad conjunta de los desempeños es una gaussiana bivariada (la densidad es la escala de grises en la figura 1.13(b)), la probabilidad del resultado es el volumen de un lado y del otro de la diagonal que representa $p_i = p_j$.



(a) Red bayesiana del modelo causal

(b) Probabilidad de desempeños y resultado

Figura 1.13: (a) Modelo generativo en el que las habilidades causan los resultados observables a través de la diferencia de los rendimientos ocultos, $d = p_i - p_j$, ambas variables aleatorias centradas en la verdadera habilidad, $p \sim \mathcal{N}(s, \beta^2)$. Quien haya obtenido mayor rendimiento gana, $r = (d > 0)$. Las variables observables se pintan de gris, las ocultas en blanco, y las constantes se muestran como puntos negros. (b) La probabilidad conjunta de los desempeños es una gaussiana bivariada. La escala de grises representa la densidad. Luego, la probabilidad del resultado se puede calcular como el volumen de un lado y del otro de la diagonal $p_i = p_j$.

La metodología desarrollada por Arpad Elo implementa un procedimiento ad hoc de selección de las hipótesis de habilidad basado en las predicciones del modelo (continuamos con la versión provista en el trabajo de Herbrich et. al [58]). Sea s_i^{old} y s_j^{old} las estimaciones de habilidad seleccionadas previamente, la predicción a priori del siguiente resultado es $P(H_R|s_i^{\text{old}}, s_j^{\text{old}})$. La idea es que la magnitud de la sorpresa está relacionada con cuán buenas son las estimaciones previas y, por lo tanto, podría usarse para actualizarlas. Sea el resultado $r \in H_R$, la **sorpresa** se define como,

$$\Delta = \underbrace{(1 - P(r|s_i^{\text{old}}, s_j^{\text{old}}))}_{\text{Sorpresa del resultado}}.$$

Si la predicción del resultado es $P(r|s_i^{\text{old}}, s_j^{\text{old}}) = 1$, luego la sorpresa es nula $\Delta = 0$, y viceversa. La metodología Elo se basa en la idea de que resultados inesperados indicarían que las estimaciones actuales no son del todo correctas. Por ello, la propuesta de Elo fue entonces utilizar la sorpresa Δ como factor de corrección de las estimaciones previas.

$$\begin{aligned} s_{\text{winner}}^{\text{new}} &= s_{\text{winner}}^{\text{old}} + \Delta \\ s_{\text{loser}}^{\text{new}} &= s_{\text{loser}}^{\text{old}} - \Delta. \end{aligned} \tag{1.24}$$

Cuanto más inesperados son los resultados, en mayor medida se actualizan las estimaciones previas. Si la “sorpresa” es nula, $P(r|s_i^{\text{old}}, s_j^{\text{old}}) = 1$, la actualización es nula. Si la “sorpresa” es total, $P(r|s_i^{\text{old}}, s_j^{\text{old}}) = 0$, se actualiza un punto de habilidad. Con varias rondas de actualización, esta solución es capaz de recuperar la escala relativa de los agentes, partiendo de valores iniciales arbitrarios. Sin embargo, el sistema Elo tiene algunas debilidades importantes.

Por un lado, la regla de actualización, expresada en la ecuación (1.24), es simétrica. Es decir, lo que gana un agente lo pierde el otro. Debido a que los agentes nuevos comienzan

con estimaciones arbitrarias (el mismo valor inicial para todos los individuos), los primeros cruces tienden a generar una alta sorpresa y, por lo tanto, pueden modificar bruscamente las estimaciones de sus oponentes a pesar de que ya hubieran convergido a valores estables. Esta debilidad ocurre por no tener en cuenta la incertidumbre sobre las estimaciones de los agentes. Para resolver este problema se propuso reducir el impacto de la sorpresa en función de la cantidad de veces que el agente ha participado previamente. Ese es el rol que desempeña el K-factor usado por la FIDE, $\Delta_i = \Delta \cdot K_i$, donde K_i es el factor de la persona i . A medida que la persona tiene más cantidad de partidas jugadas, el valor de K es menor. De esa forma se rompe la simetría y se preservan las estimaciones de las personas que se consideran conocidas. Esta definición de habilidad es usada todavía hoy por la Federación Internacional de Ajedrez (FIDE).

La metodología Elo tiene otros problemas asociados al uso de una función ad hoc de selección de hipótesis elementales de habilidad. El método TrueSkill [58–60], presentado en 2006, resuelve la inferencia de este mismo modelo aplicando estrictamente las reglas de la probabilidad y calculando la incertidumbre asociada a las hipótesis de habilidad de cada uno de los individuos. Esto resuelve naturalmente los problemas del modelo Elo. Además, se realizó una extensión del modelo causal, permitiendo estimar la habilidad de los jugadores individuales incluso cuando participan en equipo, algo que no estaba incluido en el modelo causal original.

1.4.2. Evidencia, verosimilitud y posterior exacto

Los problemas vinculados al modelo Elo se solucionan naturalmente aplicando estrictamente las reglas de la probabilidad. Para ello es necesario definir una distribución a priori sobre las habilidades. La solución bayesiana al modelo Elo (TrueSkill) define la distribución de creencias a priori sobre la habilidad de un individuo i , $s_i \in H_{S_i}$, mediante una distribución gaussiana

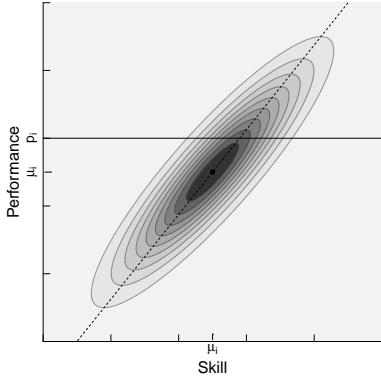
$$p(s_i) = \mathcal{N}(s_i | \mu_i, \sigma_i^2) \quad (1.25)$$

que representa la incertidumbre respecto del verdadero valor de habilidad. Para definir el valor a priori de la media, μ , hay que recordar que lo que importa no es su valor absoluto sino la diferencia con otros jugadores, pues un β de distancia entre habilidades, $s_i - s_j = \beta$ equivale a 76 % de probabilidad de ganar. Lo importante es que la incertidumbre a priori, σ , sea lo suficientemente grande para asegurarnos de cubrir todos los posibles valores de habilidad. Este valor depende del contexto. En general, se usa una incertidumbre que es tres veces el desvío estándar del desempeño, β .

En TrueSkill el desempeño de un individuo i , $p_i \in H_{P_i}$, se modela mediante la siguiente distribución de probabilidad condicional,

$$p(p_i | s_i) = \mathcal{N}(p_i | s_i, \beta^2). \quad (1.26)$$

En la figura 1.14 graficamos la distribución conjunta del desempeño y la habilidad de un individuo. Para calcular la probabilidad marginal del desempeño tenemos que integrar todos los posibles valores de habilidad, $p(p_i) = \int_{s_i} p(p_i | s_i) p(s_i) ds_i$. En términos gráficos, esto es equivalente a computar el área debajo de la línea sólida de la figura 1.14, donde la escala de grises representa la densidad de la gaussiana bivariada. Por las propiedades de la distribución gaussiana (ver anexo), esta integral se puede parametrizar usando otra gaussiana, centrada en la distribución de creencias a priori de la habilidad, μ , y con la suma de los desvíos estándar de la habilidad y del desempeño, $\mathcal{N}(p_i | \mu_i, \beta^2 + \sigma_i^2)$.



$$p(p_i) = \int \mathcal{N}(p_i | s_i, \beta^2) \mathcal{N}(s_i | \mu_i, \sigma_i^2) ds_i \\ = \mathcal{N}(p_i | \mu_i, \beta^2 + \sigma^2)$$

Figura 1.14: Distribución conjunta de habilidades y desempeños, calculada como el producto de la distribución a priori de las habilidades, $\mathcal{N}(s_i | \mu_i, \sigma_i^2)$, por la distribución condicional del desempeño, $\mathcal{N}(p_i | s_i, \beta^2)$. El área debajo de la línea sólida representa la probabilidad marginal de un cierto desempeño p_i . La línea punteada representa el valor máximo en una dimensión.

Calculada la distribución de probabilidad marginal de los desempeños, la predicción a priori del resultado de la partida entre dos individuos i y j se corresponde, nuevamente, con el volumen de un lado y del otro de la línea de nivel $p_i = p_j$ visto en la figura 1.13(b). En términos analíticos, la diferencia de desempeños, $d = p_i - p_j$, es una gaussiana centrada en la diferencia de las medias de las distribuciones de habilidad a priori ($\mu_i - \mu_j$), con una varianza que incorpora la incertidumbre de ambas estimaciones (σ) y la varianza de ambos rendimientos (β), $\mathcal{N}(d | \mu_i - \mu_j, 2\beta^2 + \sigma_i^2 + \sigma_j^2)$. Al observar que el agente i ganó, sabemos por el modelo causal que la diferencia de desempeños oculta fue en efecto positiva. Por lo tanto, la predicción a priori del resultado observado, o evidencia, es la densidad acumulada (Φ) de todos los valores positivos de la gaussiana de diferencia de desempeños:

$$\underbrace{P(r)}_{\text{Evidencia}} = 1 - \Phi(0 | \mu_i - \mu_j, 2\beta^2 + \sigma_i^2 + \sigma_j^2). \quad (1.27)$$

La evidencia es la verosimilitud marginal $p(r)$, la predicción a priori del resultado observado realizada con la contribución de todas las hipótesis elementales de habilidad.

Para calcular el posterior sobre la habilidad $p(s_i | r)$ necesitamos la verosimilitud condicional, $p(r | s_i)$. En el capítulo 3 mostramos todos los detalles que se requieren para llegar a la siguiente expresión analítica exacta.

$$\underbrace{p(s_i | r)}_{\text{Posterior}} \propto \underbrace{1 - \Phi(0 | s_i - \mu_j, 2\beta^2 + \sigma_j^2)}_{\text{Verosimilitud } P(r | s_i)} \underbrace{\mathcal{N}(s_i | \mu_i, \sigma_i^2)}_{\text{Prior } p(s_i)}. \quad (1.28)$$

Donde el posterior normalizado se obtiene dividiendo el lado derecho con la evidencia, $P(r)$. Es interesante notar las similitudes y diferencias entre la verosimilitud y la evidencia. La verosimilitud cuantifica la misma densidad acumulada que la evidencia, pero centrada ahora en la diferencia entre la hipótesis que estamos evaluando s_i y la estimación media del oponente μ_j , con una varianza que incluye todas las incertidumbres salvo la de la propia hipótesis s_i .

El posterior no es más que la densidad del prior no filtrada por la verosimilitud. La sorpresa, definida como el complemento de la verosimilitud, funciona como un filtro para el prior. En la región de hipótesis de muy alta habilidad, donde el resultado ganador no nos

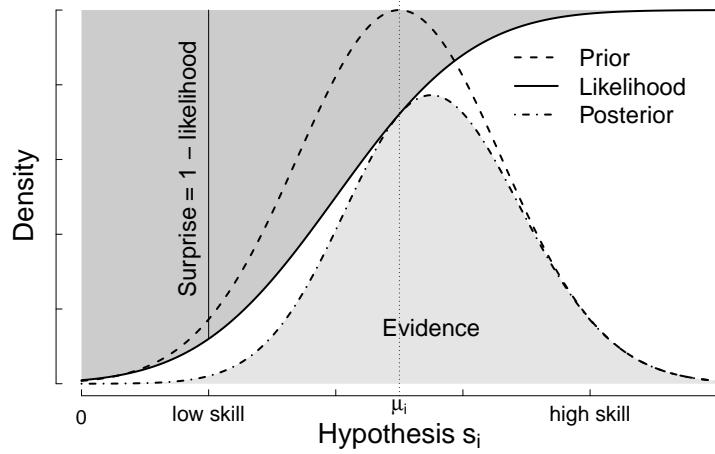


Figura 1.15: Actualización de creencias para el caso ganador. El posterior proporcional se obtiene como el producto de la distribución a priori (distribución gaussiana) y la verosimilitud (distribución gaussiana acumulada). La evidencia es la integral del posterior proporcional. Las distribuciones no están necesariamente en la misma escala: la distribución a priori integra 1, mientras que la verosimilitud va de 0 a 1.

hubiera generado casi ninguna sorpresa ($\lim_{s_i \rightarrow \infty} P(r|s_i) = 1$), el posterior recibe casi toda la densidad del prior. En cambio, en la región de hipótesis de muy baja habilidad, donde el resultado habría generado mucha sorpresa ($\lim_{s_i \rightarrow -\infty} P(r|s_i) = 0$), el posterior no recibe casi nada de la densidad del prior.

Esta es la estimación de habilidad correcta dado el modelo causal Elo. Es importante remarcar que la posterior, aunque se parezca, no es una distribución gaussiana, lo que nos impedirá usar la ecuación (1.28) iterativamente.

1.4.3. Posterior aproximado y extensión del modelo

Por la forma del posterior exacto, una gaussiana puede ser usada como una buena aproximación, permitiéndonos evitar el costo computacional de las metodologías de muestreo. Uno de los principales aportes del sistema Glicko [61] fue el desarrollo de un método eficiente para aproximar la posterior exacta con una distribución gaussiana. Sin embargo, este método no garantiza que la distribución gaussiana seleccionada sea la que mejor aproxima. El éxito de la solución TrueSkill [58] se basa en la aplicación de un método eficiente para calcular la gaussiana que mejor aproxima al posterior exacto (sección 3.2.6),

$$\hat{p}(s_i|r, s_j) = \arg \min_{\mu, \sigma} \text{KL}(p(s_i|r, s_j) || \mathcal{N}(s_i|\mu, \sigma^2)), \quad (1.29)$$

que se plantea en términos de minimización de la divergencia Kullback-Leibler entre la distribución verdadera y la aproximada. Este método nos permite aplicar eficientemente la ecuación (1.28) iterativamente sobre una secuencia de observaciones, que de otra manera sería inviable.

El enfoque adoptado por TrueSkill para tratar el proceso dinámico, conocido como *filtering*, usa el último posterior aproximado como prior del siguiente evento. Luego, el posterior

aproximado en un determinado momento se define como

$$\widehat{\text{Posterior}}_t \propto \widehat{\text{Likelihood}}_t \underbrace{\widehat{\text{Likelihood}}_{t-1} \dots \widehat{\text{Likelihood}}_1}_{\text{Posterior}_1 \text{ como Prior}_2} \underbrace{\widehat{\text{Posterior}}_{t-1} \text{ como Prior}_t}_{\widehat{\text{Posterior}}_t}, \quad (1.30)$$

donde $\widehat{\text{Posterior}}_i$ y $\widehat{\text{Likelihood}}_i$ representan las aproximaciones inducidas por la ecuación (1.29) en el i -ésimo evento. Si consideramos la verosimilitud como un filtro del prior, cada posterior puede ser visto como una acumulación de todos los filtros anteriores. De esta forma, la información se propaga de las estimaciones pasadas hacia las futuras. Debido a que las habilidades cambian en el tiempo, es importante agregar alguna incertidumbre γ luego de cada paso.

$$\widehat{p}(s_{i_t}) = \mathcal{N}(s_{i_t} | \mu_{i_{t-1}}, \sigma_{i_{t-1}}^2 + \gamma^2). \quad (1.31)$$

Modelo con equipos

Otra de las novedades de TrueSkill fue la extensión del modelo Elo para permitir estimación de habilidad cuando los individuos compiten en equipos. Para ello se agrega una hipótesis de investigación adicional, $H_{T_e} \subseteq \mathbb{R}$, que representa el conjunto posibles desempeños de los equipos. Sea A_e el conjunto de jugadores que forman parte del equipo e . El modelo supone que el desempeño de un equipo es la suma de los desempeños de sus miembros.

$$t_e = \sum_{j \in A_e} p_j. \quad (1.32)$$

El resto del modelo se preserva igual. Un equipo vence a otro cuando su desempeño es mayor al del equipo contrario. El resultado de una partida con varios equipos se modela con una desigualdad de los desempeños de los equipos, tal que $t_{o_1} < \dots < t_{o_k}$, donde o_i representa el equipo que quedó en la i -ésima posición, y k representa la cantidad total de equipos. En resumen, el modelo TrueSkill puede representarse mediante las siguientes red bayesiana (figura 1.16).

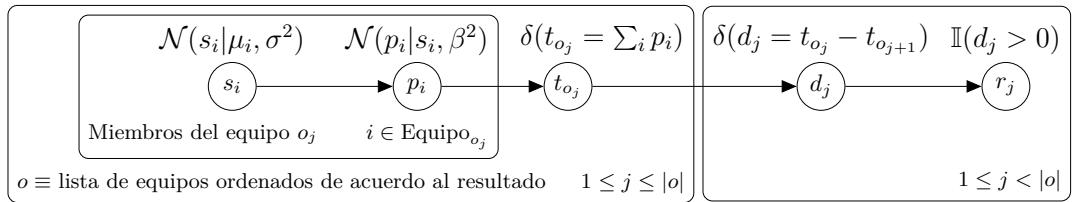


Figura 1.16: Modelo general con equipos. Los subíndices que aparecen abajo a la derecha en las placas, indican replicación. El subíndice j de la placa izquierda despliega los desempeños de los k equipos, y el subíndice i de la placa interna despliega sus jugadores. El subíndice j de la placa derecha despliega los $k - 1$ comparaciones entre equipos consecutivos. Arriba de los nodos que representan las hipótesis de investigación se especifica la distribución de probabilidad. Las hipótesis que no tienen incertidumbre dado su condicional se representan mediante la delta de dirac si la variable es continua, $\delta(\cdot = \cdot)$, o mediante la indicadora si la variable es discreta, $\mathbb{I}(\cdot = \cdot)$.

La solución multiequipos no tiene una forma cerrada, requiere una iteración para alcanzar convergencia que será explicada en detalle en el capítulo 3.

Capítulo 2

Resultado 1. Efecto de la formación de equipos sobre el aprendizaje

El problema de adquisición de habilidades es ubicuo y fundamental para la vida. La mayoría de las tareas humanas se realizan en sociedad, en cooperación con otros sujetos. A pesar de su importancia fundamental, la selección de compañeros de equipo suele pasarse por alto cuando se estudia el aprendizaje. En este trabajo aprovechamos el repositorio de comportamiento humano disponible en Internet para estudiar un tema relevante en la ciencia antropológica: cómo las estrategias de formación de equipo pueden afectar el aprendizaje individual. Analizamos el impacto de las estrategias de formación de equipos en la adquisición de habilidades mediante un juego por turnos donde los jugadores pueden participar individualmente o en equipos. Revelamos un efecto sutil pero fuerte en la adquisición de habilidades basado en la forma en que se forman y mantienen los equipos a lo largo del tiempo. La tendencia a jugar en equipos se asocia con una mejora de las habilidades a largo plazo, mientras que jugar lealmente con el mismo compañero de equipo acelera significativamente la adquisición de habilidades a corto plazo.

2.1. Introducción

Ciertamente las habilidades se adquieren principalmente a través de la experiencia. Sin embargo, por sus capacidades sociales el ser humano incorpora muchos de los conocimientos aprendiendo de los demás (*aprendizaje social*). Este aprendizaje social afecta el proceso de adquisición de habilidades esperado por la experiencia individual [4]. En este trabajo, exploramos el efecto que las estrategias de formación de equipo tienen sobre los procesos de adquisición de habilidades.

La experiencia individual ha sido durante mucho tiempo estudiada como el factor principal en la mejora del desempeño. Las investigaciones sobre toma de decisiones expertas se iniciaron con el estudio del comportamiento de los ajedrecistas profesionales [62–64]. Newell señaló en 1981 que la ley de potencia, en función del grado de práctica adquirida, describía todos los datos sobre aprendizaje analizados hasta el momento [65]. En los últimos años, algunos autores señalaron que la ley de potencia explica solamente las curvas de aprendizaje poblacionales y afirman la necesidad de proponer otras funciones para aproximar las curvas de aprendizaje individuales [66]. Sin embargo, las curvas de aprendizaje individuales son, naturalmente, más irregulares que las curvas de aprendizaje promediadas y la predicción del desempeño a gran

escala de tiempo basado en eventos de pequeña escala de tiempo demostró ser difícil [67]. La práctica es un factor de aprendizaje importante pero no es el único. Por ello se deben tener en cuenta otros componentes esenciales para comprender mejor los procesos de aprendizaje.

El aprendizaje social se define como cambios a largo plazo en el comportamiento causados por estímulos derivados de la observación o la interacción con otros individuos [68, 69]. Nuestra especie estuvo involucrada en un proceso de coevolución genético-cultural único, que provocó el surgimiento de nuestras singulares capacidades de aprendizaje social: una maquinaria cognitiva costosa que permite la adquisición eficiente de tradiciones complejas [70]. Los seres humanos aprendemos conocimientos de los demás, los modificamos y los transmitimos a la siguiente generación, produciendo una acumulación cultural que no puede ser desarrollada por ningún individuo aislado en un período de vida [71]. La capacidad de adquirir comportamientos basados en la experiencia de otros, permite evitar reconstruir los conocimientos por ensayo y error, lo que permite a las poblaciones humanas adaptarse rápidamente a los cambios y nuevos entornos [4]. Sin embargo se requiere un grado de credulidad para que este proceso funcione y, por lo tanto, existe el riesgo de adquirir información inapropiada incluso en entornos uniformes y estables [72, 73]. Descifrar cómo aprovechar la información social mientras se manejan los inconvenientes que se derivan de su uso se ha convertido en uno de los temas principales en la investigación sobre estrategias de aprendizaje social [74, 75].

Se han propuesto muchos modelos sobre estrategias de aprendizaje social y su dinámica poblacional emergente. Los investigadores han identificado varias estrategias teóricas [75, 76], que pueden clasificarse entre: aquellas que especifican las circunstancias bajo las cuales los individuos copian a otros; y aquellas que identifican de quién aprenden los individuos. Normalmente, los estudios de aprendizaje social se realizan utilizando métodos como observaciones de campo [77], experimentos controlados de laboratorio [78–81] y experimentos de campo [82–85]. Todos estos métodos tienen una dificultad inherente para obtener muestras de datos grandes y longitudinales.

Con el advenimiento de las comunidades virtuales comienza a ser posible estudiar datos masivos sobre aprendizaje social sobre largos períodos de tiempo. Esto permite buscar efectos potencialmente sutiles, detectables solo con un número grande de observaciones, que pueden pasar desapercibidos en muestras pequeñas, típicas de los estudios de laboratorio [33]. Nuestro estudio también incorpora la capacidad actual para estimar las habilidades con alta precisión. Estos resultados se obtienen en condiciones experimentales únicas en la que los jugadores se involucran en relaciones naturales, libres de elegir a sus compañeros de equipo y oponentes, produciendo resultados confiables que se pueden medir directamente sin depender de métodos indirectos como los cuestionarios.

Los juegos en línea ya se han utilizado como modelo para estudiar procesos cooperativos complejos en las ciencias sociales [86, 87], neurociencias [33, 88], y ciencias sociales computacionales [89, 90]. El ajedrez ha sido, por su complejidad y reglas claras, un modelo privilegiado para el estudio del aprendizaje y la toma de decisiones. Los datos de ajedrez masivos permitieron el análisis de la influencia de la edad, las cohortes, el género y otras características en el aprendizaje [91–93].

Aquí nos propusimos investigar el impacto de las estrategias de formación de equipo en la adquisición de habilidades en *Conquer Club*, un juego multijugador en línea por turnos inspirado en el juego de mesa RISK o TEG. A diferencia de la naturaleza individual del ajedrez, en *Conquer Club* se puede jugar individualmente o en equipos. Además, en Conquer Club hay un fuerte incentivo para la colaboración pues los resultados de los juegos son por equipos: todos los miembros de un equipo ganan o pierden juntos. Un jugador que es eliminado

durante un juego aún puede terminar ganando si sus compañeros de equipo derrotan al resto de los equipos. Por lo tanto, es fundamental que los compañeros de equipo puedan coordinar sus acciones. A diferencia de otras plataformas, no hay contenido pago, ofreciendo las mismas condiciones para todos los jugadores. No existe un mecanismo que defina la composición de los equipos en función de la probabilidad de ganar de los jugadores.

Los investigadores que estudian aprendizaje en ajedrez se basan en Elo para estimar las habilidades, mientras que nosotros adoptamos una extensión bayesiana llamada TrueSkill [58]. En primer lugar, TrueSkill utiliza una distribución de creencias a priori, en lugar de un escalar, para representar las estimaciones de habilidades. En segundo lugar, TrueSkill modela el desempeño de los equipos, lo que le permite estimar la habilidad de las personas individuales a pesar de que el resultado sea el mismo para todos sus miembros. Finalmente, TrueSkill utiliza una función de actualización no arbitraria, el posterior del modelo bayesiano, que se calcula marginalizando la distribución conjunta [94]. Como resultado, TrueSkill puede identificar con precisión la habilidad de los jugadores con la menor cantidad de partidas posibles.

Con nuestro enorme conjunto de datos, podemos investigar el impacto de las estrategias de juego en equipo en la adquisición de habilidades individuales que, de otro modo, no sería posible estudiar.

2.2. Materiales y métodos

Todos los juegos se descargaron de *Conquer Club*, un servicio gratuito que ofrece juegos en línea tipo RISK o TEG. El sitio web permite a cualquier persona, y no solo a los participantes registrados, explorar los partidos y buscar sus datos relacionados. Los usuarios registrados se identifican por sus apodos y, para registrarse en la plataforma deben aceptar que sus partidas se almacenen en un servidor de acceso público. Además, durante el proceso de descarga, cada jugador fue identificado por un número de identificación interno, anonimizando los datos. En consecuencia, no fue necesario el consentimiento individual debido a esta doble capa de anonimato y la naturaleza abierta del sitio web. Adicionalmente, nos comunicamos con los propietarios de *Conquer Club* para obtener la autorización para realizar esta investigación, cumpliendo así con los términos de uso de este sitio.

La aplicación consta de un *script Python* que se conecta al servidor de *Conquer Club*. Los datos entre 2006/01/03 y 2009/07/12 se almacenan en una base de datos PostgreSQL. Hay cerca de 4,4 millones de partidas jugadas por casi 270 mil usuarios diferentes. Debido a una doble capa de anonimato y la naturaleza abierta del sitio web pudimos publicar la base de datos en Dryad [95], una plataforma para la reutilización de datos de investigación.

Para calcular la habilidad, usamos el paquete *TrueSkill 0.4.4* de *Python*. Todos los jugadores comienzan con una habilidad media $\mu = 25$ y una desviación estándar de habilidad de $\sigma = 8,33$. El valor de probabilidad de empate se estableció en 0 ya que no hay posibilidad de empate en *Conquer Club*.

Gameplay Los juegos de mesa tipo RISK o TEG contienen mapas con regiones. Al comienzo de cada partida, las regiones del mapa se distribuyen aleatoriamente entre los jugadores a las que le asignan una cierta cantidad de tropas. Cada turno consiste en i) desplegar nuevas tropas, ii) conquistar las regiones vecinas del oponente y iii) reforzar las regiones. El entorno del juego tiene cuatro elementos relevantes: el tablero actual, un panel con el estado del juego, un chat público y un registro de movimientos.

La figura 2.1 ejemplifica la interfaz de una partida de dos equipos de dos jugadores. Los nodos del gráfico representan las regiones, el color indica a quién pertenece, los números representan el número de tropas y las formas indican la agrupación de regiones por zonas. Al lado hay un panel con el número de ronda, el jugador activo y el tiempo restante para jugar la ronda actual, y un resumen de las tropas totales y las regiones controladas totales para cada jugador. El color junto al apodo identifica el número total de regiones que tiene cada jugador en el mapa. Un servicio de chat público está disponible dentro de cada juego. El *log* registra todos los movimientos de los jugadores.

Al comienzo de cada turno, los jugadores reciben nuevas tropas. La cantidad de tropas resulta del número de *regions* ocupadas y la bonificación de la *zona* controlada y eventualmente por el intercambio de *spoils*. Podemos ver en la primera línea del *log* en la Fig. 2.1(d) que el jugador *c* recibió tropas por mantener el Continente 4. Estas tropas pueden desplegarse en sus regiones ocupadas. Por ejemplo, el jugador *c* despliega todas sus tropas en la región *M*.

Una vez finalizado el despliegue comienza la fase de asalto. Un jugador puede asaltar la región de cualquier oponente siempre que ambas estén adyacentes y la región asaltante tenga un mínimo de dos tropas. El motor del juego tira un dado por cada tropa que asalta, excepto por una tropa que necesita permanecer en la región, hasta un máximo de tres tropas. Luego, el sistema lanza un dado por cada tropa defensora, hasta un máximo de dos tropas. Los valores obtenidos de cada dado se ordenan de forma creciente y luego se comparan uno por uno. Si el dado de asalto es más alto, la región defensora pierde una tropa. Si el dado defensor es mayor o igual, la región atacante pierde una tropa. Si el atacante destruye todas las tropas defensoras, algunas de las tropas restantes deben moverse para ocupar la región recién conquistada. En nuestro ejemplo, el jugador *c* asalta la región *N* desde la región *M* y la conquista del jugador *a*. Luego, este jugador usa las tropas movidas recientemente para conquistar otra región.

Cuando el jugador termina los asaltos, se pueden usar algunas tropas para reforzar la posición de defensa. El jugador puede mover algunas (pero no todas) las tropas de una de sus regiones a cualquier otra de sus regiones que estén conectadas. La configuración de las reglas del juego de refuerzos determina cuántas de estas jugadas de refuerzo están permitidas. En nuestro ejemplo, el jugador *c* refuerza la región *P* moviendo dos tropas de la región *M*. Finalmente, el jugador *c* termina el turno y el jugador *b* comienza con su ronda.

Matchmaking La plataforma tiene una sección “Unirse a un juego”, donde todos los jugadores pueden ver todos los juegos abiertos. Cuando un usuario crea una nueva partida debe elegir: a) opciones de juego, b) tipo de juego (todos contra todos o juego en equipo), c) el número de participantes, y d) el método de participación, público, público con espacios reservados, y privado. Los juegos públicos son aquellos a los que cualquiera puede unirse. Los juegos públicos con espacios reservados tienen espacios asignados a jugadores particulares, y el resto está abierto a jugadores generales. En los juegos privados solo las personas que tengan la contraseña pueden jugar.

En esta plataforma, no existe un mecanismo de formación de equipos automático basado en la probabilidad de ganar entre jugadores. Hay una clasificación interna y un sistema de puntos que los jugadores pueden usar como referencia para estimar la habilidad de los demás. El sistema de puntos se actualiza como $\Delta = \min(\frac{\text{Loser's score}}{\text{Winner's score}} \cdot 20, 100)$. Sin embargo, no son indicadores precisos de la habilidad de los jugadores ni de la probabilidad de ganar entre oponentes. El ranking interno es la conjunción entre el número de partidos jugados y los

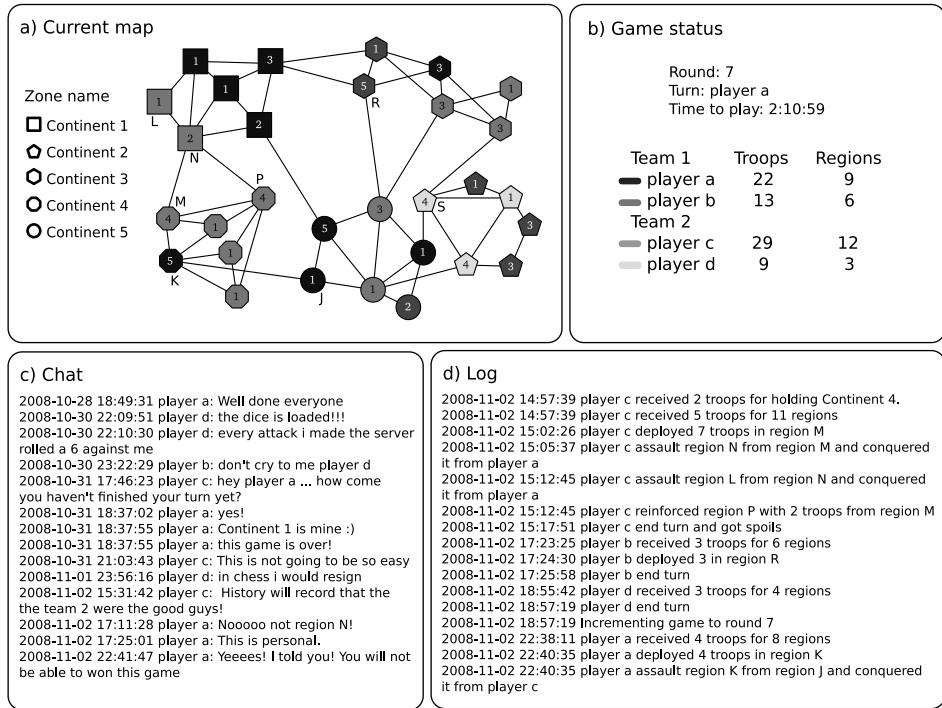


Figura 2.1: Esquema del juego Conquer Club. a) El mapa representado como un grafo con continentes (nodos con misma forma), jugadores (identificados con la escala de grises) y un número de tropas en cada región. b) Estado general del juego: ronda actual, jugador activo y tiempo restante para jugar; y un resumen del total de tropas y regiones controladas para cada jugador. c) Ejemplo de sesión de chat durante un juego. d) Registro del juego utilizado para extraer información a través del web scraper.

puntos alcanzados.

Cuando un jugador selecciona un juego, puede ver los nombres de los que ya se han unido. Aparece un ícono y una estrella junto a los nombres. Los íconos representan el *ranking* de habilidad interno. Al final de un juego, los jugadores pueden reportar, en una escala de 1 a 5, el comportamiento del resto de jugadores en cuanto a *Fair Play*, *Gameplay* y *Actitud*. Las estrellas resumen la opinión sobre la persona indicada por algunos de sus anteriores contrincantes.

2.3. La ley de la práctica

Primero, estudiamos cómo las personas mejoran su desempeño a medida que ganan experiencia, es decir, la ley de la práctica. Estimamos la experiencia de cada jugador por el número de partidas jugadas. La diferencia de habilidad entre los oponentes (medida con *TrueSkill* [58]) indica con alta precisión la probabilidad de ganar. La probabilidad de ganar cuando los oponentes tienen la misma habilidad es $1/2$, y una diferencia de 4 tsp (puntos *TrueSkill*) aumenta la probabilidad de ganar a $2/3$.

En nuestro contexto, la curva de aprendizaje es la progresión de habilidades a medida que se adquiere experiencia (es decir, la cantidad de partidas jugadas). Como se mencionó, las

curvas de aprendizaje de la población siguen una función de ley de potencia [65]

$$\text{habilidad} = \text{habilidad}_0 \cdot \text{Experiencia}^\alpha, \quad (2.1)$$

donde α es la tasa de aprendizaje característica de la población y habilidad_0 la habilidad de la población después del primer juego.

Para analizar la ley de la práctica, dividimos a los jugadores según su actividad total: (1) jugadores con al menos ocho partidas y menos de 16, (2) jugadores al menos 16 partidas y menos de 32, etc. Por lo tanto, ajustamos los parámetros de la ley de la práctica a cada uno de los conjuntos de jugadores según su nivel de actividad. De acuerdo con la ley de la práctica, observamos una dependencia lineal en las curvas de aprendizaje en escala logarítmica en todos los segmentos poblacionales (Figura 2.2).

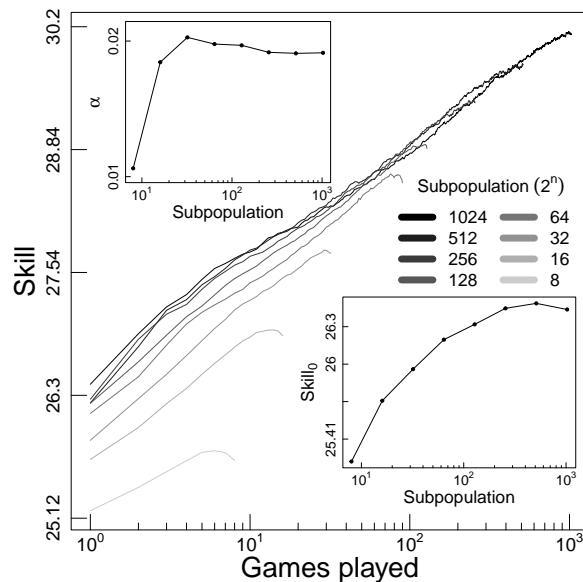


Figura 2.2: Ley de la práctica. Curva de aprendizaje de las subpoblaciones de jugadores según su nivel de actividad total. Cada curva de aprendizaje muestra la habilidad de las primeras 2^n partidas jugadas de la subpoblación con al menos 2^n partidas jugadas y menos de 2^{n+1} . Las subfiguras muestran los valores de los parámetros (es decir, α y Skill_0) de las curva de aprendizaje de cada subpoblación (eje x) siguiendo la ecuación (2.1).

Las curvas de aprendizaje dependen del abandono, exhibiéndose una habilidad más baja para las subpoblaciones con una actividad total más baja. Sin embargo, la tasa de aprendizaje (α) se mantiene estable para las subpoblaciones con al menos 32 partidas jugadas (panel superior izquierdo de la figura 2.2). La diferencia entre ellos causa poca variación en la adquisición de habilidades a largo plazo, con menos de 0,24 tsp después de 1000 partidas jugadas. La habilidad inicial de la subpoblación (Skill_0) también se ve afectada por el abandono (subfigura abajo a la derecha en la figura 2.2). Sin embargo, todas las subpoblaciones con al menos 64 juegos jugados, no tienen una habilidad inicial significativamente diferente (Wilcoxon rank-sum test en Anexo).

Por lo tanto, todas las cohortes con al menos 64 partidas jugadas tienen curvas de aprendizaje casi equivalentes. Son lo que llamamos la “curva de aprendizaje esperada por la experiencia”. Este aprendizaje básico puede verse alterado por muchos factores. Por ejemplo, el

compromiso de terminar los juegos es sin duda un factor relevante en el proceso de adquisición de habilidades. De hecho, los jugadores que siempre terminan sus juegos tienen una curva de aprendizaje más alta que el resto de la población, alrededor de 0,5 tsp (Fig. 7.7(a) Anexo).

2.4. Aprendizaje social

El aprendizaje social es esencial en los animales prosociales. Nuestra hipótesis es que la curva de aprendizaje esperada por la experiencia individual podría verse alterada por diferentes estrategias de formación de equipo. Para estudiarlo, analizamos el comportamiento de los jugadores en lo que respecta a la selección de equipos.

En la plataforma de juego, los usuarios pueden elegir entre jugar individualmente o en equipos. Definimos el concepto *Team-oriented behavior* (TOB), como el número de partidas en las que participa en equipo, dividido por el número total de partidas jugadas:

$$\text{Team-oriented behavior} = \frac{\text{Partidas de equipo jugadas}}{\text{Partidas jugadas}}. \quad (2.2)$$

Para evaluar la influencia de TOB en las curvas de aprendizaje, dividimos la población en TOB fuerte, medio y débil (es decir, $0,8 < TOB \leq 1$, $0,4 < TOB \leq 0,6$ y $0 < TOB \leq 0,2$, respectivamente). En adelante, excluimos a los jugadores con menos de cuatro partidas en equipo.

En el largo plazo, entre 200 y 500 juegos de experiencia, las curvas de aprendizaje se ordenan según su nivel de TOB, exhibiendo un mayor nivel de habilidad para las poblaciones con mayor TOB (Figura 2.3). Los jugadores con TOB altos muestran, después de 250 partidas, una habilidad final significativamente más alta en comparación con TOB medio y débil (Wilcoxon rank-sum test, $p < 1 \times 10^{-4}$).

En este intervalo, las poblaciones de TOB fuerte y medio están distanciadas por aproximadamente 1 tps. Un comportamiento TOB alto adquiere, a largo plazo, un mayor valor de habilidad incluso en comparación con jugadores sin juegos de equipo.

2.4.1. Faithfulness-boost effect

Los jugadores pueden elegir entre jugar con el mismo compañero de equipo o seleccionar diferentes jugadores en cada juego. Nuestra hipótesis es que un comportamiento leal puede afectar el aprendizaje (aumentar o disminuir la tasa de adquisición de habilidades) cuando se juega en equipo. La mayoría de las veces que dos personas aparecen recurrentemente en la misma partida, vemos que la inmensa mayoría son compañeros de equipo en lugar de oponentes. Por lo tanto, enfocamos nuestro análisis solo en la lealtad de los compañeros de equipo ya que la lealtad en los oponentes no está presente en nuestra base de datos. Definimos el concepto *loyalty* de las personas como la proporción de veces que juegan con el compañero de equipo más recurrente dividida por la cantidad de partidas de equipo jugadas:

$$\text{Loyalty} = \frac{\text{Máximo de partidas jugadas con un mismo compañero}}{\text{Partidas de equipo jugadas}}. \quad (2.3)$$

Para evaluar la influencia del *loyalty* sobre el aprendizaje, examinamos la evolución de las habilidades de los jugadores con TOB fuerte en función de su valor de *loyalty*. Definimos un jugador como *leal* cuando $loyalty > 0,5$, y un jugador como *casual* cuando $loyalty \leq 0,2$.

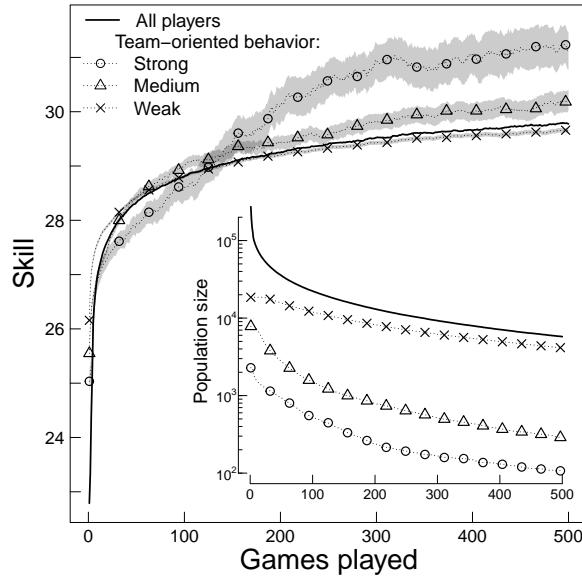


Figura 2.3: Aprendizaje social. La curva de aprendizaje para TOB fuerte, medio y débil. La banda representa el intervalo de confianza de 95 % del Wilcoxon rank-sum y la línea central representa la pseudomediana. Como referencia, mostramos la curva de aprendizaje de toda la población. Los resultados son análogos a los obtenidos con la media y el intervalo de confianza de 95 % del t-test.

Si comparamos las curvas de aprendizaje de jugadores leales y casuales, obtenemos una separación sustancial entre ellos en los primeros juegos de experiencia. Los jugadores leales muestran un incremento en la habilidad media de aproximadamente 4 tsp sobre los jugadores casuales (Fig. 2.4). La distribución de habilidades en cada punto de la curva de aprendizaje es significativamente diferente hasta 386 partidas jugadas (Wilcoxon rank-sum test $p < 0,01$). Un comportamiento análogo entre las subclases leales y casuales ocurre también para los TOB medios y débiles (Fig. 7.8 en Anexo), aunque menos intensos ya que están menos orientados a equipos.

Para estudiar la interacción entre TOB y *loyalty* en la adquisición de habilidades, fijamos el número de partidas de experiencia en 100. Al aislar el efecto de la experiencia, encontramos que un aumento en *loyalty* siempre implica un aumento en la habilidad, más prominente para valores más altos de TOB (Fig. 2.5). Por el contrario, el aumento de los valores de TOB muestra una disminución de la habilidad para los niveles bajos de *loyalty*, y solo implica un aumento de la habilidad para los niveles altos de *loyalty*. La diferencia de habilidad del mínimo al máximo es mayor a 4,5 tsp.

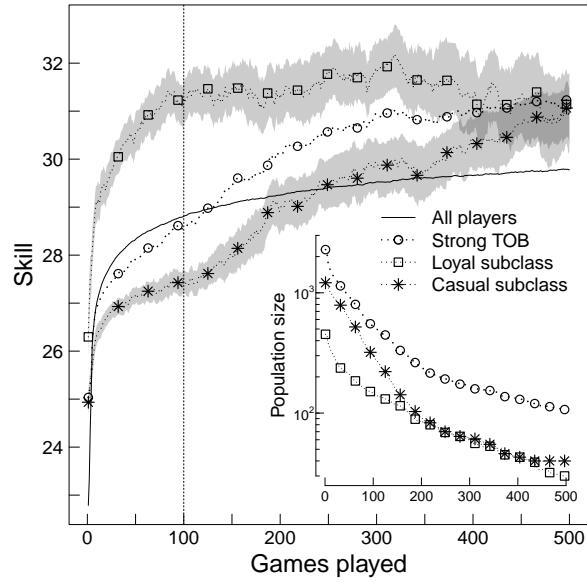


Figura 2.4: Curvas de aprendizaje de las subclases leal y casual de la clase TOB fuerte. La banda representa el intervalo de confianza 95 % Wilcoxon rank-sum, y la línea media representa la pseudomediana. Como referencia, mostramos la curva de aprendizaje de toda la población y la clase TOB fuerte. Los resultados son análogos a los obtenidos con la media y el intervalo de confianza 95 % t-test. La línea vertical en 100 partidas jugadas indica el análisis realizado en la Fig. 2.5.

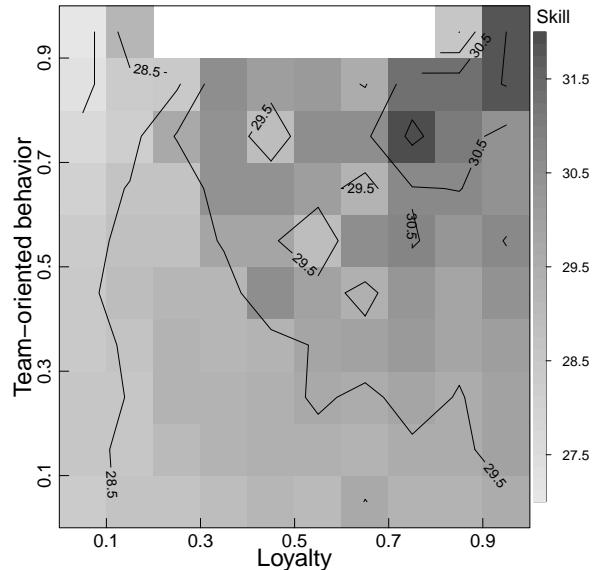


Figura 2.5: Interacción de la habilidades entre *loyalty* y TOB para todos los jugadores. El papel de la experiencia se aisló tomando la habilidad de los jugadores en el mismo punto de experiencia. Todos los jugadores tienen 100 de partidas jugadas. La habilidad promedio de cada parcela se informa mediante la escala de grises. Se muestran las curvas de nivel. Las parcelas vacías tienen menos de cinco jugadores.

La interacción entre *loyalty* y TOB se puede resumir como $Loyalty \cdot TOB = faithfulness$, lo que queda definido como

$$Faithfulness = \frac{\text{Máximo de partidas jugadas con un mismo compañero}}{\text{Partidas jugadas}}, \quad (2.4)$$

que es simplemente la proporción de veces jugadas con el compañero de equipo más leal sobre todas las partidas jugadas.

Para medir la influencia sobre la adquisición de habilidades de *loyalty*, TOB y *faithfulness*, construimos un modelo lineal que será resuelto por mínimos cuadrados. La correlación entre las variables *loyalty* y TOB es baja (0,11) y *Variance Inflation Factor* es nulo (1,01), lo que sugiere que no hay evidencia de colinealidad.

$$\text{skill}_i \sim \beta_1 \text{loyalty}_i + \beta_2 \text{TOB}_i + \beta_3 \text{faithfulness}_i. \quad (2.5)$$

Con 100 partidas jugadas, la variable *loyalty* tiene una pendiente positiva significativa, la variable TOB tiene una negativa significativa, mientras que la variable *faithfulness* tiene una pendiente positiva pronunciada significativa (Tabla 2.1). La variable *faithfulness* es lo suficientemente fuerte como para revertir la contribución negativa de TOB sobre la habilidad a una positiva cuando $loyalty > 0,27$. El *faithfulness-boost effect* es de aproximadamente 3,7 tsp, lo que genera una diferencia de habilidades entre jugadores con la misma experiencia extremadamente relevante en términos de probabilidad de ganar.

	Estimate	Std. Error	t value	Pr(> t)
Intercept	28.5707	0.0405	705.59	$p < 2e^{-16}$
Loyalty	0.7594	0.0972	7.82	$p < 2e^{-14}$
Team-oriented	-1.0042	0.1088	-9.23	$p < 2e^{-16}$
Faithfulness	3.7077	0.2611	14.20	$p < 2e^{-16}$

Tabla 2.1: Influencia sobre el aprendizaje de las variables de *loyalty*, TOB y *faithfulness* (modelo lineal). Reportamos el valor de la pendiente estimada, su desviación estándar y la significancia respecto a una pendiente cero. Todos los jugadores tienen 100 partidas de experiencia.

Repetimos este procedimiento para jugadores con la misma experiencia, a partir de 100 a 1300 partidas jugadas (Fig. 7.9 en Anexo). El *faithfulness-boost effect* sigue siendo significativo hasta 400 partidas jugadas, siempre por encima de 3 tsp. A partir de 500 partidas jugadas, la variable *faithfulness* deja de ser significativa pero la pendiente de la variable TOB revierte su contribución a una significativa positiva. La variable *loyalty* tiene un efecto positivo significativo en cualquier nivel de experiencia (≥ 100 partidas jugadas). Aunque el efecto de interacción del modelo lineal (*faithfulness*) ya no es significativo a partir de 500 partidas de experiencia, el punto de máxima habilidad siempre se alcanza maximizando tanto la variable *loyalty* como TOB. La magnitud de esta contribución siempre es relevante en términos de probabilidad de ganar, con más de 2 tsp.

Para integrar todas las observaciones parciales realizadas hasta aquí, ajustamos un modelo general a todos los datos, incluyendo la experiencia, *loyalty* y TOB como predictores y la identidad de la persona como efecto aleatorio (Tabla 2.2). Elegimos un modelo mixto lineal porque la relación entre la experiencia y la habilidad es lineal en una escala logarítmica

siguiendo la expresión (2.1). La variable dependiente, habilidad de los jugadores, se transformó a una escala logarítmica,

$$\log_{10}(skill) = \log_{10}(experience) + loyalty + TOB + individual\ player + \varepsilon$$

El conjunto de datos utilizado contiene los valores para todos los jugadores entre 10 y 500 partidas jugadas. La colinealidad entre variables es nula en términos del *Variance Inflation Factor* calculado a partir de un modelo lineal resuelto por mínimos cuadrados (es decir, todos los VIF son inferiores a 1,5). Por tanto, podemos ajustar el modelo lineal sin incurrir en resultados artificiales.

	Estimate	Norm.Est.	t value	[0.005	0.995]
Intercept	1.415	26.00	5966.264	1.414	1.415
exp	0.016	0.98	225.628	0.016	0.017
loyal	-0.007	-0.42	-28.846	-0.008	-0.006
tob	-0.044	-2.51	-91.773	-0.045	-0.043
(faithful) loyal:tob	0.090	5.99	91.801	0.088	0.093
exp:tob	0.016	0.98	70.829	0.015	0.016
exp:loyal	0.004	0.24	29.678	0.003	0.004
exp:loyal:tob	-0.026	-1.51	-56.129	-0.027	-0.025
Group Var	0.002				

Tabla 2.2: Modelo mixto lineal ajustado para todos los datos entre 10 y 500 partidas jugadas de experiencia individual. En la columna Norm.Est. transformamos los estimadores, en escala logarítmica, a su valor normalizado (e.g. Intercept = $10^{1,415}$, exp = $10^{1,415+0,016} - 10^{1,415}$). Método: REML converged. Número de grupos: 65335. Tamaño máximo de grupo: 491. Tamaño medio de grupos: 99,5

Este modelo general confirma las observaciones ya introducidas. La experiencia es el principal predictor en tanto tiene la pendiente más significativa. Inicialmente, la variable *loyalty* solo tiene un efecto positivo en contextos de orientados a equipos (loyal:tob). Ser leal sin un comportamiento orientado a equipo tiene un efecto muy marginal (loyal). Sin un compañero de equipo estable, jugar en equipos es un mal plan (tob). Sin embargo a medida que se gana mayor experiencia, y el efecto de la interacción *faithfulness* pierde fuerza (exp:loyal:tob), el efecto de la variable *loyalty* (exp:loyal) y TOB (exp:tob) revierten su contribución a una positiva. Debido a esta dinámica el valor máximo de habilidad se alcanza siempre cuando se maximizan ambas variables TOB y *loyalty*, es decir maximizando el *faithfulness*.

2.5. Discusión

Tradicionalmente el aprendizaje ha sido modelado en función de la experiencia individual. En este artículo exploramos cómo la curva de aprendizaje esperada por la experiencia individual se ve alterada por diferentes estrategias de agrupamiento. Para ello, analizamos el juego en línea *Conquer Club* que, a diferencia del ajedrez, se puede jugar en equipos. Nuestro estudio se basa en datos masivos, que permitieron realizar un estudio longitudinal con una precisión muy alta para detectar cambios sutiles. Allí verificamos la ley de la práctica en nuestro conjunto de datos, que llamamos la “curva de aprendizaje esperada por la experiencia”,

la cual tomamos como *baseline* para cuantificar el efecto de los diferentes comportamientos de agrupamiento sobre la adquisición de habilidades.

En primer lugar, encontramos que los comportamientos orientados a equipo están relacionados con una mejora significativa del nivel de habilidad a largo plazo a pesar de tener una tasa de crecimiento inicial más lenta. Este resultado se alinea con las expectativas previas. El *aprendizaje social* es un factor fundamental en los seres humanos para la adquisición de habilidades. Jugar en equipo ofrece un espacio privilegiado para la socialización y la transmisión de conocimientos. Pero también implica un desafío. Coordinar correctamente las actividades con otros compañeros de equipo requiere la adquisición de una habilidad adicional, que puede explicar la lenta tasa de crecimiento inicial.

En segundo lugar, encontramos que la estabilidad en la formación de equipos se asocia con una mejora de las habilidades a corto plazo. Este resultado se alinea con algunas expectativas previas. Si el aprendizaje asociado a la coordinación de equipo debe renovarse cada vez que se cambia de compañeros de equipo, entonces es esperable que la estabilidad de compañeros de equipo amortigue ese impacto negativo, acelerando los efectos positivos relacionados con el aprendizaje social.

En tercer lugar, hemos visto que la combinación de estos dos efectos, definido como *faithfulness*, intensifica ambas observaciones previas. Es decir, a medida que las personas juegan más partidas en equipo el aprendizaje a largo plazo es mayor. Y si además de este efecto positivo, las personas forman equipo lealmente, se observa un impulso temprano de las habilidades. El efecto positivo producto de la conjunción de estos factores es equivalente al aprendizaje que adquiere luego de miles de partidas de práctica de experiencia individual.

Un aspecto nos resultó sorprendente respecto de nuestras expectativas previas. Antes de comenzar este trabajo esperábamos encontrar algún efecto positivo vinculado a cambiar de equipo. Esta idea estaba basada en una hipótesis ingenua, que suponía que el acceso a diferentes grupos ofrecería un mayor repertorio cultural que sería positivo para el aprendizaje. Sin embargo, estudios recientes han cuestionado la hipótesis de que todo lo que maximice el flujo de información cultural debería repercutir positivamente en el aprendizaje. En particular, el exceso en la interconexión reduciría las posibilidades de innovación [5]. Por el contrario, la formación de subgrupos beneficiaría el desarrollo de innovaciones [5]. En este sentido, nuestro resultado se alinea con los trabajos teóricos y experimentales recientes.

También es importante señalar que la probabilidad de victoria de un equipo es independiente de la diferencia entre compañeros (Fig. 7.10 en Anexo). Asociarse con un compañero de equipo con menos habilidades implicará efectivamente una disminución en la probabilidad de ganar, pero no necesariamente implicará una disminución en la habilidad. Si la colaboración es sólida, ambos jugadores se beneficiarán del efecto positivo sobre la adquisición de habilidades.

La evidencia deja importantes preguntas abiertas que pueden tener implicaciones prácticas para la planificación de estrategias de capacitación. Nuestra hipótesis sugiere que la sociabilidad es el factor de aprendizaje subyacente de diferentes tácticas de agrupación. Sin embargo, se necesita más trabajo para poder formular explicaciones y recomendaciones confiables. La investigación experimental es necesaria para determinar con certeza las causas de esos efectos observados. Creemos que el efecto positivo de la asociación surge del compromiso social. Los derivados socio-cognitivos de la lealtad como la confianza, la constancia y la comunicación fluida superan los costes de coordinación y la reducción del abanico de relaciones que se pueden establecer.

Capítulo 3

Resultado 2. Implementación del estimador de habilidad estado del arte

Si bien las redes bayesianas han mostrado ser útiles para evaluar el aprendizaje en los sistemas educativos, hace falta más trabajo en el desarrollo de redes bayesianas dinámicas [96]. Por ejemplo, los estimadores de habilidad más utilizados en la industria del video juego, como Elo y TrueSkill, no permiten obtener estimaciones iniciales fiables ni garantizar la comparabilidad entre estimaciones distantes en el tiempo y el espacio debido a que propagan la información en una sola dirección a través del sistema, utilizando el último posterior como prior del siguiente evento.

Durante la realización de esta tesis de doctorado, descubrimos la especificación matemática de un modelo conocido como TrueSkill Through Time (TTT), que propaga toda la información histórica a través del sistema, lo que permite realizar estimaciones fiables de la habilidad inicial y garantizar la comparabilidad histórica. Si bien el modelo TTT fue publicado hace más de una década, no se encontraba disponible hasta entonces en ningún lenguaje de programación. Por ello, decidimos realizar nuestra propia implementación con la que ofrecemos el primer software para **Julia**, **Python** y **R**, acompañado por una explicación intuitiva comprensible para el público general, y otra explicación en profundidad específica para las personas con conocimientos básicos en probabilidad. La mejora de este modelo se hace evidente a partir de las predicciones a priori, que son varios órdenes de magnitud más precisas que las obtenidas con los modelos mencionados. Con no más de tres hiperparámetros intuitivos, el modelo TTT logra resultados similares y hasta mejores que modelos más complejos como KickScore, de forma más eficiente. Luego de ilustrar su modo de uso básico, mostramos cómo estimar la curvas de aprendizaje de los jugadores históricos de la Asociación de Tennis Profesional (ATP). Este algoritmo requiere pocas iteraciones para converger, permitiendo analizar millones de observaciones usando cualquier ordenador de gama baja.

3.1. Introducción

Todos los estimadores de habilidad ampliamente utilizados en la actualidad comparten alguna variante del modelo causal propuesto por Elo [58, 97, 98]. La figura 3.1 ofrece un modelo causal en el que las habilidades generan el resultado observable. Los agentes exhiben distintos

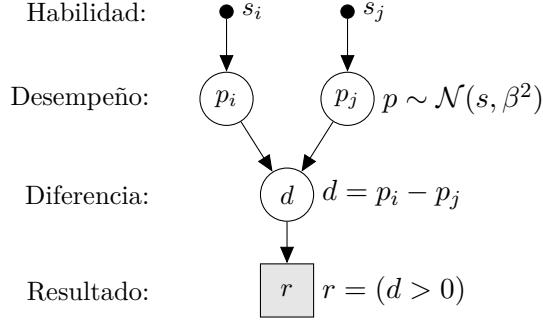


Figura 3.1: Modelo generativo en el que las habilidades causan los resultados observables a través de la diferencia de rendimientos ocultos, $d = p_i - p_j$, ambas variables aleatorias centradas en la verdadera habilidad, $p \sim \mathcal{N}(s, \beta^2)$. Quien haya obtenido mayor rendimiento gana, $r = (d > 0)$. Las variables observables se pintan de gris, las ocultas en blanco, y las constantes se muestran como puntos negros.

desempeños en cada evento, que varían alrededor de su verdadera habilidad, $\mathcal{N}(p | s, \beta^2)$. El modelo supone que gana el agente con mayor rendimiento, $r = (p_i > p_j)$. El parámetro β^2 , al ser el mismo para todos los agentes, actúa como la escala de las estimaciones: una habilidad a una distancia de un β por encima de otra implica 76 % probabilidad de ganar, independiente del valor absoluto de las estimaciones

Los estimadores de habilidad comúnmente utilizados en la industria del video juego y la academia, como Elo, Glicko [97], TrueSkill [58] e IRT [98, 99], no pueden obtener estimaciones iniciales fiables ni asegurar la comparación histórica debido a que propagan la información en una sola dirección a través del sistema, utilizando el último posterior como prior del siguiente evento (enfoque *filtering*). Luego, el posterior aproximado en un determinado momento se define como,

$$\widehat{\text{Posterior}}_t \propto \widehat{\text{Likelihood}}_t \underbrace{\widehat{\text{Posterior}}_{t-1} \text{ como Prior}_t}_{\widehat{\text{Posterior}}_1 \text{ como Prior}_2} \underbrace{\widehat{\text{Likelihood}}_{t-1} \dots \widehat{\text{Likelihood}}_1 \text{ Prior}_1}_{\widehat{\text{Likelihood}}_1 \text{ Prior}_1}. \quad (3.1)$$

Donde $\widehat{\text{Posterior}}_i$ y $\widehat{\text{Likelihood}}_i$ representan la aproximaciones inducidas por la ecuación (1.29) en el i -ésimo evento. Si consideramos la verosimilitud como un filtro del prior, cada posterior puede ser visto como una acumulación de todos los filtros anteriores. De esta forma, la información propaga de estimaciones pasadas hacia futuras. Debido a que las habilidades cambian en el tiempo, es importante agregar alguna incertidumbre γ luego de cada paso.

$$\widehat{p}(s_{i_t}) = \mathcal{N}(s_{i_t} | \mu_{i_{t-1}}, \sigma_{i_{t-1}}^2 + \gamma^2). \quad (3.2)$$

Debido a que el enfoque de filtrado es un procedimiento ad hoc que no surge de ningún modelo probabilístico, sus estimaciones exhiben una serie de problemas. El más obvio es que el inicio de toda secuencia de estimaciones siempre tiene alta incertidumbre. Pero también pueden producirse desacoplos temporales y espaciales que impidan comparar estimaciones distantes. Aunque la diferencia relativa entre estimaciones contemporáneas al interior de comunidades bien conectadas sea correcta, las estimaciones separadas en el tiempo y entre comunidades poco conectadas pueden ser incorrectas. Todos estos problemas están relacionados al hecho

de que la información propaga en una sola dirección a través del sistema, cuando la inferencia debería realizarse con toda la información disponible, también de eventos que ocurren de forma paralela como de eventos futuros disponibles.

3.1.1. Modelo TrueSkill Through Time

Para corregir los problemas de TrueSkill, es necesario realizar la inferencia dentro de una red bayesiana que incluya todas las actividades históricas, de modo que la información pueda propagarse por todo el sistema. De esta forma se garantiza tanto buenas estimaciones iniciales como la comparación de las estimaciones que están separadas en el tiempo y el espacio. La conectividad entre eventos surge de suponer que la habilidad de un jugador en un tiempo t depende de su propia habilidad en un tiempo anterior $t - 1$, generando una red que adquiere su estructura dependiendo de quienes participan en cada evento. Algoritmos similares fueron implementados por Coulom [100] y Maystre et. al [101] basados en aproximaciones laplacianas y procesos gaussianos respectivamente. Excluyendo el aspecto dinámico, $\gamma = 0$, el prior de un agente i en el t -ésimo evento es el producto de todas sus verosimilitudes, salvo la del t -ésimo evento.

$$\text{Prior}_{i_t} = \text{Prior}_{i_0} \underbrace{\prod_{k=1}^{t-1} \text{Verosimilitud}_{i_k}}_{\text{Información pasada}} \underbrace{\prod_{k=t+1}^{T_i} \text{Verosimilitud}_{i_k}}_{\text{Información futura}}, \quad (3.3)$$

donde T_i es la cantidad total de eventos del agente i , siendo Prior_{i_0} su prior inicial. Esto produce una mutua dependencia entre estimaciones que nos obliga a usar iterativamente las últimas verosimilitudes disponibles hasta alcanzar convergencia (detalles en la sección 3.2.8). En la figura 3.2 mostramos cómo convergen las estimaciones en una red bayesiana de dos

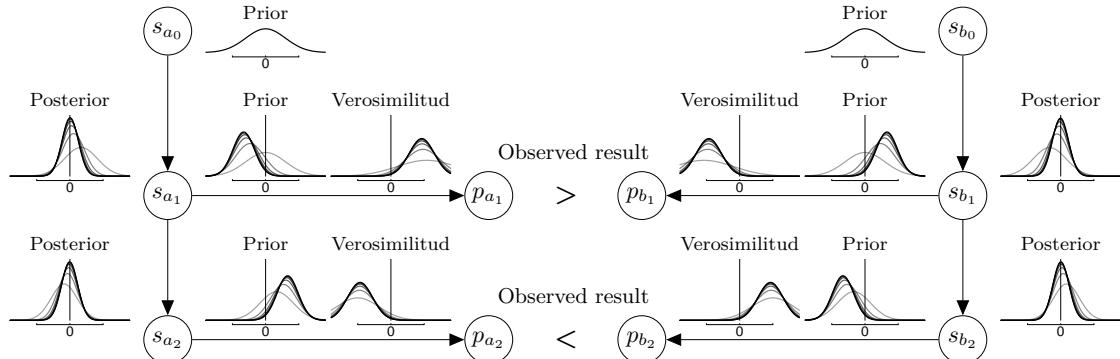


Figura 3.2: Convergencia de una red Bayesiana con dos eventos y dos agentes: la primera partida la gana el jugador a y la segunda la gana el jugador b . La luminosidad de las curvas indican el orden: la primera (la más clara) corresponde a las estimaciones de TrueSkill, y la última (la más oscura) corresponde con las estimaciones de TrueSkill Through Time.

agentes y dos eventos. TrueSkill Through Time recupera, de acuerdo a lo que sugieren los datos (una victoria cada uno), las verdaderas diferencias entre habilidades indicando que ambos jugadores tienen misma habilidad (posterior centrado en cero), a diferencia de TrueSkill que ofrece estimaciones sesgadas.

La ventaja de TrueSkill Through Time radica en que el modelo causal temporal permite que la información propague correctamente por todo el sistema. A diferencia de las redes

neuronales que tienen estructuras regulares, estas redes bayesianas adquieren siempre una estructura compleja, creciendo típicamente a millones de parámetros (e.g., videojuego). El procedimiento converge con unas pocas iteraciones lineales sobre los datos. La corrección de los sesgos es un paso fundamental para construir estimadores confiables que sirvan tanto para la toma de decisiones en áreas sensibles como para la evaluación de teorías científicas que utilicen la habilidad como dato observable. Con este artículo ponemos a disposición los primeros paquetes de TrueSkill Through Time para **Julia**, **Python** y **R**, junto con su documentación científica completa [102].

3.1.2. Evaluación de modelos

El modelo TrueSkill Through Time (TTT) [103] propaga la información histórica a través de toda la red, proporcionando estimaciones con baja incertidumbre en cualquier momento, ofreciendo estimaciones iniciales de habilidad fiables y garantizando la comparación de las estimaciones distantes en el tiempo y el espacio (enfoque *smoothing*). Enfoques similares fueron implementados en *Whole History Rating* [100] y *KickScore* [101], basados en aproximaciones laplacianas y procesos gaussianos respectivamente.

En la tabla 3.1 comparamos nuestra implementación del algoritmo TTT contra KickScore TrueSkill, Elo, y un modelo “Constante” en el que las habilidades no cambian en el tiempo. Para evaluar los modelos debemos calcular su probabilidad dados los datos,

$$P(\text{Modelo}|\text{Datos}) = \frac{P(\text{Datos}|\text{Modelo})P(\text{Modelo})}{P(\text{Datos})}. \quad (3.4)$$

Cuando no tenemos acceso a todos los modelos, no podremos calcular $P(\text{Modelo}|\text{Datos})$, pero sí podremos comparar modelos.

$$\frac{P(\text{Modelo}_i|\text{Datos})}{P(\text{Modelo}_j|\text{Datos})} = \frac{P(\text{Datos}|\text{Modelo}_i)P(\text{Modelo}_i)}{P(\text{Datos}|\text{Modelo}_j)P(\text{Modelo}_j)} \stackrel{\text{si } *}{=} \frac{P(\text{Datos}|\text{Modelo}_i)}{P(\text{Datos}|\text{Modelo}_j)}. \quad (3.5)$$

Esta expresión se conoce como el *Bayes factor*, el cual se suele expresar en escala logarítmica para reportar la diferencia en órdenes de magnitud. En el caso especial en el que no tenemos preferencia a priori sobre ningún modelo ($\stackrel{\text{si } *}{=}$) la comparación de modelos depende de

$$P(\text{Datos}|\text{Modelo}) = P(d_1|\text{Modelo})P(d_2|d_1, \text{Modelo}) \dots P(d_n|d_1, \dots, d_{n-1}, \text{Modelo}), \quad (3.6)$$

donde cada elemento de la productoria es una predicción del siguiente dato en base a los datos previos y el modelo. En este trabajo también vamos a reportar la media geométrica debido a que puede interpretarse como la predicción “característica” del modelo,

$$P(\text{Datos}|\text{Modelo}) = \text{geometric mean}(P(\text{Datos}|\text{Modelo}))^{|\text{Datos}|}, \quad (3.7)$$

donde $|\text{Datos}|$ representa el tamaño del conjunto de datos.

Para realizar la comparación seguiremos la metodología empleada por Maystre et. al [101] analizando las cuatro bases de datos sobre las cuales ellos reportan sus predicciones a priori: ATP (tenis), NBA (baloncesto), FIFA (fútbol), y FIDE (ajedrez). Usamos el 70 % inicial de la base de datos para entrenar los hiperparámetros y el 30 % final para evaluar los modelos. Para predecir el resultado de una observación en el tiempo t , ellos utilizan todos los datos (en los conjuntos de entrenamiento y de prueba) hasta el día anterior a t . La tabla 3.1 resume las

Dataset Test size	Constante [†] GM log ₂ BF		Elo [†] GM log ₂ BF		TrueSkill GM log ₂ BF		KickScore [†] GM log ₂ BF		TTT GM	TTT LOOCV
Tenis 186 361	0,5593	7910	0,5695	3051	0,5722	1780	0,5758	93	0,5760	0,5908
Basketball 20 300	0,5006	1771	0,5305	72	0,5316	11	0,5328	-55	0,5318	0,5382
Chess 92 004	0,3570	520	0,3552	1190	0,3580	148	0,3584	0	0,3584	0,3641
Fútbol 5759	0,3949	30	0,3867	204	0,3921	89	0,3961	4	0,3963	0,3974

Tabla 3.1: Comparaciones de modelos en cuatro base de datos. Las columnas GM reportan la media geométrica de las predicciones a priori de los modelos (ecuación 3.7). Las columnas log₂BF reportan el Bayes Factor, siguiendo la expresión (3.5), entre TTT y otros modelos en escala logarítmica. Como referencia, la columna LOOCV reporta la media geométrica de las predicciones usando toda la información histórica, $p(d_i|d_1, \dots, d_{i-1}, d_{i+1}, \dots, d_n, M)$. La media geométrica de los modelos señalados con el símbolo [†] fue obtenida por Maystre et. al [101].

comparaciones de los modelos en las cuatro base de datos. El modelo TTT supera *KickScore* en la base de datos de tenis por 93 órdenes de magnitud, mientras que *KickScore* supera a TTT en la base de basket por 55 órdenes de magnitud. Consideramos que los modelos empatan en las bases de datos de ajedrez y de fútbol en tanto la diferencia es menor a 10 órdenes de magnitud. Unificando las cuatro base de datos en una sola, el modelo TTT supera a *KickScore* por más de 40 órdenes de magnitud.

Para computar las predicciones, Maystre et. al utilizan un cluster para crear en paralelo un modelo diferente por cada día t , cada uno requiere cerca de 100 iteraciones para alcanzar convergencia. En cambio, en nuestro caso fue suficiente usar una computadora de escritorio para crear un único modelo al que le agregamos los datos día a día, realizando una única iteración en cada paso. La selección de modelo la hemos realizado optimizando los hiperparámetros, pero una selección de modelo bayesiana completa debería computar las predicciones integrando todo el espacio de hiperparámetros. Este procedimiento suele penalizar a los modelos complejos cuando la búsqueda en el espacio de hiperparámetros es innecesaria. Con no más de tres hiperparámetros intuitivos (incertidumbre a priori, incertidumbre dinámica y probabilidad de empate), el modelo TTT logra resultados similares y hasta mejores que *KickScore* de forma más eficiente, debido a que el último debe optimizar en un espacio de parámetros mucho más grande, que incluye la definición de un *kernel* y luego los hiperparámetros específicos a ese *kernel*.

Cuando el objetivo es estimar las curvas de aprendizaje en el tiempo con la mayor precisión posible, en vez de forzar al modelo a hacer la inferencia usando solamente la información del pasado (como hicimos para comparar el desempeño de los modelos), conviene hacer la inferencia usando toda la información histórica. Esto es útil debido a que los modelos probabilísticos pueden usar información del futuro para inferir eventos del pasado, algo que intuitivamente hacemos cuando sospechamos que los deportistas destacados también eran hábiles un tiempo antes de hacerse conocidos. La columna LOOCV de la tabla 3.1 reporta la media geométrica de las predicciones que usan toda la información histórica excepto el evento que se está prediciendo, $p(d_i|d_1, \dots, d_{i-1}, d_{i+1}, \dots, d_n, M)$. Si bien no usamos estas predicciones *offline* para comparar modelos (Bayes Factor), es interesante notar que incorporar toda la información histórica al modelo TTT mejora sus propias estimaciones centenas de órdenes de magnitud.

Debido a que la estimación de habilidad es un problema sensible para los individuos, en las siguientes secciones ofrecemos una explicación lo más intuitiva y completa posible. Las comunidades que son evaluadas tanto en la industria del video juego como en los sistemas educativos necesitan conocer cómo funciona el algoritmo que mide su habilidad. Además de desarrollar una herramienta de software, ofrecemos un reporte científico completo y accesible al público en general.

3.2. Software: implementación del modelo

En esta sección ofrecemos la documentación matemática completa del modelo TrueSkill Through Time. La ventaja de este modelo reside en la aplicación estricta de la teoría de la probabilidad: todos los supuestos se hacen explícitos a través de un modelo generativo, y la inferencia se resuelve sólo con las regla de la probabilidad, nada más que las reglas de la suma y el producto. En la sección 3.2.1 introducimos el *sum-product algorithm*, que nos permite aplicar eficientemente estas reglas para computar las distribuciones marginales, e.g., el posterior y la predicción a priori. En la sección 3.2.2 enumeramos las propiedades que necesitaremos para derivar las distribuciones marginales de interés. En la sección 3.2.3 introducimos las operaciones de la clase **Gaussian**, la que realiza la mayor parte del cómputo. En las secciones 3.2.4, 3.2.5, 3.2.6, mostramos cómo resolver la predicción a priori y el posterior exacto de un evento, incorporamos empate al modelo y explicamos cómo aproximar el posterior exacto en partidas con dos equipos. En la sección 3.2.7 explicamos la solución general multi-equipos, que requiere la aplicación de un algoritmo iterativo. En la sección 3.2.8 justificamos los pasos matemáticos requeridos para resolver el modelo TrueSkill Through Time completo.

Debido a que la herramienta fue desarrollada con tres lenguajes de programación, identificaremos las diferentes sintaxis usando el siguiente formato:

Syntax common to Julia, Python and R		
Julia syntax	Python syntax	R syntax

donde la línea completa la usamos cuando la sintaxis de los tres lenguajes coinciden, y las columnas las usamos cuando los lenguajes difieren: Julia a la izquierda, Python al centro, y R a la derecha.

3.2.1. Sum-product algorithm

Cualquier modelo puede factorizarse en el producto de probabilidades condicionales. El *sum-product algorithm* [94] aprovecha la estructura de la distribución de probabilidad conjunta, impuesta por el modelo causal (figura 3.1), para aplicar eficientemente las reglas de la probabilidad. El problema de investigación relativo a una partida con dos jugadores contiene las hipótesis $\mathcal{H} = \{H_{S_1}, H_{S_2}, H_{P_1}, H_{P_2}, H_D, H_R\}$, donde $s_1 \in H_{S_1}$ y $s_2 \in H_{S_2}$ representan habilidades, $p_1 \in H_{P_1}$ y $p_2 \in H_{P_2}$ desempeños, $d \in H_D$ diferencia de desempeños y $r \in H_R$ el resultado. Haciendo uso de esas las independencias entre hipótesis especificada en la figura 3.1, la probabilidad conjunta puede factorizarse como

$$p(\mathbf{s}, \mathbf{p}, d, r) = p(s_1)p(s_2)p(p_1|s_1)p(p_2|s_2)p(d|\mathbf{p})P(r|d). \quad (3.8)$$

En la figura 3.3 mostramos esta factorización gráficamente. Este tipo de representaciones, conocidas como *factor graph*, son grafos con dos tipos de nodos: nodos variables (círculos

blancos), y nodos funciones (cuadrados negros). Los ejes entre nodos variables y nodos funciones representan la relación matemática “la variable v es argumento de la función f ”. En

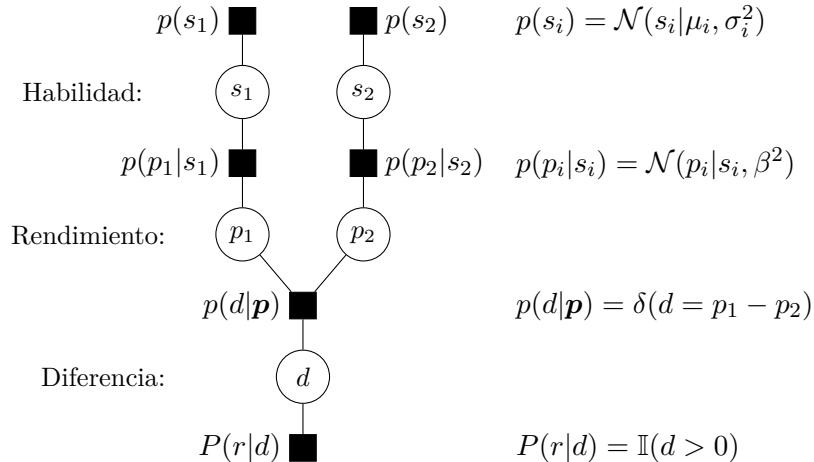


Figura 3.3: Forma gráfica de representar la factorización de la distribución conjunta inducida por el modelo causal básico (ecuación (3.8)). Los cuadrados negros representan las funciones, los círculos blancos representan las variables, y los ejes entre ellos representan la relación matemática “la variable es argumento de la función”.

nuestro caso, buscamos computar dos marginales, el posterior proporcional de las habilidades $p(s_i, r)$ y la probabilidad a priori del resultado $p(r)$. El *sum-product algorithm* es una forma general de descomponer las reglas de la probabilidad como mensajes que se envían los nodos del *factor graph*. Hay dos tipos de mensajes: los mensajes que envían los nodos variables a sus funciones vecinas ($m_{v \rightarrow f}(v)$); y los mensajes que envían los nodos funciones a sus variables vecinas ($m_{f \rightarrow v}(v)$). El primero codifica una porción de la regla del producto.

$$m_{v \rightarrow f}(v) = \prod_{h \in n(v) \setminus \{f\}} m_{h \rightarrow v}(v), \quad (\text{paso del producto})$$

donde $n(v)$ representa el conjunto de vecinos del nodo v . Es decir, el mensaje que envía una variable v a un factor f es simplemente la multiplicación de los mensajes que recibe del resto de sus vecinos $h \in n(v)$ salvo f . Los mensajes que envían los nodos funciones codifican una parte de la regla de la suma.

$$m_{f \rightarrow v}(v) = \int \cdots \int \left(f(\mathbf{h}, v) \prod_{h \in n(f) \setminus \{v\}} m_{h \rightarrow f}(h) \right) d\mathbf{h}, \quad (\text{paso de la suma})$$

donde $\mathbf{h} = n(f) \setminus \{v\}$ es el conjunto de todos los vecinos de f salvo v , y $f(\mathbf{h}, v)$ representa la función f , evaluada en todos sus argumentos. En pocas palabras, el mensaje que envía una función f a una variable v también es la multiplicación de los mensajes que recibe del resto de sus vecinos $h \in n(f)$ salvo v , incluyendo en el producto el factor mismo $f(\cdot)$, integrando (o sumando) sobre los vecinos \mathbf{h} . Finalmente, la distribución de probabilidad marginal de una variable v es simplemente la multiplicación de los mensajes que v recibe de todos sus vecinos.

$$p(v) = \prod_{h \in n(v)} m_{h \rightarrow v}. \quad (\text{probabilidad marginal})$$

Este algoritmo codifica la mínima cantidad de pasos que se requieren para calcular cualquier distribución de probabilidad marginal.

3.2.2. Propiedades matemáticas y notación

La eficiencia de TrueSkill Through Time se obtiene gracias a que las marginales se pueden computar de forma analítica. En esta sección enumeramos las propiedades que necesitamos para derivar los mensajes exactos y aproximados que surgen del *sum-product algorithm*. La primera propiedad establece que el producto de dos distribuciones gaussianas, ambas evaluadas en el mismo punto x , pueden expresarse como el producto de otras dos distribuciones gaussianas con sólo una de ellas evaluada en x .

$$\mathcal{N}(x|\mu_1, \sigma_1^2) \mathcal{N}(x|\mu_2, \sigma_2^2) \stackrel{7.5.1}{=} \mathcal{N}(\mu_1|\mu_2, \sigma_1^2 + \sigma_2^2) \mathcal{N}(x|\mu_*, \sigma_*^2), \quad (\text{Producto de gaussianas})$$

con $\mu_* = \frac{\mu_1}{\sigma_1^2} + \frac{\mu_2}{\sigma_2^2}$ y $\sigma_*^2 = \left(\frac{1}{\sigma_1^2} + \frac{1}{\sigma_2^2}\right)^{-1}$. Algo similar ocurre con la división de dos distribuciones gaussianas, ambas evaluadas en el mismo punto x .

$$\mathcal{N}(x|\mu_1, \sigma_1^2) / \mathcal{N}(x|\mu_2, \sigma_2^2) \stackrel{7.5.4}{\propto} \mathcal{N}(x|\mu_{\nabla.}, \sigma_{\nabla.}^2) / \mathcal{N}(\mu_1|\mu_2, \sigma_1^2 + \sigma_2^2), \quad (\text{División de gaussianas})$$

con $\mu_{\nabla.} = \frac{\mu_1}{\sigma_1^2} - \frac{\mu_2}{\sigma_2^2}$ y $\sigma_{\nabla.}^2 = \left(\frac{1}{\sigma_1^2} - \frac{1}{\sigma_2^2}\right)^{-1}$. La función indicadora $\mathbb{I}(\cdot = \cdot)$ vale 1 cuando la igualdad es verdadera y 0 en caso contrario. Se usa para representar distribuciones de probabilidad de variables discretas no aleatorias, como el resultado de las partidas dada la diferencia de desempeños $p(r|d)$. De la misma forma, la función delta de Dirac $\delta(\cdot = \cdot)$ se usa para representar distribuciones de probabilidad de variables continuas no aleatorias, es la diferencia de desempeños dados los rendimientos de los agentes $p(d|\mathbf{p})$. Cuando permite remplazar variable dentro de una integral,

$$\iint \delta(x = h(y, z)) f(x) g(y) dx dy = \int f(h(y, z)) g(y) dy \quad (\text{Función delta de dirac})$$

la dimensionalidad del problema se reduce. Usaremos además las propiedades que se derivan de la simetría de gaussianas.

$$\mathcal{N}(x|\mu, \sigma^2) = \mathcal{N}(\mu|x, \sigma^2) = \mathcal{N}(-\mu|x, \sigma^2) = \mathcal{N}(-x|\mu, \sigma^2). \quad (\text{Simetría de gaussianas})$$

La estandarización de la gaussiana,

$$\mathcal{N}(x|\mu, \sigma^2) = \mathcal{N}((x - \mu)/\sigma|0, 1). \quad (\text{Estandarización de gaussianas})$$

La igualdad entre la distribución gaussiana y la derivada de la su acumulada,

$$\frac{\partial}{\partial x} \Phi(x|\mu, \sigma^2) = \mathcal{N}(x|\mu, \sigma^2), \quad (\text{Derivada de la gaussiana acumulada})$$

que vale por definición. La simetría de la distribución gaussiana acumulada.

$$\Phi(0|\mu, \sigma^2) = 1 - \Phi(0|\mu, \sigma^2). \quad (\text{Simetría de la gaussiana acumulada})$$

3.2.3. La clase gaussiana

La clase **Gaussian** realiza la mayor parte del cómputo en todos los paquetes. Se representa mediante dos parámetros, la media y el desvío estándar.

```
N1 = Gaussian(mu = 1.0, sigma = 1.0); N2 = Gaussian(1.0, 2.0)
```

Código 3.1: Inicialización de distribuciones gaussianas

La clase sobreescribe los operadores suma (+), resta (-), producto (*) y división (/) con las principales propiedades requeridas para computar las distribuciones marginales en el modelo TrueSkill Through Time.

$$\mathcal{N}(x|\mu_1, \sigma_1^2)\mathcal{N}(x|\mu_2, \sigma_2^2) \stackrel{7.5.1}{\propto} \mathcal{N}(x|\mu_*, \sigma_*^2). \quad (\text{N1} * \text{N2})$$

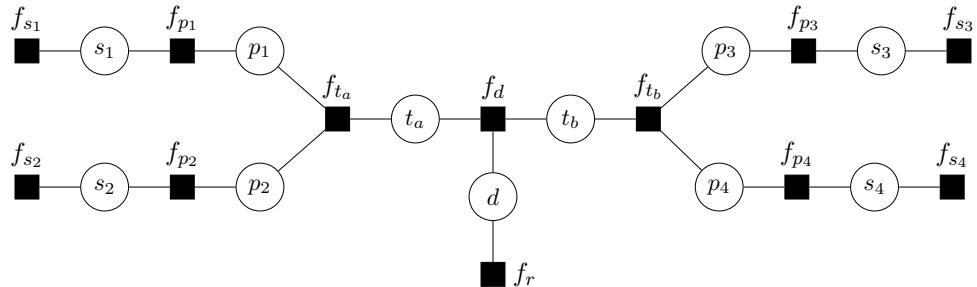
$$\mathcal{N}(x|\mu_1, \sigma_1^2)/\mathcal{N}(x|\mu_2, \sigma_2^2) \stackrel{7.5.4}{\propto} \mathcal{N}(x|\mu_{\nabla}, \sigma_{\nabla}^2). \quad (\text{N1} / \text{N2})$$

$$\begin{aligned} \iint \delta(t = x + y) \mathcal{N}(x|\mu_1, \sigma_1^2) \mathcal{N}(y|\mu_2, \sigma_2^2) dx dy &\stackrel{7.5.2}{=} \mathcal{N}(t|\mu_1 + \mu_2, \sigma_1^2 + \sigma_2^2). & (\text{N1} + \text{N2}) \\ \iint \delta(t = x - y) \mathcal{N}(x|\mu_1, \sigma_1^2) \mathcal{N}(y|\mu_2, \sigma_2^2) dx dy &\stackrel{7.5.2}{=} \mathcal{N}(t|\mu_1 - \mu_2, \sigma_1^2 + \sigma_2^2). & (\text{N1} - \text{N2}) \end{aligned}$$

Aunque estas propiedades son ampliamente conocidas, adjuntamos sus demostraciones completas en el anexo.

3.2.4. Solución exacta para eventos con dos equipos

En presencia de equipos, el modelo Elo supone que el desempeño de los equipos t es la suma de los desempeños de sus integrantes, y que el equipo con mayor desempeño gana, $r = (t_i > t_j)$. En la figura 3.4 mostramos la factorización gráfica del modelo Elo que incorpora equipos. En este ejemplo, tenemos dos equipos con dos jugadores cada uno. Toda partida con



$$f_{s_i} = \mathcal{N}(s_i|\mu_i, \sigma^2) \quad f_{p_i} = \mathcal{N}(p_i|s_i, \beta^2) \quad f_{t_e} = \delta(t_e = \sum_{i \in A_e} p_i) \quad f_d = \delta(d = t_a - t_b) \quad f_r = \mathbb{I}(d > 0)$$

Figura 3.4: Factorización gráfica de una partida con dos equipos de dos jugadores. El modelo Elo incorpora una variable nueva, t , que modela el desempeño de los equipos.

dos equipos tiene solución analítica exacta.

En esta sección mostramos los pasos para calcular la predicción a priori (o evidencia) exacta y los likelihoods exactos de una partida con dos equipos. Sólo necesitamos el *sum-product algorithm* y las propiedades arriba mencionadas. Empezaremos primero con los mensajes “descendentes”, desde los priors al resultado, hasta calcular la evidencia, y seguiremos con los mensajes “ascendentes”, desde el resultado observado a los priors, hasta calcular el posterior de cada agente. Además, acompañaremos la derivación matemática con código en Julia, python y R.

```
using TrueSkillThroughTime      import trueskillthroughtime      library(TrueSkillThroughTime)
mu = 0.0; sigma = 6.0; beta = 1.0; gamma = 0.03
```

```
a1 = Player(Gaussian(mu, sigma), beta, gamma); a2 = Player(); a3 = Player(); a4 = Player()
```

Código 3.2: Inicialización de los jugadores

Aquí creamos 4 jugadores mediante el método `Player()`.

Mensajes descendentes

Siguiendo el paso de la suma del *sum-product algorithm* y la factorización del modelo presentado en la figura 3.4, podemos ver que los mensajes que envían los factores de habilidad f_{s_i} a la variable s_i no son otra cosa más que el prior.

$$m_{f_{s_i} \rightarrow s_i}(s_i) = \mathcal{N}(s_i | \mu_i, \sigma_i^2). \quad (\text{prior})$$

Tenemos acceso a este mensaje llamando al atributo `prior` de la clase `Player`.

```
a1.prior      a1.prior      a1@prior
> Gaussian(mu=0, sigma=6.0)
```

Código 3.3: Prior de uno de los agentes

Además, siguiendo el paso del producto del *sum-product algorithm* y la factorización del modelo, el mensaje que envía la variable s_i al factor rendimiento f_{p_i} , $m_{s_i \rightarrow f_{p_i}}(s_i)$, es nuevamente el prior. Debido a que es trivial calcular los mensajes que envían las variables (siempre es el producto de los mensajes que reciben de atrás), vamos a evitar escribirlos. Veamos entonces el mensaje que envían los factores rendimiento f_{p_i} a su variable p_i .

$$m_{f_{p_i} \rightarrow p_i}(p_i) = \int \mathcal{N}(p_i | s_i, \beta^2) \mathcal{N}(s_i | \mu_i, \sigma_i^2) ds_i = \mathcal{N}(p_i | \mu_i, \beta^2 + \sigma_i^2). \quad (\text{performance}())$$

Tenemos acceso a estos mensajes mediante el método `performance()` de la clase `Player`.

```
p1 = performance(a1)      p1 = a1.performance()      p1 = performance(a1)
p2 = performance(a2)      p2 = a2.performance()      p2 = performance(a2)
p3 = performance(a3)      p3 = a3.performance()      p3 = performance(a3)
p4 = performance(a4)      p4 = a4.performance()      p4 = performance(a4)
```

Código 3.4: Computando el desempeño individual a priori

El mensaje que envían los factores equipos f_{t_e} a la variable equipo t_e es una integral sobre todas las variables de desempeño individual,

$$\begin{aligned} m_{f_{t_e} \rightarrow t_e}(t_e) &= \iint \delta(t_e = p_i + p_j) \mathcal{N}(p_i | \mu_i, \beta^2 + \sigma_i^2) \mathcal{N}(p_j | \mu_j, \beta^2 + \sigma_j^2) dp_i dp_j \\ &= \mathcal{N}(t_e | \underbrace{\mu_i + \mu_j}_{\mu_e}, \underbrace{2\beta^2 + \sigma_i^2 + \sigma_j^2}_{\sigma_e^2}), \end{aligned} \quad (\text{ta} = p1 + p2)$$

donde la función delta de Dirac impone la restricción de que la suma de los desempeños individuales sea igual a un valor de rendimiento del equipo t_e constante. Haciendo uso de las propiedades podemos resolver esta integral de forma analítica, obteniendo como resultado que el desempeño a priori de los equipos es una distribución gaussiana centrada en la suma de las estimaciones medias $\mu_e = \mu_i + \mu_j$ con una varianza que incluye tanto las incertidumbres de las estimaciones, $\sigma_i^2 + \sigma_j^2$, como la varianza de los rendimientos individuales, $\beta^2 + \beta^2$. Tenemos acceso a este mensaje cuando usamos el operador `+` de la clase `Gaussian` para sumar los desempeños de los agentes,

```
ta = p1 + p2; tb = p3 + p4
```

Código 3.5: Computando el desempeño a priori de los equipos

El siguiente mensaje, que envía el factor diferencia f_{d_1} a la variable diferencia d_1 es,

$$\begin{aligned} m_{f_d \rightarrow d}(d) &= \iint \delta(d = t_a - t_b) \mathcal{N}(t_a | \mu_a, \sigma_a^2) \mathcal{N}(t_b | \mu_b, \sigma_b^2) dt_a dt_b \\ &= \mathcal{N}(d | \underbrace{\mu_a - \mu_b}_{\substack{\text{Diferencia} \\ \text{esperada}}}, \underbrace{\sigma_a^2 + \sigma_b^2}_{\substack{\text{incertidumbre} \\ \text{total}}}) = \mathcal{N}(d | \psi, \vartheta^2). \end{aligned} \quad (d = ta - tb)$$

La diferencia de desempeños a priori es una distribución gaussiana centrada en la diferencia esperada a priori $\psi = \mu_a - \mu_b$ con una varianza $\vartheta^2 = \sigma_a^2 + \sigma_b^2$ que incluye la incertidumbre de ambos equipos. Tenemos acceso a este mensaje cuando usamos el operador `-` de la clase `Gaussian` para obtener la diferencia de los desempeños de los equipos,

```
d = ta - tb
```

Código 3.6: Computando la diferencia de desempeños a priori

El último mensaje descendente, el que envía el factor f_r a la variable r , permite computar la evidencia, es decir la predicción a priori del resultado observado.

$$m_{f_r \rightarrow r}(r) = \int \mathbb{I}(d > 0) \mathcal{N}(d | \psi, \vartheta^2) dd = 1 - \Phi(0 | \psi, \vartheta^2). \quad (\text{evidence})$$

Tenemos acceso a este mensaje calculando el valor acumulado desde 0 hasta ∞ de la distribución de diferencia de desempeños de los equipos.

```
e = 1.0 - cdf(d, 0.0)      e = 1 - cdf(0, d.mu, d.sigma)      e = 1 - cdf(0, d@mu, d@sigma)
```

Código 3.7: Computando la predicción a priori del resultado observado (o evidencia) donde `e` contiene el valor de la ecuación `evidence`.

Mensajes ascendentes

Examinemos ahora los mensajes ascendentes. El primer mensaje ascendente lo envía el factor de resultados f_r a la variable de diferencia d .

$$m_{f_r \rightarrow d}(d) = \mathbb{I}(d > 0). \quad (3.9)$$

El mensaje contiene la función indicadora del factor f_r , que transmite la información del resultado observado. El mensaje que el factor diferencia f_d envía la variable de desempeño del equipo ganador t_a es,

$$\begin{aligned} m_{f_d \rightarrow t_a}(t_a) &= \iint \delta(d = t_a - t_b) \mathbb{I}(d > 0) \mathcal{N}(t_b | \mu_b, \sigma_b^2) dd dt_b \\ &= \int \mathbb{I}(t_a > t_b) \mathcal{N}(t_b | \mu_b, \sigma_b^2) dt_b = 1 - \Phi(0 | t_a - \mu_b, \sigma_b^2) = \Phi(t_a | \mu_b, \sigma_b^2). \end{aligned} \quad (3.10)$$

En este caso se integra el mensaje ascendente anterior junto con el mensaje descendente del otro equipo. Este mensaje, parametrizado en t_a , es la acumulada de la distribución gaussiana de los rendimientos del equipo contrario desde t_a hasta ∞ , y codifica la verosimilitud de las

hipótesis de rendimiento de equipo ganador. El mensaje enviado por el factor de rendimiento del equipo f_{t_a} a la variable del rendimiento individual p_1 es,

$$\begin{aligned}
 m_{f_{t_a} \rightarrow p_1}(p_1) &= \iint \delta(t_a = p_1 + p_2) N(p_2 | \mu_2, \beta^2 + \sigma_2^2) \Phi(t_a | \mu_b, \sigma_b^2) dt_a dp_2 \\
 &= \int N(p_2 | \mu_2, \beta^2 + \sigma_2^2) \Phi(p_1 + p_2 | \mu_b, \sigma_b^2) dp_2 \\
 &= 1 - \Phi(0 | p_1 + \underbrace{\mu_2 - \mu_b}_{\mu_1 - \psi}, \underbrace{\beta^2 + \sigma_2^2 + \sigma_b^2}_{\vartheta^2 - (\sigma_1^2 + \beta^2)}).
 \end{aligned} \tag{3.11}$$

Otra vez, el mensaje ascendente anterior se integra junto con un mensaje descendente, el desempeño a priori de su compañero de equipos. El mensaje, parametrizado en p_1 , codifica la verosimilitud de las hipótesis de rendimiento individual del jugador ganador. El mensaje enviado por el factor de rendimiento individual f_{p_1} a la variable de habilidad s_1 es,

$$\begin{aligned}
 m_{f_{p_1} \rightarrow s_1}(s_1) &= \int N(p_1 | s_1, \beta^2) \Phi(p_1 | \mu_1 - \psi, \vartheta^2 - (\sigma_1^2 + \beta^2)) dp_1 \\
 &= 1 - \Phi(0 | \underbrace{(s_1 + \mu_2) - (\mu_3 + \mu_4)}_{\text{Diferencia esperada parametrizada en } s_1}, \underbrace{\vartheta^2 - \sigma_1^2}_{\text{Incertidumbre total salvo la de } s_1}).
 \end{aligned} \tag{3.12}$$

Esta es la verosimilitud exacta discutida en la ecuación (1.28), que computa la probabilidad a priori de un resultado ganador si la verdadera habilidad del jugador fuera s_1 .

3.2.5. Modelo básico de empates

El modelo supone que ocurre un empate cuando la diferencia de rendimientos d no supera un cierto margen, $|t_a - t_b| \leq \varepsilon$. En la figura 3.5(a) se muestra la distribución de probabilidad de la diferencia de desempeños, $\mathcal{N}(d | \psi, \vartheta^2)$, junto con los tres resultados posibles. Este mo-

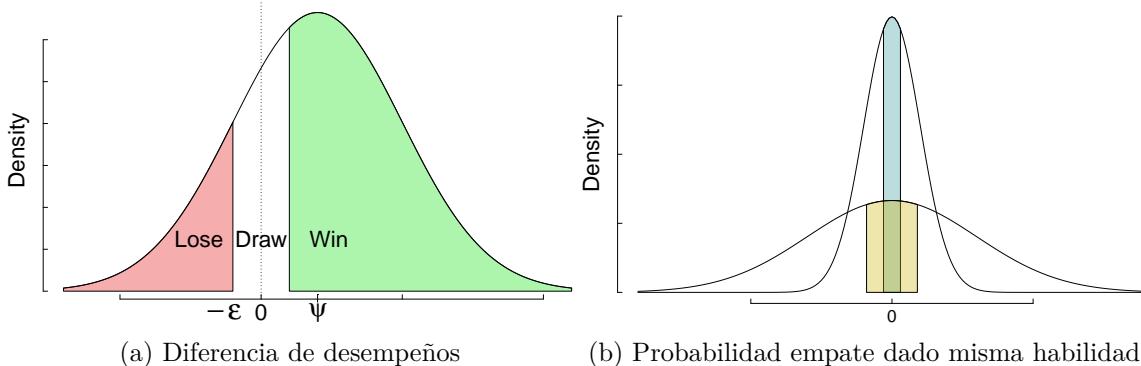


Figura 3.5: Distribución de diferencia de desempeño bajo el modelo de empate. En la figura (a) mostramos un ejemplo de las áreas correspondientes a la probabilidad de perder, empatar y ganar. En la figura (b) mostramos cómo el margen de empate ε se debe adaptar para mantener la probabilidad de empate constante (dado equipos con misma habilidad) cuando la incertidumbre de la distribución cambia.

delo básico requiere determinar el valor usado para determinar si hubo empate. El artículo original [58] propone usar la frecuencia empírica de empates como indicio para definirlo. Sin

embargo, este valor depende de la diferencia de habilidad real, que justamente no conocemos. Suponiendo que podemos definir la “probabilidad de empate entre equipos con misma habilidad”, es importante notar que el margen también depende de la cantidad de jugadores. En la figura 3.5(b) se puede ver que para mantener el área de empates constante, es necesario adaptar el margen dependiendo de la incertidumbre. Como los resultados observados son independientes de nuestras creencias, la única fuente de incertidumbre proviene de la varianza de los rendimientos β . Así es que podemos definir una ecuación que vincula el margen con la probabilidades de empate.

$$\text{Draw probability} = \Phi\left(\frac{\varepsilon}{\sqrt{n_1 + n_2}\beta}\right) - \Phi\left(\frac{-\varepsilon}{\sqrt{n_1 + n_2}\beta}\right), \quad (3.13)$$

donde Φ representa la acumulada de una Gaussiana estandarizada, con media 0 y desvío 1. En el siguiente código usamos la función `compute_margin()` para calcular el tamaño del margen de empate.

```
na = length(team_a)          na = len(team_a)          na = length([a1, a2])
nb = length(team_b)          nb = len(team_b)          nb = length([a3, a4])
sd = sqrt(na + nb)*beta      sd = math.sqrt(na + nb)*beta  sd = sqrt(na + nb)*beta
p_draw = 0.25
margin = compute_margin(p_draw, sd)
```

Código 3.8: Computando el margen de empate

donde los agentes `a` fueron inicializados en el código 3.21, y `beta` en el código 3.20.

3.2.6. Aproximación óptima del posterior exacto

En la sección 3.2.4 hemos visto cómo encontrar el posterior exacto. En esta sección mostraremos cómo encontrar la distribución gaussiana que mejor aproxima al posterior exacto, considerando la posibilidad de empates. Los paquetes lo resuelven con las siguientes dos líneas de código.

```
g = Game(teams, p_draw = 0.25)
post = posteriors(g)          post = g.posteriors()          post = posteriors(g)
```

Código 3.9: Computando el posterior aproximado

donde la variable `teams` fue inicializada en el código 3.22. La necesidad de aproximar el posterior surge debido a que la distribución de diferencia de desempeños es una gaussiana truncada.

$$p(d) = \begin{cases} \mathcal{N}(d|\psi, \vartheta^2)\mathbb{I}(-\varepsilon < d < \varepsilon) & \text{tie} \\ \mathcal{N}(d|\psi, \vartheta^2)\mathbb{I}(d > \varepsilon) & \text{not tie} \end{cases}. \quad (3.14)$$

Se sabe que la familia exponencial, a la que pertenece la distribución gaussiana, minimizan la divergencia Kullback-Leibler respecto de la verdadera distribución p , $KL(p||q)$, cuando ambas tienen los mismos momentos [104]. La esperanza y la varianza de una gaussiana truncada $\mathcal{N}(x|\mu, \sigma^2)$ en un intervalo $[a, b]$ son,

$$E(X|a < X < b) = \mu + \sigma \frac{\mathcal{N}(\alpha) - \mathcal{N}(\beta)}{\Phi(\beta) - \Phi(\alpha)} \quad (3.15)$$

$$V(X|a < X < b) = \sigma^2 \left(1 + \left(\frac{\alpha N(\alpha) - \beta N(\beta)}{\Phi(\beta) - \Phi(\alpha)} \right) - \left(\frac{N(\alpha) - N(\beta)}{\Phi(\beta) - \Phi(\alpha)} \right)^2 \right), \quad (3.16)$$

donde $\beta = \frac{b-\mu}{\sigma}$ y $\alpha = \frac{a-\mu}{\sigma}$. Con un único truncamiento, se pueden simplificar como,

$$E(X|a < X) = \mu + \sigma \frac{\mathcal{N}(\alpha)}{1 - \Phi(\alpha)} , \quad V(X|a < X) = \sigma^2 \left(1 + \left(\frac{\alpha \mathcal{N}(\alpha)}{1 - \Phi(\alpha)} \right) - \left(\frac{\mathcal{N}(\alpha)}{1 - \Phi(\alpha)} \right)^2 \right).$$

Luego, la gaussiana que mejor aproxima a $p(d)$ es

$$\hat{p}(d) = \mathcal{N}(d|\hat{\psi}, \hat{\vartheta}^2) = \begin{cases} \mathcal{N}(d|E(d) - \varepsilon < d < \varepsilon), V(d) - \varepsilon < d < \varepsilon) & \text{tie} \\ \mathcal{N}(d|E(d|d > -\varepsilon), V(d|d > -\varepsilon) & \text{not tie.} \end{cases} \quad (\text{approx}())$$

```
tie = true          tie = True          tie = T
d_approx = approx(d, margin, !tie)
```

Código 3.10: Computando la aproximación de la diferencia de desempeños

donde la distribución de diferencias d fue inicializada en el código 3.6, y `margin` en el código 3.8. Dada $\hat{p}(d)$, podemos calcular el resto de los mensajes ascendentes usando las operaciones de la clase `Gaussian`. Para derivar el primer mensaje ascendente aproximado, recordar que cualquier distribución marginal se puede calcular como el producto de los mensajes que le envían todos los factores vecinos.

$$\begin{aligned} m_{d \rightarrow f_d}(d) &= \frac{p(d)}{m_{f_d \rightarrow d}(d)} \approx \frac{\hat{p}(d)}{m_{f_d \rightarrow d}(d)} \\ &= \frac{\mathcal{N}(d|\hat{\psi}, \hat{\vartheta}^2)}{\mathcal{N}(d|\psi, \vartheta^2)} \propto N(d, \psi_{\nabla.}, \vartheta_{\nabla.}^2), \end{aligned} \quad (\text{approx_lh_d})$$

donde $m_{f_r \rightarrow d}(d) = m_{d \rightarrow f_d}(d)$ vale por la factorización del modelo. Tenemos acceso a este mensaje cuando usamos el operador `/` de la clase `Gaussian`, para dividir las gaussianas indicadas,

```
approx_lh_d = d_approx / d
```

Código 3.11: Computando el primer mensaje aproximado

Todos estos mensajes aproximados pueden ser interpretados como verosimilitudes porque contienen la información del resultado observado. El mensaje aproximado que el factor diferencia f_d envía a la variables de desempeño del equipo ganador t_a es,

$$\begin{aligned} \hat{m}_{f_d \rightarrow t_a}(t_a) &= \iint \delta(d = t_a - t_b) \mathcal{N}(d| \psi_{\nabla.}, \vartheta_{\nabla.}^2) \mathcal{N}(t_b | \mu_b, \sigma_b^2) dd dt_b \\ &= \int \mathcal{N}(t_a - t_b | \psi_{\nabla.}, \vartheta_{\nabla.}^2) \mathcal{N}(t_b | \mu_b, \sigma_b^2) dt_b = \mathcal{N}(t_a | \mu_b + \psi_{\nabla.}, \vartheta_{\nabla.}^2 + \sigma_b^2). \end{aligned} \quad (3.17)$$

El mensaje aproximado que el factor desempeño de equipo f_{t_a} envía a la variables de desempeño individual p_1 es,

$$\begin{aligned} \hat{m}_{f_{t_a} \rightarrow p_1}(p_1) &= \iint \delta(t_a = p_1 + p_2) \mathcal{N}(t_a | \mu_b + \psi_{\nabla.}, \vartheta_{\nabla.}^2 + \sigma_b^2) \mathcal{N}(p_2 | \mu_2, \sigma_2^2 + \beta^2) dt_a dp_2 \\ &= \int \mathcal{N}(p_1 + p_2 | \mu_b + \psi_{\nabla.}, \vartheta_{\nabla.}^2 + \sigma_b^2) \mathcal{N}(p_2 | \mu_2, \sigma_2^2 + \beta^2) dp_2 \\ &= \mathcal{N}(p_1 | \underbrace{\mu_b - \mu_2}_{\mu_1 - \psi} + \psi_{\nabla.}, \underbrace{\vartheta_{\nabla.}^2 + \sigma_b^2 + \sigma_2^2 + \beta^2}_{\vartheta^2 - (\sigma_1^2 + \beta^2)}). \end{aligned} \quad (3.18)$$

El mensaje aproximado que el factor de desempeño individual f_{p_1} envía a la variables de habilidad s_1 es,

$$\begin{aligned}\widehat{m}_{f_{p_1} \rightarrow s_1}(s_1) &= \int \mathcal{N}(p_1|s_1, \beta^2) \mathcal{N}(p_1|\mu_1 - \psi + \psi_{\nabla.}, \vartheta_{\nabla.}^2 + \vartheta^2 - \sigma_1^2 - \beta^2) dp_1 \\ &= \mathcal{N}(s_1|\mu_1 - \psi + \psi_{\nabla.}, \vartheta_{\nabla.}^2 + \vartheta^2 - \sigma_1^2).\end{aligned}\quad (3.19)$$

Finalmente, el posterior proporcional aproximado de la variable s_1 se obtiene multiplicando los mensajes que recibe de sus factores vecinos.

$$\widehat{p}(s_1, r) = \mathcal{N}(s_1|\mu_1, \sigma_1^2) \mathcal{N}(s_1|\mu_1 - \psi + \psi_{\nabla.}, \vartheta_{\nabla.}^2 + \vartheta^2 - \sigma_1^2). \quad (\text{posterior})$$

Tenemos acceso al posterior normalizado usando el operador $*$ de la clase `Gaussian`, o tomando el primer elemento de la lista `post` computada en el código 3.9.

```
mu = a1.prior.mu
sigma2 = a1.prior.sigma^2
phi = d.mu
v2 = d.sigma^2
phi_div = approx_lh_d.mu
v2_div = approx_lh_d.sigma^2
prior = a1.prior
posterior = post[1][1]
print( prior * Gaussian(mu-phi+phi_div, sqrt(v2 + v2_div - sigma2)) )
> Gaussian(mu=2.461, sigma=5.507)
print(posterior)
> Gaussian(mu=2.461, sigma=5.507)
```

```
mu = a1.prior.mu
sigma2 = a1.prior.sigma**2
phi = d.mu
v = d.sigma**2
phi_div = approx_lh_d.mu
v2_div = approx_lh_d.sigma**2
prior = a1.prior
posterior = post[0][0]
```

```
mu = a1@prior@mu
sigma2 = a1@prior@sigma^2
phi = d@mu
v2 = d@sigma^2
phi_div = approx_lh_d@mu
v2_div = approx_lh_d@sigma^2
prior = a1@prior
posterior = post[[1]][[1]]
```

Código 3.12: Accediendo al posterior aproximado

con `a1`, `d` y `approx_lh_d` computadas en los códigos 3.21, 3.6 y 3.11. Se puede ver que el posterior que nosotros hemos computado es el mismo que devuelve la clase `Gaussian`.

3.2.7. Varios equipos

La interfaz de los paquetes no distinguen entre partidas de dos equipos o de varios equipos, como se puede ver en los códigos de ejemplo 3.22 y 3.25. Sin embargo, en presencia de más de dos equipos nos vemos obligados a implementar un algoritmo iterativo debido a una dependencia mutua entre resultados. Gracias a la transitividad de los resultados, si participan k equipos en un evento, es suficiente con evaluar $k - 1$ diferencias de desempeño d_i entre equipos en posiciones consecutivas. Para ello definimos una lista o en la que los equipos están ordenados según el resultado observado, con o_1 el equipo ganador, y en general con o_i el equipo ubicado en la posición i . En la figura 3.6 mostramos la factorización del modelo general de TrueSkill. La idea básica es actualizar repetidamente hacia adelante y hacia atrás todos los mensajes en el camino más corto entre dos marginales $p(d_j)$ hasta la convergencia.

Veamos el algoritmo involucrado en resolver una partida con tres equipos. En vez de utilizar la notación de mensajes propuesta por el *sum-product algorithm*, le ponemos nombres a los mensajes como se muestra en la figura 3.7. Antes de empezar preparamos el escenario: computamos el desempeño a priori de los equipos usando la función `performance()`; inicializamos los mensajes que aún no están definidos con una forma neutra, como una distribución gaussiana con varianza infinita; y calculamos los márgenes de cada comparación d_j .

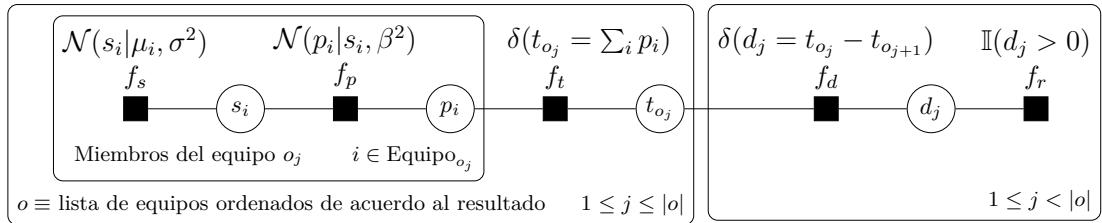


Figura 3.6: Factorización general del modelo de equipos. Los subíndices que aparecen abajo a la derecha en las placas, indican replicación. El subíndice j de la placa izquierda abre los k rendimientos de equipos, y el subíndice i de la placa interna despliega sus jugadores. El subíndice j de la placa derecha abre los $k - 1$ comparaciones entre equipos consecutivos.

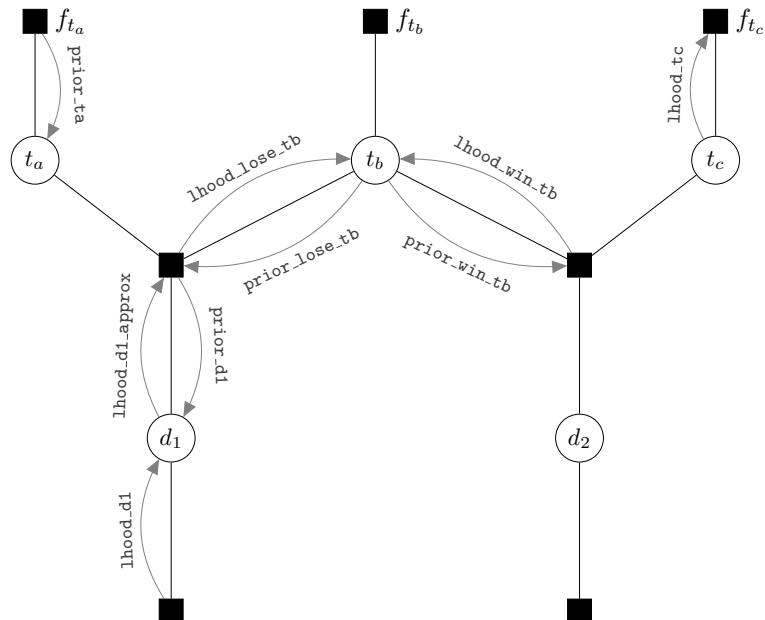


Figura 3.7: Factorización de una partida con tres equipos. Mostramos sólo factores desde los equipos hasta los resultados. Los nombres se usarán para explicar el procedimiento iterativo conocido como *loopy belief propagation*.

```

team_a = [a1]           team_a = [a1]           team_a = c(a1)
team_b = [a2, a3]       team_b = [a2, a3]       team_b = c(a2, a3)
team_c = [a4]           team_c = [a4]           team_c = c(a4)
prior_ta= performance(team_a); prior_tb= performance(team_b); prior_tc= performance(team_c)
N_inf = Gaussian(0., Inf)  N_inf = Gaussian(0, inf)  N_inf = Gaussian(0, Inf)
lhood_win_ta = N_inf; lhood_lose_tb = N_inf; lhood_win_tb = N_inf; lhood_lose_tc = N_inf
margin = compute_margin(p_draw, sqrt(3)*beta)

```

Código 3.13: Preparación del escenario

donde los equipos fueron definidos en el código 3.22. Como en ambas comparaciones hay tres jugadores, ajustamos ambos márgenes con ese mismo tamaño. Empecemos el proceso iterativo aproximando la distribución d_1 . Recuerden que cualquier distribución marginal es el producto de los mensajes que esta recibe de sus vecinos.

```
prior_lose_tb = prior_tb * lhood_win_tb
```

```

prior_d1 = prior_ta - prior_lose_tb
lhood_d1_approx = approx(prior_d1, margin, !tie) / prior_d1

```

Código 3.14: Aproximando la distribución d_1 con el último mensajes enviado por t_b
En la primera línea inicializamos el mensaje que las variable t_b envía al nodo factor f_{d_1} : la multiplicación de los mensajes que reciben de atrás. Notar que en la primera vuelta es equivalente a `prior_tb` debido a que la variable `lhood_win_tb` ha sido definida con un valor neutro. En la segunda línea computamos el mensaje que envía el factor f_{d_1} a la variable d_1 . En la última línea, calculamos el mensaje aproximado enviado por la variable d_1 al factor f_{d_1} . Esto nos permite actualizar el mensaje que recibe la variable t_b del factor f_{d_1} .

```

lhood_lose_tb = prior_ta - lhood_d1_approx

```

Código 3.15: Actualizando la distribución t_b con la última aproximación de d_1
Acá computamos el mensaje que envía el factor f_{d_1} a la variable t_b . Luego aproximamos la distribución d_2 usando los mensajes actualizados.

```

prior_win_tb = prior_tb * lhood_lose_tb
prior_d2 = prior_win_tb - prior_tc
lhood_d2_approx = approx(prior_d2, margin, tie) / prior_d2

```

Código 3.16: Aproximando la distribución d_2 con el último mensaje enviado por t_b
En la primera línea inicializamos el mensaje que la variable t_b envía al nodo factor f_{d_2} . En la segunda línea computamos el mensaje que envía el factor f_{d_2} a la variable d_2 . En la última línea, calculamos el mensaje aproximado enviado por la variable d_2 al factor f_{d_2} . Esto nos permite actualizar el mensaje que recibe la variable t_b del factor f_{d_2} .

```

lhood_win_tb = prior_lose_tc + lhood_d2_approx

```

Código 3.17: Actualizando la distribución t_b con la última aproximación de d_2
Acá computamos el mensaje que envía el factor f_{d_2} a la variable t_b . Para alcanzar convergencia hay que iterar los códigos 3.14, 3.15, 3.16, 3.17. Una vez terminada, enviamos los mensajes ascendentes a los equipos de ambos extremos.

```

lhood_win_ta = posterior_lose_tb + lhood_d1_approx
lhood_lose_tc = posterior_win_tb - lhood_d2_approx

```

Código 3.18: Cómputo de los mensajes que reciben los factores f_t del equipo ganador y perdedor

Estos son los mensajes que envían el factor f_{d_1} y f_{d_2} a la variable t_a y t_c respectivamente. Finalmente computar los likelihood de cada equipo.

```

lhood_ta = lhood_win_ta
lhood_tb = lhood_lose_tb * lhood_win_tb
lhood_tc = lhood_lose_tc

```

Código 3.19: Mensajes ascendentes de los factores f_t a las variables t
Los siguientes mensajes ascendentes se calculan como se describe en la sección 3.2.6.

3.2.8. Propagación de la información en la clase History

En esta sección explicamos cómo la información provista por los datos se propaga por toda la red bayesiana que contiene la historia de los eventos. TrueSkill Through Time crea un modelo causal único en el que todas las actividades históricas están vinculadas. La conectividad entre eventos surge de suponer que la habilidad de un jugador en un tiempo t depende de

su propia habilidad en un tiempo anterior $t - 1$. El modelo admite que en un mismo paso temporal t (e.g., día, semana, mes, año) un agente i participe en más de un evento. La figura 3.8 mostramos una parte de la factorización gráfica de la historia de eventos: los nodos vecinos a una variable de habilidad y los mensajes entre estos nodos. Por el *sum-product algorithm*

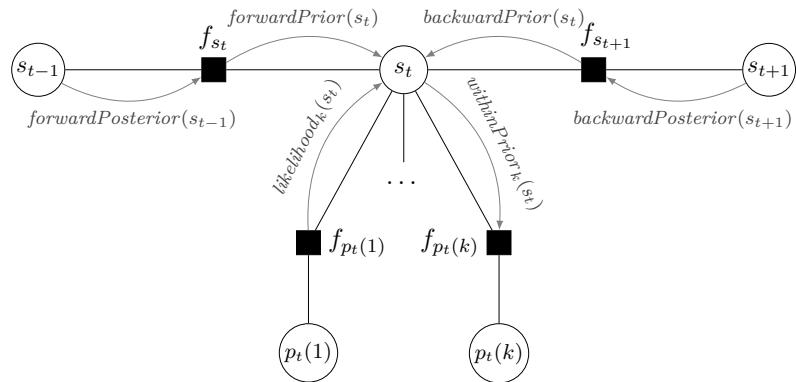


Figura 3.8: Nodos del grafo de factorización de la historia de eventos que son vecinos a la variable de habilidad de un agente en el paso temporal t . Las variables $p_t(j)$ representan el rendimiento p que ese agente tuvo en la j -ésima partida al interior del paso temporal t . Los nombres de las flechas representan los mensajes computados por el algoritmo de sum-product.

sabemos que la distribución marginal de cualquier variable es el producto de los mensajes que esta recibe de sus vecinos. Usando los nombres seleccionados en la figura 3.8, sabemos que la distribución posteriori de la habilidad de un agente i en un tiempo t es:

$$\text{posterior}(s_t) = \text{forwardPrior}(s_t) \cdot \text{backwardPrior}(s_t) \cdot \prod_{k=1}^{K_t} \text{likelihood}_k(s_t), \quad (3.20)$$

donde K_t es la cantidad de eventos en los que participa el agente en el paso temporal t . Los mensajes *likelihood* son las verosimilitudes de los eventos descritas en las secciones 3.2.6 y 3.2.7, y los mensajes *forwardPrior* y *backwardPrior* son las estimaciones de habilidad vecinas, a las que se le agrega cierta incertidumbre γ por el paso temporal.

$$forwardPrior(s_t) = \mathcal{N}(s_t | \mu_f, \sigma_f^2 + \gamma^2) \quad , \quad backwardPrior(s_t) = \mathcal{N}(s_t | \mu_b, \sigma_b^2 + \gamma^2), \quad (3.21)$$

donde $forwardPosterior(s_{t-1}) = \mathcal{N}(s_{t-1} | \mu_f, \sigma_f^2)$ y $backwardPosterior(s_{t+1}) = \mathcal{N}(s_{t-1} | \mu_b, \sigma_b^2)$. El mensaje $forwardPrior(s_t)$ es el $forwardPosterior(s_{t-1})$ luego de que se le agrega la incertidumbre dinámica del tiempo t , f_{s_t} . Estos mensajes coinciden con las definiciones de prior y posterior del modelo básico de TrueSkill. La coincidencia surge de aplicar el *sum-product algorithm*. Pero de su aplicación también aparece la existencia del mensaje $backwardPrior(s_t)$, que es el $backwardPosterior(s_{t+1})$ luego de que se le agrega la incertidumbre dinámica del tiempo $t+1$, $f_{s_{t+1}}$. La cantidad de incertidumbre dinámica agregada a los mensajes $forwardPrior$ y $backwardPrior$ en la ecuación (3.21) es la misma. Sin embargo, a veces quisiéramos que la incertidumbre dinámica dependa de la distancia temporal entre variables de habilidad adyacentes. Nuestros paquetes admiten ambas opciones. Cuando inicializamos la clase `History` sin especificar el tiempo de los eventos, como hicimos en los códigos 3.26 y 3.30, los factores dinámicos incorporan siempre un γ entre variables que son adyacentes según la composición

de los eventos. Cuando especificamos el tiempo de los eventos, como hicimos en el código 3.31, la adyacencia depende de esos tiempos y la incertidumbre dinámica depende de la distancia temporal entre dos variables adyacentes s_t y s_{t-1} .

$$f_{s_t} = \mathcal{N}(s_t | s_{t-1}, \text{distanciaTemporal}(s_t, s_{t-1}) \cdot \gamma^2). \quad (3.22)$$

Los priors que se usan para computar las verosimilitudes de los eventos son los mensajes descendentes *withinPrior*. Siguiendo el *sum-product algorithm* sabemos que los mensajes que envían las variables son la multiplicación de los mensajes que reciben de atrás, por lo que el mensaje *withinPrior* es,

$$\begin{aligned} \text{withinPrior}_q(s_t) &= \text{forwardPrior}(s_t) \cdot \text{backwardPrior}(s_t) \cdot \prod_{\substack{k=1 \\ q \neq k}}^{K_t} \text{likelihood}_k(s_t) \\ &= \frac{\text{posterior}(s_t)}{\text{likelihood}_q(s_t)}. \end{aligned} \quad (3.23)$$

El *withinPrior* contiene toda la información del *posterior*(s_t), salvo la información que proviene de la verosimilitud del evento q para la que es prior. Para resolver la mutua dependencia entre verosimilitudes, actualizamos repetidas veces los mensajes hacia adelante y hacia atrás hasta alcanzar convergencia. En cada pasada hacia adelante guardamos el último mensajes que los agentes envían para adelante, *forwardPosterior*, y en cada pasada hacia atrás guardamos el último mensajes que los agentes envían para atrás, *backwardPosterior*.

$$\begin{aligned} \text{forwardPosterior}(s_t) &= \frac{\text{posterior}(s_t)}{\text{backwardPrior}(s_t)} = \text{forwardPrior}(s_t) \cdot \prod_{k=1}^{K_t} \text{likelihood}_k(s_t) \\ \text{backwardPosterior}(s_t) &= \frac{\text{posterior}(s_t)}{\text{forwardPrior}(s_t)} = \text{backwardPrior}(s_t) \cdot \prod_{k=1}^{K_t} \text{likelihood}_k(s_t). \end{aligned} \quad (3.24)$$

Los mensajes que no están todavía definidos, como por ejemplo el *backwardPrior*(s_t) en la primer pasada, deben ser reemplazados por una forma neutral como la distribución gaussiana con varianza infinita. Este algoritmo requiere sólo unas pocas iteraciones lineales sobre los datos para converger y permite escalar a millones de observaciones en pocos minutos.

3.3. Interfaz y aplicaciones

En esta sección mostramos cómo usar los paquetes de **Julia**, **Python** y **R** para resolver un evento, una secuencia de tres eventos, la evolución de habilidad de un jugador, y la historia de la Asociación de Tenis Profesional (ATP). Presentamos las soluciones de los modelos TrueSkill y TrueSkill Through Time, y los pasos para obtener los posteriors, las curvas de aprendizaje, y la predicción a priori del dato observado (i.e. evidencia).

3.3.1. Único evento

La clase **Game** la utilizamos para modelar eventos y realizar la inferencia a partir de los datos de los equipos, el resultado y la probabilidad de empate típica de esos eventos (**p_draw**). Las características de los agentes se definen al interior de la clase **Player**: la distribución gaussiana a priori caracterizada por la media (**mu**) y el desvío estándar (**sigma**); el desvío

estándar de los rendimientos (**beta**); y el factor dinámico de la habilidad (**gamma**). En el siguiente código definimos las variables que utilizaremos más adelante, con los valores por defecto.

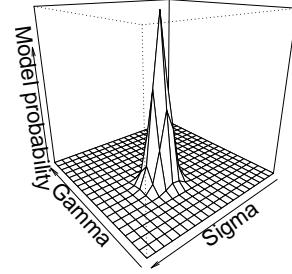
```
mu = 0.0; sigma = 6.0; beta = 1.0; gamma = 0.03; p_draw = 0.0
```

Código 3.20: Parámetros de los paquetes y sus valores por defecto

El valor inicial de **mu**, común a todos los jugadores, puede elegirse a gusto debido que es la diferencia de habilidades lo que realmente importa y no su valor absoluto. El desvío estándar del prior **sigma** debe ser suficientemente grande para incluir todas las posibles hipótesis de habilidad. El valor de **beta** quizás sea el más importante debido a que funciona como la escala de las estimaciones. Una diferencia de habilidad real de un β , $s_i - s_j = \beta$, equivale a 76 % de probabilidad de ganar. Como es la unidad de medida, elegimos **beta=1.0**. El factor dinámico **gamma** es en general una pequeña fracción de **beta**. Y la probabilidad de empate de las partidas (**p_draw**) se suele inicializar con la frecuencia observada de empates. En la figura

Parameter	Default value
mu	0,0
sigma	6,0
beta	1,0
gamma	0,03
p_draw	0,0

(a) Parámetros



(b) Optimización

Figura 3.9: (a) presentamos todos los parámetros del modelo y sus valores por defecto. (b) buscamos la combinación de valores **sigma** y **gamma** que maximice la probabilidad del modelo dado los datos.

3.9 resumimos estos valores por defecto y mostramos la posibilidad de optimizar dos de ellos que sus valores dependen del conjunto de datos. Con estos valores creamos cuatro jugadores idénticos.

```
a1 = Player(Gaussian(mu, sigma), beta, gamma); a2 = Player(); a3 = Player(); a4 = Player()
```

Código 3.21: Inicialización de los jugadores

El primer jugador fue creado haciendo explícitos los parámetros, mientras que el resto se inicializan con los valores por defecto del constructor. La clase **Gaussian** se usa para modelar las operaciones estándar de las distribuciones gaussianas, incluyendo multiplicación, suma, división y resta (detalles en la sección 3.2.3). En el siguiente paso creamos una partida con dos equipos de dos jugadores. En presencia de equipos, el rendimiento observado depende de la suma de los rendimientos de cada equipo (detalles en la sección 3.2.4).

```
team_a = [ a1, a2 ]
team_b = [ a3, a4 ]
teams = [team_a, team_b]
g = Game(teams)
```

```
team_a = [ a1, a2 ]
team_b = [ a3, a4 ]
teams = [team_a, team_b]
```

```
team_a = c(a1, a2)
team_b = c(a3, a4)
teams = list(team_a, team_b)
```

Código 3.22: Equipos e inicialización del juego

donde el resultado de la partida queda definido implícitamente por el orden de los equipos en

la lista `teams`: los equipos que aparecen primero en la lista (menor índice) le ganan a los que aparecen después (mayor índice). Este es el ejemplo de uso más simple. Más adelante veremos cómo especificar explícitamente el resultado. Durante la inicialización, la clase `Game` calcula la predicción a priori del resultado observado (`evidence`) y las verosimilitud aproximada de cada jugador (`likelihoods`).

```
lhs = g.likelihoods[1][1]           lhs = g.likelihoods[0][0]           lhs = g.likelihoods
ev = g.evidence                     ev = g.evidence                     ev = g.evidence
ev = round(ev, digits=3)           ev = round(ev, 3)                   ev = round(ev, 3)
print(ev)                           print(ev)                           print(ev)
> 0.5
```

Código 3.23: Consulta de la evidencia y las verosimilitudes

En este caso, la evidencia es de 0,5 porque ambos equipos tenían misma estimaciones de habilidad a priori. Los posteriors se pueden obtener multiplicando manualmente las verosimilitudes y los priors, o podemos llamar al método `posterior()` de la clase `Game` para que los compute. Las verosimilitudes y posteriors mantienen el orden en el que los jugadores y equipos fueron cargados durante la inicialización de la clase `Game`.

```
pos = posteriors(g)               pos = g.posterior()           pos = posteriors(g)
print(pos[1][1])                 print(pos[0][0])             print(pos[[1]][[1]])
> Gaussian(mu=2.361, sigma=5.516)
print(lhs[1][1] * a1.prior)       print(lhs[0][0] * a1.prior)   print(lhs[[1]][[1]]*a1@prior)
> Gaussian(mu=2.361, sigma=5.516)
```

Código 3.24: Consulta de los posteriors y su cómputo manual

donde el posterior impreso corresponde al primer jugador del primer equipo. Debido al resultado ganador, la estimación del jugador tiene ahora una media más grande y una incertidumbre más chica. El producto de gaussianas, la verosimilitud por el prior, genera el mismo posterior normalizado.

Ahora analizamos un ejemplo más complejo en el que los mismos cuatro jugadores participan en un juego de varios equipos. Los jugadores se organizan en tres equipos de diferente tamaño: dos equipos con un solo jugador, y el otro con dos jugadores. El resultado tiene un único equipo ganador y un empate entre los otros dos equipos perdedores. A diferencia del ejemplo anterior, ahora necesitamos usar una probabilidad de empate mayor a cero.

```
ta = [a1]                         ta = [a1]                         ta = c(a1)
tb = [a2, a3]                      tb = [a2, a3]                      tb = c(a2, a3)
tc = [a4]                         tc = [a4]                         tc = c(a4)
teams_3 = [ta, tb, tc]             teams_3 = [ta, tb, tc]             teams_3 = list(ta, tb, tc)
result = [1., 0., 0.]               result = [1, 0, 0]                   result = c(1, 0, 0)
g = Game(teams_3, result, p_draw=0.25)
```

Código 3.25: Juego con múltiples equipos de diferente tamaño y posibilidad de empate

donde `teams` contiene a los jugadores distribuidos en diferentes equipos, mientras que `result` indica ahora la puntuación obtenida por cada equipo. El equipo con la mayor puntuación es el ganador y los equipos con misma puntuación están empatados. De este modo podemos especificar cualquier resultado, incluidos empates globales. La evidencia y el posterior se obtienen de la misma forma que hemos visto.

3.3.2. Secuencia de eventos

La clase `History` se usa para computar los posteriors y la evidencia de secuencias de eventos. En el primer ejemplo, inicializamos la clase `History` con tres jugadores ("a", "b" y "c") y tres partidas. En la primera partida "a" le gana a "b", en la segunda "b" le gana a "c" y en la tercera "c" le gana a "a". En resumen, todos los agentes ganan una partida y pierden otra.

```
c1 = [[["a"], ["b"]]]           c1 = [[["a"], ["b"]]]           c1 = list(c("a"),c("b"))
c2 = [[["b"], ["c"]]]           c2 = [[["b"], ["c"]]]           c2 = list(c("b"),c("c"))
c3 = [[["c"], ["a"]]]           c3 = [[["c"], ["a"]]]           c3 = list(c("c"),c("a"))
composition = [c1, c2, c3]      composition = [c1, c2, c3]      composition = list(c1,c2,c3)
h = History(composition, gamma=0.0)
```

Código 3.26: Inicialización de una instancia de `History` con una secuencia de tres eventos donde la variable `c1`, `c2` y `c3` modela la composición de cada partida usando los nombres de los agentes (i.e. sus identificadores), la variable `composition` es una lista que contiene las tres partidas, y el valor nulo del parámetro `gamma` especifica que las habilidad no cambian en el tiempo. El resultado queda definido implícitamente por el orden en el que la composición de las partidas fueron inicializadas: los equipos que aparecen primero en la lista vencen a los que aparecen después. El resto de los parámetros se inicializan con los valores por defecto vistos en el código 3.20.

En este ejemplo, en el que todos los agentes se ganan mutuamente y sus habilidades no cambian en el tiempo, los datos sugieren que todos tienen la misma habilidad. Al inicializarse, la clase `History` inmediatamente instancia un nuevo jugador por cada nombre y activa el cómputo de las estimaciones TrueSkill, usando los posteriors de cada partida como prior de la siguiente. Para acceder a ellas podemos llamar al método `learning_curves()` de la clase `History`, que devuelve un diccionario indexado por los nombres de los agentes. Las curvas de aprendizaje individuales son listas de tuplas: el primer elemento indica el tiempo, y el segundo la estimación correspondiente a ese tiempo.

```
lc = learning_curves(h)          lc = h.learning_curves()          lc = h$learning_curves()
print(lc["a"])                   print(lc["a"])                   print(lc[["a"]])
> [(1, Gaussian(mu=3.339, sigma=4.985)), (3, Gaussian(mu=-2.688, sigma=3.779))]
print(lc["b"])                   print(lc["b"])                   print(lc[["b"]])
> [(1, Gaussian(mu=-3.339, sigma=4.985)), (2, Gaussian(mu=0.059, sigma=4.218))]
```

Código 3.27: Curvas de aprendizaje de los jugadores que participan en una secuencia de eventos

Las curvas de aprendizaje de los jugadores "a" y "b" contienen una tupla por partida jugada (no incluye el prior inicial). A pesar de que en este ejemplo ningún jugador muestra ser más fuerte que los demás, las estimaciones de TrueSkill varían mucho entre jugadores. Las estimaciones obtenidas luego de la primera partida, en la que "a" vence a "b", tienen misma incertidumbre y misma media en valor absoluto, siendo positiva para el jugador ganador y negativo para el perdedor. Las estimaciones calculadas luego de los eventos en los que juega "c", tienen menor incertidumbre y medias que se encuentran más cerca del cero.

TrueSkill Through Time resuelve la incapacidad de TrueSkill para obtener las estimaciones correctas permitiendo que la información propague por todo el sistema. Para computarlas es necesario llamar al método `convergence()` de la clase `History`.

```
convergence(h)                  h.convergence()                  h$convergence()
```

```

lc = learning_curves(h)      lc = h.learning_curves()      lc = h.learning_curves()
print(lc["a"])                print(lc["a"])                lc.print(lc[["a"]])
> [(1, Gaussian(mu=0.0, sigma=2.395)), (3, Gaussian(mu=-0.0, sigma=2.395))]
print(lc["b"])                print(lc["a"])                lc.print(lc[["a"]])
> [(1, Gaussian(mu=-0.0, sigma=2.395)), (3, Gaussian(mu=0.0, sigma=2.395))]

```

Código 3.28: Cómputo de curvas de aprendizaje de TrueSkill Through Time

TrueSkill Through Time no sólo devuelve las estimaciones correctas (la misma para todos los jugadores), también tienen menos incertidumbre.

3.3.3. Evolución de habilidad

Ahora analizamos un escenario en el que un jugador nuevo se integra a una comunidad grande de jugadores ya conocidos. En este ejemplo queremos ver cuán cercana es la estimación de la verdadera habilidad. Para eso establecemos que la habilidad del jugador objetivo cambia en el tiempo siguiendo una función logística. La comunidad se genera asegurando que cada oponente tenga una habilidad similar a la del jugador objetivo durante toda su evolución. En el siguiente código generamos la curva de aprendizaje del agente objetivo y 1000 oponentes aleatorios.

```

import math; from numpy.random import normal, seed; seed(99); N = 1000
def skill(experience, middle, maximum, slope):
    return maximum/(1+math.exp(slope*(-experience+middle)))
target = [skill(i, 500, 2, 0.0075) for i in range(N)]
opponents = normal(target, scale=0.5)

```

Código 3.29: Inicialización de la curva de aprendizaje objetivo y la comunidad de oponentes

Aquí incluimos sólo la versión de Python (las versiones de Julia y R se encuentran en el anexo 7.1.1). La lista `target` contiene la habilidad del agente en cada momento: los valores comienzan en cero y crecen suavemente hasta que la habilidad del jugador objetivo llega a dos. La lista `opponents` incluye los oponentes generados mediante una distribución gaussiana centrada en cada una de las habilidades del jugador objetivo con un desvío estándar de 0,5.

```

composition = [[[["a"], [str(i)]] for i in range(N)]
results = [[1,0] if normal(target[i]) > normal(opponents[i]) else [0,1] for i in range(N)]
times = [i for i in range(N)]
priors = dict([(str(i), Player(Gaussian(opponents[i], 0.2))) for i in range(N)])
h = History(composition, results, times, priors, gamma=0.015)
h.convergence()
mu = [tp[1].mu for tp in h.learning_curves()["a"]]

```

Código 3.30: Estimando la curva de aprendizaje simulada a partir de los resultados aleatorios

En este código definimos cuatro variables para instanciar la clase `History` con el objetivo de obtener la curva de aprendizaje del agente. La variable `composition` contiene 1000 partidas entre el jugador objetivo y los distintos oponentes. Los resultados de la lista `results` se obtienen de forma aleatoria, muestreando los desempeños de los jugadores a través de distribuciones gaussianas centradas en sus habilidades, siendo ganador el jugador con mayor rendimiento. La variable `tiempo` es una lista de valores enteros que van de 0 a 999 y que representan el lote temporal en el que está ubicada cada partida: la clase `History` utiliza la distancia temporal entre eventos para determinar la cantidad de incertidumbre dinámica (γ^2) que se añade entre partidas. La variable `priors` es un diccionario usado para personalizar los atributos de los jugadores: asignamos baja incertidumbre a los priors de los oponentes, simulando que conocemos sus habilidades de antemano.

La clase `History` recibe estos cuatro parámetros e inicializa al agente objetivo con los valores por defecto y una incertidumbre dinámica $\text{gamma}=0.015$. Usando el método `convergence()` obtenemos las estimaciones de TrueSkill Through Time y la curva de aprendizaje del agente objetivo. Debido a que las estimaciones dependen de los resultados aleatorios, repetimos la ejecución del código 3.30 para tener en cuenta su variabilidad. La figura 3.10 muestra la evolución de la curva de aprendizaje real (línea sólida) y estimada (línea punteada) del jugador objetivo. En la figura 3.10(a) vemos que las medias de las curvas de aprendizaje estimada se

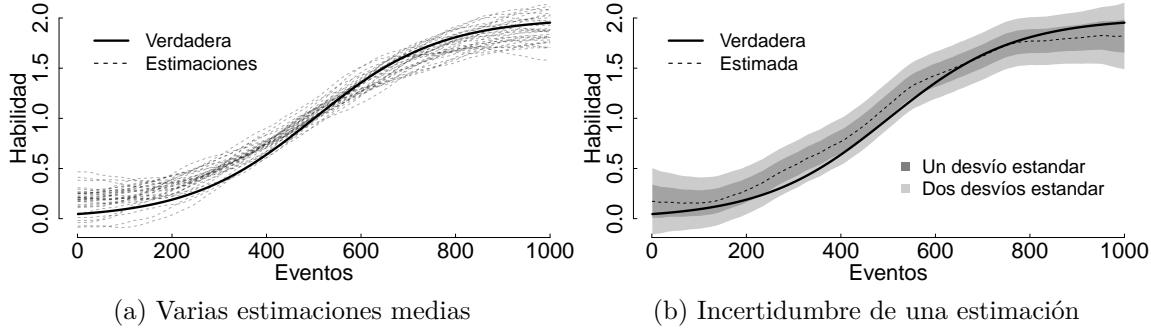


Figura 3.10: Curva de aprendizaje sintética y estimada de un jugador nuevo que se incorpora a una comunidad grande de jugadores ya conocidos. La línea continua representa la habilidad real del jugador objetivo, mientras que la línea discontinua muestra la estimación media. Las zonas oscuras y claras ilustran una y dos veces la incertidumbre de la estimación.

mantienen siempre cerca de la habilidad real durante toda la evolución. En la figura 3.10(b) vemos que la curva real está siempre contenida en el intervalo de incertidumbre de cualquiera de las estimaciones. Este ejemplo evidencia que TrueSkill Through Time es capaz de seguir la evolución de las habilidades de los jugadores nuevos.

3.3.4. Historia de la Asociación de Tenistas Profesionales (ATP)

En este último ejemplo utilizamos toda la historia de partidas de la Asociación de Tenistas Profesionales (ATP). La base de datos cuenta con 447 000 mil partidas que van desde el año 1915 hasta el 2020, en las que participan 19 000 jugadores. La información resumida en un archivo `csv`¹ contiene tanto partidas simples como dobles: si la columna `double` tiene la letra `t` la partida es doble. Cada partida tiene un identificador (`match_id`) y su número de ronda (`round_number`), donde 0 representa la partida final, 1 la semi-final, etc. El archivo también contiene los identificadores y los nombres de los jugadores: por ejemplo `w2_id` es el identificador del segundo jugador del equipo ganador y `11_name` es el nombre del primer jugador del equipo perdedor. Finalmente tenemos el nombre del torneo (`tour_name`), su identificador (`tour_id`), la fecha de inicio del torneo (`time_start`) y el tipo de piso (`ground`). Aquí mostramos sólo el código de `Julia` debido a que es la versión más eficiente (los códigos de `Python` y `R` se pueden encontrar en el anexo 7.1.2, y la comparación de rendimiento se puede encontrar en la sección 3.4).

```
using CSV; using Dates
data = CSV.read("atp.csv")
```

¹ Archivo disponible en https://github.com/glandfried/tennis_atp/releases/download/atp/history.csv.zip. Una fuente de datos actualizada en https://github.com/JeffSackmann/tennis_atp

```

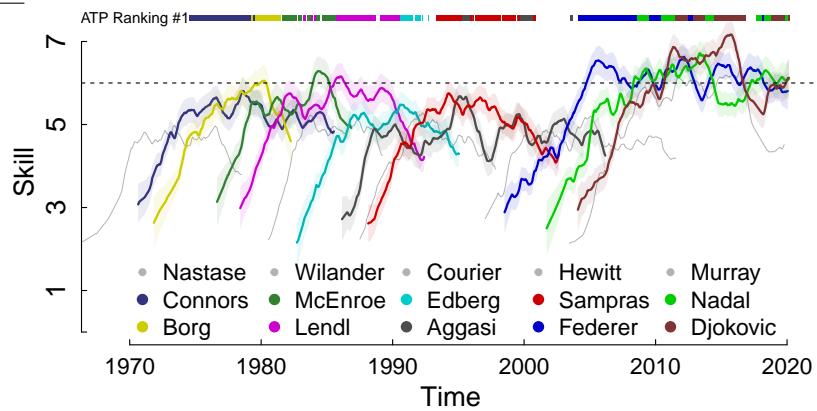
dates = Dates.value.(data[:, "time_start"] .- Date("1900-1-1"))
matches = [ r.double == "t" ? [[r.w1_id,r.w2_id],[r.l1_id,r.l2_id]] : [[r.w1_id],[r.l1_id]]
            for r in eachrow(data) ]
h = History(composition = matches, times = dates, sigma = 1.6, gamma = 0.036)
convergence(h, epsilon = 0.01, iterations = 10)

```

Código 3.31: La historia de la Asociación de Tenistas Profesionales

En este código abrimos el archivo csv, creamos las variables `times` y `composition` e instanciamos la clase `History`. Los tiempos de los eventos se definen como los días transcurridos desde una fecha de referencia hasta el inicio del torneo, asumiendo que la habilidad es la misma al interior de cada torneo. Al generar la lista `composition` discriminamos si los juegos son dobles o individuales en base a la columna `double`. Los resultados se determinan por el orden de la composición, poniendo siempre primero al equipo ganador. Al inicializar la clase `History` modificamos los valores de `sigma` y `gamma` basados en un procedimiento de optimización realizado previamente (figura 3.9(b)). Finalmente, usamos el método `convergence()` para obtener las estimaciones de TrueSkill Through Time, indicando explícitamente el criterio de corte: cuando el cambio entre iteraciones sea menor a 0,01, o cuando se alcancen las 10 iteraciones. La tabla 3.11(a) presenta el ranking histórico de acuerdo a la cantidad de semanas

Pos.	Player	Weeks
1	Novak Djokovic	320
2	Roger Federer	310
3	Pete Sampras	286
4	Ivan Lendl	270
5	Jimmy Connors	268
6	Rafael Nadal	209
7	John McEnroe	170
8	Björn Borg	109
9	Andre Agassi	101
10	Lleyton Hewitt	80
11	Stefan Edberg	72
12	Jim Courier	58
13	Gustavo Kuerten	43
14	Andy Murray	41
15	Ilie Năstase	40
16	Mats Wilander	20



(a) Semanas como No. 1 (b) Curvas de aprendizaje estimadas de algunos jugadores masculinos.

Figura 3.11: (a) Ranking histórico de acuerdo a la cantidad de semanas que los jugadores alcanzaron la primera posición del ranking ATP hasta el 10 de marzo del 2020. (b) Habilidad estimada de algunos de los líderes históricos de la ATP. El área sombreada representa un desvío estándar de incertidumbre. La barra superior indica cual jugador estaba en el primer puesto del ranking ATP. La línea punteada, ubicada a los seis puntos de habilidad, se incluye para ayudar a comparar las curvas.

que los jugadores alcanzaron la primera posición del ranking ATP hasta el día de confección del presente documento. En la figura 3.11(b) graficamos las curvas de aprendizaje de algunos jugadores famosos de la historia de la ATP, que identificamos usando diferentes colores. La barra superior indica cual jugador estaba en el primer puesto del ranking ATP (la barra no tiene color cuando el jugador número 1 no está incluido en nuestro análisis).

Los puntos de la clasificación de la ATP se actualizan semanalmente en función del prestigio del torneo y la fase alcanzada. Sólo durante cortos períodos de tiempo no hay concordancia entre las estimaciones de habilidad y quién está en el primer puesto del ranking ATP, mostrando una buena acuerdo entre ambas metodologías. Sin embargo, hay algunas diferencias

notables con respecto al ranking histórico que se muestra en la tabla 3.11(a). La posición que obtiene Lleyton Hewitt en el ranking histórico de la ATP es producto de una ventana de oportunidad abierta cerca del año 2000, pues su habilidad es relativamente baja en términos históricos. Por otro lado, Andy Murray, siendo el cuarto jugador más hábil, sólo ocupa el puesto 14 en la clasificación histórica de la ATP, sólo un lugar por encima de Ilie Năstase.

TrueSkill Through Time permite comparar la habilidad relativa de los jugadores en el tiempo, a diferencia del ranking histórico de la ATP y de los estimadores basados en el enfoque de filtrado (como TrueSkill, Glicko e IRT). Estas curvas de aprendizaje comparten un patrón similar: comienzan con un crecimiento rápido, alcanzan una meseta inestable, y terminan con una caída lenta (por razones de visualización hemos ocultado la última porción de los jugadores que tienen etapas finales largas). Las curvas de aprendizaje individuales permiten reconocer los períodos de crisis, estabilidad y éxito de los jugadores, e incluso los efectos de los bajones emocionales como los que sufrieron Agassi y Djokovic. Cabe destacar que la habilidad de los tenistas no aumentó bruscamente a lo largo de los años: en contra de lo que cabría esperar, los jugadores de la década de 1980 eran más hábiles que los de la década de 1990, y alcanzaron una habilidad similar a la que tenían Federer, Nadal y Djokovic en 2020, aunque estos últimos alcanzaron valores más altos durante más tiempo.

3.3.5. Habilidades multidimensionales

En el ejemplo anterior resumimos la habilidad de los jugadores en una única dimensión. TrueSkill Through Time permite estimar habilidades multi-dimensionales. Sabemos que la habilidad de los jugadores de tenis puede variar significativamente según el tipo de suelo. Para cuantificar este fenómeno, proponemos modelar cada jugador como un equipo compuesto por un jugador genérico, que se incluye en todos los eventos, y otro jugador que representa la habilidad del jugador en una superficie particular. Por ejemplo, Nadal se representará como un equipo de dos jugadores: *Nadal_generic* y *Nadal_clay* cuando juegue en este tipo de superficie, y *Nadal_generic* y *Nadal_grass* cuando participe en el torneo de Wimbledon.

```
players = Set(vcat((composition...)))
priors = Dict([(p, Player(Gaussian(0., 1.6), 1.0, 0.036) ) for p in players])
composition_ground = [ r.double == "t" ? [[r.w1_id, r.w1_id*r.ground, r.w2_id,
    r.w2_id*r.ground], [r.l1_id, r.l1_id*r.ground, r.l2_id, r.l2_id*r.ground]] : [[r.w1_id,
    r.w1_id*r.ground], [r.l1_id, r.l1_id*r.ground]] for r in eachrow(data) ]
h_ground = History(composition = composition_ground, times = dates, sigma = 1.0, gamma =
    0.01, beta = 0.0, priors = priors)
convergence(h_ground, epsilon = 0.01, iterations=10)
```

Código 3.32: Modelando habilidades multi-dimensionales en la historia de la ATP

En este ejemplo mantenemos el mismo prior para todos los jugadores genéricos, pero en esta ocasión los definimos usando la variable **priors**. Al igual que en el código anterior (3.31), creamos los equipos en función de si la partida es doble o simple, pero ahora añadiendo las habilidades específicas de cada jugador como su compañero de equipo (usamos el operador ***** para concatenar strings). Como las habilidades específicas no están definidas en el diccionario **prior**, ellas van a ser inicializadas con los valores por defecto que le indicamos a la clase **History**. Para ellos elegimos un **beta** nulo de modo tal de no agregar más ruido al rendimiento de los jugadores, manteniendo estable la escala de las estimaciones. Elegimos un **sigma** que consideramos suficientemente grande y una factor dinámico **gamma** que represente el 1% de la incertidumbre a priori. En la figura 3.12 mostramos la diferencia de habilidad que Nadal y

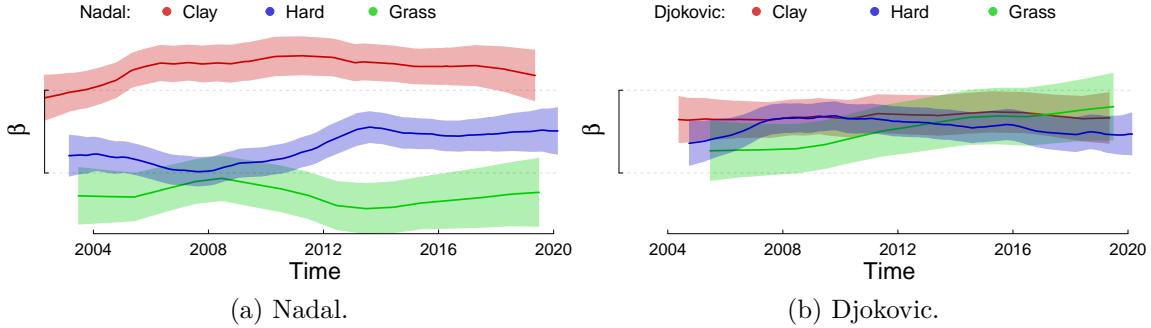


Figura 3.12: Diferencia de habilidad en los tres tipos de suelo principales. Cada punto del eje y representa una β de distancia, es decir 76 % de probabilidad de ganar.

Djokovic tienen entre los tres tipos de piso. Podemos ver que Nadal tiene una diferencia de habilidad grande entre suelos, a diferencia de Djokovic que tiene habilidades muy similares en los tres tipos de suelo. La diferencia de habilidad de Nadal entre piso de polvo de ladrillo y pasto es mayor a un β , lo que significa una diferencia de al menos 76 % de probabilidad de ganar si lo comparamos consigo mismo. En el caso de Nadal parece importante modelar la multi-dimensionalidad de la habilidad, mientras que en el caso de Djokovic parece razonable resumir la habilidad en una única dimensión. Para evaluar si la complejidad que se agrega al modelar la multidimensional conviene en términos generales, podemos comparar la predicción a priori conjunta de los modelos, llamando al método `log_evidence()` de la clase `History`. En tenis es suficiente resumir las habilidades en una única dimensión, dado que la predicción a priori se maximiza cuando los parámetros de los factores de piso, σ y γ , se desvanecen. En otros ejemplos, donde la multidimensionalidad de las habilidades sea más relevante, convendrá separar la habilidad de todos los agentes en diferentes componentes. Si consideramos sólo las partidas en las que participa Nadal, la optimización se alcanza cuando los parámetros toman los valores $\sigma = 0,35$ y $\gamma = 0$, indicando que conviene modelar habilidades multidimensionales ($\sigma > 0$) pero considerando que su efecto no cambia en el tiempo ($\gamma = 0$). En este escenario, la habilidad de Nadal en polvo de ladrillo es $0,87\beta$ mayor a la de cemento, y $1,05\beta$ mayor a la de pasto.

3.4. Detalles computacionales

En esta sección reportamos los tiempos de ejecución de los ejemplos presentados en la sección 3.3. Los códigos fuentes de los paquetes TrueSkill Through Time se encuentran en:

- github.com/glandfried/TrueSkillThroughTime.jl (Julia)
- github.com/glandfried/TrueSkillThroughTime.py (Python)
- github.com/glandfried/TrueSkillThroughTime.R (R)

El análisis fue realizado en dos computadoras representativas de uso domiciliario. Los tiempos en Julia, Python y R fueron analizados con `@time`, `timeit` y `microbenchmark` respectivamente. **Evento único:** En la sección 3.3.1 presentamos dos ejemplos: en el código 3.22 y 3.25 presentamos una partida con dos y tres equipos respectivamente. Aquí evaluamos la

eficiencia de la inicialización de la clase `Game` y el método `posterior()`. En el paquete básico de TrueSkill estos dos pasos los resuelve la función `rate()`. La tabla 3.2(a) muestra los tiempos de ejecución. Nuestros paquetes de `Python` y `Julia` son 10 y 20 veces más rápido que

Versión	Tiempo	Mejora
<code>trueskill 0.4.5</code>	1,45 ms	1,0X
<code>R 3.4.4</code>	4,35 ms	0,33X
<code>Python 3.6.9</code>	0,14 ms	10,4X
<code>Julia 1.5.0</code>	0,064 ms	22,7X

(a) : Two-teams game

Versión	Tiempo	Mejora
<code>trueskill 0.4.5</code>	2,45 ms	1,0X
<code>R 3.4.4</code>	31,3 ms	0,078X
<code>Python 3.6.9</code>	0,93 ms	2,63X
<code>Julia 1.5.0</code>	0,096 ms	25,5X

(b) : Three-teams game

Tabla 3.2: Tiempos de ejecución de la inicialización de la clase `Game` y del método `posterior()` para la partida de dos equipos, y la mejora es respecto al paquete original `trueskill 0.4.5` de `Python`.

el paquete de `trueskill` original, mientras que el paquete de `R` es tres veces más lento. La tabla 3.2(b) presenta los tiempos de ejecución del evento con tres equipos. Siempre que hay más de dos equipos en la partida es necesario ejecutar un algoritmo iterativo que aumenta los tiempos requeridos para computar los posteriors. Nuevamente, nuestros paquetes de `Python` y `Julia` son más rápido que el paquete de `trueskill` original, mientras que el paquete de `R` es más lento. **Secuencia de eventos:** En la sección 3.3.2 presentamos la inicialización de la clase `History` y el método `convergence()` en los códigos 3.26 y 3.28 respectivamente. Si bien la inicialización de la clase `History` calcula las estimaciones para las cuales está pensado el paquete original `trueskill 0.4.5`, no hacer más comparaciones debido a que este paquete no cuenta con una función para resolver las secuencias de eventos automáticamente. En la tabla 3.3 presentamos los tiempos de ejecución para inicializar la clase `History` y realizar una única iteración del método `convergence()`. La inicialización de la clase `History` incluye la

Versión	<code>History</code>	<code>convergence()</code>
<code>R 3.4.4</code>	74,4 ms	46,8 ms
<code>Python 3.6.9</code>	1,09 ms	1,88 ms
<code>Julia 1.5.0</code>	0,31 ms	0,58 ms

Tabla 3.3: Tres eventos de dos equipos: tiempos de ejecución de la inicialización de la clase `History` y una única iteración del método `convergence()`.

creación de la red bayesiana y una pasada por todas las partidas, mientras que la convergencia realiza dos pasadas por la secuencia de eventos, una para atrás y otra para adelante. La inicialización en el paquete `Python` es todavía más rápida que el cómputo de un único evento aislado mediante el paquete original `trueskill`. El paquete de `Julia` aumenta la velocidad respecto de nuestro paquete `Python`. Y el paquete de `R`, si bien mantiene la diferencia de tiempo respecto de `Python` durante la convergencia, tiene una demora adicional durante la inicialización debida a la creación de “clases por referencia”. **Evolución de habilidad:** En la sección 3.3.3 analizamos la evolución de habilidad de un jugador que juega contra 1000 oponentes distintos. Los tiempos de ejecución para inicializar la clase `History` y para una única iteración de convergencia fueron, En este caso el paquete de `Julia` es diez veces más rápido que el paquete de `Python`. Esto sugiere que la cantidad de trabajo mejora la eficiencia

Lenguaje	History	convergence()
R 3.4.4	31 000 ms	26 000 ms
Python 3.6.9	380,2 ms	876,3 ms
Julia 1.5.0	38,1 ms	75,7 ms

Tabla 3.4: Inicialización de la clase `History` y una única iteración del método `convergencia()` en una secuencia de 1000 partidas que un jugador tiene contra oponentes distintos.

relativa del paquete de `Julia`. La versión `R` no aumenta su tiempo de ejecución durante la convergencia, lo que pone de manifiesto el impacto que tiene la inicialización de la clase en este lenguaje. **Historia de la ATP:** En la sección 3.3.4 analizamos una base de datos real con 447 000 partidas. En este escenario realista, realizamos el análisis utilizando una estación de trabajo adicional para mostrar el impacto de diferentes hardware en el tiempo de ejecución. Incluimos la misma computadora que antes (llamada **A** ahora), y una nueva computadora con (llamada **B**) con 16 GB de RAM y un procesador Intel (i5-3330, 3,00 GHz, cache total 6144 kB). La table 3.5 presenta los tiempos de ejecución totales de inicialización de la clase `History` y del método `convergence()`. La inicialización más las diez iteraciones de convergen-

Plataforma	Versión	Tiempo
A	Python 3.6.9	4498,8 s
B	Python 3.6.8	1368,6 s
A	Julia 1.5.0	387,5 s
B	Julia 1.5.3	138,5 s

Tabla 3.5: Inicialización de la clase `History` y diez iteraciones del método `convergence()` del ejemplo ATP.

cia evalúan en total 447000×21 eventos: una pasada sobre todas las partidas correspondiente al computar la estimaciones de TrueSkill durante la inicialización, y dos pasadas sobre todas las partidas por cada iteración (backward y forward). Dados los tiempos de ejecución de los eventos reportados en la tabla 3.2(a), el tiempo de ejecución mínimo que se necesitaría para computar las estimaciones TrueSkill Through Time mediante el paquete original `trueskill` 0.4.5 no podría ser menor a 13 611 s en la computadora A, tres veces más de lo que tarda nuestro paquete `Python` para realizar la totalidad del cómputo. Nuestro paquete de `Julia` demora menos de lo esperado considerando los tiempos reportados en la tabla 3.2(a) debido a optimizaciones que automáticamente realiza `Julia`.

3.5. Conclusiones y discusión

La principal contribución es la implementación de uno de los modelos más importantes de la industria de los video juegos, que aún no estaba disponible en los lenguajes de programación con mayores comunidades, como `Julia`, `Python` y `R`.

Que un método tan atractivo como éste haya estado ausente hasta ahora en estos lenguajes de programación creemos que ha sido la consecuencia de la falta de documentación que permitiera a los desarrolladores de software entender todos los aspectos teóricos que se requieren para su implementación. Esta información no está presente ni en los artículos originales Herbrich et al. [58] y Dangauthier et al. [103] ni en reportes técnicos informales famosos como

el desarrollado por Mosser [105]. Por este motivo, hemos dedicado la sección metodología para documentar todos los argumentos teóricos paso a paso, de modo que sea comprensible para cualquier persona con conocimientos básicos de probabilidad. Pero además, como la estimación de habilidades es un tema muy sensible para muchas personas, hemos dedicado la sección introductoria a explicar de la forma más accesible posible al público en general los detalles de este modelo de estimación de habilidades, contextualizando la importancia y las ventajas del modelo de habilidades que proponemos.

La mayoría de los servidores de juegos en línea y artículos científicos siguen utilizando algunas estimaciones de habilidad basadas en un enfoque de filtrado. Estos modelos comparten un problema común: ellos propagan la información histórica en una sola dirección a través del sistema, del pasado al futuro. Esta estrategia, además de producir malas estimaciones iniciales de habilidad, impide comparar estimaciones que están alejadas en el tiempo y el espacio.

La aplicación estricta de las reglas de la probabilidad nos obliga a realizar la inferencia usando toda la información disponible, incluso la que se encuentra en el futuro, algo que intuitivamente hacemos cuando concluimos que los deportistas famosos también eran hábiles un tiempo antes de hacerse famosos. Por este motivo el modelo TTT, al propagar toda la información histórica a través de una única red causal, proporciona estimaciones con baja incertidumbre en todo la serie temporal, permitiendo estimaciones iniciales de habilidad fiables y asegurando su comparación histórica. De esta forma, sus predicciones a priori son varios órdenes de magnitud más precisas que las obtenidas con los modelos mencionados (Tabla 3.1). Con no más de tres hiperparámetros intuitivos (incertidumbre a priori, incertidumbre dinámica y probabilidad de empate), el modelo TTT logra resultados similares y hasta mejores que modelos más complejos como KickScore.

Propagar la información histórica correctamente por toda la red causal de eventos produce una mutua dependencia entre las estimaciones que obliga a implementar un algoritmo iterativo. Gracias a los métodos de aproximación analítica y a los algoritmos de pasaje de mensajes, el modelo TTT puede resolverse eficazmente con cualquier ordenador de gama baja, incluso en redes causales con millones de nodos y estructuras irregulares. Hemos mostrado que este procedimiento requiere pocas iteraciones para converger incluso en bases de datos grandes como la de la Asociación de Tennis Profesional. En la sección *Detalles computacionales* estudiamos los tiempos de ejecución de los tres paquetes que presentamos en este trabajo. Nuestro paquete de **Python** resuelve los eventos individuales diez veces más rápido que el paquete original **trueskill 0.4.5** [106]. A su vez, nuestro paquete de **Julia** resuelve las secuencias de eventos diez veces más rápido que nuestro paquete de **Python**. Por el contrario, nuestro paquete de **R** es varias veces más lento que el paquete básico **trueskill 0.4.5**.

Esperamos que esta primera implementación abierta en **Julia**, **Python** y **R**, de uno de los modelos de estimación de habilidad más importantes, al estar acompañada además por una explicación científica en profundidad, ofrezca las condiciones para generar una proceso colectivo continuo de mejora. Por ejemplo, la implementación actual no cuenta todavía con la extensión realizada por Guo et al. [107], en el que los datos observables ya no son los valores ordinales de los resultados (perder/empatar/ganar), sino la cantidad de puntos que obtuvo cada equipo. Una de las dificultades más grandes del modelo actual es distinguir la habilidad de los individuos de un mismo equipo usando los resultados del equipo como único observable. Esto podría resolverse incorporando otras variables como observables, específicas a los individuos. Esta es la estrategia utilizada por el modelo TrueSkill 2 [6], que desarrolla un modelo particular para los llamados “shooters” en línea el número de logros individuales, su tendencia a abandonar y su habilidad en otros modos de juego. De hecho, TrueSkill 2

ha sido desarrollado como una extensión del modelo TrueSkill Through Time, que mejora significativamente su rendimiento predictivo. Ahora que ofrecemos la primera implementación de TrueSkill Through Time en [Julia](#), [Python](#), [R](#), el desafío es extenderlo, desarrollando un software con la mayor flexibilidad posible, que permita adaptarse a la mayor variedad de situaciones posibles.

Capítulo 4

Resultado 3. Efecto en el aprendizaje de la posición en la red de intercambios

Los seres humanos crecen y se desarrollan en entornos culturales dinámicos, haciendo que el aprendizaje humano dependa en gran medida de las interacciones sociales y la transmisión cultural. Aunque está ampliamente aceptado que la estructura de esas “redes culturales” afecta los procesos de aprendizaje social y acumulación cultural, estudios recientes han cuestionado la hipótesis de que todo lo que maximice el flujo de información cultural debería repercutir positivamente en el aprendizaje. La razón por la que resulta difícil para los estudios antropológicos evaluar estas hipótesis es la falta de datos granulares sobre la interacción de todos los individuos de una comunidad humana a lo largo del tiempo. En este capítulo estudiamos la estructura de interacciones y la evolución de las habilidades de todos los individuos de una de las comunidades virtuales de Go más importantes, a lo largo de ocho años. Dado que la tasa de aprendizaje depende de la etapa en la que se encuentran los individuos al inicio del proceso, usamos el modelo de habilidad del estado del arte para obtener estimaciones precisas a lo largo de toda la serie temporal, permitiendo comparar por separado a individuos novatos, intermedios y expertos. Analizando la posición topológica que ocupan los individuos en la red de interacciones a lo largo del tiempo mediante técnicas clásicas de grafos dinámicos, descubrimos que los valores moderados de centralidad tienen un efecto positivo solo cuando los individuos están en la mitad del proceso de aprendizaje. Por el contrario, los jugadores novatos aprenden principalmente por experiencia individual, y los jugadores expertos no tienen nada más que aprender. Por último, discutimos nuestros resultados en relación con trabajos anteriores.

4.1. Introducción

La integración de los procesos biológicos, cognitivos y sociales que permite a los humanos desarrollar culturas complejas se debió a una coevolución genético-cultural desencadenada por el surgimiento previo de la crianza cooperativa [14, 108]. El surgimiento de la comprensión mutua produjo cambios radicales para nuestra especie. Esas cadenas de transmisión de información producen innovaciones más complejas de lo que cualquier persona podría crear individualmente [109]. Emergió así un nuevo sistema de información, autónomo del sistema

de información biológico, lo que nos permitió ocupar todos los nichos ecológicos de la tierra como ningún otro vertebrado terrestre había logrado antes [4].

La cultura es un fenómeno poblacional que surge como consecuencia del intercambio de información entre individuos. Todos los procesos de aprendizaje social modifican en cierta medida la variante cultural adoptada. La descendencia, la mutación y la selección de la información cultural se producen cuando un individuo adopta el comportamiento de otros, ya sea porque es el comportamiento de la mayoría (estrategia basada en la frecuencia), porque el otro individuo es especialmente exitoso o prestigioso (estrategia basada en la retribución) o en función de diferentes sesgos de selección [75, 81]. En este proceso, algunas de las variantes culturales se extienden más rápidamente o son más estables que otras, lo que afecta al proceso general de adquisición de habilidades y acumulación cultural.

La estructura de estas “redes culturales” permite o impide el acceso a la información cultural, afectando entonces los procesos de aprendizaje social y acumulación cultural [5]. Aunque se ha demostrado que la capacidad de acumulación cultural es proporcional al tamaño efectivo de la población [16, 110, 111], estudios recientes han cuestionado la suposición de que todo lo que maximice el flujo de información cultural debería repercutir positivamente [5]. La hipótesis afirma que las redes totalmente conectadas producen una homogeneización cultural que reduce las posibilidades de innovación, mientras que las redes fragmentadas tienden a perder variantes culturales complejas con el tiempo [27, 28]. La razón por la que es difícil evaluar estas hipótesis en datos del mundo real es que debemos registrar todas las interacciones al interior de una comunidad a nivel individual, lo que suele ser difícil [5]. Existen algunos estudios, sin embargo, que demuestran que las sociedades humanas suelen desarrollar redes parcialmente conectadas, lo que maximizaría la adaptabilidad al garantizar tanto contactos regulares con grupos externos como el mantenimiento de la diferenciación local [26, 29].

Para responder cómo la estructura de la red de intercambio afecta los procesos de aprendizaje social a nivel individual, decidimos estudiar en detalle la evolución de una de las mayores comunidades de jugadores de Go en línea (OGS <https://online-go.com>) durante ocho años. El Go es un juego de mesa para dos jugadores creado en China hace aproximadamente 3000 años. Aunque las reglas son sencillas, el Go presenta una enorme complejidad en comparación con el ajedrez: fue en 1996 cuando el primer algoritmo de inteligencia artificial venció a uno de los mejores jugadores de ajedrez [112], mientras que en el juego del Go este objetivo se consiguió 20 años después [113]. Aprender a jugar al Go no solo implica adquirir una tradición cultural milenaria y un conjunto de estrategias complejas acumuladas de generación en generación, sino también adaptarse a las innovaciones que surgen cada día.

Aunque nuestra base de datos cuenta con más de 7 millones de jugadores y casi 15 millones de partidas entre noviembre de 2005 y julio de 2018, hemos decidido trabajar exclusivamente con los ocho primeros años (con 446 000 jugadores y 818 000 partidas) por los siguientes motivos:

1. Pudimos comprobar a través de la información proporcionada por la plataforma OGS, que no hay *bots* en ese período.
2. A finales de 2013, se produce una fusión entre OGS y Nova, otro portal de Go, que produce un cambio repentino en la estructura de la red.

El juego Go ya ha permitido estudiar cómo las variantes culturales particulares son impulsadas por una mezcla de procesos de aprendizaje individuales y sociales [86].

Dado que los individuos aprenden principalmente cuando están directamente implicados en tareas de resolución de problemas en los que se ven comprometidos personalmente [12], nos enfocamos en la red de partidas, definida como un grafo en el que los nodos representan a los jugadores y los juegos jugados entre ellos son las aristas. Para calcular la posición topológica de los individuos en la red, nos basamos en una metodología clásica y robusta de ventanas deslizantes de análisis de grafos dinámicos [114, 115]. Esta técnica divide la red en pequeñas subredes compuestas únicamente por nodos y vértices activos en un periodo determinado. Luego, cada una de estas subredes se analizan utilizando métodos altamente aceptados en grafos clásicos, con lo que podemos estar seguros acerca de su robustez metodológica.

Según los principales hallazgos de la bibliografía sobre aprendizaje social y acumulación cultural [5], el aprendizaje humano depende de al menos cuatro variables fundamentales: el *entorno*, la *experiencia individual*, la *estructura de red* que da acceso a la información cultural, y la *habilidad inicial*.

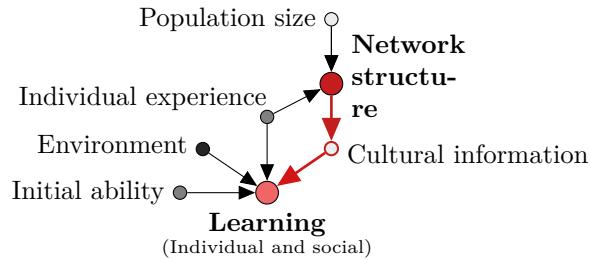


Figura 4.1: Modelo causal. El aprendizaje humano depende de la habilidad inicial, la experiencia individual, la información cultural y el ambiente. Y el acceso a la información cultural depende de la estructura de la red de intercambios. Los colores son los roles de las variables en el análisis estadístico: causa (rojo), efecto (rosa), control (gris), constante (negro), no observables (blanco).

En la figura 4.1 resumimos el modelo causal en términos gráficos, donde las flechas representan relaciones causales entre las variables y los colores representan el rol que las variables cumplen en el análisis estadístico: causa (rojo), efecto (rosa), control (gris), constante (negro), no observable (blanco). Dado que queremos explorar el efecto que la estructura de la red tiene en el aprendizaje humano (camino rojo), la información cultural debe tratarse como una variable oculta [48]. En nuestra base de datos, el ambiente es constante ya que las reglas del juego Go no cambian con el tiempo, y el tamaño de la red permanece relativamente estable durante el periodo estudiado. Entonces, para evaluar el efecto, necesitamos observar cómo cambia la habilidad dada la estructura de la red para diferentes valores de experiencia individual y habilidad inicial. La diferencia en el aprendizaje que observamos entre las distintas estratificaciones basadas en la posición de los individuos en la red, manteniendo fijas todas las demás variables, puede considerarse el efecto del modelo causal propuesto.

Si bien las redes bayesianas han mostrado ser útiles para evaluar el aprendizaje en los sistemas educativos, hace falta más trabajo en el desarrollo de redes bayesianas dinámicas [96]. Por ejemplo, todos los estimadores de habilidad habitualmente utilizados en la industria del video juego y la academia (e.g. Elo, Glicko, Item Response Theory y TrueSkill) no pueden obtener estimaciones iniciales fiables debido a que usan el último posterior como prior del siguiente evento, propagando la información en una sola dirección por el sistema, del pasado al futuro. En cambio, TrueSkill Through Time [103, 116], propaga la información histórica a

través de toda la red, proporcionando estimaciones con baja incertidumbre en cualquier momento, ofreciendo estimaciones iniciales de habilidad fiables y garantizando la comparabilidad de las estimaciones distantes en el tiempo y el espacio. Como este modelo no estaba disponible en los lenguajes de programación más populares, lo implementamos en **Julia**, **Python** y **R** [116].

Conocer las habilidades iniciales de los individuos nos permite comparar curvas de aprendizaje que se encuentran en el mismo estadio. La diferencia de aprendizaje que observemos entre diferentes estratificaciones basadas en la posición de los individuos en la red, manteniendo fijas el resto de las variables, puede ser considerado el efecto causal para el modelo causal propuesto. Esto nos permitiría detectar el efecto de la red de intercambio de información sobre el proceso de aprendizaje individual dado el modelo causal propuesto. Hasta donde sabemos, éste es el primer trabajo que explora el efecto que la posición en una red dinámica tiene sobre el aprendizaje individual. Nuestra base de datos está disponible para que otros grupos de investigación evalúen modelos causales alternativos.

4.2. Materiales y métodos

Esta sección está dividida en cuatro partes: describimos el conjunto de datos; explicamos intuitivamente el modelo de estimación de habilidad; elaboramos la metodología para estudiar los grafos en el tiempo; detallamos las medidas específicas que utilizamos para el análisis.

4.2.1. Base de datos

En este capítulo usamos métodos no experimentales, realizando análisis sobre datos de archivo, provistos por la plataforma de juego de Go en línea OGS. El conjunto de datos se ha hecho público hace varios años, y se encuentra disponibles para realizar trabajos de investigación científica. Para preservar el anonimato de los participantes, en ningún caso estudiamos el comportamiento de los individuos, sino el comportamiento agregado en conjuntos de partidas.

El servidor de OGS se encuentra en los Estados Unidos y actualmente la base de datos pública cuenta con decenas de millones de partidas. La plataforma OGS es, junto a KGS, la más utilizada para jugar al Go en línea en el hemisferio occidental. Si bien la composición de sus participantes por características socio-culturales es desconocida, sabemos que el uso de este tipo de plataformas en línea produce naturalmente sesgos por sexo, edad, nacionalidad, entre otras. Sin embargo, el hecho de que la página oficial tenga la opción de mostrar sus contenidos en casi 30 idiomas indica que los participantes son de distintos países alrededor del mundo.

Para los objetivos de este trabajo decidimos analizar solamente las partidas que se encuentran en el período 2005 al 2013. Esto reduce el conjunto de datos a 446 000 participantes y 818 000 partidas. La decisión se tomó por dos razones. Por un lado, excluimos del análisis el impacto que la fusión entre las plataformas OGS y Nova tuvo sobre la base de datos, la que se produjo a fines del 2013. Por otro lado, restringiendo el análisis al período previo a 2013 garantizamos que la base de datos estuviera libre de *bots*, algo que pudimos constatar a través de consultas que le hicimos a la API que proveé OGS para interactuar con la base de datos. Finalmente, si bien analizamos la base de datos completa, decidimos enfocar el estudio sobre los individuos que jugaron al menos 150 partidas, para estudiar curvas de aprendizaje suficientemente largas. Debido a que la cantidad de partidas jugadas por persona sigue una

distribución exponencial, este filtro basado en la actividad produce un conjunto de individuos de tamaño cercano a 1000.

4.2.2. Modelo de estimación de habilidad

Los estimadores de habilidad habitualmente utilizados en la industria del video juego y la academia (e.g. Elo, Glicko, Item Response Theory y TrueSkill) no permiten obtener estimaciones iniciales fiables debido a que usan el último posterior como prior del siguiente evento, propagando la información en una sola dirección por el sistema, del pasado al futuro. En cambio, TrueSkill Through Time [103, 116], propaga la información histórica a través de toda la red, proporcionando estimaciones con baja incertidumbre en cualquier momento, ofreciendo estimaciones iniciales de habilidad fiables y garantizando la comparabilidad de las estimaciones distantes en el tiempo y el espacio.

El modelo causal común a todos los estimadores supone que los resultados observados de las partidas son una consecuencia directa de las habilidades ocultas (figura 4.2 izquierda). Según este modelo, gana la persona con mayor desempeño, $r = (p_i > p_j)$, siendo los desem-

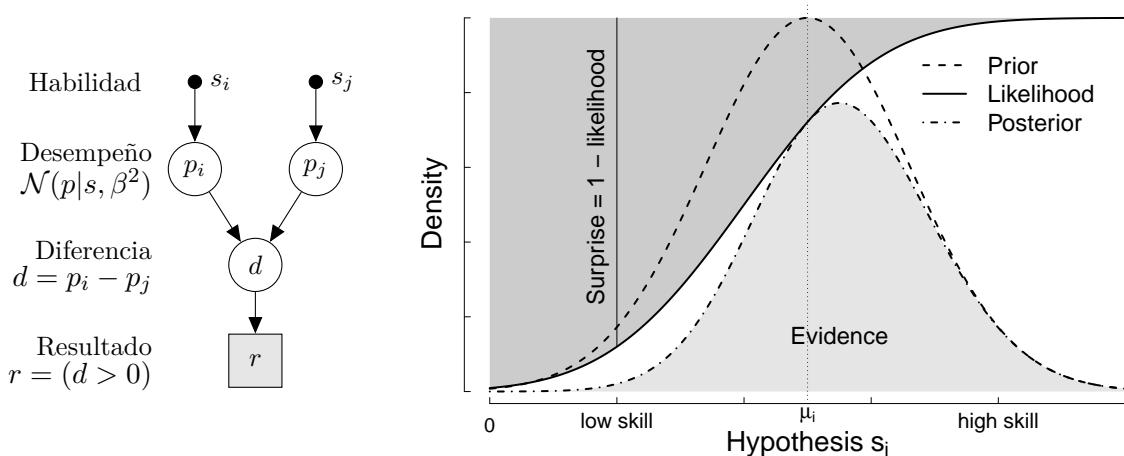


Figura 4.2: (Izquierda) Modelo generativo en el que las habilidades causan los resultados observables a través de la diferencia de rendimientos ocultos. Las variables observables se pintan de gris, las ocultas en blanco, y las constantes se muestran como puntos negros. (Derecha) Actualización de creencias para un caso ganador. El posterior proporcional se obtiene como el producto de la distribución a priori y la verosimilitud.

peños variables aleatorias centradas en las verdaderas habilidades, $\mathcal{N}(p|s, \beta^2)$. La escala de las estimaciones está determinada por el parámetro β : habilidades a distancia de un β tienen 76 % de probabilidad de ganar. Aquí usamos $\beta = 1$.

En la figura 4.2 derecha mostramos cómo se actualiza un prior gaussiano luego de que observamos a un jugador ganar una partida. En la región de hipótesis de muy alta habilidad, donde el resultado ganador no nos hubiera generado casi ninguna sorpresa ($\lim_{s_i \rightarrow \infty} P(r|s_i) = 1$), el posterior recibe casi toda la densidad del prior. En cambio, en la región de hipótesis de muy baja habilidad, ($\lim_{s_i \rightarrow -\infty} P(r|s_i) = 0$), posterior no recibe casi nada de la densidad del prior. Debido a que las habilidades cambian en el tiempo, debemos agregar alguna incertidumbre luego de cada paso temporal, $\mathcal{N}(s_t|s_{t_1}, \gamma^2)$.

Para analizar el aprendizaje en el tiempo, la mayoría de los modelos utilizan el último

posterior como prior del siguiente evento. Este enfoque ad hoc, sin embargo, tiene varios problemas que el modelo TrueSkill Through Time resuelve aplicando estrictamente las reglas de la probabilidad, lo que obliga a propagar toda la información histórica a través de la red. En la figura 4.3 mostramos las estimaciones obtenidas con TrueSkill (rojo) y con TrueSkill Through Time (verde) luego de haber visto a dos jugadores competir dos veces. Las esti-

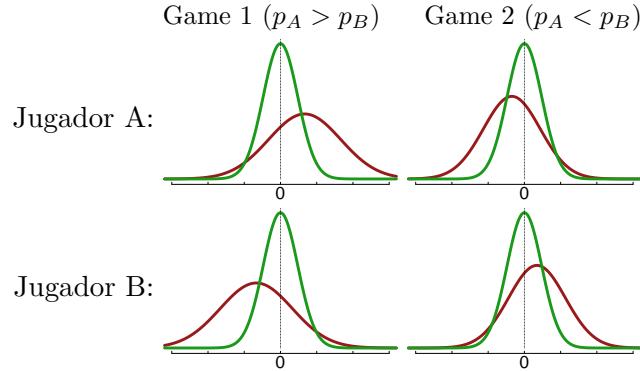


Figura 4.3: Estimaciones obtenidas con TrueSkill (rojo) y con TrueSkill Through Time (verde) luego de haber visto a dos jugadores competir dos veces: la primera gana el jugador A y en la segunda el jugador B.

maciones de TrueSkill Through Time son correctas de acuerdo a lo que sugieren los datos (una victoria cada uno), teniendo ambos jugadores la misma habilidad (centradas en 0). Los estimadores de habilidad usados comúnmente, en cambio, ofrecen estimaciones sesgadas y con mayor incertidumbre.

4.2.3. Una metodología clásica de análisis de grafos dinámicos

Nos basamos en una metodología clásica y robusta de grafos dinámicos [114, 115] para calcular la posición topológica de los individuos en la red a lo largo del tiempo (donde los jugadores son los nodos y las partidas jugadas entre ellos son las aristas). Esta técnica divide la red en pequeñas subredes compuestas únicamente por nodos y vértices activos en un periodo determinado. A continuación, cada una de estas subredes se analiza utilizando diversas métricas de red.

Para determinar el tamaño de la ventana, Kossinets y Watts [114] estudian cómo cambia el comportamiento de un conjunto de indicadores a través del tiempo. Siguiendo esta metodología, en la figura 4.4 analizamos el grado medio, el tamaño de la componente más grande, el largo promedio de los caminos mínimos, y el coeficiente de clustering, para diferentes tamaños de ventana: una semana, dos semanas, y un mes de largo. Queremos un tamaño de ventana que tenga suficiente granularidad para detectar los pequeños cambios, pero que sea lo suficientemente suave para evitar ruidos locales.

Podemos ver que el grado medio está fuertemente influenciado por el tamaño de la ventana. A medida que aumentamos el tamaño de la ventana aumenta el grado promedio, porque observamos más partidas en promedio por jugador, pero el ruido nunca se reduce. En segundo lugar, la proporción de la componente principal no se ve muy afectada por el tamaño de la ventana, salvo durante los primeros meses cuando todavía había poca actividad en el sitio, y en un breve falla del servidor ocurrida a fines del año 2010. La ventana de dos semanas es capaz de detectar la misma componente conexa que la ventana de un mes con mayor

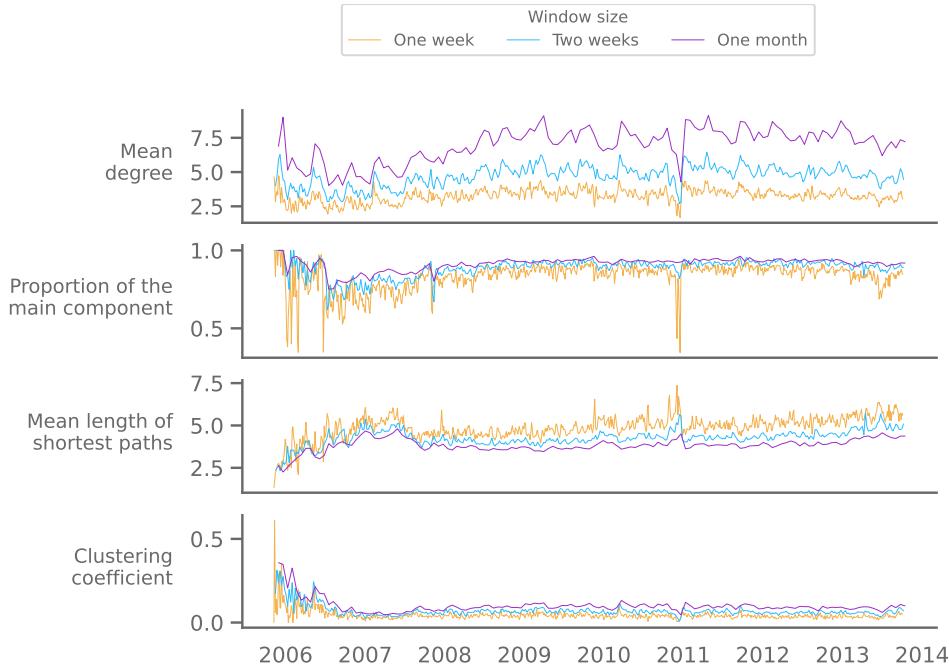


Figura 4.4: Grado medio, tamaño de la componente más grande, largo promedio de los caminos mínimos y el coeficiente de clustering para ventanas de una semana, dos semanas y un mes.

granularidad. En tercer lugar, el largo promedio de los caminos mínimos de la componente más grande muestra que, en general, los jugadores se encuentran a cuatro saltos de distancia. Por último, el coeficiente de clustering muestra que la densidad del grafo (la proporción de triángulos) es baja utilizando para cualquier tamaño de ventana debido a que la plataforma estudiada favorece los vínculos sin necesidad que las personas tengan conocidos en común, evitando así la emergencia de comunidades cerradas.

Decidimos utilizar una ventana de tamaño medio debido a que nos permite evitar ruido sin perder la capacidad de detectar tendencias coyunturales relevantes.

4.2.4. La posición de los individuos en la red

Para describir la posición de los individuos analizamos 6 medidas de centralidad, que pueden ser clasificadas en tres grupos.

- I) Las medidas de popularidad, como *degree* y *eigenvector*, están basadas en la cantidad de ejes que tiene los nodos (y sus vecinos): las personas que conocen a todo el mundo (y sus conocidos también) tienen valores altos de popularidad.
- II) Las medidas de cercanía, como *closeness* y *harmonic*, están basadas en la distancia típica que un nodo tiene hacia el resto: las personas que están cerca de todo el mundo tienen valores altos de cercanía.
- III) Y medidas de intermediación, como *betweenness* o *communicability*, están basadas en la influencia que un nodo tiene para los flujos de información: las personas que conectan varias comunidades separadas tienen valores altos de intermediación.

Las definiciones están basadas en las implementaciones del paquete `Networkx` de `Python` que usamos durante el estudio [117].

Indicadores de popularidad o prestigio

Llamamos indicadores de popularidad a aquellos que están basados en la cantidad de contactos que tienen las personas. Conocer *directamente* a muchas otras puede ser un indicador de su popularidad. Bajo esta idea, la centralidad conocida como *grado* define la popularidad como la fracción de contactos que tiene cada nodo p sobre todos los contactos posibles.

$$\text{degree}(p) = \frac{\text{Contactos efectivos de } p}{\text{Contactos potenciales}}. \quad (4.1)$$

Esta definición de grado no tiene en cuenta el peso de los ejes.

La popularidad (o el prestigio) también puede verse influenciada por la cantidad de contactos *indirectos*, la cantidad de contactos que tienen mis contactos. Este concepto es detectado por la centralidad conocida como *eigenvector*. La solución iterativa ofrece una interpretación intuitiva. En el primer paso, multiplicamos la matriz de adyacencias A por el vector unitario, $A \cdot \mathbf{1} = x'$, y obtenemos la suma de contactos directos de los nodos. En el segundo paso, $A \cdot x' = x''$, obtendremos la suma de los contactos de los vecinos. Y así sucesivamente. En cada paso, sin embargo, se normaliza el vector para que tenga longitud 1, lo que produce un decaimiento en la importancia a medida que nos alejamos del nodo de origen. Una vez que el vector converge, x^* , el valor de eigenvalue del nodo p es su p -ésimo elemento.

$$\text{eigenvalue}(p) = x_p^*. \quad (4.2)$$

Indicadores de cercanía

Llamamos indicadores de cercanía a aquellos que están basados en la distancia que los individuos tienen hacia el resto de los nodos. La medida de cercanía paradigmática es *closeness*, definida como el recíproco de las distancias mínimas promedio que un individuo tiene con todos los otros nodos. Cuando hay más de una componente, sólo se calcula el camino más corto entre nodos de la misma componente, ponderando por la proporción de la componente para no asignar valores de centralidad elevados a nodos pertenecientes a componentes periféricas.

$$\text{closeness}(p) = \text{Proporción de la componente} \cdot \text{promedio invertido}. \quad (4.3)$$

Esta solución es conocida como Wasserman y Faust closeness. Otra forma de resolver el problema es calcular la *harmonic centrality* (la suma de los recíprocos de las distancias mínimas).

$$\text{harmonic}(p) = \sum_r (\text{Distancia mínima entre } p \text{ y } r)^{-1}. \quad (4.4)$$

Esta medida maneja apropiadamente ∞ , asumiendo $\infty^{-1} = 0$. En Harmonic centrality cada componente resulta independiente del resto, lo que no ocurre con closeness.

Indicadores de intermediación

Llamamos indicadores de intermediación a aquellos que están basados en la influencia que un nodo tiene para los flujos de información. La medida de intermediación paradigmática es

betweenness, definida como la suma de las proporciones de caminos mínimos entre todos los pares q y r que pasan por p .

$$\text{betweenness}(p) = \sum_{q,r} \frac{\text{Caminos mínimos entre } q \text{ y } r \text{ que pasan por } p}{\text{Caminos mínimos entre } q \text{ y } r}. \quad (4.5)$$

Es una medida que beneficia a los nodos “puente”.

Load Centrality es una variante de *betweenness*, pero su definición formal usa el concepto de flujo. El flujo que envía un nodo q hacia un nodo r se transmite por los caminos mínimos, dividiendo el flujo en partes iguales cada vez que aparece más de una alternativa disponible.

$$\text{load}(p) = \sum_{q,r} \text{Cantidad de flujo enviada por el nodo } q \text{ al } r \text{ que pasa por } p. \quad (4.6)$$

Si hay un solo camino con distancia mínima, todo el flujo irá por ahí. Estas dos soluciones están basadas en caminos mínimos.

4.3. Análisis

En esta sección exploramos el efecto causal que la posición topológica de los individuos en la red de partidas tiene sobre sus procesos de aprendizaje individual. En la subsección 4.3.1 estimamos la habilidad inicial de los individuos a través del modelo estado del arte de la industria y la academia (TrueSkill Through Time), que hemos publicado por primera vez desde su creación en **Julia**, **Python** y **R** [116]. En la subsección 4.3.2 medimos la posición topológica de los individuos en la red de intercambios de información (indicadores de popularidad, cercanía e intermediación) utilizando la metodología clásica de análisis de grafos dinámicos [114, 115]. Finalmente, en la subsección 4.3.3 evaluamos cómo cambia el aprendizaje en función de la posición topológica los individuos en el red, manteniendo la habilidad inicial y la experiencia individual fijas.

4.3.1. La habilidad inicial de las curvas de aprendizaje individuales

Conocer la habilidad inicial es importante debido a que en todos los modelos causales estructurales (e.g. funciones sigmoideas, leyes de potencia o de von Bertalanffy) la tasa de aprendizaje depende del estadío en el que se encuentra cada uno de los individuos al empezar el proceso. Sin embargo, los modelos de habilidad más utilizados en la industria del video juego y la academia no son capaces de ofrecer buenas estimaciones iniciales fiables. Por el contrario, TrueSkill Through Time ofrece estimaciones con baja incertidumbre a lo largo de toda la curva de aprendizaje, proporcionando estimaciones precisas de la habilidad inicial y garantizando la comparación de las estimaciones distantes en tiempo y espacio. Si bien la especificación de este modelo había sido publicada más de una década atrás [103], no existía ninguna versión de código abierto disponible hasta el lanzamiento de nuestros paquetes en **Julia**, **Python** y **R** [116]. Sin nuestra implementación, no hubiera sido posible distinguir el estadío inicial en el que se encuentran los individuos antes de comenzar el proceso de aprendizaje y, por lo tanto, hubiera sido imposible evaluar el efecto causal que la posición en la red tiene sobre el aprendizaje.

En la figura 4.5 mostramos la distribución de habilidades iniciales de los jugadores con al menos 150 partidas jugadas. Antes de explorar los posibles efectos causales, discriminamos

a la población en grupos según el nivel de su habilidad inicial: “novatos”, “intermedios” y “expertos”.

En la figura 4.5 mostramos la distribución de habilidades iniciales de los jugadores con al menos 150 partidas jugadas. Antes de explorar los posibles efectos causales, discriminamos a la población en grupos según el nivel de su habilidad inicial: “novatos”, “intermedios” y “expertos”. Definimos como novatos a los individuos con habilidades iniciales entre 7 y 10,

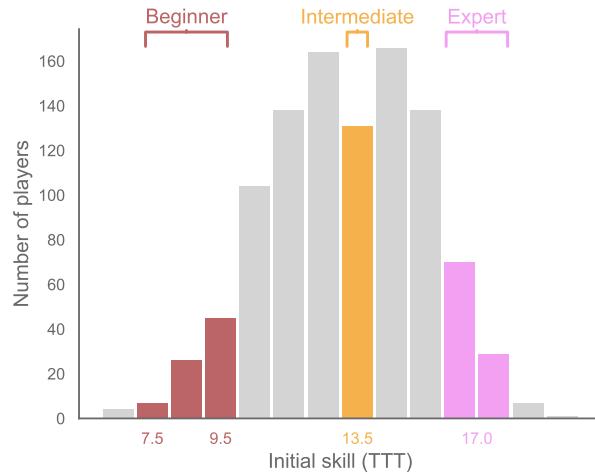


Figura 4.5: Distribución de habilidad inicial para individuos con al menos 150 partidas jugadas. En color señalamos los estadíos iniciales de las curvas de aprendizaje individuales que analizamos en este trabajo: novatos, intermedio y experto.

como intermedios a los individuos con habilidades iniciales entre 12 y 13 , y como expertos a los individuos con habilidades iniciales entre 16 y 18. La distancia que existe entre grupos nos permite distinguir claramente las clases.

Las personas que recién empiezan tienen mucho que aprender, por lo que sus curvas de aprendizaje suelen ser pronunciadas. Las personas con cierta experiencia necesitan trabajar más para adquirir nuevos conocimientos, por lo que sus curvas de aprendizaje suelen crecer más lentamente. Las personas expertas alcanzan un límite en la adquisición de nuevo conocimiento, por lo que sus curvas de aprendizaje suelen ser estables o incluso levemente decrecientes. Por ello, es esperable que el efecto de la posición en la red sea diferente en cada caso.

4.3.2. La posición topológica de los individuos en la red

Para describir la posición topológica de los individuos en cada una de las ventanas deslizantes computamos seis medidas de centralidad individual, que clasificamos en tres grupos: popularidad, cercanía e intermediación (descritos en la sección metodología). En la figura 4.6 mostramos la distribución de medianas de *eigenvalue*, *harmonic* y *load*, una medida de centralidad por cada tipo de indicador (popularidad, cercanía e intermediación). Las figuras 4.6(a) y 4.6(c) tienen el eje *x* en escala logarítmica. Aquí mostramos el valor mediano de centralidad en las primeras 50 partidas de los jugadores activos que alcanzaron al menos 150 partidos de experiencia. En todos los casos las distribuciones se encuentran centradas alrededor de un valor típico, con diferentes tipos de dispersión. Otras combinaciones de valores fueron exploradas, y los resultados que aquí presentamos no se ven afectados. La distribución

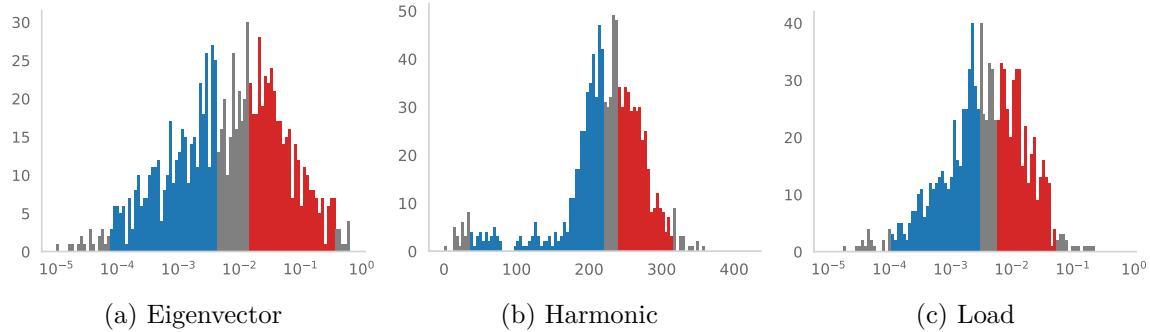


Figura 4.6: Distribución de medianas de las centralidades *eigenvalue*, *harmonic* y *load*, teniendo en cuenta únicamente las primeras 50 partidas jugadas de usuarios con al menos 150 partidas jugadas totales. Las líneas punteadas muestran los percentiles 2, 40, 60 y 98.

de valores medianos de centralidad de quienes tienen experiencia individual entre 50 y 150 partidas jugadas es similar a la que se muestra en la figura 4.6.

Para evaluar el efecto que tienen las posiciones de los individuos en la red sobre el aprendizaje, identificamos cohortes de centralidad baja y alta. Eliminamos del análisis a los jugadores con valores de centralidad inferiores al percentil 2 y superiores al 98. Y para distinguir con claridad las cohortes estudiadas, eliminamos también el grupo de jugadores con valores de centralidad entre los percentiles 40 y 60. De esa forma, los jugadores clasificados en las cohortes de baja y alta centralidad son aquellos que tienen sus valores de centralidad entre los percentiles 2 y 40, y entre los percentiles 60 y 98 respectivamente. Es esperable que el efecto sobre el aprendizaje sea diferente en cada caso.

4.3.3. El impacto que la posición en la red tiene sobre el aprendizaje individual.

Para explorar el efecto que la posición en la red tiene sobre el aprendizaje individual, comparamos las curvas de aprendizaje de individuos pertenecientes a cohortes de centralidad baja o alta, dados los mismos valores de habilidad y experiencia iniciales. Observando el comportamiento de las distintas curvas de aprendizaje dado un valor de habilidad inicial, podemos comparar tasas de aprendizaje que según la experiencia individual deberían ser iguales. De esta forma, la diferencia entre las curvas de aprendizaje puede ser interpretada como el efecto causal producido por la posición de los individuos en la red. La habilidad inicial, nos permite evaluar por separado este efecto en las diferentes etapas de aprendizaje (principiantes, intermedios y expertos).

En la figura 4.7 podemos observar las diferentes curvas de aprendizaje dados mismos los valores de habilidad inicial y experiencia. Cada fila contiene un determinado nivel de habilidad inicial: novato, intermedio y experto. Y en cada columna se evalúa una misma medida de centralidad, una por cada tipo de indicador: popularidad (*eigenvector*), cercanía (*harmonic*) e intermediación (*load*). Las curvas de aprendizaje azul y naranja representan la habilidad en poblaciones con niveles de centralidad baja y alto respectivamente. Al interior de cada figura, el eje *x* permite comparar las curvas de aprendizaje para los mismos valores de experiencia. En el eje *y*, las curvas de aprendizaje están medidas usando el modelo de habilidad estado del arte de la industria del video juego, bajo una escala en la que un punto diferencia de habilidad

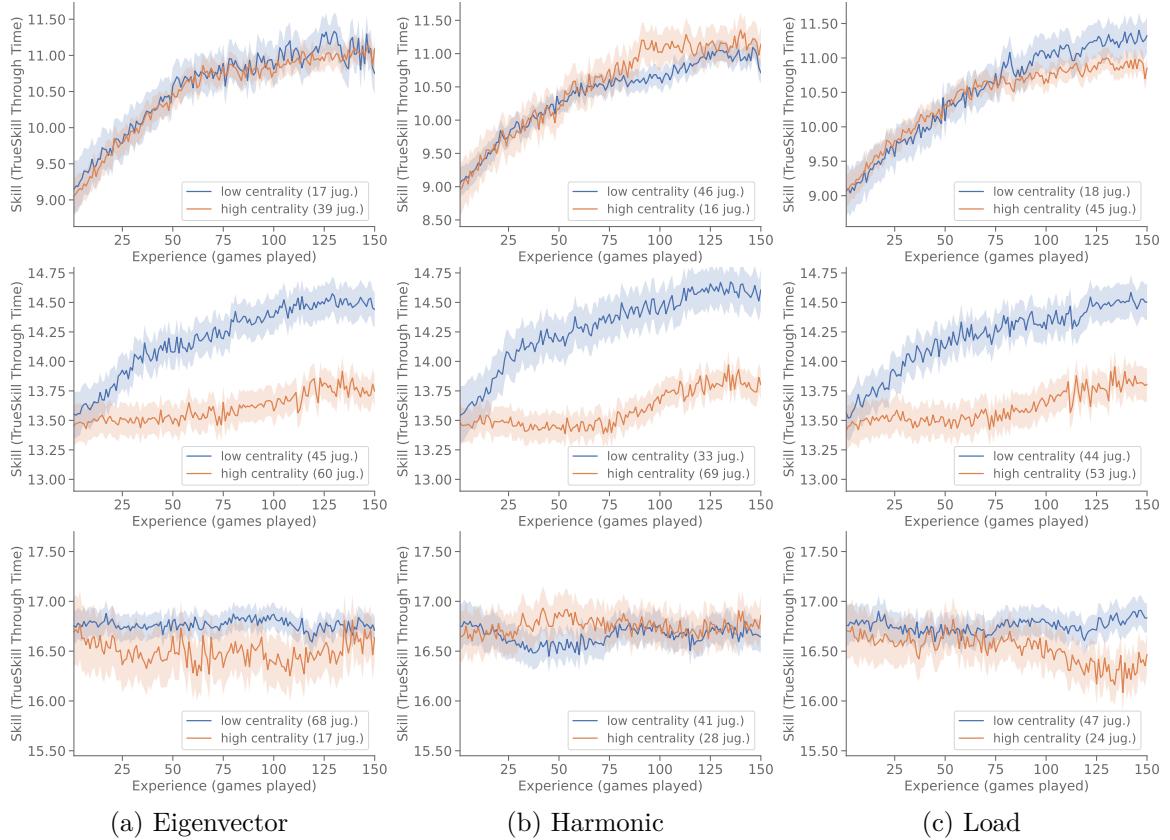


Figura 4.7: Curvas de aprendizaje dado niveles de centralidad baja (azul) y alto (naranja). Cada columna contiene un medida de centralidad. Y cada fila contiene un nivel de habilidad inicial: alrededor de 9 para los principiantes, 13,5 para los intermedios, y 17,75 para los expertos.

equivalga a 76 % de probabilidad de ganar. La leyenda indica el tamaño de la población en cada una de las curvas de aprendizaje. El área sombreada representa dos desvíos estándar de la curva de aprendizaje estimada.

Curiosamente, el efecto de la centralidad sobre las curvas de aprendizaje solo se observa en la población con habilidad inicial intermedia. El impacto en las curvas de aprendizaje no es significativo sólo en términos estadísticos, es además relevante en términos cualitativos en tanto la diferencia de habilidad entre las poblaciones con baja y alta centralidad alcanza al final de la serie una distancia de 1 punto, lo que equivale a 76 % de probabilidad de ganar. No es sorprendente que este efecto no se observe en la población experta pues, como ellas no tienen nada más que aprender, sus curvas de aprendizaje se mantienen planas. Lo que puede parecer sorprendente es que las curvas de aprendizaje de la población de novatos, que es la que tiene más por aprender, dependa solo de la experiencia y es independiente de la posición de los individuos en la red.

4.4. Discusiones

En este capítulo, exploramos el efecto que la posición topológica de los individuos en una red de intercambios culturales tiene sobre el aprendizaje individual. Estudiando la red de partidas de una de las comunidades virtuales más importantes de Go durante sus primeros 8 años de vida (2005-2013), pudimos detectar la existencia de un efecto positivo de la posición de los individuos en la red sobre el aprendizaje, que solo se activa cuando los jugadores tienen una habilidad inicial intermedia. Para interpretar los resultados, especificamos un modelo causal que busca resumir los principales acuerdos alcanzados en el área de aprendizaje social y acumulación cultural.

A diferencia del efecto de la experiencia en el aprendizaje, que decrece continuamente, el efecto de este factor social parece regirse por un crecimiento inicial y un decaimiento final, una especie de patrón en forma de U invertida. Por un lado, explotar los factores sociales puede requerir conocimientos previos para decodificar las estrategias del adversario e incorporar algunos de sus aspectos a la estrategia propia. Los novatos no tienen la perspicacia necesaria para lograrlo, pero los jugadores intermedios y expertos sí. Por otra parte, todo proceso de aprendizaje se satura en algún momento, lo que también produce un decaimiento del efecto de los factores de aprendizaje social. Así, aunque los jugadores expertos pueden entender mejor que nadie las estrategias de sus oponentes, al no tener mucho más que aprender no se observa por lo tanto el efecto del factor social en el aprendizaje.

Esto parece indicar que la posición de los individuos en la red tendría un efecto de segundo orden que sólo se activa cuando las personas han adquirido suficientes conocimientos para aprovechar la información cultural. Pero, ¿por qué es mejor una centralidad baja que una centralidad alta? Es importante señalar que la plataforma online analizada facilita el contacto entre personas, con una tendencia a establecer juegos basados en el azar. En este contexto, no existen barreras sociales o ambientales que produzcan la aparición de comunidades aisladas. En otras palabras, lo que llamamos baja centralidad está lejos de representar situaciones de aislamiento, asociadas a la pérdida de conocimiento cultural. Por esta razón, nuestra cohorte de centralidad baja podría interpretarse como de centralidad media para la literatura sobre aprendizaje social y acumulación cultural.

Nuestro resultado se alinea con recientes trabajos teóricos y experimentales, que muestran que las estructuras con conexión moderada favorecen los procesos de aprendizaje social y acumulación cultural [5]. Aunque todavía no es común encontrar en la literatura académica la formalización gráfica de los modelos causales, en la introducción especificamos un modelo causal que pretende resumir los principales acuerdos alcanzados en el campo de la evolución cultural acumulativa, considerando las características particulares de nuestra base de datos. Siempre podemos aumentar la complejidad de los modelos causales añadiendo variables de importancia de segundo orden o detalles específicos de contextos particulares. Sin embargo, su evaluación mediante la aplicación estricta de las reglas de probabilidad, $P(\text{Modelo}|\text{Datos})$, arroja un equilibrio natural a favor de los modelos que tienen la complejidad necesaria y suficiente, penalizando tanto a los modelos que no tienen la complejidad necesaria para predecir los datos observados como a los modelos que introducen una complejidad innecesaria que no mejora la predicción previa de los datos. En lugar de comparar el rendimiento de modelos causales alternativos presentes en la literatura, en este trabajo extraemos las ideas centrales compartidas por los principales modelos.

La estimación de la capacidad inicial ha sido uno de los problemas centrales de este trabajo. Los modelos comúnmente utilizados en la industria y en el mundo académico propagan la

información en una sola dirección, del pasado al futuro, produciendo estimaciones de la habilidad inicial con una incertidumbre muy elevada. Si bien trabajos recientes intentan mejorar estas estimaciones analizando los chats entre jugadores mediante técnicas de procesamiento de lenguaje natural moderna [118], sus estimaciones tienen una nivel de granularidad muy bajo en tanto sólo permiten clasificar la habilidad en tres categorías (expertos, novatos o desconocida). Por el contrario, el desarrollo reciente de redes bayesianas dinámicas, como TrueSkill Through Time [103], produce estimaciones precisas con baja incertidumbre incluso al principio de las series temporales de cada individuo. Debido a que este modelo no estaba disponible en ninguno de los lenguajes de programación más populares, nos vimos obligados a implementarlo [116]. A través de nuestro software de código abierto, en **Julia**, **Python**, y **R**, pudimos conocer con precisión la habilidad inicial de los individuos, permitiéndonos analizar por separado el efecto que la estructura de red tiene sobre los procesos de aprendizaje.

Estudiar la dinámica estructural de una red que contiene medio millón de nodos durante un periodo de ocho años fue otro de los retos que enfrentó este trabajo. El análisis de grafos es una tarea compleja e intensiva desde el punto de vista computacional. En este contexto, decidimos seguir la metodología clásica y robusta [114, 115] que estudia la dinámica del grafo utilizando ventanas deslizantes. Dado que las redes sociales online suelen tener las propiedades fundamentales de las redes del ego humano [119], nuestras observaciones pueden eventualmente ser representativas del comportamiento humano en otros dominios.

Nuestros resultados también están alineados con otros análisis nuestros anteriores, en el que observamos que repetir compañeros de equipo en juegos en línea también mejora los procesos de aprendizaje a largo plazo a pesar de estar asociados con valores de centralidad bajos [120]. A pesar del aparente acuerdo con la literatura previa, los resultados de nuestro trabajo deben ser considerados sólo como exploratorios y no como conclusiones causales definitivas.

Utilizando el modelo de habilidad de estado del arte y técnicas clásicas de análisis dinámico de grafos, descubrimos que los valores moderados de centralidad tienen una relación de segundo orden con el aprendizaje, que sólo se activa para los individuos con experiencia intermedia. Los jugadores novatos aprenden principalmente por experiencia individual, y los jugadores expertos no tienen nada más que aprender.

Capítulo 5

Conclusiones

Esta es una tesis interdisciplinaria que, desde su concepción, ha tenido como objetivo sacar provecho de la relación entre las ciencias de la computación y las ciencias antropológicas. La antropología no es una ciencia social más. Cuando se establece finalmente la hegemonía europea a nivel global, hace menos de 200 años, se crea la sociología, destinada al estudio de la propia sociedad europea, y sus especializaciones como las ciencias políticas, la economía y la criminología, enfocadas en el estudio de ciertos aspectos de la sociedad europea. El resto, y por lo tanto *todo*, quedó a cargo de la antropología. Así es que la antropología es la única ciencia social que recibió el objeto social completo: todas las culturas del mundo, en el tiempo y el espacio, desde el origen de la humanidad hasta el presente, desde sociedades nómadas cazadoras-recolectoras actuales, hasta las sociedades agricultoras estatales antiguas y modernas, en sus aspectos ecológicos, económicos, políticos, religiosos y otros no contemplados por la cosmovisión colonial-moderna.

El objeto de estudio principal de la antropología es la cultura. El ser humano se destaca por su capacidad de transmitir conocimientos por vía social (*aprendizaje social*), de un individuo a otro, entre poblaciones y a lo largo de las generaciones. La transmisión y acumulación de adaptaciones en el tiempo producen la emergencia del sistema de información no-genético denominado cultura [3, 4]. Comprender las propiedades de este sistema de información es fundamental para comprender el aprendizaje humano. Si bien el aprendizaje humano ocurre a través de los cuerpos biológicos, la cultura es un aprendizaje colectivo que emerge en el tiempo del intercambio de información entre individuos. Entender las relaciones que vinculan el nivel poblacional (como la dinámica y estructura de la red de intercambios de información cultural) con el nivel individual (el aprendizaje efectivo que se produce en los cuerpos), ha sido la pregunta que motivó esta tesis de doctorado.

Este plan quedó enmarcado inicialmente dentro del área denominada como Ciencias Sociales Computacionales [30,31], que en la actualidad ya alcanzó reconocimiento en los gigantes tecnológicos relacionados a la industria del software. Sin embargo, a medida que fue avanzando su desarrollo, el enfoque bayesiano de la probabilidad fue adquiriendo cada vez más protagonismo.

5.1. Enfoque bayesiano de la probabilidad

El doctorado en computación me permitió responder las preguntas metodológicas y epistemológicas que traía de mi formación previa en antropología. Al mismo tiempo que surge la

corriente económica neoliberal, surge la corriente filosófica posmoderna que fue hegemónica en la década de los 90 y seguía presente en el Departamento de Antropología de la Universidad de Buenos Aires en la primera década de los años 2000. El posmodernismo pone en duda la existencia de verdades universales. Según esta corriente, lo que se considera verdad en un momento dado no sería más que una expresión coyuntural de las relaciones de poder. Criticar los universalismos aparece como políticamente correcto en relación a la historia reciente, en la que valores arbitrarios de la sociedad colonial fueron impuestos a sangre y fuego en nombre de la “verdad”. Sin embargo, si fuera imposible alcanzar acuerdos intersubjetivos válidos universalmente, tampoco podríamos enunciar juicio categórico respecto de los crímenes más atroces de la humanidad, incluyendo los innumerables genocidios de la propia colonial-modernidad. Esta idea queda bien resumida en la siguiente tira cómica 5.1.

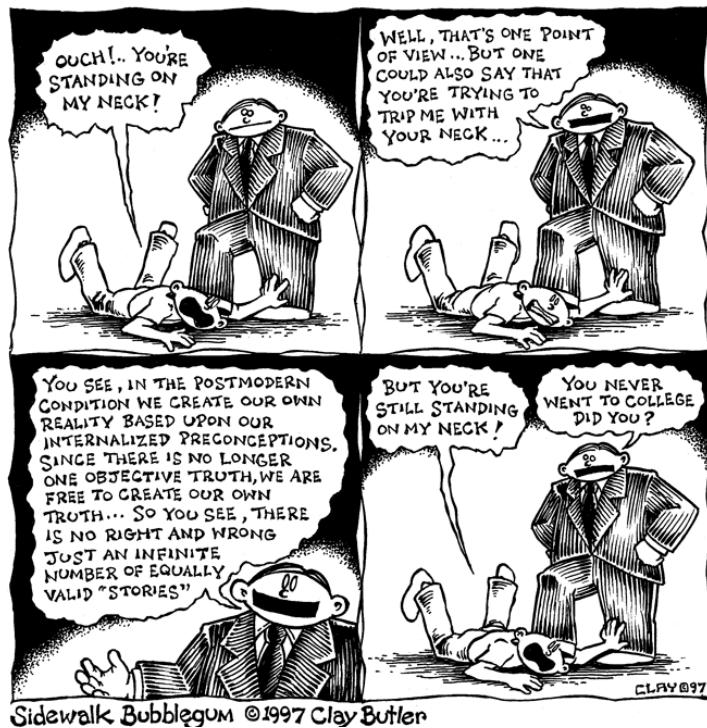


Figura 5.1: Parodia del posmodernismo, corriente filosófica contemporánea del neoliberalismo.

En antropología aprendí a programar y a pensar el problema social como un sistema complejo en el que las propiedades del sistema emergen de la interacción entre las partes. La cuestión metodológica y epistemológica acerca de cómo alcanzar acuerdos intersubjetivos válidos universalmente se presentó como la pregunta más profunda y difícil en antropología. Comencé mi formación en ciencias de la computación con el objetivo de encontrar herramientas para las ciencias sociales. Jamás me imaginé que me encontraría con el enfoque bayesiano de la probabilidad, que respondió de forma simple y directa todas las preguntas metodológicas y epistemológicas que traía de mi formación en antropología.

En la sección *Aplicación estricta de las reglas de la probabilidad* del capítulo 1 se presentaron los aspectos más significativos: la definición formal de acuerdos intersubjetivos en contextos de incertidumbre (maximizar entropía o incertidumbre dadas las restricciones), la evaluación honesta de argumentos causales alternativos y las consecuencias de la ruptura de

las reglas de razonamiento.

5.2. Efecto de la formación de equipos sobre el aprendizaje.

Uno de los objetivos principales que nos habíamos propuesto en el plan de doctorado era el análisis de base de datos reales. Para el primer trabajo desarrollamos una pieza de software mediante la cual exploramos la comunidad en línea vinculada a la plataforma Conquer Club, que ofrecía juegos línea tipo RISK o TEG. En base a las características del juego, implementamos una base de datos relacional utilizando utilizando el motor PostgreSQL. Además, desarrollamos una script Python que ingresaba a cada una de las partidas y descargaba los datos públicos que se encontraban en formato html. Obtuimos la autorización de los propietarios de Conquer Club para descargar cerca de 4,4 millones de partidas jugadas por casi 270 mil usuarios diferentes entre 2006/01/03 y 2009/07/12. Debido a una doble capa de anonimato y la naturaleza abierta del sitio web pudimos publicar la base de datos en Dryad [95], una plataforma para la reutilización de datos de investigación.

Si bien nuestro objetivo inicial era el estudio de la evolución de las estrategias en los juegos en línea, esta cuestión nos obligó a resolver primero un problema más sencillo pero importante: estimar la “adaptabilidad” de los individuos a lo largo del tiempo. Considerar sólo la frecuencia éxitos y fracasos como indicador de la habilidad de los individuos conduce a estimaciones de habilidad erróneas debido a que el valor de un éxito o un fracaso depende de la dificultad de los desafíos. Para solucionar este problema acudimos al estimador de habilidad Elo, famoso por ser el sistema oficial de habilidad de la Federación Internacional de Ajedrez (FIDE). Sin embargo, encontramos que existía un estimador de habilidad considerado estado-del-arte en aquel momento (TrueSkill), que era una solución bayesiana al modelo Elo. Gracias a que este modelo estaba disponible como paquetes en varios lenguajes de programación, pudimos llevar adelante ese primer trabajo (capítulo 2) sin conocer todos sus detalles técnicos.

Debido a que la base de datos Conquer Club contiene juegos por turnos donde los jugadores pueden participar individualmente o en equipos, decidimos analizar el impacto de las estrategias de formación de equipos en la adquisición de habilidades. Allí encontramos que la tendencia a jugar en equipos se asocia con una mejora de las habilidades a largo plazo, mientras que jugar lealmente con el mismo compañero de equipo acelera significativamente la adquisición de habilidades a corto plazo [120]. El análisis estadístico de este primer artículo lo realizamos aplicando métodos frecuentistas. Un tiempo después de publicado el artículo, el profesor de Ciencias Cognitivas en University of Sheffield, Tom Stafford, nos escribió una mensaje confirmando que su grupo de investigación había podido replicar exitosamente los resultados de nuestro trabajo.

Dear Drs Landfried, Slezak, and Mocsos

I enjoyed your Plos ONE paper immensely [1] and thought I would write to say that, thanks to your generous data sharing policy, we were able to independently replicate your result. You won't find this surprising, but I always think it is comforting to hear that a complex analysis pipeline can be independently replicated, and is testimony to the quality of the original research.

5.3. Una hipótesis que fue rechazada

Una de las hipótesis que se desprendió de este primer trabajo era la idea de que el efecto positivo de tener compañeros estables estuviera producido por una especie de “sinergia” que se genera cuando dos personas aprenden a jugar juntas. Hemos visto que TrueSkill modela el desempeño de los equipos como una suma lineal de los desempeños de los jugadores, sin considerar interacción, por lo que no es capaz de detectar la presencia de sinergia. En un segundo trabajo, no publicado en esta tesis, me propuse el objetivo de extender el modelo TrueSkill de modo tal que permitiera distinguir la habilidad de los individuos de la “sinergia” entre pares de individuos.

Incorporo al Anexo 7.2 algunos análisis realizados en esta etapa. En esta etapa comencé a interiorizarme con la matemática detrás del modelo TrueSkill. Diseñé un modelo alternativo, que llamamos TrueSynergy, en el cual los desempeños de los equipos se modelan como una suma lineal de las habilidades de los individuos s_i y las sinergias o habilidades entre pares de individuos s_{ij} . En el modelo propuesto, la sinergia se incorpora de la misma forma que las habilidades de los individuos, por lo que la matemática necesaria para realizar la inferencia no cambia. La única diferencia respecto del modelo TrueSkill es que por cada par de jugadores i y j se incorpora una componente de habilidad adicional. Este modelo es similar al utilizado en el capítulo 3 para estimar las habilidades de los tenistas profesionales en los diferentes pisos. En ese caso, modelamos las habilidades de los tenistas como una suma lineal de su habilidad “general” de jugar al tenis y su habilidad “específica” relativa a cada uno de los pisos (polvo de ladrillo, pasto, etc.).

Para cuantificar el error de estimación de habilidad de TrueSkill cuando los equipos tienen un componente de sinergia, generamos datos sintéticos utilizando como *ground truth* el modelo con sinergias TrueSynergy. Se encontró una función (7.1) que predice la sobreestimación que hace el modelo TrueSkill de la habilidad de los individuos cuando existe sinergia entre jugadores. Evaluamos también la mejora en la estimación del modelo que incorpora sinergias (figura 7.3). Ambos modelos tienen una limitación común. Ninguno de los dos modelos puede diferenciar las habilidades de dos individuos (y su sinergia) si los individuos juegan siempre juntos en el mismo equipo.

Al haber producido un modelo alternativo, apareció por primera vez la pregunta respecto de cómo evaluar correctamente modelos alternativos sobre los datos reales. Este punto fue una bisagra para mi doctorado. En el área de inteligencia artificial hay muchas magnitudes ad hoc para evaluar el desempeño de los modelos (*accuracy*, *precision*, *Fscore*, área ROC, etc.). En ese momento, mi codirector me propuso algunas opciones que no me convencían. Esto me impulsó a revisar cuál era la forma correcta de evaluar modelos alternativos. La respuesta era simple, computar el posterior de los modelos M dado los datos D , $P(M|D)$, mediante la aplicación estricta de las reglas de la probabilidad, o en su defecto comparar pares de modelos mediante el Bayes Factor, $P(M_1|D)/P(M_2|D)$. Cuando le dije a mi codirector “*Encontré la forma correcta de evaluar modelos*”, me respondió “*No hay una forma correcta de evaluar modelos*”. La misma frase que había escuchado en antropología tantos años, respecto de la imposibilidad de alcanzar acuerdos intersubjetivos, se me había presentado nuevamente, esta vez en el seno del área de inteligencia artificial.

El modelo que había desarrollado era bastante más complejo que TrueSkill, debido a que por cada par de jugadores era necesario agregar una componente de sinergia. Hemos visto en la sección 1.3.2 que la forma correcta de evaluar modelos penaliza naturalmente la complejidad si ella no mejora la predicción a priori de los datos. Y eso fue lo que ocurrió

con el modelo propuesto, TrueSynergy. Luego de confirmar que mi modelo no era mejor en desempeño predictivo, y sin apoyo de la dirección en el trabajo, las estimaciones realizadas no fueron analizadas en detalle y la hipótesis original que había motivado este trabajo no alcanzó a ser respondida de forma positiva o negativa.

Sin embargo, esta es una línea de investigación que sería interesante finalizar. Algo similar ocurrió cuando estimamos las habilidades que los tenistas de la ATP tienen en los diferentes tipos de suelo. Si bien sabemos que algunos jugadores de tenis tienen habilidades muy diferentes entre pisos (como Nadal), el desempeño del modelo que creamos en el capítulo 3 para estimarlas fue menor que el modelo que hace de cuenta que no existen diferencias entre pisos. Si bien la predicción conjunta de los modelos complejos fue menor tanto en Conquer Club (para estimar sinergia) como en ATP (para estimar habilidad por piso), las estimaciones ofrecen información interesante para analizar.

Una estrategia interesante hubiera sido descomponer la predicción y evaluar para qué individuos la complejidad añadida sí era necesaria. En ATP, por ejemplo, pudimos ver que la complejidad añadida aumentaba la capacidad predictiva para ciertos jugadores, como el caso de Nadal, quien tiene habilidades que varían significativamente entre pisos. Un análisis similar se podría haber realizado con los datos de Conquer Club, lo que permitiría distinguir compañeros de equipo para los cuales la sinergia es un factor relevante. En cualquier caso, nuestro modelo, al igual que TrueSkill tiene una dificultad para distinguir de dónde proviene el desempeño cuando dos personas siempre juegan juntas, por lo que el análisis hubiera quedado limitado a pares de jugadores no estables, y la hipótesis inicial no hubiera podido responderse.

El modelo con sinergias lo pudimos aplicar gracias a que existía una forma de implementarlo usando el paquete disponible, agregando las sinergias como si fueran miembros de los equipos. Si yo quería realizar nuevas modificaciones al modelo TrueSkill, comenzaba a ser una necesidad comprender todos sus detalles.

5.4. La primera implementación del modelo de habilidad estado del arte

Esta tesis se destaca, en particular, por haber realizado la primera implementación del modelo de estimación de habilidad estado del arte en la industria del video juego, que si bien ya estaba especificado matemáticamente, no se encontraba todavía disponible como paquete para ser utilizado en algún lenguaje de programación. Como hemos visto en el capítulo 3, el modelo TrueSkill Through Time mejora significativamente las estimaciones respecto de TrueSkill gracias a que propaga la información histórica a través de toda la red de eventos, lo que le permite realizar estimaciones iniciales de habilidad fiables y garantizar su comparabilidad en el tiempo y el espacio. La novedad no es tanto el modelo causal de base (que se ve en detalle en el capítulo 1 y 3), el cual es tan simple como en los anteriores. La mejora que introduce proviene de integrar todos los modelos causales de base (los eventos donde ocurren los éxitos o los fracasos) en una única red causal histórica (la habilidad del pasado es una causa de la habilidad del presente) que permite propagar la información por toda la red, obteniendo así estimaciones con menor incertidumbre en todas las series temporales.

El modelo TrueSkill ofrece una aproximación al posterior, no su solución exacta. Tanto el paper original [58] y como sus patentes [59, 60] no muestran cómo se deriva la aproximación propuesta ni tampoco la solución exacta. Debido a que la matemática detrás del algoritmo no estaba disponible de forma completa en los artículos oficiales, ni siquiera en las fuentes

alternativas que pretendían mostrar la matemática detrás de TrueSkill [105], comencé a dudar si efectivamente la solución propuesta era la mejor aproximación al posterior exacto. Leyendo y releyendo el paper original y las patentes apareció una cita al algoritmo *sum-product* [94]. Hemos visto que este algoritmo permite computar las marginales de una distribución de probabilidad conjunta través de pasaje de mensajes entre los nodos de la red bayesiana expresada como un *factor graph* (sección 3.2.1). Con este algoritmo comencé resolver los mensajes uno a uno. Una vez alcanzada la solución exacta pude comprobar mediante métodos numéricos que la aproximación propuesta era efectivamente la mejor en términos de divergencia KL. Pero quizás más importante, de esta forma había logrado comprender absolutamente todos los detalles del modelo TrueSkill.

Luego de comprender todos los detalles del modelo TrueSkill, encuentro la existencia de una extensión conocida como TrueSkill Through Time que no estaba implementada hasta el momento como paquete de ningún lenguaje de programación. De hecho, existía un *issue* abierto en el paquete de TrueSkill de Python en el cuál se solicitaba su implementación¹. Su desarrollador Heungsub Lee respondió que no implementaría la extensión, pero dejaba abierta la posibilidad para que otra persona lo hiciera. Uno de los colaboradores activos del paquete respondió “*Having just read the paper I admit I don't see TTT is not something that can be implemented meaningfully without some significant qualification and work.*”

Efectivamente, ya había pasado una década desde su publicación [103] y este algoritmo seguía sin estar disponible en ningún lenguaje de programación a pesar de que mejoraba significativamente las estimaciones de todos los modelos disponibles hasta el momento. Aprender todos los detalles del modelo TrueSkill fue un camino largo y arduo. Sin embargo, cuando leí la especificación del modelo TrueSkill Through Time [103] inmediatamente entendí cómo resolverlo. Debido a que no estaba implementado en ningún lenguaje, repliqué la implementación en Python, Julia y R. Y aproveché esta ocasión para documentar no solo toda la matemática detrás de TrueSkill, sino también la de TrueSkill Through Time.

Este trabajo ha sido aceptado para su publicación en el *Journal of Statistical Software*.

5.5. Efecto en el aprendizaje individual de la posición en el red cultural

El aprendizaje humano depende, en gran medida, del contexto, las interacciones sociales y la transmisión de información cultural entre individuos, por lo que está ampliamente aceptado que la estructura de las “redes culturales” afecta los procesos de aprendizaje individual y colectivo. Estudios recientes, sin embargo, han cuestionado la hipótesis de que todo lo que maximice el flujo de información cultural debería repercutir positivamente en el aprendizaje. Para explorar este problema decidimos estudiar la relación entre el aprendizaje de los individuos en el tiempo y la estructura de interacciones en una importante comunidad virtual de Go a lo largo de ocho años.

Para el desarrollo de la base de datos, diseñamos un *script* (escrito en Python) para extraer la información de cada una de las partidas del sitio OGS. Dado el enfoque de este trabajo y la naturaleza de los datos, decidimos usar una base de datos no relacional basada en grafos: Neo4j. Sin embargo, ciertas consultas son extremadamente costosas (por ejemplo, calcular centralidades), y pueden ser prohibitivas en como encontrar caminos mínimos. Por esta razón,

¹<https://github.com/sublee/trueskill/issues/14>

decidimos exportar los datos de la base de datos a un formato que pudiera ser interpretado por alguna biblioteca en `Python`, como `NetworkX` que ofrece métodos que resultaron más eficientes para calcular algoritmos complejos, como centralidades, *clustering*, caminos mínimos, etc.

En este trabajo exploramos el vínculo entre la red de partidas y la evolución de las habilidades de todos los individuos a lo largo de 8 años, con 446 000 jugadores y 818 000 partidas, período libre de *bots*. Dado que para evaluar la tasa de aprendizaje de los individuos debemos conocer el valor inicial de habilidad, este trabajo sólo pudo ser llevado a cabo gracias a nuestra implementación del modelo TrueSkill Through Time. Al resolver la inferencia en una única red bayesiana con toda la historia de partidas, obtuvimos buenas estimaciones de habilidad incluso al inicio de las series temporales. Esto nos permitió clasificar a los usuarios en tres categorías, novatos, intermedios y expertos.

Para analizar la posición topológica que ocupan los individuos en la red de partidas a lo largo del tiempo utilizamos una técnicas de grafos dinámicos que divide la red en pequeñas subredes, compuestas únicamente por nodos y vértices activos en un periodo determinado, y analiza mediante métricas clásicas de grafos. Para describir la posición de los individuos analizamos seis medidas de centralidad, que pueden ser clasificadas en tres grupos: medidas de popularidad, como *degree* y *eigenvector*; medidas de cercanía, como *closeness* y *harmonic*; medidas de intermediación, como *betweenness* o *communicability*.

Siguiendo los principales hallazgos de la bibliografía sobre aprendizaje social y acumulación cultural (figura 4.1), para evaluar el efecto de la posición en la red sobre el aprendizaje individual se observó el cambio en la habilidad en función de la posición de los individuos en la red (alta o baja centralidad) para valores determinados de experiencia individual y habilidad inicial. De esta forma encontramos que sólo los individuos que están en la mitad del proceso de aprendizaje se ven afectados por la posición en la red de partidas. Es interesante que el aprendizaje de la población de novatos, que es la que tiene más por aprender, depende solo de la experiencia y es independiente de la posición de los individuos en la red. Esto puede estar relacionado con el hecho de que el aprendizaje social requiere una maduración para comprender las estrategias del adversario e incorporar algunos de sus aspectos a la estrategia propia. Además, nuestro resultado se alinea con la literatura antropológica reciente que afirma que las estructuras con conexión moderada favorecen en mayor medida los procesos de aprendizaje social y acumulación cultural que las estructuras de red hiperconectadas.

Capítulo 6

Trabajo futuro

La aplicación estricta de las reglas de probabilidad (o enfoque bayesiano) constituye la base de la inteligencia artificial y de las ciencias empíricas que aspiran a alcanzar verdades en contextos de incertidumbre. Si bien estas reglas se conocen desde el siglo 18, su aplicación se ha visto limitada históricamente debido al costo computacional asociado a la evaluación de todo el espacio de hipótesis. Aunque estos obstáculos han sido superados gracias a los avances computacionales y algorítmicos modernos, la inercia histórica hace que su desarrollo siga siendo incipiente, especialmente en países periféricos. Las cuestiones computacionales, sin embargo, son solo el primer paso de la inteligencia artificial.

Para explicar el mundo y tomar decisiones, todas las ciencias que trabajan con datos, desde la física hasta las ciencias sociales, desarrollan teorías causales justamente porque no es posible sacar conclusiones causales y predecir las consecuencias de las intervenciones sin proponer modelos que tengan una interpretación causal [48, 121]. Las teorías causales son hipótesis, y como tales pueden ser evaluadas mediante la aplicación estricta de las reglas de la probabilidad [122], $P(\text{Teoría causal}|\text{Datos})$, que penaliza naturalmente la complejidad innecesaria. Hoy la inteligencia artificial tiene la oportunidad de hacer uso de los modelos causales que se desarrollan en diversas ciencias empíricas, y las ciencias empíricas tienen la oportunidad de crear y evaluar modelos a la medida de cada problema específico [49, 123–125].

6.1. Comunidad

La relevancia de la aplicación estricta de las reglas de la probabilidad contrasta con la realidad actual en la que el enfoque bayesiano es marginal incluso en las universidades más importantes de la región, y prácticamente inexistente en ciencias empíricas que no cuentan con formación específica en matemática y programación. Este diagnóstico fue emergiendo durante la segunda mitad del doctorado. A partir de la poca cantidad de referentes bayesianos en la propia facultad, comencé un proceso de búsqueda activa. Este diagnóstico era compartido por otros referentes bayesianos de Argentina. En el año 2022 creamos la comunidad Bayes Plurinacional (figura 6.1) con el objetivo de impulsar las comunidades bayesianas de la región.

Una de las primeras actividades de organización consistió en analizar sistemáticamente y en profundidad la comunidad científica argentina que tuviera publicaciones utilizando métodos bayesianos. Para ello, desarrollé otra pieza de software mediante la cual exploramos la red de co-autorías de todos los artículos científicos (fuente Scopus). En el anexo se pueden visualizar



Figura 6.1: Símbolo de la comunidad Bayes Plurinacional

las redes bipartitas entre “artículos” y “autores” (figura 7.14). En la siguiente figura 6.2 se detalla la cantidad de artículos científicos vinculados al enfoque bayesiano publicados cada año con al menos una autoría con filiación en América Latina y el Caribe.

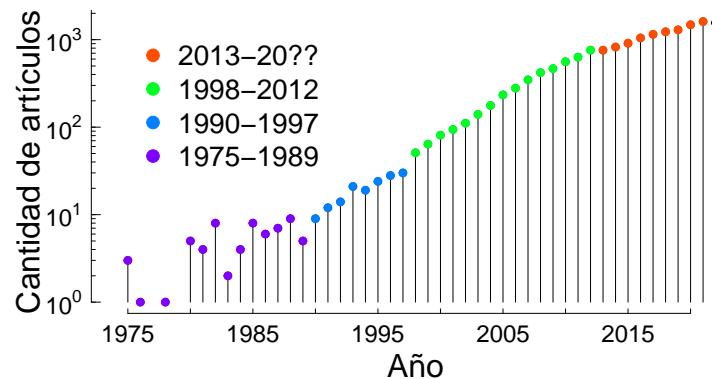


Figura 6.2: Cantidad de artículos científicos que fueron detectados con al menos una afiliación en Latinoamérica y alguna palabra claves en el título o abstract vinculada al enfoque bayesiano. Dejamos en blanco el último punto que representa al año 2022 (año en que se realizó la búsqueda), para el cual tenemos datos parciales.

Descargamos 16500 papers de 1975 en adelante. En el año 1975 se publican simultáneamente 3 artículos “Optimal Policies for Identification of Stochastic Linear Systems” de Alejandro Antonio Lopez-Toledo, “Unfolding of physical spectra by sequential bayes methods, Part I: Linear spline models” y “Part II. Estimation of parameters in non-linear models” de Manuel Mendès. En el año 1976 se publica el artículo “Bayesian full information analysis of simultaneous equations” de Juan Antonio Morales Anaya. En el año 1978 se publica el artículo “An extension of Colton’s model for comparing two medical treatments”. Entre el año 1980 y 1990 se publican entre dos y nueve artículos por año.

El crecimiento exponencial comienza en 1990, acompañando el período de masificación de las computadoras personales. En 1991 se rompe por primera vez la frontera de los 10 artículos por año, y comienza la primera etapa del crecimiento exponencial de las publicaciones bayesianas con co-autores de latinoamérica y el caribe. En el año 1998, se produce un aumento del 66 % en la cantidad de publicaciones respecto del año anterior y el inicio de la segunda etapa de crecimiento exponencial del enfoque bayesiano, que perdura hasta el año 2012. En este período la inferencia bayesiana estaba en el centro de atención en el área de la inteligencia artificial, hasta que en el año 2012, se presenta una red neuronal convolucional profunda entrenada sobre GPU que mejora significativamente el desempeño de los modelos previos [126]. A partir del año 2013 el crecimiento exponencial de las publicaciones bayesianas con afiliación en Latinoamérica continúa, alcanzando los 1611 artículos en el año 2021, pero

su tasa de crecimiento se ve reducida significativamente respecto de la etapa anterior.

Con el surgimiento de las redes neuronales profundas, nuevamente la complejidad computacional de la aplicación estricta de las reglas de la probabilidad actúa como un límite al enfoque bayesiano. Aún así, el auge de las redes neuronales profundas no produjo la aparición de un sistema de razonamiento bajo incertidumbre nuevo, con mejor desempeño que el de las reglas de la probabilidad. El enfoque bayesiano sigue siendo el corazón de la Inteligencia Artificial y de toda ciencia empírica. En este contexto, NeurIPS realiza una competencia en el año 2021 denominada *“Approximate inference in Bayesian Deep Learning”*. En ese año se publica el artículo *“What Are Bayesian Neural Network Posteriors Really Like?”* [127] en el que aproximan las distribuciones de probabilidad de redes neuronales mediante *full-batch Hamiltonian Monte Carlo* (HMC). Allí se confirma la observación previa [128], en la que se muestra que los métodos de *deep ensambles* proporcionan un mecanismo eficaz de aproximación de la aplicación estricta de las reglas de la probabilidad.

Gracias a que la mitad de los autores proveen en sus artículos científicos algún medio de contacto, pudimos conversar personalmente con una gran cantidad de investigadores bayesianos de la región y verificar que sus protagonistas no se suelen conocer todavía. Esto ratificó nuestro objetivo principal a largo plazo, de contribuir en el desarrollo de las comunidades bayesianas de la región. En este contexto, decidimos organizar el Congreso Bayesiano Plurinacional 2023.

El Congreso Bayesiano Plurinacional 2023

La comunidad Bayes Plurinacional realizó con éxito el Congreso Bayesiano Plurinacional 2023 los días 4, 5 y 6 de agosto en las instalaciones de la Secretaría de Ciencia y Tecnología de la provincia de Santiago del Estero, Argentina. La diversidad disciplinar fue extraordinaria: ecología, medicina, astronomía, geología, psicología, cuántica, economía, ingeniería, filosofía, biología, estadística, computación, antropología, entre otras. La diversidad de regiones también: Chile, Paraguay, Uruguay, Colombia, Honduras, Brasil, Argentina (Córdoba, Salta, Mendoza, Entre Ríos, Santa Fe, Santiago del Estero, Buenos Aires), tanto del sector privado, del público y de la academia. Además, durante el congreso nos visitó el gobernador de la provincia de Santiago del Estero, Gerardo Zamora (figura en anexo 7.11).



Figura 6.3: Congreso Bayesiano Plurinacional, agosto 2023 en La Banda, Santiago del Estero, Argentina.

Tuvimos cinco sesiones de conversaciones interpersonales de 40 minutos cada una; cuatro

bloques con tres exposiciones en profundidad de 25 minutos cada una (12 en total); cuatro bloques de taller de 50 minutos; y sesiones de póster que incluyeron la exposición oral breve a todo el auditorio. Las sesiones de conversaciones en grupos lograron crear una ambiente ameno, de apertura y conocimiento mutuo. Las exposiciones abarcaron una variedad de temas,

1. Gabriel A Alzamendi: *Métodos basados en inferencia Bayesiana para la evaluación de los mecanismos vocales de la fonación humana.*
2. Nicolás A Comay: *Abordaje Bayesiano a la toma de decisiones perceptuales y su confianza asociada.*
3. Javier Arellana: *Bayesian retrieval scheme and its application to Cassini Mission SAR observations in an area of Titan, Saturn's largest moon.*
4. Yasmín E Navarrete Díaz: *Sistemas frágiles: una formulación bayesiana de la teoría cuántica.*
5. Diego JR Sevilla: *Regresión bayesiana para el análisis de series temporales con apilamiento en Astronomía de rayos X.*
6. Andrea P Goijman: *Modelos de ocupación multiespecie con enfoque bayesiano: una herramienta valiosa para estudios de biodiversidad en ecología.*
7. Hossein Dinani: *Bayesian estimation for quantum sensing.*
8. Albert R Ortiz: *Aplicación de métodos bayesianos en ingeniería estructural.*
9. Gonzalo Ríos: *Bayesian Machine Learning Applied to Marketing.*
10. Sergio M Davis Irarrázabal: *Probabilidad bayesiana a partir de la estimación plausible de cantidades.*
11. Tomás A Olego: *Métodos robustos Bayesianos Generalizados para los modelos de regresión logística.*
12. Luciano Moffatt. *Modelado Bayesiano Computacional de Sistemas Biológicos.*

Durante el congreso quedó de manifiesto una imperiosa necesidad de contar con mayor cantidad de instancias de formación teórica y práctica en métodos bayesianos. Por ello, en la sesión de cierre decidimos comenzar a organizar la Escuela Bayesiana Plurinacional. Gracias a esta actividad inaugural, actualmente existe una activa comunidad bayesiana. A futuro me propongo seguir impulsando la comunidad Bayes Plurinacional.

6.2. Enseñaje

Recientemente, la comunidad Bayes Plurinacional a enviado un proyecto a la organización de Inteligencia Artificial de América Latina *Khipu* para la realización de la Escuela Bayesiana Plurinacional 2024 en la Universidad Nacional de Salta, Argentina. Esta propuesta contiene el apoyo más de 50 investigadores de diversos países. El nombre elegido para esta edición es “Aprender a enseñar a aprender”. Por una parte, esta frase es una extensión del concepto de “enseñaje”, que es una contracción entre las palabras enseñanza y aprendizaje empleada por el psiquiatra Enrique Pichon-Rivière y denota la naturaleza dialéctica entre enseñanza y aprendizaje: quien enseña aprende y quien aprende enseña. Por otro lado, “enseñar a aprender” puede ser interpretado como “enseñar bayes”, pues el enfoque bayesiano de la probabilidad es en sí mismo un método de aprendizaje. Por último, aprender a “enseñar bayes” hace referencia al objetivo principal de la Escuela Bayesiana Plurinacional: ampliar la oferta de formación en métodos bayesianos en América latina.

La Escuela Bayesiana Plurinacional se realizará al menos durante tres días, en principio, del 25 al 27 de julio de 2024 en la Universidad Nacional de Salta, Argentina. Contará con

talleres prácticos de niveles inicial e intermedio durante la mañana, talleres prácticos de nivel avanzado durante la tarde, y exposiciones y mesas de debate durante la noche. La búsqueda activa de disertantes se realizará durante el primer cuatrimestre del año 2024. La escuela finalizará con un Hackatón que tiene una duración cuatrimestral hasta la publicación de los resultados. Esperamos contar con la presencia de aproximadamente 100 personas, con una gran proporción de estudiantes iniciales universitarios, con investigadores formados de diversos países y disciplinas, tanto del sector privado, del público y de la academia. A futuro me propongo seguir impulsando la organización de eventos de formación en métodos bayesianos.

Taller “Verdades empíricas”

Durante el congreso se presentó un taller sobre inferencia bayesiana denominado “Verdades empíricas”, que estuvo dividido en 5 unidades y un hackatón. Se puede acceder en línea a través de <https://github.com/BayesPlurinacional/tallerBP-2023>.

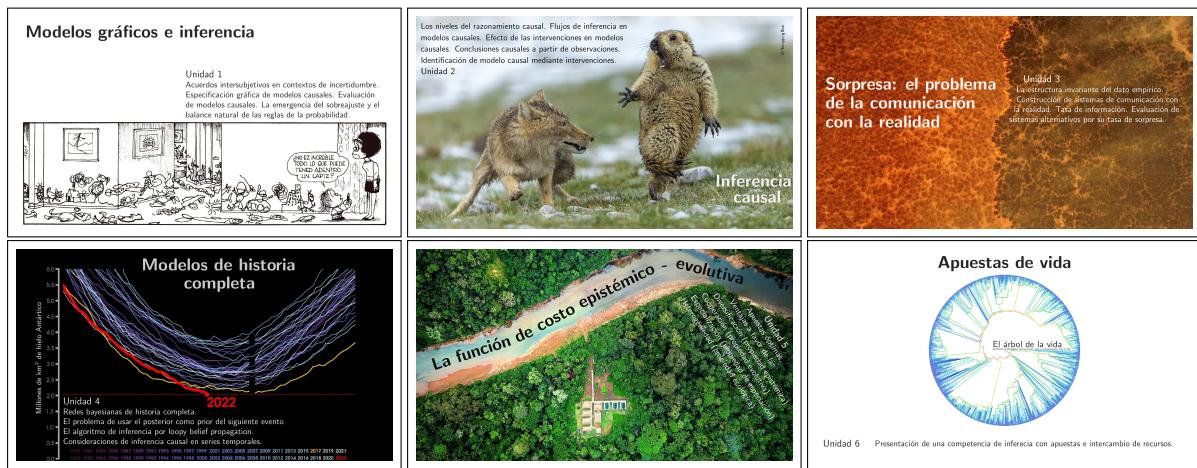


Figura 6.4: Unidades del Taller “Verdades empíricas”.

En la unidad 1 se presenta una introducción a los *Modelos gráficos e inferencia*. Allí se justifican las reglas de la probabilidad, se presentan los métodos gráfico de especificación de modelos probabilísticos y se aplican las reglas de la probabilidad para evaluar dos modelos causales alternativos. Se discute el problema computacional de las reglas de la probabilidad y se muestran soluciones bayesianas del siglo 18 y soluciones de la física estadística del siglo 19. Finalmente se revisa la emergencia del *overfitting* en el enfoque frequentista y se muestra que la aplicación estricta de las reglas de la probabilidad resuelve naturalmente este problema. Aplicaciones: Monty Hall extendido y regresión lineal bayesiana.

En la unidad 2 se presenta una introducción a la *Inferencia causal*. Se discute las dificultades que existen: para estimar correctamente el efecto causal cuando no se cuenta con un modelo causal; y para identificar el verdadero modelo causal cuando solo se cuenta con datos observacionales. Se muestra cómo los niveles de razonamiento causal (asociacional, intervencional y contrafactual) emergen naturalmente del proceso generativo de los datos. Se revisa en qué condiciones se abren y cierran los flujos de inferencia en los modelos causales (*d-separation*). Se presenta el criterio *backdoor* para la identificación de efectos causales a partir de observaciones (sin intervenciones). Se evalúan diversos escenarios de estimación de efectos causales. Finalmente definimos el concepto de teorías causales y mostramos cómo

identificarlas a partir de intervenciones. Aplicaciones: cómputo del efecto causal en modelos lineales bayesianos y evaluación de teorías causales alternativas con datos sintéticos.

En la unidad 3 se presenta una introducción a *El problema de comunicación con la realidad*. Se comienza discutiendo los niveles de base empírica y se identifica el rol que cumplen las hipótesis en la construcción de los datos de base empírica metodológica. Se revisa la estructura invariante los datos de base empírica metodológica y su equivalencia con el esquema emisor-receptor de la teoría de la información: hipótesis oculta (fuente), hipótesis observable (transmisor), procedimientos (canal), base empírica (receptor), y estimación o probabilidad (destino). Se muestra que la tasa de predicción de dos sistemas de comunicación medida en órdenes de magnitud, resulta ser el negativo de la entropía (cuando el modelo causal es correcto) o el negativo de la entropía cruzada (cuando el modelo causal es incorrecto). Se revisa la propiedad de la entropía como medida de la información. Se define formalmente el principio de no mentir como maximización de la entropía dada restricciones y se presenta la familia de distribuciones exponenciales. Se evalúan sistemas de comunicación con la realidad que difieren, no en los modelos causales, sino en los procedimientos que se usan para transmitir las señales. Aplicaciones: evaluación de estimadores de habilidad alternativos y evaluación de los test diagnósticos de Chagas.

En la unidad 4 se presenta una introducción a los *Modelos de historia completa*. Se motiva el problema mediante los modelos habitualmente utilizados en la industria del videojuego, como TrueSkill, que usan el posterior de un evento como prior del siguiente, propagando la información en una única dirección, del pasado al futuro. Se presenta el algoritmo suma-producto, que permite realizar la inferencia mediante pasaje de mensaje entre los nodos de la red causal (expresada como un *factor graph*). Se usan esas reglas para derivar el posterior exacto en el modelo TrueSkill, y se revisan los problemas de estimación que efectivamente tiene el modelo TrueSkill. Se especifica el modelo causal de historia completa, que vincula todos los eventos en una misma red causal. Se revisa el algoritmo iterativo que propaga la información histórica por toda la red causal, y se verifica la mejora de las estimaciones. Se finaliza discutiendo la evaluación de efectos causales en series temporales. Aplicaciones: estimador de habilidad estado del arte en la industria del videojuego y apuestas en tenis en base a pagos históricos de una casa de apuestas.

En la unidad 5 se presenta una introducción a la teoría (ergódica) de *toma de decisiones*. Revisamos la naturaleza multiplicativa de la función de costo de la teoría de la probabilidad (la regla del producto) y de la teoría de la evolución (la secuencia de tasas de reproducción y supervivencia). Presentamos un problema de apuestas binario, en el que nos vemos obligados a apostar todos nuestros recursos en cada vuelta, dividiendo una proporción b por cara y $(1 - b)$ por sello. Mostramos en ese ejemplo que no cualquier función de costo permite calcular la tasa de crecimiento que efectivamente se obtiene en el tiempo e introducimos el concepto de ergodicidad. Derivamos la propiedad epistémica que explica el funcionamiento de la teoría de la probabilidad. Derivamos la propiedad evolutiva, que le permite a la vida aumentar la tasa de crecimiento reduciendo fluctuaciones mediante la cooperación. Derivamos la propiedad de especiación, que le permite a grupos cooperativos aumentar un más la tasa de crecimiento mediante la especialización cooperativa. Discutimos el error de la teoría de decisiones clásica, mostrando que no existe un dilema del prisionero bajo procesos como los epistémicos o evolutivos, que son de naturaleza multiplicativa. Aplicaciones: un problema de inferencia con apuestas e intercambio de recursos (“apuestas de vida”).

A futuro me propongo seguir extendiendo los contenidos teóricos y prácticos de este curso.

6.3. Praxis

Las áreas de investigación en aprendizaje automático, inteligencia artificial y ciencia de datos siguen desarrollando una multitud de algoritmos genéricos sin interpretación causal, que si bien permiten usar las dependencias entre variables para predecir, no permiten sacar conclusiones causales y, por lo tanto, tampoco recomendaciones para la intervención [48]. Al mismo tiempo, quienes trabajan en ciencias sociales (y otras ciencias empíricas) proponen un enorme cantidad de argumentos causales en lenguaje natural, que si bien están fundados en el conocimiento experto del área, al no estar especificados en términos probabilísticos no pueden ser evaluados correctamente.

La inferencia causal es un área en crecimiento dentro de la inteligencia artificial [48, 121], fundamental para tomar decisiones en contextos de incertidumbre. En las últimas décadas se fueron desarrollando técnicas que permiten aprovechar toda la potencia del aprendizaje automático probabilístico [123, 124] de forma flexible e intuitiva, para crear modelos personalizados a la medida de cada problema específico. A este enfoque también se lo conoce como “aprendizaje automático basado en modelos” [49, 125]. El desarrollo de los métodos gráficos ofrece hoy la capacidad de desarrollar flujos de trabajo interdisciplinarios, basados en un lenguaje intuitivo común (las redes bayesianas causales). El desarrollo de la programación probabilística por su parte, permite resolver eficientemente las consecuencias lógicas de los modelos causales.

Existe así una complementariedad que todavía no está siendo explotada: las ciencia de datos tiene la oportunidad de hacer uso de los modelos causales que se desarrollan en diversas ciencias empíricas, y las ciencias empíricas tienen la oportunidad de crear y evaluar modelos a la medida de cada problema específico [49, 123–125]. Los argumentos causales provenientes de las ciencias sociales (y otras ciencias empíricas) le va a permitir a la ciencia de datos sacar conclusiones causales. Y la especificación probabilística de los argumentos causales le va a permitir a la ciencias sociales evaluar correctamente sus hipótesis, poniendo en valor el conocimiento experto actual. La complejidad del problema pone en tensión las metodologías actuales de las ciencias de datos y, por lo tanto, se presenta como una oportunidad para la innovación.

Bibliografía

- [1] Aplin L. Culture in birds. *Current Biology*. 2022;32(20):R1136–R1140.
- [2] Boyd R, Richerson PJ. Why culture is common, but cultural evolution is rare. In: *Proceedings-British Academy*. vol. 88. OXFORD UNIVERSITY PRESS INC.; 1996. p. 77–94.
- [3] Tomasello M, Kruger AC, Ratner HH. Cultural learning. *Behavioral and Brain Sciences*. 1993;16(3):495–511. doi:10.1017/S0140525X0003123X.
- [4] Boyd R, Richerson PJ, Henrich J. The cultural niche: Why social learning is essential for human adaptation. *Proceedings of the National Academy of Sciences of the United States of America*. 2011;108(Supplement 2):10918–10925. doi:10.1073/pnas.1100290108.
- [5] Derex M, Mesoudi A. Cumulative Cultural Evolution within Evolving Population Structures. *Trends in Cognitive Sciences*. 2020;24(8):654–667. doi:10.1016/j.tics.2020.04.005.
- [6] Minka T, Cleven R, Zaykov Y. TrueSkill 2: An Improved Bayesian Skill Rating System. Microsoft; 2018. MSR-TR-2018-8. Available from: <https://www.microsoft.com/en-us/research/publication/trueskill-2-improved-bayesian-skill-rating-system/>.
- [7] Czégel D, Zachar I, Szathmáry E. Multilevel selection as Bayesian inference, major transitions in individuality as structure learning. *Royal Society open science*. 2019;6(8):190202.
- [8] Czégel D, Giaffar H, Tenenbaum JB, Szathmáry E. Bayes and Darwin: How replicator populations implement Bayesian computations. *BioEssays*. 2022; p. 2100255.
- [9] Taylor PD, Jonker LB. Evolutionary stable strategies and game dynamics. *Mathematical biosciences*. 1978;40(1-2):145–156.
- [10] Harper M. The replicator equation as an inference dynamic. arXiv preprint arXiv:09111763. 2009;.
- [11] Shalizi CR. Dynamics of Bayesian updating with dependent data and misspecified models. *Electronic Journal of Statistics*. 2009;3:1039–1074.
- [12] Ormrod JE. *Human Learning* (Seventh Edition). U.S.A.: Pearson Education, Inc; 2017.
- [13] Hrdy SB. *Mothers and others: The Evolutionary Origins of Mutual Understanding*. Harvard University Press; 2009.

- [14] Hrdy SB, Burkart JM. The emergence of emotionally modern humans: implications for language and learning. *Philosophical Transactions of the Royal Society B-Biological Sciences*. 2020;375(1803):20190499. doi:10.1098/rstb.2019.0499.
- [15] Jones JH. Primates and the evolution of long, slow life histories. *Current Biology*. 2011;21(18):R708–R717.
- [16] Henrich J. Demography and cultural evolution: how adaptive cultural processes can produce maladaptive losses—the Tasmanian case. *American Antiquity*. 2004;69(2):197–214. doi:10.2307/4128416.
- [17] Dussel E. Sistema mundo y transmodernidad. In: *Modernidades coloniales*. El Colegio de México México DF; 2004. p. 201–226.
- [18] Weber M. *La ética protestante y el espíritu del capitalismo*. Fondo de cultura económica; 2012 [1905].
- [19] Romero JL. *La revolución burguesa en el mundo feudal*. Sudamericana; 1967.
- [20] Henrich J. *The WEIRDest people in the world: How the West became psychologically peculiar and particularly prosperous*. Penguin UK; 2020.
- [21] Dussel E. Charla Magistral de Enrique Dussel en la Casa Central de la Universidad de Chile; 2017.
- [22] Bar-On YM, Phillips R, Milo R. The biomass distribution on Earth. *Proceedings of the National Academy of Sciences*. 2018;115(25):6506–6511.
- [23] Ostrom E. Beyond markets and states: polycentric governance of complex economic systems. *American economic review*. 2010;100(3):641–72.
- [24] Ostrom E. *Governing the commons. The Evolution of Institutions for Collective Action*. Cambridge: Cambridge University Press; 1990.
- [25] Segato RL. *La crítica de la colonialidad en ocho ensayos*. Prometeo libros; 2013.
- [26] Padilla-Iglesias C, Atmore LM, Olivero J, Lupo K, Manica A, Arango Isaza E, et al. Population interconnectivity over the past 120,000 years explains distribution and diversity of Central African hunter-gatherers. *Proceedings of the National Academy of Sciences*. 2022;119(21):e2113936119.
- [27] Derex M, Perreault C, Boyd R. Divide and conquer: intermediate levels of population fragmentation maximize cultural accumulation. *Philosophical Transactions of the Royal Society B-Biological Sciences*. 2018;373(1743):20170062. doi:10.1098/rstb.2017.0062.
- [28] Creanza N, Kolodny O, Feldman MW. Greater than the sum of its parts? Modelling population contact and interaction of cultural repertoires. *Journal of the Royal Society Interface*. 2017;14(130):20170171. doi:10.1098/rsif.2017.0171.
- [29] Migliano A, Page A, Gómez-Gardeñes J, Salali G, Viguier S, Dyble M, et al. Characterization of hunter-gatherer networks and implications for cumulative culture. *Nature Human Behaviour*. 2017;1:0043.

- [30] Lazer D, Pentland A, Adamic L, Aral S, Barabási AL, Brewer D, et al. Computational Social Science. *Science*. 2009;323(5915):721–723.
- [31] Lazer DMJ, Pentland A, Watts DJ, Aral S, Athey S, Contractor N, et al. Computational social science: Obstacles and opportunities. *Science*. 2020;369(6507):1060–1062. doi:10.1126/science.aaz8170.
- [32] Miu E, Gulley N, Laland KN, Rendell L. Innovation and cumulative culture through tweaks and leaps in online programming contests. *Nature Communications*. 2018;9.
- [33] Slezak DF, Sigman M. Do not fear your opponent: Suboptimal changes of a prevention strategy when facing stronger opponents. *Journal of Experimental Psychology: General*. 2012;141(3):527.
- [34] Dafoe A, Hughes E, Bachrach Y, Collins T, McKee KR, Leibo JZ, et al. Open problems in cooperative AI. DeepMind preprint (arXiv:201208630). 2020;.
- [35] Dafoe A, Bachrach Y, Hadfield G, Horvitz E, Larson K, Graepel T. Cooperative AI: machines must learn to find common ground. *Nature*. 2021;.
- [36] Kramár J, Eccles T, Gemp I, Tacchetti A, McKee KR, Malinowski M, et al. Negotiation and honesty in artificial intelligence methods for the board game of Diplomacy. *Nature Communications*. 2022;13(1):1–15.
- [37] (FAIR) MFARDT, Bakhtin A, Brown N, Dinan E, Farina G, Flaherty C, et al. Human-level play in the game of Diplomacy by combining language models with strategic reasoning. *Science*. 2022;378(6624):1067–1074.
- [38] Halpern JY. Reasoning about uncertainty. 2nd ed. MIT press; 2017.
- [39] Ramsey FP. Truth and probability. In: Readings in Formal Epistemology: Sourcebook. Springer; 1926. p. 21–45.
- [40] Cox RT. Probability, frequency and reasonable expectation. *American Journal of Physics*. 1946;14(1):1–13. doi:10.1119/1.1990764.
- [41] Kolmogorov A. Foundations of the theory of probability. Chelsea Publishing Co.; 1950.
- [42] Samaja J. Epistemología y metodología: elementos para una teoría de la investigación científica. 165/168. EUDEBA; 1999.
- [43] Devlin K. The Unfinished Game: Pascal, Fermat, and the Seventeenth-Century Letter that Made the World Modern. 1st ed. Basic Books; 2008.
- [44] Jaynes ET. Bayesian methods: General background; 1984.
- [45] Jaynes ET. Probability Theory: The Logic of Science (edited by Larry Bretthorst). Cambridge University Press; 2003.
- [46] Jaynes ET. Information theory and statistical mechanics. *Physical review*. 1957;106(4):620.

- [47] Jaynes ET. Information theory and statistical mechanics. II. Physical review. 1957;108(2):171.
- [48] Pearl J. Causality. Cambridge university press; 2009.
- [49] Bishop CM. Pattern Recognition and Machine Learning. 1st ed. Information Science and Statistics. New York, NY: Springer-Verlag; 2006.
- [50] Thurstone LL. Psychophysical Analysis. American Journal of Psychology. 1927;38(3):368–389. doi:10.2307/1415006.
- [51] Zermelo E. The Calculation of the Results of a Tournament as a Maximum Problem in the Calculus of Probability. In: Ebbinghaus HD, Kanamori A, editors. Ernst Zermelo - Collected Works Volume II. vol. 23 of Schriften der Mathematisch-naturwissenschaftlichen Klasse. Berlin Heidelberg: Springer-Verlag; 2013. p. 616–671.
- [52] Bradley RA, Terry ME. Rank Analysis of Incomplete Block Designs: I. The Method of Paired Comparisons. Biometrika. 1952;39(3/4):324–345. doi:10.2307/2334029.
- [53] Mosteller F. Remarks on the Method of Paired Comparisons: I. The Least Squares Solution assuming Equal Standard Deviations and Equal Correlations. Psychometrika. 1951;16(1):3–9. doi:10.1007/BF02313422.
- [54] Mosteller F. Remarks on the method of paired comparisons: II. The Effect of an Aberrant Standard Deviation when Equal Standard Deviations and Equal Correlations are assumed. Psychometrika. 1951;16(2):203–206. doi:10.1007/BF02289115.
- [55] Mosteller F. Remarks on the Method of Paired Comparisons: III. A Test of Significance for Paired Comparisons when Equal Standard Deviations and Equal Correlations are assumed. Psychometrika. 1951;16(2):207–218. doi:10.1007/BF02289116.
- [56] Elo AE. The Rating of Chess Players, Past and Present. Ishi Press; 2008.
- [57] Glickman ME. Dynamic Paired Comparison Models with Stochastic Variances. Journal of Applied Statistics. 2001;28(6):673–689. doi:10.1080/02664760120059219.
- [58] Herbrich R, Minka R, Graepel T. TrueSkillTM: A Bayesian Skill Rating System. In: Schölkopf B, Platt J, Hoffman T, editors. Proceedings of the Conference in Advances in Neural Information Processing Systems. Cambridge, MA: MIT Press; 2006. p. 569–576. Available from: <https://proceedings.neurips.cc/paper/2006/file/f44ee263952e65b3610b8ba51229d1f9-Paper.pdf>.
- [59] Graepel TKH, Herbrich R. Bayesian scoring; 2006.
- [60] Minka T, Graepel TK, Herbrich R. Determining Relative Skills of Players; 2009.
- [61] Glickman M. Example of the Glicko-2 System; 2013. <http://www.glicko.net/glicko/glicko2.pdf>.
- [62] De Groot A. Thought and Choice in Chess. 2nd ed. Psychological Studies. Mouton De Gruyter; 1978.

- [63] Chase WG, Simon HA. Perception in chess. *Cognitive psychology*. 1973;4(1):55–81.
- [64] Simon HA. How big is a chunk? *Science*. 1974;183(4124):482–488.
- [65] Newell A, Rosenbloom PS, Anderson JR. Mechanisms of skill acquisition and the law of practice. In: *Cognitive skills and their acquisition*. Psychology Press; 1981. p. 1–55.
- [66] Heathcote A, Brown S, Mewhort DJ. The power law repealed: The case for an exponential law of practice. *Psychonomic bulletin & review*. 2000;7(2):185–207.
- [67] Howard RW. Learning curves in highly skilled chess players: a test of the generality of the power law of practice. *Acta psychologica*. 2014;151:16–23.
- [68] Hoppitt W, Laland KN. Social learning: an introduction to mechanisms, methods, and models. Princeton University Press; 2013.
- [69] Bandura A. *Social Learning Theory*. Prentice-Hall; 1977.
- [70] Richerson PJ, Boyd R, Henrich J. Gene-culture coevolution in the age of genomics. *Proceedings of the National Academy of Sciences of the United States of America*. 2010;107(Supplement 2):8985–8992. doi:10.1073/pnas.0914631107.
- [71] Boyd R, Richerson PJ. Culture and the evolutionary process. University of Chicago press; 1985.
- [72] Feldman MW, Aoki K, Kumm J. Individual versus social learning: evolutionary analysis in a fluctuating environment. *Anthropological Science*. 1996;104(3):209–231.
- [73] Giraldeau LA, Valone TJ, Templeton JJ. Potential disadvantages of using socially acquired information. *Philosophical Transactions of the Royal Society of London B: Biological Sciences*. 2002;357(1427):1559–1566.
- [74] Henrich J, McElreath R. The evolution of cultural evolution. *Evolutionary Anthropology: Issues, News, and Reviews*. 2003;12(3):123–135.
- [75] Rendell L, Fogarty L, Hoppitt WJ, Morgan TJ, Webster MM, Laland KN. Cognitive culture: theoretical and empirical insights into social learning strategies. *Trends in cognitive sciences*. 2011;15(2):68–76.
- [76] Rendell L, Boyd R, Cownden D, Enquist M, Eriksson K, Feldman MW, et al. Why copy others? Insights from the social learning strategies tournament. *Science*. 2010;328(5975):208–213.
- [77] Henrich J, Broesch J. On the nature of cultural transmission networks: evidence from Fijian villages for adaptive learning biases. *Philosophical Transactions of the Royal Society of London B: Biological Sciences*. 2011;366(1567):1139–1148.
- [78] Mesoudi A. An experimental comparison of human social learning strategies: payoff-biased social learning is adaptive but underused. *Evolution and Human Behavior*. 2011;32(5):334–342.

- [79] Toelch U, Bruce MJ, Newson L, Richerson PJ, Reader SM. Individual consistency and flexibility in human social information use. *Proceedings of the Royal Society of London B: Biological Sciences*. 2014;281(1776):20132864.
- [80] Caldwell CA, Cornish H, Kandler A. Identifying innovation in laboratory studies of cultural evolution: rates of retention and measures of adaptation. *Phil Trans R Soc B*. 2016;371(1690):20150193.
- [81] Muthukrishna M, Morgan TJ, Henrich J. The when and who of social learning and conformist transmission. *Evolution and Human Behavior*. 2016;37(1):10–20.
- [82] Henrich J, Boyd R, Bowles S, Camerer C, Fehr E, Gintis H, et al. In search of homo economicus: behavioral experiments in 15 small-scale societies. *The American Economic Review*. 2001;91(2):73–78.
- [83] Efferson C, Richerson PJ, McElreath R, Lubell M, Edsten E, Waring TM, et al. Learning, productivity, and noise: an experimental study of cultural transmission on the Bolivian Altiplano. *Evolution and Human Behavior*. 2007;28(1):11–17.
- [84] Wisdom TN, Song X, Goldstone RL. Social learning strategies in networked groups. *Cognitive Science*. 2013;37(8):1383–1425.
- [85] Glowacki L, Molleman L. Subsistence styles shape human social learning strategies. *Nature Human Behavior*. 2017;1(5):0098. doi:10.1038/s41562-017-0098.
- [86] Beheim BA, Thigpen C, McElreath R. Strategic social learning and the population dynamics of human behavior: The game of Go. *Evolution and Human Behavior*. 2014;35(5):351–357. doi:10.1016/j.evolhumbehav.2014.04.001.
- [87] Johnson NF, Xu C, Zhao Z, Ducheneaut N, Yee N, Tita G, et al. Human group formation in online guilds and offline gangs driven by a common team dynamic. *Phys Rev E*. 2009;79:066117.
- [88] Sigman M, Etchemendy P, Fernandez Slezak D, Cecchi G. Response Time Distributions in Rapid Chess: A Large-Scale Decision Making Experiment. *Frontiers in Neuroscience*. 2010;4:60.
- [89] DeLong C. Team chemistry: the missing link in skill assessment for teams; 2013.
- [90] Shim KJ, Srivastava J. Team performance prediction in massively multiplayer online role-playing games (MMORPGs). In: *Social Computing (SocialCom), 2010 IEEE Second International Conference on*. IEEE; 2010. p. 128–136.
- [91] Chabris CF, Glickman ME. Sex differences in intellectual performance analysis of a large cohort of competitive chess players. *Psychological Science*. 2006;17(12):1040–1046.
- [92] Roring RW, Charness N. A multilevel model analysis of expertise in chess across the life span. *Psychology and aging*. 2007;22(2):291.
- [93] Vaci N, Bilalić M. Chess databases as a research vehicle in psychology: Modeling large data. *Behavior Research Methods*. 2016; p. 1–14.

- [94] Kschischang FR, Frey BJ, Loeliger HA. Factor Graphs and the Sum-Product Algorithm. *IEEE Transactions on Information Theory*. 2001;47(2):498–519. doi:10.1109/18.910572.
- [95] Landfried G, Slezak DF, Mocskos EE. Data from: Faithfulness-boost effect: loyal teammate selection correlates with skill acquisition improvement in online games [Dataset]. Dryad. <https://doi.org/10.5061/dryad.888gm50>. Dryad. 2020;doi:doi:10.5061/dryad.888gm50.
- [96] Almond RG, Mislevy RJ, Steinberg LS, Yan D, Williamson DM. Bayesian networks in educational assessment. Springer; 2015.
- [97] Glickman ME. Parameter Estimation in Large Dynamic Paired Comparison Experiments. *Journal of the Royal Statistical Society Series C-Applied Statistics*. 1999;48(3):377–394. doi:10.1111/1467-9876.00159.
- [98] van der Linden WJ. Handbook of Item Response Theory, Volume 1: Models. Statistics in the Social and Behavioral Sciences. Chapman and Hall/CRC; 2016.
- [99] Fox JP. Bayesian item response modeling: Theory and applications. Statistics for Social and Behavioral Sciences. Springer; 2010.
- [100] Coulom R. Whole-History Rating: A Bayesian Rating System for Players of Time-Varying Strength. In: van den Herik HJ, Xu X, Ma Z, Winands MHM, editors. Proceedings of the 6th International Conference on Computers and Games. Berlin, Heidelberg: Springer; 2008. p. 113–124.
- [101] Maystre L, Kristof V, Grossglauser M. Pairwise Comparisons with Flexible Time-Dynamics. In: Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. New York, NY, USA: Association for Computing Machinery; 2019. p. 1236–1246.
- [102] Landfried G. TrueSkill Through Time: the Julia, Python and R packages.; 2021. <https://github.com/glandfried/TrueSkillThroughTime>.
- [103] Dangauthier P, Herbrich R, Minka T, Graepel T. TrueSkill Through Time: Revisiting the History of Chess. In: Platt JC, Koller D, Singer Y, Roweis ST, editors. Proceedings of the Twenty-First Annual Conference on Neural Information Processing Systems. vol. 20. New York: Curran Associates, Inc.; 2007. p. 337–344.
- [104] Minka T. Divergence Measures and Message Passing. Microsoft Research; 2005. MSR-TR-2005-173. Available from: <https://www.microsoft.com/en-us/research/publication/divergence-measures-and-message-passing/>.
- [105] Mosser J. The Math Behind TrueSkill; 2011. Available from: <http://www.moserware.com/assets/computing-your-skill/The%20Math%20Behind%20TrueSkill.pdf>.
- [106] Lee H. TrueSkill. The Video Game Rating System.; 2012. Available from: <https://github.com/sublee/trueskill/>.
- [107] Guo S, Sanner S, Graepel T, Buntine W. Score-Based Bayesian Skill Learning. In: Flach PA, De Bie T, Cristianini N, editors. Proceedings of the Joint European Conference on

Machine Learning and Knowledge Discovery in Databases (ECML PKDD). vol. 7523 of Lecture Notes in Computer Science. Berlin, Heidelberg: Springer-Verlag; 2012. p. 106–121.

- [108] Koster J, McElreath R, Hill K, Yu D, Shepard G, Van Vliet N, et al. The Life History of Human Foraging: Cross-Cultural and Individual Variation. *Science Advances*. 2020;6(26). doi:10.1126/sciadv.aax9070.
- [109] Derex M, Boyd R. The foundations of the human cultural niche. *Nature Communications*. 2015;6(1):8398. doi:10.1038/ncomms9398.
- [110] Shennan S. Demography and cultural innovation: a model and its implications for the emergence of modern human culture. *Cambridge archaeological journal*. 2001;11(1):5–16.
- [111] Kempe M, Mesoudi A. An experimental demonstration of the effect of group size on cultural accumulation. *Evolution and Human Behavior*. 2014;35(4):285–290.
- [112] Campbell M, Hoane Jr AJ, Hsu Fh. Deep Blue. *Artificial Intelligence*. 2002;134(1):57–83. doi:10.1016/S0004-3702(01)00129-1.
- [113] Silver D, Huang A, Maddison CJ, Guez A, Sifre L, Van Den Driessche G, et al. Mastering the game of Go with deep neural networks and tree search. *Nature*. 2016;529(7587):484–489. doi:10.1038/nature16961.
- [114] Kossinets G, Watts DJ. Empirical Analysis of an Evolving Social Network. *Science*. 2006;311(5757):88–90. doi:10.1126/science.1116869.
- [115] McGregor A. Graph stream algorithms: a survey. *ACM SIGMOD Record*. 2014;43(1):9–20.
- [116] Landfried G, Mocsos E. TrueSkill Through Time: reliable initial skill estimates and historical comparability with Julia, Python and R.; 2023. Preprint at <https://github.com/glandfried/TrueSkillThroughTime/releases/download/doc/landfried-learning.pdf>.
- [117] Hagberg A, Conway D. NetworkX: Network Analysis with Python. URL: <https://networkxgithubio>. 2020;
- [118] Rezapour MM, Fatemi A, Nematbakhsh MA. Learning experience assessment through players chat content in multiplayer online games. *Computers in Human Behavior*. 2023; p. 108003.
- [119] Arnaboldi V, Conti M, Passarella A, Dunbar RI. Online social networks and information diffusion: The role of ego networks. *Online Social Networks and Media*. 2017;1:44–55.
- [120] Landfried G, Slezak DF, Mocsos E. Faithfulness-boost effect: Loyal teammate selection correlates with skill acquisition improvement in online games. *PLoS ONE*. 2019;14(3):e0211014.
- [121] Peters J, Janzing D, Schölkopf B. Elements of causal inference: foundations and learning algorithms. The MIT Press; 2017.

- [122] Winn J. Causality with gates. In: Artificial Intelligence and Statistics. Proceedings of Machine Learning Research; 2012. p. 1314–1322.
- [123] Murphy KP. Probabilistic Machine Learning: An introduction. MIT Press; 2022. Available from: probml.ai.
- [124] Murphy KP. Probabilistic Machine Learning: Advanced Topics. MIT Press; 2023. Available from: <http://probml.github.io/book2>.
- [125] Bishop CM. Model-based machine learning. Philosophical Transactions of the Royal Society of London Series A-Mathematical Physical and Engineering Sciences. 2013;371(1984):20120222. doi:10.1098/rsta.2012.0222.
- [126] Krizhevsky A, Sutskever I, Hinton GE. Imagenet classification with deep convolutional neural networks. Advances in neural information processing systems. 2012;25.
- [127] Izmailov P, Vikram S, Hoffman MD, Wilson AGG. What are Bayesian neural network posteriors really like? In: International conference on machine learning. PMLR; 2021. p. 4629–4640.
- [128] Wilson AG, Izmailov P. Bayesian deep learning and a probabilistic perspective of generalization. Advances in neural information processing systems. 2020;33:4697–4708.

Capítulo 7

Anexo

7.1. TrueSkill Through Time

7.1.1. Evolución de habilidad

Se adjuntan los códigos de *Julia* y *R* que resuelven el ejemplo presentado en la sección 3.3.3 sobre la evolución de habilidad de un jugador nuevo.

Código 7.1: Código *Julia*

```
using Random; Random.seed!(999); N = 1000
function skill(experience, middle, maximum, slope)
    return maximum/(1+exp(slope*(-experience+middle)))
end
target = skill.(1:N, 500, 2, 0.0075)
opponents = Random.randn.(1000)*0.5 .+ target

composition = [[["a"], [string(i)]] for i in 1:N]
results = [r? [1.,0.]:[0.,1.] for r in (Random.randn(N).+target.>Random.randn(N).+opponents)]
times = [i for i in 1:N]
priors = Dict{String,Player}()
for i in 1:N priors[string(i)] = Player(Gaussian(opponents[i], 0.2)) end

h = History(composition, results, times, priors, gamma=0.015)
convergence(h)
mu = [tp[2].mu for tp in learning_curves(h) ["a"]]
```

Código 7.2: Código *R*

```
N = 1000
skill <- function(experience, middle, maximum, slope){
    return(maximum/(1+exp(slope*(-experience+middle)))) }
target = skill(seq(N), 500, 2, 0.0075)
opponents = rnorm(N,target,0.5)

composition = list(); results = list(); times = c(); priors = hash()
for(i in seq(N)){composition[[i]] = list(c("a"), c(toString(i)))}
for(i in
    seq(N)){results[[i]]=if(rnorm(1,target[i])>rnorm(1,opponents[i])){c(1,0)}else{c(0,1)}}
for(i in seq(N)){times = c(times,i)}
for(i in seq(N)){priors[[toString(i)]] = Player(Gaussian(opponents[i],0.2))}

h = History(composition, results, times, priors, gamma=0.015)
```

```

h$convergence(); lc_a = h$learning_curves()$a; mu = c()
for(tp in lc_a){mu = c(mu,tp[[2]]@mu)}

```

7.1.2. Historia de la Asociación de Tenistas Profesionales (ATP)

Se adjuntan los códigos de Python y R que resuelven el ejemplo presentado en la sección 3.3.4 sobre la estimación de habilidad de los tenistas profesionales.

Código 7.3: Código Python

```

import pandas as pd; from datetime import datetime
df = pd.read_csv('input/history.csv')

columns = zip(df.w1_id, df.w2_id, df.l1_id, df.l2_id, df.double)
composition = [[[w1,w2],[l1,l2]] if d=='t' else [[w1],[l1]] for w1, w2, l1, l2, d in columns]
days = [ datetime.strptime(t, "%Y-%m-%d").timestamp()/(60*60*24) for t in df.time_start]

h = History(composition = composition, times = days, sigma = 1.6, gamma = 0.036)
h.convergence(epsilon=0.01, iterations=10)

```

Código 7.4: Código R

```

data = read.csv("input/history.csv", header=T)

get_composition = function(x){
  res = list()
  if (x["double"]=="t"){
    res[[1]] = c(x["w1_name"],x["w2_name"])
    res[[2]] = c(x["l1_name"],x["l2_name"])
  }else{
    res[[1]] = c(x["w1_name"])
    res[[2]] = c(x["l1_name"])
  }
  return(res)
}
composition = apply(data, 1, get_composition )
days = as.numeric(as.Date(data[,"time_start"], format = "%Y-%m-%d"))

h = History(composition = composition, times = days, sigma = 1.6, gamma = 0.036)
h$convergence(epsilon=0.01, iterations=10)

```

En la figura 7.1 se incluye la estimación de habilidad TrueSkill de ATP (con 0 iteraciones).

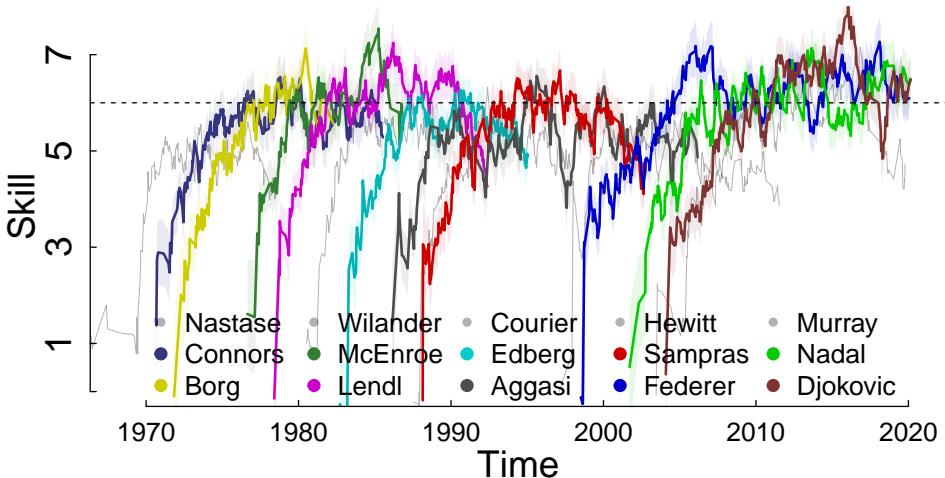


Figura 7.1: ATP TrueSkill

7.2. TrueSynergy

Uno de los supuestos del modelo TrueSkill es considerar el rendimiento de los equipos como una suma lineal de los rendimientos de los jugadores, sin considerar interacción. El modelo TrueSkill, que fue diseñado específicamente para estimar la habilidad de los jugadores individuales cuando participan en equipos, no es capaz de detectar cuándo el rendimiento de un equipo se debe a synergia entre jugadores, y cuándo se debe a la habilidad individual de cada uno de ellos.

En la primera parte realizamos simulaciones sintéticas para cuantificar el error de estimación de habilidad de TrueSkill cuando los equipos tienen un componente de sinergia. En la segunda parte evaluamos el desempeño del modelo TrueSynergy, una modificación del modelo.

7.2.1. TrueSkill: sobreestimación de habilidad en contextos de sinergia

TrueSkill fue diseñado para clacular la habilidad de los jugadores en equipo, sin considerar la posibilidad de que exista sinergia entre jugadores. En la página oficial de TrueSkill, en la sección preguntas frecuentes,

Q: You are saying that the TrueSkill ranking system assumes that the skill of a team is the sum of the skills of its players. I think this model is not appropriate: I am usually playing much better with people from my friends list rather than with random players. Will this assumption lead to incorrect rankings?

The assumption that the team skill is the sum of the skills of its players is exactly that: an assumption. The TrueSkill ranking system will use the assumption to adopt the skill points of individual players such that the team outcome can be best predicted based on the additive assumptions of the skills. Provided that you and your friends also play team games with other players now and then, *the TrueSkill ranking system will assign you a skill belief that is somewhere between the skill when you are playing with your friends and the skill when you are playing as an individual*. So, in the worst case, every other game is not with your friends: then you are slightly ranked too high when you play with random team players and slightly ranked too low when you

play with your friends. But, if you mostly play with your friends only the system will identify your skill correctly for most of your games.

El contexto de la simulación es el siguiente. Tenemos un juego donde las personas pueden jugar en equipo o en individuales. Ejemplo concreto son la palataforma ConquerClub y los campeonatos profesionales de tenis ATP y WTF. Hay un grupo de n jugadores nuevos, a quienes no se le conoce la habilidad. Estos jugadores siempre que juegan en equipo entonces juegan juntos. Una proporción de las veces, además, juegan partidas individuales. Compiten con gente siempre distinta (sistema abierto), a la que se le conoce la habilidad. Las simulaciones realizadas con sinergia permiten formular la siguiente hipótesis: *La sinergia se incorpora como skill de los jugadores en la proporción de partidas jugadas juntos.*

$$\text{sobreestimación por jugador} = \frac{\text{sinergia total del equipo}}{\text{tamaño del equipo}} (\text{proporción de partidas en equipo}). \quad (7.1)$$

A continuación detallamos los experimentos realizados

1. Dos jugadores, con sinergia equivalente a 8 tsp, juegan la mitad de las veces partidas individuales. Observación: la sobreestimación de habilidad de cada jugador es de 2 TrueSkill Points (tsp) de acuerdo a la ecuación 7.1 (Fig. 7.2(a))
2. Dos jugadores, con sinergia equivalente a 8 tsp, juegan $\frac{1}{3}$ de las veces partidas individuales. Observaciones: La sinergia se absorbe de acuerdo a la ecuación 7.1, $\frac{8}{2} \cdot \frac{2}{3} = 2,667$ (Fig. 7.2(b))
3. Dos jugadores, con sinergia equivalente a 8 tsp, juegan $\frac{1}{5}$ de las veces partidas individuales. Observaciones: La sinergia se absorbe de acuerdo a la ecuación 7.1. $\frac{8}{2} \cdot \frac{4}{5} = 3,2$ (Fig. 7.2(c)).
4. Dos jugadores, con sinergia equivalente a 8 tsp, juegan $\frac{1}{10}$ de las veces partidas individuales. Observaciones: La sinergia se absorbe de acuerdo a la ecuación 7.1. $\frac{8}{2} \cdot \frac{9}{10} = 3,6$ (Fig. 7.2(d))
5. Tres jugadores, una par de jugadores aportan sinergia equivalente a 8 tsp, las otras aportan 0. Juegan $\frac{1}{2}$ de las veces partidas individuales. Observaciones: La sinergia se absorbe de acuerdo a la ecuación 7.1 (Fig. 7.2(e))
6. Tres jugadores, las tres aristas aportan cada una sinergia equivalente a 8 tsp. Juegan $\frac{1}{2}$ de las veces partidas individuales. Observaciones: La sinergia se absorbe de acuerdo a la ecuación 7.1 (Fig. 7.2(f))

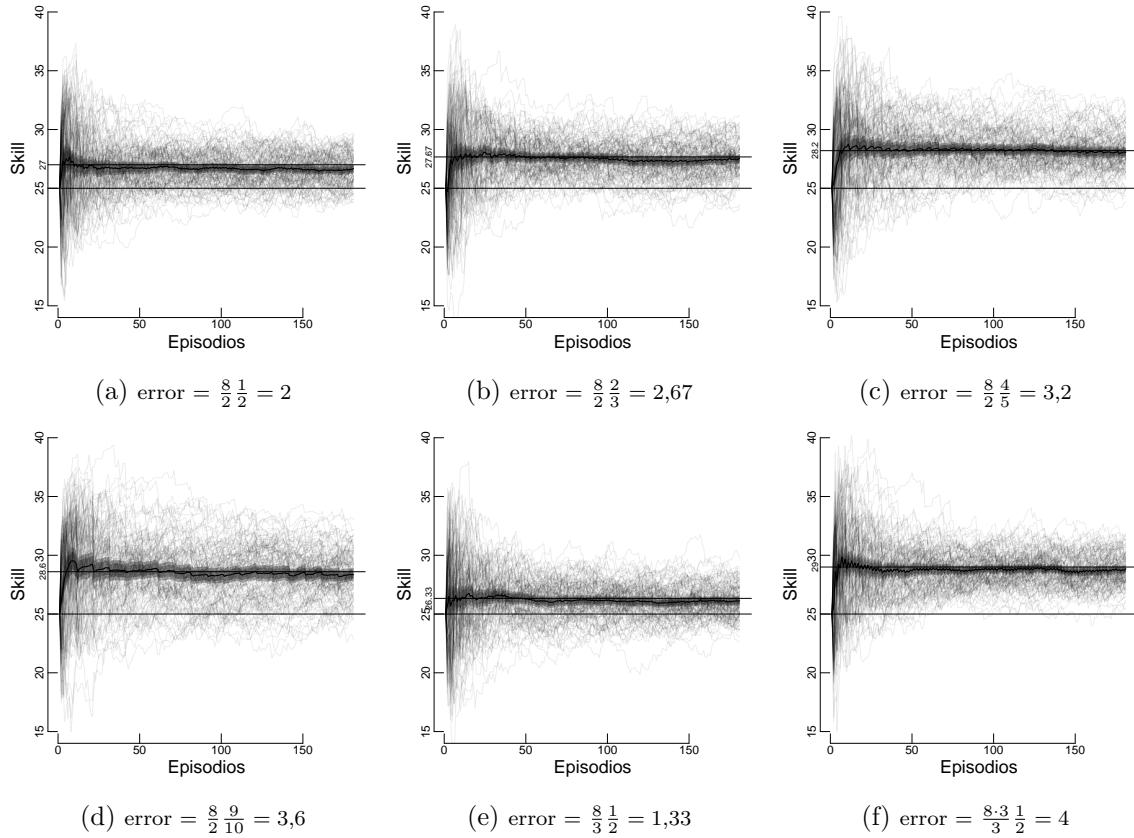


Figura 7.2: Cuantificación del error de estimación de habilidad del modelo TrueSkill cuando los equipos tienen un componente de sinergia.

Mediante ejemplos sintéticos (Fig. 7.2) observamos que en el modelo TrueSkill la sinergia se asigna equivocadamente como habilidad de los jugadores. El error de estimación del modelo TrueSkill parece estar definido por la ecuación 7.1. La magnitud del error de estimación de habilidad de los jugadores puede ser arbitrariamente grande, proporcional a la sinergia del equipo. Es necesario entonces contar con un modelo capaz de distinguir qué componentes del rendimiento de un equipo provienen de la habilidad individual de cada jugador, y qué componentes del rendimiento de un equipo provienen de la sinergia entre jugadores.

7.2.2. Modelo TrueSynergy

En esta sección proponemos el modelo TrueSynergy, una modificación del modelo TrueSkill capaz de distinguir la habilidad de los jugadores de la sinergia entre pares jugadores.

En este modelo se agrega una componente de habilidad por cada par de jugadores. Afortunadamente, la estructura de árbol no cambia, con lo cual todas las derivaciones matemáticas hechas sobre el modelo TrueSkill valen para el modelo TrueSynergy

Si hacemos una analogía con la regresión lineal multiple, el modelo TrueSkill es una regresión que tiene un parámetro por cada uno de los predictores pero que no estima ninguna de las interacciones, mientras que TrueSynergy es tiene un parámetro por cada uno de los predictores y cada una de los pares de interacciones.

Decimos que es un modelo de sinergia de primer nivel debido a que permite estimar solo la sinergia entre pares de jugadores, y no permite estimar la sinergia que se activa ante la presencia simultánea de tres o más jugadores.

7.2.3. Experimentos sintéticos con TrueSynergy

A continuación detallamos los experimentos realizados

Convergencia de habilidad y sinergia

1. Dos jugadores, con sinergia equivalente a 8 tsp, juegan 75 % de las partidas individuales (Fig. 7.3(a) y 7.3(d))
2. Dos jugadores, con sinergia equivalente a 8 tsp, 50 % de las partidas individuales (Fig. 7.3(b) y 7.3(e)).
3. Dos jugadores, con sinergia equivalente a 8 tsp, 25 % de las partidas individuales (Fig. 7.3(c) y 7.3(f)).

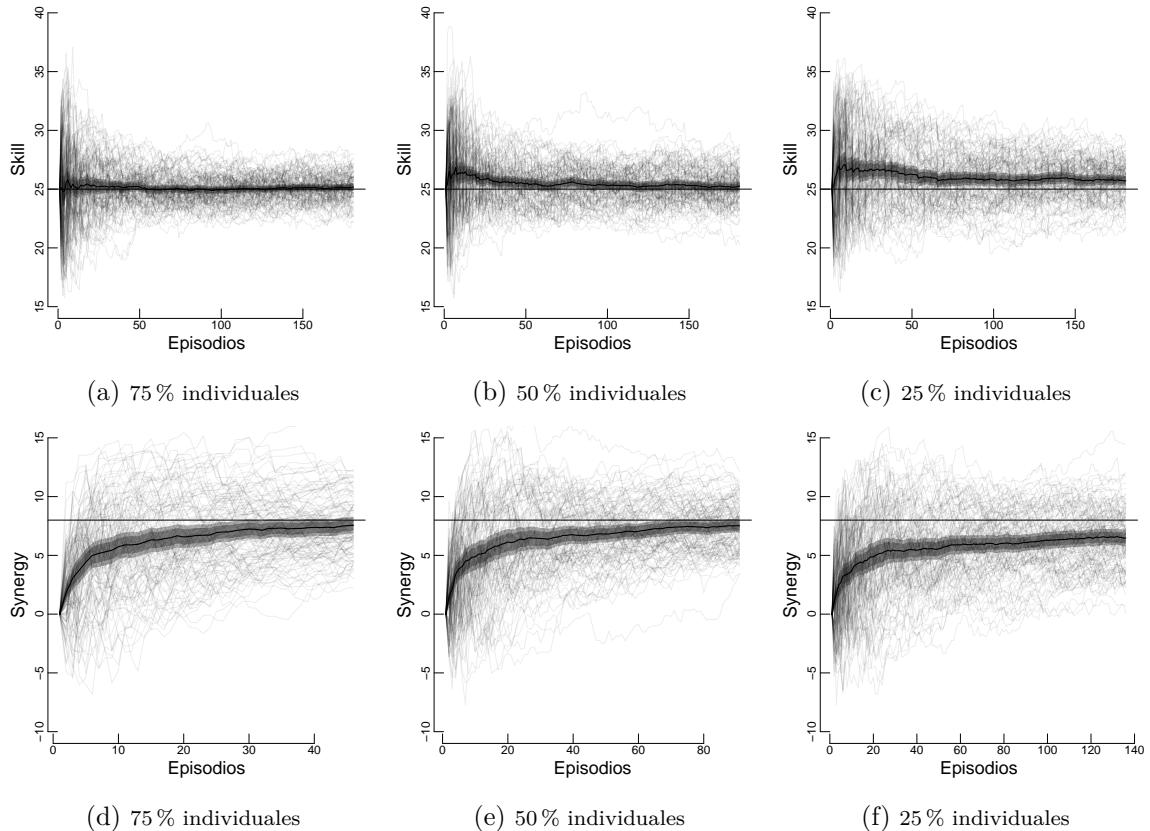


Figura 7.3

Podemos ver que el modelo TrueSynergy es capaz de distinguir el componente de sinergia respecto del componente de habilidad. La media del jugador converge aproximadamente al

valor verdadero. La convergencia es más rápida a mayor información individual de los jugadores. Con 75 % de las partidas individuales, la estimación de habilidad (Fig. 7.3(a)) y de sinergia (Fig. 7.3(d)) convergen rápidamente a su verdadero valor.

Limites a la estimación de habilidad y sinergia

A medida que contamos con menos información del jugador fuera de su equipo (50 % y 25 % de partidas individuales), la convergencia de habilidad y de sinergia se vuelve más lenta. En el límite, cuando el jugador juega el 100 % de las partidas dentro del mismo equipo, es imposible distinguir los componentes de sinergia del de habilidad. Esto no es un problema nuevo. Al igual que el modelo TrueSkill, si dos jugadores juegan siempre juntos, no es posible distinguir el componente de habilidad de cada jugador.

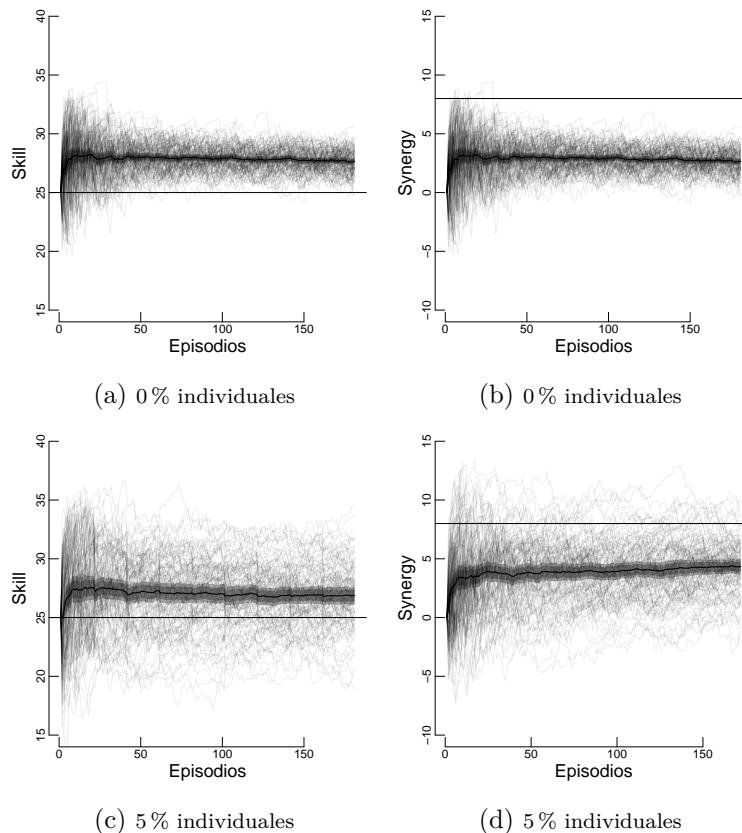


Figura 7.4

Sin más información adicional que permita romper la simetría original (i.e. los jugadores comienzan con el mismo prior y comparten todos los resultados), el rendimiento del equipo se divide en partes iguales. Esto ocurre en TrueSkill así como en TrueSynergy, y está bien que así sea. Si dos jugadores siempre juegan juntos, no es posible saber si existe entre ellos existe o no sinergia, ni a quien pertenece la habilidad. En el peor de los casos, cuando el 100 % de las partidas son dentro del mismo equipo, la habilidad y la sinergia se reparte en partes iguales. Si bien la estimación individual no es precisa, si lo es la estimación de la habilidad del equipo como un todo.

7.3. Efectos sociales sobre el aprendizaje

7.3.1. Formación de grupos

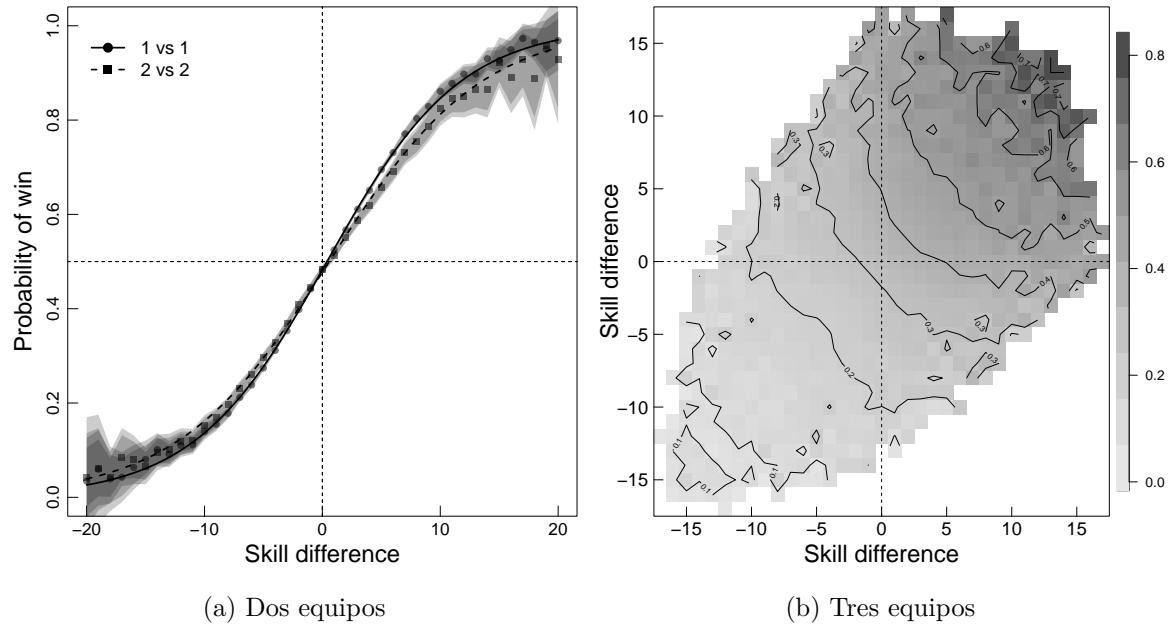


Figura 7.5: Probabilidad de ganar en función de la diferencia de habilidad estimada.

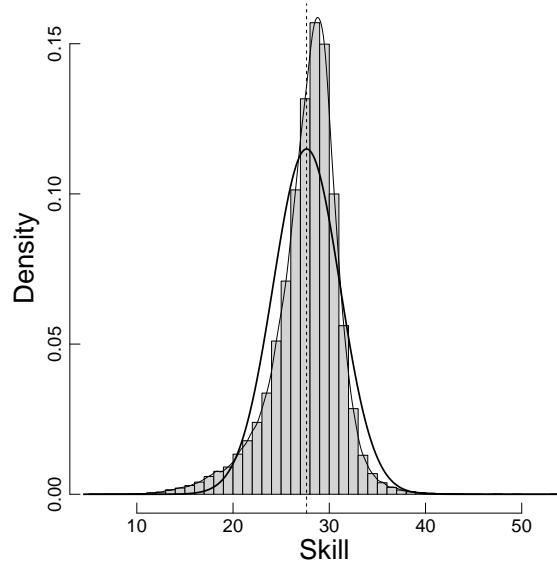


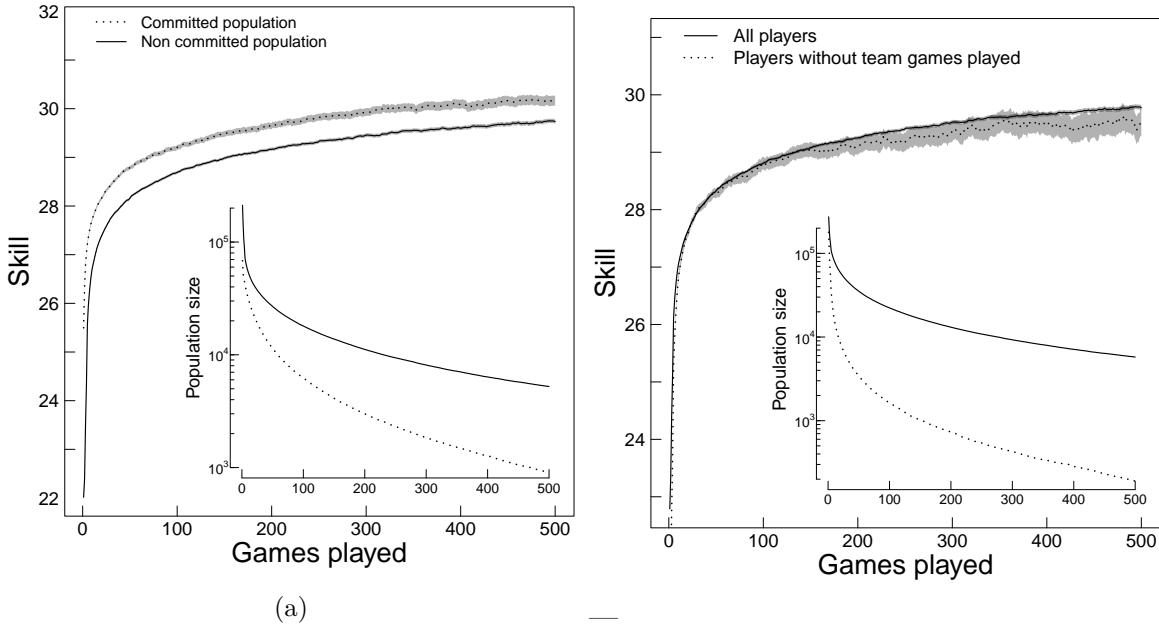
Figura 7.6: Histograma de habilidades al final (12-07-2009). La distribución empírica no es exactamente una distribución Normal. Un test Kolmogorov-Smirnov entre el histograma y la distribución Normal con misma media y varianza rechaza la hipótesis nula ($D = 0,0599$ and $p\text{-value} \approx 0$).

n	Interval	Size	9	8	7	(b)	6	(a)	5	4	3
10	1024- ∞	2997	0	0	0	0	1	1	1	1	1
9	512-1024	3141		0	0	0	0	0	1	1	1
8	256-512	5410			0	0	1	1	1	1	1
7	128-256	8490				0	0	1	1	1	1
(b)	(90-128)	5624					0	0	1	1	1
6	64-128	12108						0	1	1	1
(a)	(64-90)	6484							0	1	1
5	32-64	16481								1	1
4	16-32	20746									1
3	8-16	25319									

Tabla 7.1: Wilcoxon rank-sum test entre curvas de aprendizaje de la figura. El cero 0 representa una diferencia no significativa, uno 1 representa una diferencia significativa. Las comparaciones se realizan entre subpoblaciones: personas con al menos 8 partidas jugadas y menos de 16; con al menos 16 y menos de 32, y así sucesivamente. Agregamos dos subpoblaciones para un análisis más detallado.

TOB	size	p-value (vs Weak)	p-value (vs Medium)
Strong	103	$4,5e^{-17}$	$1,1e^{-5}$
Medium	271	$6,7e^{-9}$	
Weak	3845		

Tabla 7.2: Wilcoxon rank-sum test entre las distribuciones de habilidad empírica después de 500 partidas jugadas entre TOB fuerte, medio y débil.



(a)

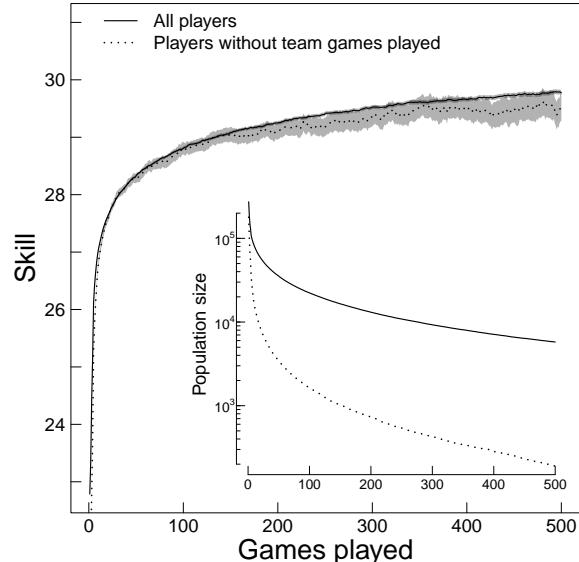


Figura 7.7: Habilidad media, la banda representa 2 errores estándar. (a) Curva de aprendizaje de la población comprometida (i.e. personas que no abandonan ninguna partida), de la población no comprometida (i.e. personas con al menos una partida abandonada). (b) Curva de aprendizaje de la población sin partidas jugadas. Como referencia mostramos la curva de aprendizaje de toda la población.

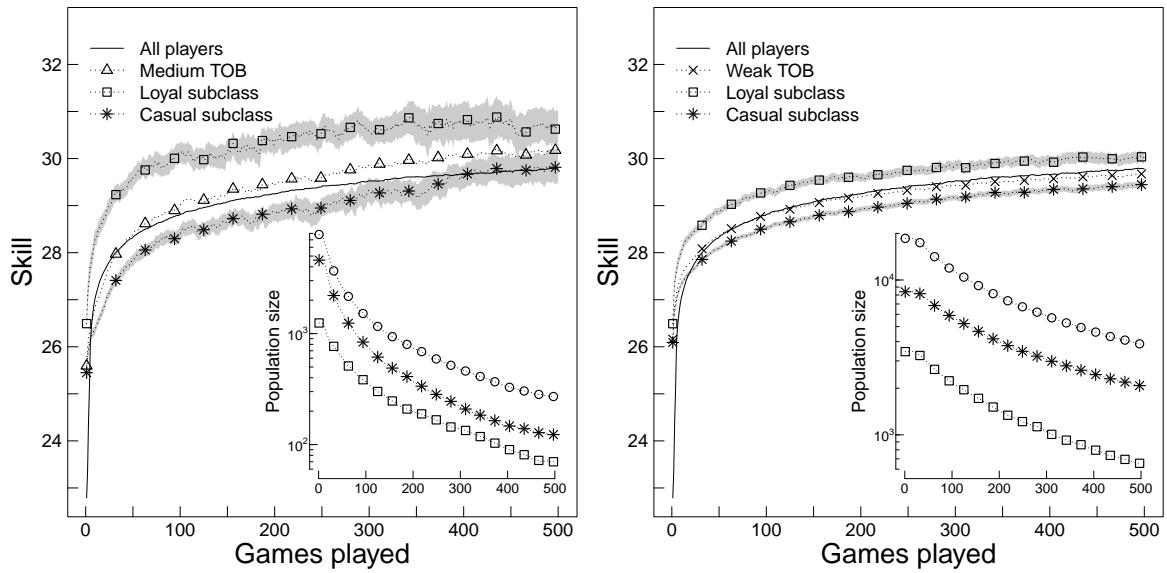


Figura 7.8: Learning curve of loyal and casual subclasses for medium (a) and weak (b) team-oriented behavior. For reference, we show the learning curve of the whole population and team-oriented behavior. The average skill is reported. The band represent 2 times the standard error.

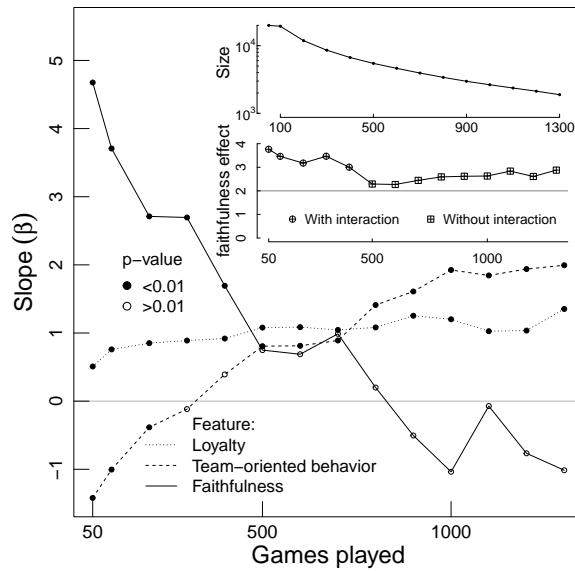


Figura 7.9: Influencia de las variables loyalty, TOB, y la interacción faithfulness sobre la adquisición de habilidades evaluada bajo misma experiencia individual. Modelo: $\text{skill}_i \sim \beta_1 \text{loyalty}_i + \beta_2 \text{TOB}_i + \beta_3 \text{faithfulness}_i$. Ajustamos un modelo lineal cada 100 partidas de experiencia, desde 100 to 1300. Agregamos una población extra con 50 partidas de experiencia. La subfigura superior muestra el tamaño de la población para cada punto de experiencia. La subfigura inferior muestra la contribución al aprendizaje cuando la interacción es $\text{faithfulness} = 1$ (i.e el efecto del faithfulness). Mostramos la contribución de la interacción hasta 400 partidas de experiencia cuando la interacción deja de ser significativa. Luego de las 400 partidas de experiencia reportamos la contribución del modelo sin interacción (i.e. $\text{skill}_i \sim \beta_1 \text{loyalty}_i + \beta_2 \text{TOB}_i$).

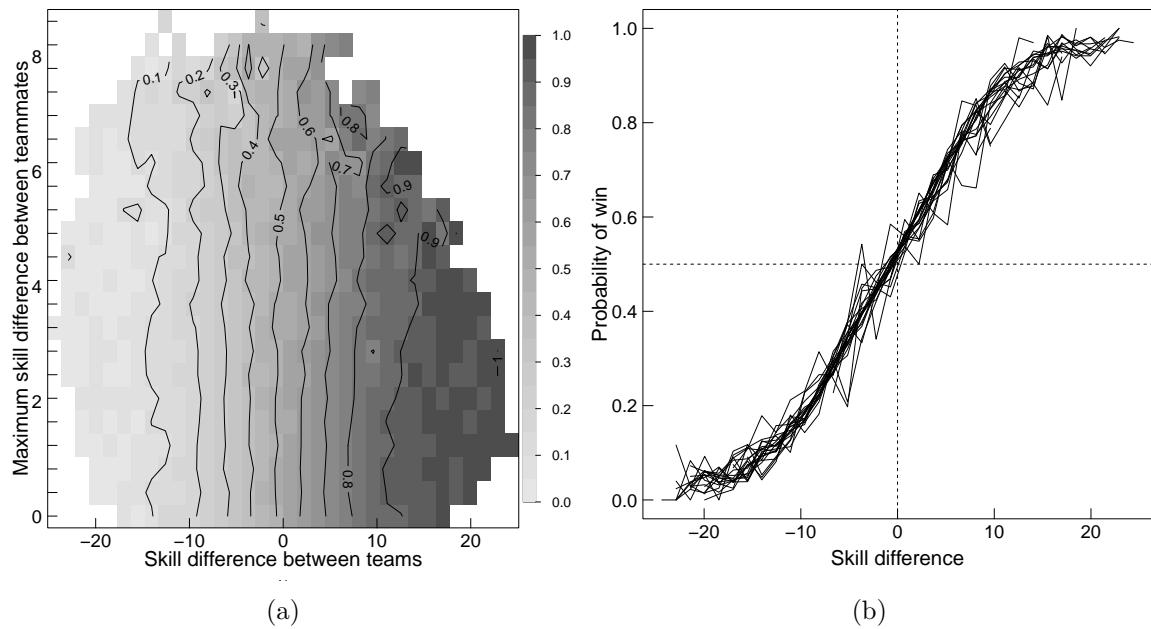
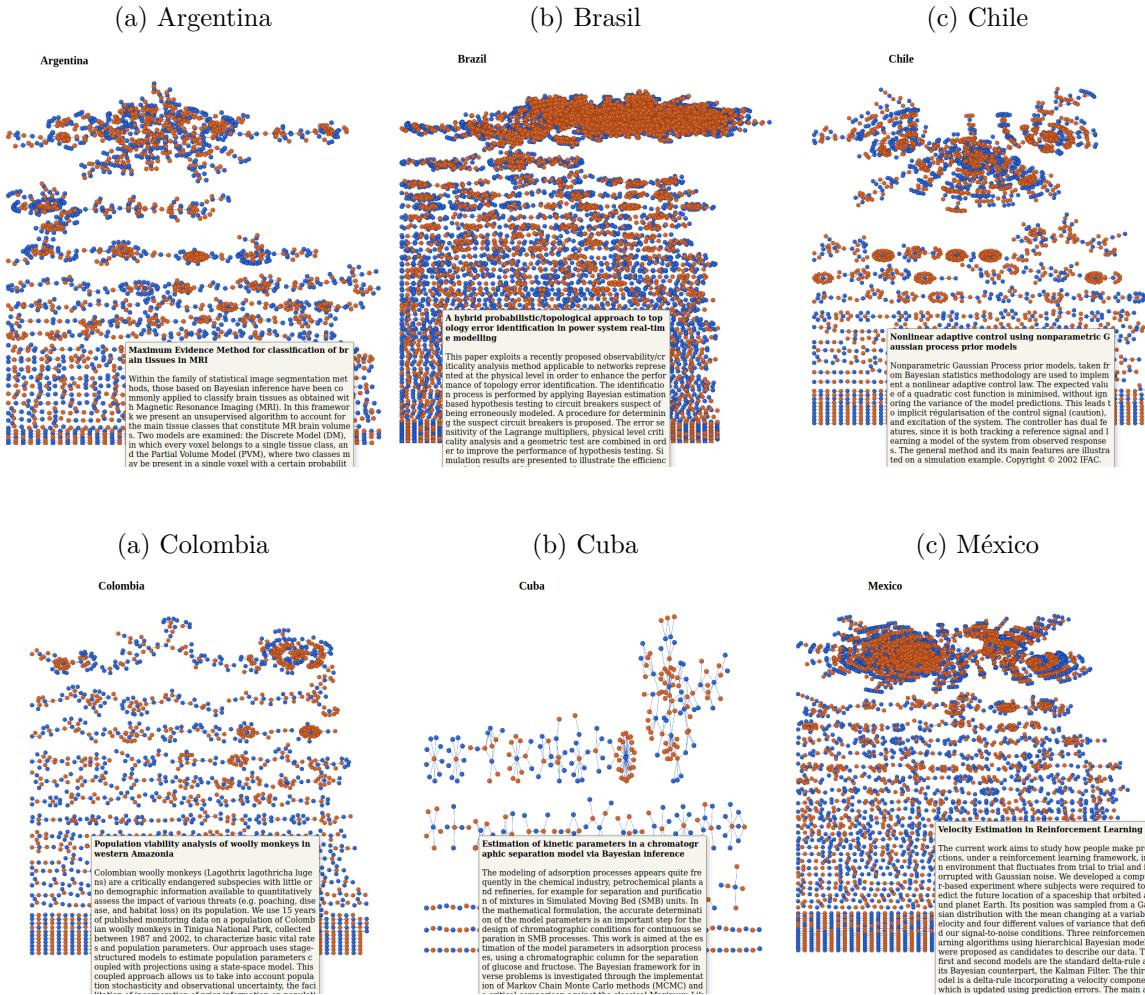


Figura 7.10: La probabilidad de ganar de los equipos en función de la diferencia de habilidad entre los miembros del mismo equipo y la diferencia entre equipos. (a) Cada parcela reporta la frecuencia observada de victorias. En cada parcela no vacía hay al menos 32 partidas. (b) Cada curva corresponde con una diferencia de habilidad determinada entre los miembros del equipo, observando la frecuencia de victorias en función de la diferencia de habilidad dentro de equipos.

7.4. Bayes Plurinacional



Figura 7.11: El gobernador de la provincia de Santiago del Estero visita a organizadores del Congreso Bayesiano Plurinacional. De izquierda a derecha, Edmundo Vizgarra, Sergio Davis Irarrázabal, el gobernador Gerardo Zamora, Gustavo Landfried, Adrián Suarez, Ricardo Montenegro, Gimena Serrano, María Della Aringa



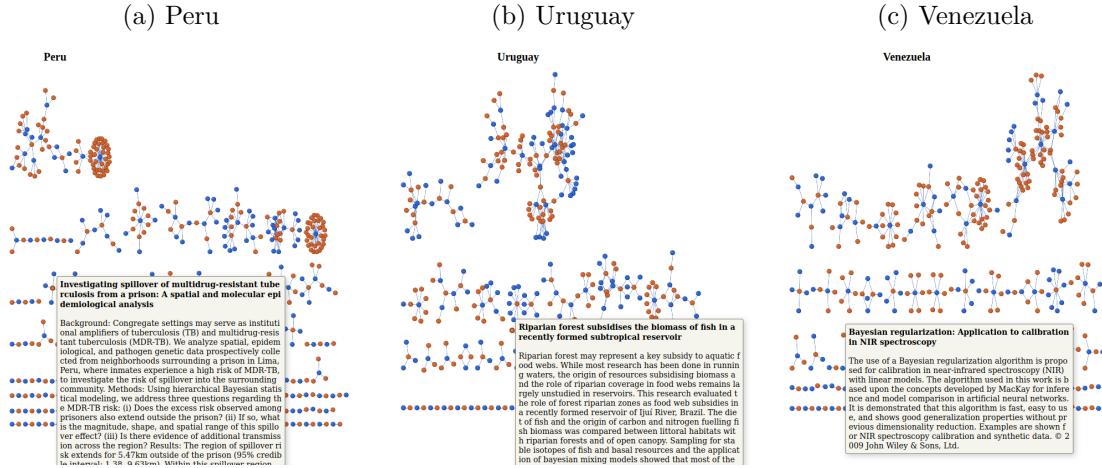


Figura 7.14: Red de co-autorías bayesianas realizadas con la colaboración de Ariel Salgado. Una versión interactiva se encuentra disponible en la web (<http://bayesdelsur.com.ar/comunidad.html>). Permite desplegar los datos de los autores (nombre y afiliaciones) y de los artículos (título, abstract, journal y año).

7.5. Propiedad de las distribución gaussiana

7.5.1. Producto de gaussianas

El problema que tenemos que resolver es

$$\int \mathcal{N}(x|\mu_1, \sigma_1^2) \mathcal{N}(x|\mu_2, \sigma_2^2) dx. \quad (7.2)$$

Por definición,

$$\begin{aligned} \mathcal{N}(x|y, \beta^2) \mathcal{N}(x|\mu, \sigma^2) &= \frac{1}{\sqrt{2\pi}\sigma_1} e^{-\frac{(x-\mu_1)^2}{2\sigma_1^2}} \frac{1}{\sqrt{2\pi}\sigma_2} e^{-\frac{(x-\mu_2)^2}{2\sigma_2^2}} \\ &= \frac{1}{2\pi\sigma_1\sigma_2} \exp\left(-\underbrace{\left(\frac{(x-\mu_1)^2}{2\sigma_1^2} + \frac{(x-\mu_2)^2}{2\sigma_2^2}\right)}_{\theta}\right). \end{aligned} \quad (7.3)$$

Luego,

$$\theta = \frac{\sigma_2^2(x^2 + \mu_1^2 - 2x\mu_1) + \sigma_1^2(x^2 + \mu_2^2 - 2x\mu_2)}{2\sigma_1^2\sigma_2^2}. \quad (7.4)$$

Expando y reordeno los factores por potencias de x

$$\frac{(\sigma_1^2 + \sigma_2^2)x^2 - (2\mu_1\sigma_2^2 + 2\mu_2\sigma_1^2)x + (\mu_1^2\sigma_2^2 + \mu_2^2\sigma_1^2)}{2\sigma_1^2\sigma_2^2}. \quad (7.5)$$

Divido al numerador y el denominador por el factor de x^2

$$\frac{x^2 - 2\frac{(\mu_1\sigma_2^2 + \mu_2\sigma_1^2)}{(\sigma_1^2 + \sigma_2^2)}x + \frac{(\mu_1^2\sigma_2^2 + \mu_2^2\sigma_1^2)}{(\sigma_1^2 + \sigma_2^2)}}{2\frac{\sigma_1^2\sigma_2^2}{(\sigma_1^2 + \sigma_2^2)}}. \quad (7.6)$$

Esta ecuación es cuadrática en x , y por lo tanto es proporcional a una función de densidad gausiana con desvío

$$\sigma_x = \sqrt{\frac{\sigma_1^2 \sigma_2^2}{\sigma_1^2 + \sigma_2^2}}, \quad (7.7)$$

y media

$$\mu_x = \frac{(\mu_1 \sigma_2^2 + \mu_2 \sigma_1^2)}{(\sigma_1^2 + \sigma_2^2)}. \quad (7.8)$$

Dado que un término $\varepsilon = 0$ puede ser agregado para completar el cuadrado en θ , esta prueba es suficiente cuando no se necesita una normalización.

$$\varepsilon = \frac{\mu_x^2 - \mu_x^2}{2\sigma_x^2} = 0. \quad (7.9)$$

Al agregar este término a θ tenemos

$$\theta = \frac{x^2 - 2\mu_x x + \mu_x^2}{2\sigma_x^2} + \underbrace{\frac{\frac{(\mu_1^2 \sigma_2^2 + \mu_2^2 \sigma_1^2)}{(\sigma_1^2 + \sigma_2^2)} - \mu_x^2}{2\sigma_x^2}}_{\varphi}. \quad (7.10)$$

Reorganizando φ

$$\begin{aligned} \varphi &= \frac{\frac{(\mu_1^2 \sigma_2^2 + \mu_2^2 \sigma_1^2)}{(\sigma_1^2 + \sigma_2^2)} - \left(\frac{(\mu_1 \sigma_2^2 + \mu_2 \sigma_1^2)}{(\sigma_1^2 + \sigma_2^2)}\right)^2}{2\frac{\sigma_1^2 \sigma_2^2}{\sigma_1^2 + \sigma_2^2}} \\ &= \frac{(\sigma_1^2 + \sigma_2^2)(\mu_1^2 \sigma_2^2 + \mu_2^2 \sigma_1^2) - (\mu_1 \sigma_2^2 + \mu_2 \sigma_1^2)^2}{\sigma_1^2 + \sigma_2^2} \frac{1}{2\sigma_1^2 \sigma_2^2} \\ &= \frac{(\mu_1^2 \sigma_1^2 \sigma_2^2 + \mu_2^2 \sigma_1^4 + \mu_1^2 \sigma_2^4 + \mu_2^2 \sigma_1^2 \sigma_2^2) - (\mu_1^2 \sigma_2^4 + 2\mu_1 \mu_2 \sigma_1^2 \sigma_2^2 + \mu_2^2 \sigma_1^4)}{\sigma_1^2 + \sigma_2^2} \frac{1}{2\sigma_1^2 \sigma_2^2} \\ &= \frac{(\sigma_1^2 \sigma_2^2)(\mu_1^2 + \mu_2^2 - 2\mu_1 \mu_2)}{\sigma_1^2 + \sigma_2^2} \frac{1}{2\sigma_1^2 \sigma_2^2} = \frac{\mu_1^2 + \mu_2^2 - 2\mu_1 \mu_2}{2(\sigma_1^2 + \sigma_2^2)} = \frac{(\mu_1 - \mu_2)^2}{2(\sigma_1^2 + \sigma_2^2)}. \end{aligned} \quad (7.11)$$

Luego,

$$\theta = \frac{(x - \mu_x)^2}{2\sigma_x^2} + \frac{(\mu_1 - \mu_2)^2}{2(\sigma_1^2 + \sigma_2^2)}. \quad (7.12)$$

Colocando θ en su lugar

$$\begin{aligned} \mathcal{N}(x|y, \beta^2) \mathcal{N}(x|\mu, \sigma^2) &= \frac{1}{2\pi\sigma_1\sigma_2} \exp\left(-\underbrace{\left(\frac{(x - \mu_x)^2}{2\sigma_x^2} + \frac{(\mu_1 - \mu_2)^2}{2(\sigma_1^2 + \sigma_2^2)}\right)}_{\theta}\right) \\ &= \frac{1}{2\pi\sigma_1\sigma_2} \exp\left(-\frac{(x - \mu_x)^2}{2\sigma_x^2}\right) \exp\left(-\frac{(\mu_1 - \mu_2)^2}{2(\sigma_1^2 + \sigma_2^2)}\right). \end{aligned} \quad (7.13)$$

Multiplicando por $\sigma_x \sigma_x^{-1}$

$$\underbrace{\frac{\sigma_x}{\sqrt{\sigma_1^2 + \sigma_2^2}}}_{\sigma_x} \frac{1}{\sigma_x} \frac{1}{2\pi\sigma_1\sigma_2} \exp\left(-\frac{(x - \mu_x)^2}{2\sigma_x^2}\right) \exp\left(-\frac{(\mu_1 - \mu_2)^2}{2(\sigma_1^2 + \sigma_2^2)}\right). \quad (7.14)$$

Luego,

$$\frac{1}{\sqrt{2\pi}\sigma_{\times}}\exp\left(-\frac{(x-\mu_{\times})^2}{2\sigma_{\times}^2}\right)\frac{1}{\sqrt{2\pi(\sigma_1^2+\sigma_2^2)}}\exp\left(-\frac{(\mu_1-\mu_2)^2}{2(\sigma_1^2+\sigma_2^2)}\right). \quad (7.15)$$

Retomando a la integral

$$\begin{aligned} I &= \int \mathcal{N}(x|\mu_{\times}, \sigma_{\times}^2) \underbrace{\mathcal{N}(\mu_1|\mu_2, \sigma_1^2 + \sigma_2^2)}_{\substack{\text{Escalar independiente de } x \\ \text{Integra 1}}} dx \\ &= \mathcal{N}(\mu_1|\mu_2, \sigma_1^2 + \sigma_2^2) \underbrace{\int \mathcal{N}(x|\mu_{\times}, \sigma_{\times}^2) dx}_{\text{Integra 1}} \\ &= \mathcal{N}(\mu_1|\mu_2, \sigma_1^2 + \sigma_2^2). \end{aligned} \quad (7.16)$$

7.5.2. Suma de gaussianas

Demostración por inducción,

Casos base

$$P(1) := \int \delta(t_1 = x_1) \mathcal{N}(x_1|\mu_1, \sigma_1^2) dx_1 = \mathcal{N}(t_1|\mu_1, \sigma_1^2). \quad (7.17)$$

La proposición $P(1)$ es verdadera dada las propiedades de la función delta de dirac.

$$\begin{aligned} P(2) &:= \iint \delta(t_2 = x_1 + x_2) \mathcal{N}(x_1|\mu_1, \sigma_1^2) \mathcal{N}(x_2|\mu_2, \sigma_2^2) dx_1 dx_2 \\ &\stackrel{7,18,1}{=} \int \mathcal{N}(x_1|\mu_1, \sigma_1^2) \mathcal{N}(t_2 - x_1|\mu_2, \sigma_2^2) dx_1 \\ &\stackrel{7,18,2}{=} \int \mathcal{N}(x_1|\mu_1, \sigma_1^2) \mathcal{N}(x_1|t_2 - \mu_2, \sigma_2^2) dx_1 \\ &\stackrel{*}{=} \underbrace{\int N(t_2|\mu_1 + \mu_2, \sigma_1^2 + \sigma_2^2)}_{\text{const.}} \underbrace{\mathcal{N}(x_1|\mu_*, \sigma_*^2) dx_1}_1 \\ &= \mathcal{N}(t_2|\mu_1 + \mu_2, \sigma_1^2 + \sigma_2^2). \end{aligned} \quad (7.18)$$

Donde $\stackrel{7,18,1}{=}$ vale por las propiedades de la función delta de dirac, $\stackrel{7,18,2}{=}$ vale por la simetría de las gaussianas, y $\stackrel{*}{=}$ vale por la demostración de multiplicación de normales en la sección 7.5.1. Luego, vale $P(2)$.

Paso inductivo $P(n) \Rightarrow P(n+1)$

Dado,

$$P(n) := \int \cdots \int \delta(t_n = \sum_{i=1}^n x_i) \left(\prod_{i=1}^n \mathcal{N}(x_i|\mu_i, \sigma_i^2) \right) dx_1 \cdots dx_n = \mathcal{N}(t|\sum_{i=1}^n \mu_i, \sum_{i=1}^n \sigma_i^2). \quad (7.19)$$

Queremos ver que $P(n+1)$ es válida.

$$P(n+1) := \int \cdots \int \delta(t_{n+1} = x_{n+1} + \sum_{i=1}^n x_i) \left(\prod_{i=1}^n \mathcal{N}(x_i|\mu_i, \sigma_i^2) \right) \mathcal{N}(x_{n+1}|\mu_{n+1}, \sigma_{n+1}^2) dx_1 \cdots dx_n dx_{n+1}. \quad (7.20)$$

Por independencia

$$\int \mathcal{N}(x_{n+1}|\mu_{n+1}, \sigma_{n+1}^2) \left(\int \cdots \int \delta(t_{n+1} = x_{n+1} + \sum_{i=1}^n x_i) \left(\prod_{i=1}^n \mathcal{N}(x_i|\mu_i, \sigma_i^2) \right) dx_1 \dots dx_n \right) dx_{n+1}. \quad (7.21)$$

Por hipótesis inductiva

$$\int \mathcal{N}(x_{n+1}|\mu_{n+1}, \sigma_{n+1}^2) \mathcal{N}(t_{n+1} - x_{n+1} | \sum_{i=1}^n \mu_i, \sum_{i=1}^n \sigma_i^2) dx_{n+1}. \quad (7.22)$$

Por la demostración de la sección 7.5.1

$$\mathcal{N}(t_{n+1} | \mu_{n+1} + \sum_{i=1}^n \mu_i, \sigma_{n+1}^2 \sum_{i=1}^n \sigma_i^2) dx_{n+1}. \quad (7.23)$$

Luego, $P(n+1)$ es válida.

7.5.3. gaussiana por acumulada de gaussiana.

Queremos resolver la integral

$$f(x) = \int \mathcal{N}(y; \mu_1, \sigma_1^2) \Phi(y + x; \mu_2, \sigma_2^2) dy. \quad (7.24)$$

Para ello trabajamos con la derivada $\frac{\partial}{\partial x} f(x) = \theta(x)$,

$$\theta(x) = \frac{\partial}{\partial x} \int \mathcal{N}(y; \mu_1, \sigma_1^2) \Phi(y + x; \mu_2, \sigma_2^2) dy, \quad (7.25)$$

$$\theta(x) = \int \mathcal{N}(y; \mu_1, \sigma_1^2) \frac{\partial}{\partial x} \Phi(y + x; \mu_2, \sigma_2^2) dy. \quad (7.26)$$

La derivada de Φ es justamente una gaussiana,

$$\begin{aligned} \theta(x) &= \int \mathcal{N}(y; \mu_1, \sigma_1^2) \mathcal{N}(y + x; \mu_2, \sigma_2^2) dy \\ &= \int \mathcal{N}(y; \mu_1, \sigma_1^2) \mathcal{N}(y; \mu_2 - x, \sigma_2^2) dy. \end{aligned} \quad (7.27)$$

Por la demostración de la sección 7.5.1 sabemos

$$\theta(x) = \mathcal{N}(\mu_1 | \mu_2 - x, \sigma_1^2 + \sigma_2^2). \quad (7.28)$$

Por simetría

$$\theta(x) = \mathcal{N}(x | \mu_2 - \mu_1, \sigma_1^2 + \sigma_2^2). \quad (7.29)$$

Retornando a $f(x)$

$$f(x) = \Phi(x | \mu_2 - \mu_1, \sigma_1^2 + \sigma_2^2). \quad (7.30)$$

7.5.4. División de gaussianas

$$\kappa = \frac{\mathcal{N}(x|\mu_f, \sigma_f^2)}{\mathcal{N}(x|\mu_g, \sigma_g^2)} = \mathcal{N}(x|\mu_f, \sigma_f^2) \mathcal{N}(x|\mu_g, \sigma_g^2)^{-1}. \quad (7.31)$$

Por definición

$$\begin{aligned} \kappa &= \frac{1}{\sqrt{2\pi}\sigma_f} e^{-\left(\frac{(x-\mu_f)^2}{2\sigma_f^2}\right)} \left(\frac{1}{\sqrt{2\pi}\sigma_g} e^{-\left(\frac{(x-\mu_g)^2}{2\sigma_g^2}\right)} \right)^{-1} \\ &= \frac{1}{\sqrt{2\pi}\sigma_f} e^{-\left(\frac{(x-\mu_f)^2}{2\sigma_f^2}\right)} \frac{\sqrt{2\pi}\sigma_g}{1} e^{\left(\frac{(x-\mu_g)^2}{2\sigma_g^2}\right)} \\ &= \frac{\sigma_g}{\sigma_f} \exp\left(-\underbrace{\left(\frac{(x-\mu_f)^2}{2\sigma_f^2} - \frac{(x-\mu_g)^2}{2\sigma_g^2}\right)}_{\theta} \right). \end{aligned} \quad (7.32)$$

Reorganizando θ

$$\begin{aligned} \theta &= \frac{(x-\mu_f)^2}{2\sigma_f^2} - \frac{(x-\mu_g)^2}{2\sigma_g^2} = \frac{\sigma_g^2(x-\mu_f)^2 - \sigma_f^2(x-\mu_g)^2}{2\sigma_f^2\sigma_g^2} \\ &= \frac{\sigma_g^2(x^2 + \mu_f^2 - 2\mu_f x) - \sigma_f^2(x^2 + \mu_g^2 - 2\mu_g x)}{2\sigma_f^2\sigma_g^2}. \end{aligned} \quad (7.33)$$

Expandimos y ordenamos en base x ,

$$\begin{aligned} \theta &= \left((\sigma_g^2 - \sigma_f^2)x^2 - 2(\sigma_g^2\mu_f - \sigma_f^2\mu_g)x + (\sigma_g^2\mu_f^2 - \sigma_f^2\mu_g^2) \right) \frac{1}{2\sigma_f^2\sigma_g^2} \\ &= \left(x^2 - \frac{2(\sigma_g^2\mu_f - \sigma_f^2\mu_g)}{(\sigma_g^2 - \sigma_f^2)}x + \frac{(\sigma_g^2\mu_f^2 - \sigma_f^2\mu_g^2)}{(\sigma_g^2 - \sigma_f^2)} \right) \frac{(\sigma_g^2 - \sigma_f^2)}{2\sigma_f^2\sigma_g^2}. \end{aligned} \quad (7.34)$$

Esto es cuadrático en x . Dado que un término $\varepsilon = 0$, independiente de x puede ser agregado para completar el cuadrado en θ , esta prueba es suficiente para determinar la media y la varianza cuando no es necesario normalizar.

$$\sigma_{\nabla} = \sqrt{\frac{\sigma_f^2\sigma_g^2}{(\sigma_g^2 - \sigma_f^2)}}, \quad (7.35)$$

$$\mu_{\nabla} = \frac{(\sigma_g^2\mu_f - \sigma_f^2\mu_g)}{(\sigma_g^2 - \sigma_f^2)}, \quad (7.36)$$

agregado $\varepsilon = \frac{\mu_{\nabla}^2 - \mu_{\nabla}^2}{2\sigma_{\nabla}^2} = 0$

$$\theta = \frac{x^2 - 2\mu_{\nabla}x + \mu_{\nabla}^2}{2\sigma_{\nabla}^2} + \underbrace{\frac{\frac{(\sigma_g^2\mu_f^2 - \sigma_f^2\mu_g^2)}{(\sigma_g^2 - \sigma_f^2)} - \mu_{\nabla}^2}{2\sigma_{\nabla}^2}}_{\varphi}. \quad (7.37)$$

Reorganizando φ

$$\begin{aligned}
\varphi &= \left(\frac{(\sigma_g^2 \mu_f^2 - \sigma_f^2 \mu_g^2)}{(\sigma_g^2 - \sigma_f^2)} - \left(\frac{(\sigma_g^2 \mu_f - \sigma_f^2 \mu_g)}{(\sigma_g^2 - \sigma_f^2)} \right)^2 \right) \frac{(\sigma_g^2 - \sigma_f^2)}{2\sigma_f^2 \sigma_g^2} \\
&= \left((\sigma_g^2 \mu_f^2 - \sigma_f^2 \mu_g^2)(\sigma_g^2 - \sigma_f^2) - \left((\sigma_g^2 \mu_f - \sigma_f^2 \mu_g) \right)^2 \right) \frac{1}{2\sigma_f^2 \sigma_g^2 (\sigma_g^2 - \sigma_f^2)} \\
&= \left(\cancel{\sigma_g^4 \mu_f^2} - \sigma_f^2 \sigma_g^2 \mu_f^2 - \sigma_f^2 \sigma_g^2 \mu_g^2 + \cancel{\sigma_f^4 \mu_g^2} - (\cancel{\sigma_g^4 \mu_f^2} + \cancel{\sigma_f^4 \mu_g^2} - 2\sigma_f^2 \sigma_g^2 \mu_f \mu_g) \right) \frac{1}{2\sigma_f^2 \sigma_g^2 (\sigma_g^2 - \sigma_f^2)}. \tag{7.38}
\end{aligned}$$

Cancelando $\sigma_f^2 \sigma_g^2$

$$\varphi = \frac{-\mu_g^2 - \mu_f^2 + 2\mu_f \mu_g}{2(\sigma_g^2 - \sigma_f^2)} = \frac{-(\mu_g - \mu_f)^2}{2(\sigma_g^2 - \sigma_f^2)}. \tag{7.39}$$

Luego θ

$$\theta = \frac{(x - \mu_{\nabla.})^2}{2\sigma_{\nabla.}^2} - \frac{(\mu_g - \mu_f)^2}{2(\sigma_g^2 - \sigma_f^2)}. \tag{7.40}$$

Volviendo a la expresión original

$$\begin{aligned}
\kappa &= \frac{\sigma_g}{\sigma_f} \exp \left(-\frac{(x - \mu_{\nabla.})^2}{2\sigma_{\nabla.}^2} + \frac{(\mu_g - \mu_f)^2}{2(\sigma_g^2 - \sigma_f^2)} \right) \\
&= \frac{\sigma_g}{\sigma_f} \exp \left(-\frac{(x - \mu_{\nabla.})^2}{2\sigma_{\nabla.}^2} \right) \exp \left(\frac{(\mu_g - \mu_f)^2}{2(\sigma_g^2 - \sigma_f^2)} \right). \tag{7.41}
\end{aligned}$$

Multiplicando por $\frac{\sqrt{2\pi}}{\sqrt{2\pi}} \frac{\sigma_{\nabla.}}{\sigma_{\nabla.}} \frac{\sqrt{\sigma_g^2 - \sigma_f^2}}{\sqrt{\sigma_g^2 - \sigma_f^2}} = 1$,

$$\begin{aligned}
\kappa &= \frac{1}{\sqrt{2\pi} \sigma_{\nabla.}} e^{-\frac{(x - \mu_{\nabla.})^2}{2\sigma_{\nabla.}^2}} \left(\frac{1}{\sqrt{2\pi(\sigma_g^2 - \sigma_f^2)}} e^{-\frac{(\mu_g - \mu_f)^2}{2(\sigma_g^2 - \sigma_f^2)}} \right)^{-1} \frac{\sigma_{\nabla.}}{\sqrt{\sigma_g^2 - \sigma_f^2}} \frac{\sigma_g}{\sigma_f} \\
&= \frac{\mathcal{N}(x | \mu_{\nabla.}, \sigma_{\nabla.}^2)}{\mathcal{N}(\mu_g | \mu_f, \sigma_g^2 - \sigma_f^2)} \frac{\sigma_g^2}{\sigma_g^2 - \sigma_f^2}. \tag{7.42}
\end{aligned}$$