

Versión
7 de julio de 2008

LÓGICA PRÁCTICA Y APRENDIZAJE COMPUTACIONAL

Jacinto A. Dávila Q.

UNIVERSIDAD DE LOS ANDES
MÉRIDA, VENEZUELA
Mérida, 2008

Índice general

Índice general	v
Licencia de uso	ix
1. Lógica Práctica	1
1.1. Introducción	1
1.2. La semántica de la lógica práctica	1
1.2.1. Sobre la validez de un argumento	4
1.3. Una realización computacional.	7
1.4. Lógica y Agentes	9
1.4.1. Una Teoría de Agentes en Lógica	9
1.4.2. Una Práctica de Agentes en Lógica	10

Índice de figuras

1	Lógica Práctica	1
1.1.	Burocratín	11
1.2.	El Usuario frente a Burocratín	11
1.3.	Matraquín	12
1.4.	Machotín	14
1.5.	Mataquín	17
1.6.	Agente Web Semántica	19
1.7.	Gea: El Agente Catastral	21
1.8.	Kally	23
1.9.	La Gramática Española de Kally	23
1.10.	El Léxico Español de Kally	24
1.11.	El Artista	27

Licencia de uso

Este libro contiene un cúmulo de información referencial sobre los sistemas Galatea¹, Gloria² y Bioinformantes³.

Copyright © ②2008 Jacinto A. Dávila

Se concede permiso de copiar, distribuir o modificar este documento bajo los términos establecidos por la licencia de documentación de GNU, GFDL, Version 1.2 publicada por la Free Software Foundation en los Estados Unidos, siempre que se coloquen secciones invariables explicando las diferencias con el original. Una copia de esta licencia se incluye al final del documento en el capítulo “GNU Free Documentation License”. Nos apegaremos a esta licencia siempre que no contradiga los términos establecidos en la legislación correspondiente de la República Bolivariana de Venezuela.

Según establece GFDL, se permite a cualquier modificar y redistribuir este material y los autores originales confiamos que otros crean apropiado y provechoso hacerlo. Esto incluye traducciones, bien a otros lenguajes naturales o a otros medios electrónicos o no.

En nuestro entender de GFDL, cualquiera puede extraer fragmentos de este texto y usarlos en un nuevo documento, siempre que el nuevo documento se acoja también a GFDL y sólo si se mantienen los créditos correspondientes a los autores originales (tal como establece la licencia).

¹<http://galatea.sourceforge.net>

²<http://gloria.sourceforge.net>

³<http://simulants.wiki.sourceforge.net>

Lógica Práctica

1.1. Introducción

Para decidir si la lógica es útil o no, uno tendría que producir un **argumento** (en favor o en contra). Entre lógicos, se dice, a modo de broma, que esta es la mejor **prueba** de la utilidad de la lógica. Muchas veces, sin embargo, no ocurre que se dude de la utilidad de la lógica en sí misma, sino que las formas de lógica disponibles tienen tal nivel de complejidad que su utilización **práctica** (distinta de su utilización en la especulación teórica) es muy limitada.

La lógica que se propone en este texto tiene 2 características que nos permitimos ofrecer para justificarle el atributo de práctica:

1. Una **semántica** relativamente simple.
2. Una realización computacional.

1.2. La semántica de la lógica práctica

En la **lógica matemática** clásica, el mundo (el universo entero y toda otra posibilidad) se ve reducido a un conjunto de objetos y a un conjunto de relaciones (entre los mismos objetos o entre las mismas relaciones). Eso es todo lo que existe (o puede existir): objetos y relaciones.

Esta **ontología** degenerada (respecto a la nuestra) de la lógica clásica es frecuentemente criticada por **reduccionista**. Los **argumentos** para esas críticas, muchos ampliamente aceptados como válidos, son tolerados (o mejor, ignorados) por los lógicos en virtud de una gran ventaja que ofrece esa ontología: Es menos compleja y, por tanto, es más fácil pensar con ella (o respecto a ella).

En definitiva, para darle significados a los símbolos en lógica contamos con:

OBJETOS designados por sus nombres, llamados también **términos** del lenguaje y

RELACIONES cuyos símbolos correspondientes son los nombres de predicados o, simplemente, predicados.

Por ejemplo, el símbolo **ama(romeo, julieta)** es una átomo constituido sintácticamente por el predicado **ama** y sus dos argumentos, los términos **romeo** y **julieta**.

Para saber qué significa esos términos y ese predicado, uno tiene que proveerse de una definición de la **semántica** de ese lenguaje. Una forma de hacerlo es usar (referirse a) una historia (como la que sugieren esos nombres) o, un poco más matemáticamente, definirlos:

romeo se refiere al jovencito hijo de los Montesco, en la Novela de William Shakespeare.

julieta se refiere a la jovencita, hija de los Capuletos, en la misma Novela.

Hay muchas formas de hacer esa definición ⁽¹⁾. En cada caso, quien la hace tiene en mente su propia **interpretación**. Uno podría, por ejemplo, decir:

romeo es el perrito de mi vecina.

y no habría razón para decir que esa interpretación es incorrecta.

Esas definiciones dan cuenta del nombre, pero no del predicado. Este se define también, pero de una forma diferente. Al predicado le corresponde una **relación**. Si hacemos la intepretación que pretende

¹vean http://es.wikipedia.org/wiki/Romeo_y_Julieta

Shakespeare, por ejemplo, la relación correspondiente es aquella en la que el objeto en el primer argumento ama al objeto en el segundo. Es decir, los matemáticos definen la relación *amar a* como la colección de todas las duplas en las que X ama a Y. Los computistas, siguiéndoles la corriente, dicen que eso es muy parecido a tener una **tabla** en una **base de datos** que describe todo lo que es **verdad**.

Para saber si Romeo, (el de la historia) realmente ama a Julieta (la de la historia), uno tendría que revisar la base de datos (correspondiente a la historia) y verificar que contiene la tupla correspondiente. Noten, por favor, una sutileza. Esa no es una base de datos de palabras o términos electrónicos. Es, de hecho (dicen los lógicos clásicos), una reunión (*¿imaginaria?*) de **objetos reales**.

Por simplicidad (y sin olvidar lo que acabo de decir), uno puede representarla así:

AMA	amante	amado
romeo	julieta	
julieta	romeo	
bolívar	colombia	
manuela	bolívar	
...	...	

y, colapsando todo en palabras (en la sintaxis), así:

ama(romeo, julieta)
ama(julieta, romeo).
ama(bolívar, colombia).
ama(manuela, bolívar).
...

(noten que uno puede anotar esa información de muchas maneras diferentes, e.g. *romeo ama a julieta*).

Esta explicación dispensa con algunos elementos básicos de la semántica de la lógica. Pero no con todos. Faltan los más útiles. Las llamadas **palabras lógicas**.

En la lógica, uno puede crear **oraciones compuestas** a partir de oraciones simples (como **ama(romeo, julieta)**).

Digamos que usamos los símbolos **y**, **o**, **no**, **si** (siempre acompañado de **entonces**) y **si y solo si**. (los símbolos empleados son también una

p	q	p y q
v	v	v
f	v	f
v	f	f
f	f	f

Cuadro 1.1: La semántica de **y**

decisión del diseñador del lenguaje. Algunas veces se usan \wedge , \vee , \neg , \rightarrow , \leftrightarrow y muchos otros).

Sabemos que `ama(romeo, julieta)` es verdadera al revisar la *base de datos de la interpretación* que se haya seleccionado. Pero, ¿cómo sabemos que `ama(romeo, julieta)` y `ama(julieta, romeo)` es verdadera?

La respuesta está codificada en la **tabla de verdad** de la palabra **y**:

Si podemos decir que $p = \text{ama}(\text{romeo}, \text{julieta}) = \text{verdad}$ y $q = \text{ama}(\text{julieta}, \text{romeo}) = \text{verdad}$, entonces la **conjunción** es también verdad **en esa interpretación**.

Esas tablas de verdad son útiles porque nos permiten decidir si un discurso, visto como una (gran) fórmula lógica, es cierto o falso. Esto es importante para verificar si tal discurso es coherente o, dicho de otra manera, está razonablemente estructurado. En lógica, cuando el discurso es verdadero en alguna intepretación decimos que es un **discurso consistente**.

Un discurso o teoría es inconsistente si permite derivar una proposición y su opuesta: e.g. te amo y no te amo (Ver capítulo ??).

1.2.1. Sobre la validez de un argumento

Esos ejercicios de análisis de significados con tablas de verdad tienen otro sentido práctico. Se les puede usar para evaluar la validez de un argumento.

Un argumento es válido si siempre que sus condiciones son ciertas, sus conclusiones también son ciertas.

Argumento es uno de los conceptos trascendentales en lógica. Se refiere a toda secuencia de fórmulas lógicas, conectadas entre sí (por relaciones lógicas), que establece (**prueba**) una conclusión. Un ejemplo muy conocido de argumento es este:

Todo humano es mortal. Sócrates es humano. Por lo tanto,
Sócrates es mortal.

que puede ser codificado y usado por un computador en esta forma:

```
mortal(X) :- humano(X).  
humano(socrates).
```

```
?mortal(X)  
X = socrates  
Yes
```

¿Ve lo que ocurre en el ejemplo?

Para verificar que un argumento es válido basta verificar que **CUANDOQUIERA** que sus condiciones son todas ciertas, sus conclusiones también (son todas ciertas). Vale la pena tomar un tiempo para aclarar el concepto. Algunas personas tienen dificultad para aceptar que un argumento puede ser válido (aún) si:

sus condiciones son falsas y sus conclusiones son ciertas o
sus condiciones son falsas y sus conclusiones son falsas también.

De hecho, lo único que **NO** puede ocurrir es que

sus condiciones sean ciertas y su conclusiones falsas.

Recapitulando, un argumento es válido si cuandoquiera que sus premisas (condiciones) son conjuntamente ciertas, su conclusión (o conclusiones) también son ciertas.

Esta definición, como suele pasar en matemática, sirve para describir, pero no para producir o justificar un argumento válido. El que podamos construir un argumento válido va a depender, siempre, de que podamos establecer la conexión adecuada entre sus premisas y sus conclusiones.

En particular, la validez de un argumento **NO DEPENDE** de los valores de verdad de sus componentes. Eso es lo que verificamos con los siguientes ejemplos (basados en los ejemplos de libro Introducción a la Lógica de Irving Copi y Carl Cohen. LIMUSA, 1998)[?].

Todas las ballenas son mamíferos. Todos los mamíferos tienen pulmones. Por lo tanto, todas las ballenas tienen pulmones.

Este es un argumento válido, con condición cierta y conclusión cierta.

Todas las arañas tienen diez patas. Todas las criaturas de diez patas tienen alas. Por lo tanto, todas las arañas tienen alas.

Este **también** es un argumento válido, con condiciones falsas y conclusión falsa.

Noten que la falsedad de la primer condición y de conclusión es atribuida por nuestro conocimiento de biología (*Animal Planet*, decimos en clase). Decir que la segunda condición es falsa es un poco temerario, puesto que no conocemos criaturas de diez patas (o ¿sí?), pero realmente no importa porque la conjunción se falsifica con la primera condición falsa.

Si yo tuviera todo el dinero de Bill Gates, sería rico. No tengo todo el dinero de Bill Gates. Por lo tanto, no soy rico.

Este es un argumento **inválido** (es decir, no valido) con condiciones ciertas y conclusión cierta (y le juro que es absolutamente cierta si habla del autor).

¿Por qué es inválido?. Piénselo.

Si Bin Laden tuviera todo el dinero de Bill Gates, sería rico. Bin Laden no tiene todo el dinero de Bill Gates. Por lo tanto, Bin Laden no es rico.

Este también es **inválido**, pero sus condiciones son ciertas (suponiendo que Bill es más adinerado) y su conclusión es falsa (por lo que sabemos del origen de Bin Laden).

¿Será este argumento inválido por la misma razón que el anterior?. Piénselo.

Todos los peces son mamíferos. Todas las ballenas son peces.
Por lo tanto, todas las ballenas son mamíferos.

Este es válido, con condiciones (ambas) falsas y conclusión cierta (otra cosa que sabemos gracias a *Animal Planet*).

¿Podemos llegar a la conclusión de validez, olvidándonos de peces, ballenas y mamíferos? Piénselo.

Todos los mamíferos tienen alas. Todas las ballenas tienen alas. Por lo tanto, todas las ballenas son mamíferos.

Este es inválido, con condiciones falsas y condición cierta.

Todos los mamíferos tienen alas. Todas las ballenas tienen alas. Por lo tanto, todos los mamíferos son ballenas.

Este es inválido, con condiciones falsas y conclusión también falsa.

¿Qué podemos decir de esos 7 argumentos?. ¿En qué nos basamos para el análisis?. ¿Qué ocurre con la 8va. posibilidad?. ¿Cuál es la combinación faltante?. ¿Cuál es la relación entre la validez de un argumento y la tabla de verdad de las fórmulas **si .. entonces**?

Si este es un libro guía es razonable que el lector reclame las respuestas a estas preguntas. Descubrirlas por sí solo o sola, sin embargo, será mucho más productivo. Veamos la otra ventaja interesante de la lógica que se describe en el texto.

1.3. Una realización computacional.

En lógica clásica se cultivan separadamente esas estructuras semánticas que hemos discutido en la sección anterior, bajo el nombre de **Teorías de Modelos**. Un **modelo** es una interpretación lógica que hace cierta a una fórmula. Por ejemplo, p y q es satisfecha si $p = \text{verdad}$ y $q = \text{verdad}$. Por tanto, **esa asignación** es un modelo.

La otra parte de la lógica matemática clásica es las llamadas **Teorías de Prueba** (*Proof Theories*). En lugar de girar en torno a la noción de **DEDUCCION**, como se hace al analizar argumentos y tablas de

verdad, las teorías de prueba se concentran en describir ejercicios y estrategias de **DERIVACION**.

Derivar es obtener una fórmula a partir de otras a través de transformaciones sintácticas. También se usa el término **INFERRIR**.

X es P y P es M

entonces

X es M

Una realización computacional de un razonador lógico es un programa que hace justo eso: inferir. Deriva unas fórmulas a partir de otras. Así, para el computista lógico, derivar es computar. El razonamiento lógico es una forma de computación (si bien, no la única).

A los razonadores lógicos se les llama de muchas maneras: motores de inferencia, probadores de teoremas, máquinas lógicas, método de prueba. En todo esos casos, el razonador tiene sus propias reglas para saber qué hacer con las fórmulas lógicas (que también suelen ser reglas). A las reglas de un motor de inferencia se les llama: reglas de inferencia. He aquí una muy popular que hasta tiene nombre propio: **modus ponens**

Con una fórmula como:

Si A entonces B

y otra formula como:

A

Derive

B

¿Puede ver que se trata de una regla para procesar reglas?. Por eso, algunas veces se dice que es una **meta-regla**.

Las teorías de modelos y las teorías de prueba normalmente caracterizan a la lógica clásica (a una lógica o a cada lógica, si uno prefiere hablar de cada realización por separado). Han sido objetivo de intenso debate durante largo tiempo y mucho antes de que existieran los actuales computadores. La lógica computacional, sin embargo, ha abierto un nuevo espacio para describir el propio lenguaje.

En la siguiente sección conoceremos de un primer intento por explicar el alcance expresivo de la lógica computacional y conoceremos también una lista de ejemplos de agentes modelados con y en lógica.

1.4. Lógica y Agentes

1.4.1. Una Teoría de Agentes en Lógica

Hace más de una década, en la escuela de la **programación lógica**, emprendimos un proyecto colectivo para caracterizar a la nueva noción de Agente que irrumpió en varios escenarios de la Inteligencia Artificial. Robert Kowalski es uno de los mejores exponentes de ese proyecto, con su libro *Cómo ser artificialmente inteligente*². El objetivo de Kowalski es explorar el uso de la lógica para modelar agentes. Eso lo ha llevado a bosquejar una teoría para explicar qué es un agente y cómo se le puede implementar para efectos de computación. Kowalski ha tratado de aproximar esa misma noción de agente que se ha hecho muy popular en el mundo tecnológico en las últimas décadas.

Resumiendo esa teoría, uno puede decir que un **agente** es una pieza de software para controlar un dispositivo capaz de interactuar con su entorno, percibiendo y produciendo cambios en ese entorno. La **especificación lógica** es más general que esa visión resumida y orientada a la máquina. Un agente, en esa teoría lógica, es un proceso auto sostenido y auto dirigido de intercambios entre un estado interno y un estado externo. El proceso es modelado como un programa lógico que puede convertirse en código ejecutable sobre un hardware o en otros casos, decimos nosotros, sobre **bioware**. En este sentido, la conceptualización alcanza a describir aspectos de la conducta autónoma de los humanos y otros seres en los que es posible o conveniente distinguir un estado interno de uno externo. Es el ahora clásico ejercicio de *postura intencional*[?], según la cual la condición de agente es determinada por el ojo del observador.

La teoría renuncia a toda pretensión totalizante: Un agente no es sólamente lo que allí se describe y, más aún, nunca se pretende que los humanos (y otros seres conscientes) seamos eso únicamente (pueden verse declaraciones al respecto al principio de los capítulos 1, 6 y 7 .ibid).

Sin embargo, esa teoría sí pretende restaurar aquella intención originaria de la lógica: *La lógica simbólica fue originalmente desarrollada como un herramienta para mejorar el razonamiento humano*. Ese es el sentido práctico, de utilidad, que se pretende rescatar con la teoría y

²publicado libremente en Internet y disponible en español en <http://webdelprofesor.ula.ve/ingenieria/jacinto/kowalski/logica-de-agentes.html> y como anexo de este texto

que, necesariamente, debe estar asociado a una práctica.

1.4.2. Una Práctica de Agentes en Lógica

Inspirados por la teoría de Kowalski y por su afán de demostrar que transciende los símbolos y las operaciones matemáticas, hemos venido recolectando ejemplos particulares de especificaciones, siempre parciales, de agentes en lógica. Cada uno de los ejemplos trata de capturar antes que un agente completo, el rol (papel, partitura) que desempeña cierto agente en cierta circunstancia de interés. El objetivo es, siempre, explicar a un agente y a su forma de pensar sin que la representación matemática interfiera (demasiado).

Burocratín

En la figura 1.1 se muestran las reglas que cierto empleado público en Venezuela, llamado Burocratín, emplearía para decidir cómo atender a un usuario de su servicio:

La situación problema es una en la que Ud necesita que Burocratín le suministre información con cierta urgencia. Debemos explicar las reglas de conducta seguiríamos para hacer que Burocratín le responda en el menor tiempo posible. Algo como la figura 1.2:

Con estas reglas, el usuario (uno de nosotros), de *necesitar información de burocratín, ubicaría antes una referencia al tema en alguna de las leyes, le pediría la información a burocratín y mencionaría con pasión la referencia legal*. Eso obligaría a Burocratín a *resolver inmediatamente*, una acción, por cierto, que no está definida en sus reglas (y que, por tanto, puede significar más problemas. Buena Suerte)

Matraquín

Matraquín es un agente que vive en un Universo similar al nuestro en donde sirve como fiscal de tránsito, mientras porta un arma. Sus reglas de conducta incluyen las que se muestran en la figura 1.3:

¿Qué pasaría si Ud, conduciendo y desarmado, se encuentra con Matraquín, tiene una falta evidente y Ud es de quienes creen que no se puede negociar desarmado con quienes portan armas?.

Si alguien me pide algo entonces yo sigo el procedimiento.

Si alguien me pide algo y es algo muy importante,
resuelvo inmediatamente.

Para seguir el procedimiento, consulto el manual si existe
o invento un manual si no existe.

Para inventar un manual de procedimiento,
solicite una carta de alguna autoridad y
haga lentamente lo que allí dice (para evitar errores).

Todo procedimiento se sigue lentamente
para evitar errores y trampas.

Algo es muy importante si se lo menciona en las leyes.

Para las cosas importantes
existen manuales de procedimiento.

Figura 1.1: Burocratín

Si necesito información de burocratín
entonces se la pido apropiadamente

Para pedir información apropiadamente (a burocratín),
se lo pido y le explico que es muy importante.

Para explicarle que es muy importante,
antes encuentro una referencia del asunto
en alguna de las leyes y
la menciono con pasión.

Figura 1.2: El Usuario frente a Burocratín

Si se acerca un ciudadano en su vehículo entonces lo interpelo.

Para interpelar a un ciudadano,
debo pedirle su cédula, su licencia,
su certificado médico, el carnet de circulación y
le pido que explique de donde viene y adonde va.

Si el ciudadano se pone nervioso, le retiro sus
documentos por un tiempo,
examino con cuidado el vehículo y
los documentos y luego lo interpelo.

Si logro establecer una falta o violación de la ley,
le explico el castigo y le permito negociar.

Si no negocia, ejecuto el castigo.

Si negociando, el ciudadano propone una ayuda mutua,
la acepto y le dejo pasar.

Figura 1.3: Matraquín

Digamos que esto es lo que me ocurre a mí: 1) (En cualquier caso) Matraquín me somete a la primera interpelación, al observar que me acerco. 2) Yo observo el arma y, siguiendo la mencionada creencia, no discuto (entiendo que eso significa que no negocio), pero (supongamos) tampoco me pongo nervioso. 3) Matraquín observa mi evidente falla y procede a explicarme el castigo y me deja negociar. 4) Yo ya había decidido no negociar, así que guardo silencio. 5) Matraquín observa mi silencio y procede a ejecutar el castigo. El significado de esto último queda para la imaginación del lector.

Obviamente esta no es la única forma de **desenlazar la historia**. Todo depende de cuáles otras suposiciones hagamos sobre Matraquín y nosotros mismos. Pero parece que ya hay razones suficientes para preocuparse por esta clase de sistema multi-agentes, ¿verdad?. Consideré ahora este otro escenario:

¿Qué pasaría si Ud, conduciendo, se encuentra con Matraquín, tiene una falta evidente y tiene Ud mucha prisa?.

1) Matraquín me somete a la interpelación (de rigor). 2) Esta vez no tengo aquella regla que me impide negociar con una persona armada. Supongamos que trato de explicar mi situación y, supongamos también, que tengo suerte de que Matraquín no interprete que estoy nervioso. 3) Matraquín observa mi evidente falla y procede a explicarme el castigo y me deja negociar. 4) Yo le explico (supongamos que calmadamente) que necesito continuar mi viaje y le pido que ayude!. 5) Matraquín entiende que estoy negociando y me propone que lo ayude a ayudarme. 6) Yo me muestro dispuesto a la ayuda mutua. 7) y 8) (*estas acciones deben ser censuradas*) . 9) Matraquín acepta mi parte de la ayuda mutua y me deja pasar.

¿Cuál es la MORALeja de la historia?

Machotín

Consideré la siguiente especificación parcial e informal de un agente que llamamos Machotín en figura 1.4:

Imagine que queremos escribir una historia en la que participe Machotín y en la que se activen, cada una en algún momento, todas sus **metas de mantenimiento**. Nos piden que expliquemos, en esa historia, qué está pensando, observando y haciendo Machotín en cada momento.

METAS:

si hay hembras cerca, establece tu autoridad.

si alguna hembra necesita ayuda, sondéala.

si la hembra que necesita ayuda se muestra receptiva, lánzate.

si se aproxima una hembra, abórdala.

si luego de abordar a una hembra se muestra receptiva, lánzate.

si se quiebra, suéltaselo.

CREENCIAS:

Para establecer tu autoridad, haz una bravuconada.

Para abordar a una hembra, convértsale sobre cualquier cosa.

Para sondarla, abórdala.

Para lanzarte, invítala a salir.

Para soltárselo, invítala a

Figura 1.4: Machotín

Considere la siguiente respuesta:

“Machotin es un agente en busca de una hembra, bien sea para invitarla a salir o para lanzarse en una aventura con ella. Machotin anda por la calle y observa que hay hembras cerca y establece su autoridad. Para establecer su autoridad, hace una bravuconada con su pinta de sobrado. Mientras camina con su actitud altiva, observa que una hembra necesita ayuda. La sondea, abordándola y sacándole conversación mientras le ofrece ayuda. La hembra en cuestión se muestra receptiva. Entonces, Machotin aprovecha la oportunidad de lanzarse con una invitación a salir. Luego de cuadrada la cita, Machotin continua su recorrido y observa que se aproxima una hembra. La aborda. Conversa con ella y observa que la hembra es receptiva. Luego se lanza con otra de sus cotorras para invitarla a salir. Mientra sigue con su labia, nota que la hembra se quiebra y le suelta una invitación a su casa para que le lave la ropa aprovechándolo que ella hará todo lo que él le pida.”³

¿Ve cómo y cuando se **dispara** cada regla?

Mataquín

Cierto agente llamado Mataquín está contemplando la posibilidad de cometer un crimen. Quiere vengarse de cierto enemigo suyo pero quiere, desde luego, cometer el crimen perfecto: venganza plena, impunidad total y mantener su imagen pública cubriendo bien sus huellas. Mataquín ha observado que los policías de su ciudad no tienen mucha capacidad de respuesta ante un crimen. No suelen atender sino emergencias extremas. No tienen recursos para investigaciones complejas que involucren a muchas personas, muchos lugares o que impliquen análisis técnicos de cierta sofisticación. Tienen una pésima memoria organizacional, pues llevan todos los registros en papel y los guardan en lugares inseguros. Además, los policías son muy mal pagados y se conocen casos de sobornos, especialmente en crímenes muy complejos o en los que los investigadores se arriesgan mucho y por mucho tiempo. Los ciudadanos de esa ciudad tienen tal desconfianza en su policía que nunca les ayudan, aún cuando tengan información sobre un crimen.

³Joskally Carrero, 2007

Suponga ahora que, con esas consideraciones en mente, se nos pide proponer una especificación (informal, pero tan compacta como sea posible a partir de la información dada) de las reglas de conducta que debería seguir Mataquín para alcanzar su meta.

Considere la respuesta en la figura 1.5⁴:

La historia resultante, aparte de respulsiva, es bastante obvia, ¿verdad?

Halcones, Burgueses y Palomas: Sobre las aplicaciones de la Teoría de Decisiones en lógica

Para ilustrar una posible aplicación de la teoría de Decisiones, un desarrollo matemático estrechamente asociado con los agentes y los juegos (ver en el capítulo 8 de Kowalski), consideren el siguiente ejemplo tomado de un examen. Considere los siguientes tres tipos de personas en una sociedad en la que los individuos compiten por recursos que siempre son de alguien:

La **paloma** nunca trata de hacerse con las posesiones de otros, sino que espera a que sean abandonadas y ella misma abandona un recurso propio tan pronto es atacada. Si dos de ellas compiten por el mismo recurso, entonces una de ellas lo obtendrá (por suerte o paciencia) con la misma probabilidad para cada una. El **halcón** siempre trata de apoderarse de los recursos de otros por medio de la agresión y se rinde sólo si recibe graves lesiones. El **burgués** nunca trata de hacerse con las posesiones de otros, sino que espera hasta que son abandonadas, pero defiende su posesión contraatacando hasta que tiene éxito o es derrotado⁵ [].

Cuando dos individuos se encuentran tienen idénticas probabilidades de ganar o perder y de ser, al inicio, dueños ó no del recurso. Por ejemplo, si dos halcones se encuentran, uno de ellos obtendrá el recurso con utilidad U , mientras el otro tendrá graves lesiones, con costo $-C_{pelea}$. Como ambos tienen la misma oportunidad de ganar, la utilidad esperada para cada uno es $(U - C_{pelea})/2$.

¿Cuáles son las utilidades esperadas correspondientes cuando se enfrentan una paloma y un halcón?. Explique.

⁴Basada en una propuesta de Victor Malavé, 2007

⁵Original de Martinez Coll, J.C. (1986) A bioeconomic model of Hobbes' state of nature. Social Science Information, 25: 493-505)

METAS

si enemigo se acerca entonces establece contacto.

si contacto establecido y policía cercano,
crea distracción.

si contacto establecido y policía distraída,
conduce a enemigo a piso más alto
de edificio cercano.

si enemigo en piso alto de edificio cercano, asesínalo.

si durante asesinato un policía observó, sobórnalo.

CREENCIAS

Para establecer contacto, invítalo a lugar público.

Para crear distracción, provoca emergencia extrema.

Para provocar emergencia extrema,
contrata lugareños
que incendien estación de servicio cercana.

Para conducir enemigo a piso alto de
edificio cercano, propónle un negocio atractivo.

Para asesinarlo,
provoca caída mortal desde ese piso alto.

Para sobornar policía, entrégale mucho dinero.

Figura 1.5: Mataquín

Dadas esas reglas de conducta, la paloma nunca gana contra un halcón (y el halcón nunca pierde contra una paloma). Así que la utilidad para la paloma es 0 y para el halcón es U . Noten que U es el valor *puro* del recurso a conquistar (otra simplificación muy gruesa).

¿Cuáles son las utilidades esperadas correspondientes cuando se enfrentan una paloma y un burgués?. Explique⁶

“Cuando un burgués encuentra a otro individuo, cada uno de ellos puede ser el dueño lícito del recurso rivalizado. Por ejemplo, si una paloma se encuentra con un burgués y ambos compiten por el mismo recurso, entonces tenemos dos posibilidades igualmente probables:

- Caso1: Si el burgués es el dueño legal del recurso, entonces conserva el recurso ($Beneficio=U$), y la paloma no se lleva nada ($Beneficio=0$).
- Caso2: Si la paloma es la dueña legal del recurso, ambos deben esperar hasta que alguno renuncie (lo cuál les cuesta digamos $Cespera$. Beneficio = $-Cespera$) y, entonces, cada uno de ellos tiene igual probabilidad de lograr el recurso. Así que el resultado local esperado es $U/2 - Cespera$ para cada uno.

De esta manera, el resultado global es $\frac{U+(\frac{U}{2}-Cespera)}{2}$ para el burgués y $\frac{0+(\frac{U}{2}-Cespera)}{2}$ para la paloma”.

Noten que la utilidad global (para cada agente) es igual a $Caso1 * Prob1 + Caso2 * Prob2$, donde $Prob1 = Prob2 = 1/2$, pues se nos dice que ambos casos son igualmente probables.

En el caso2, además, juega un papel la acción de esperar a que el recurso sea abandonado (sí, esperar también es una acción posible). Cómo no se sabe cuál de los dos va a renunciar primero en esa esperar, se dice que su probabilidad local asociada es $1/2$ y como el beneficio en disputa es U , la utilidad esperada local es de $U/2$, faltando por descontar el precio de esperar que ambos pagan igualmente ($Cespera$).

Este ejemplo es particularmente interesante porque muestra el uso del concepto de la utilidad esperada *anidada* (cálculo de utilidad esperada sobre otra utilidad esperada).

¿Cómo sería la utilidad global para ambos agentes?.

⁶Cito la explicación dada en Simulación para las Ciencias Sociales de Nigel Gilbert y Klaus Troitzsch)

si me preguntan *¿Quién Opina sobre Tema?* y
Quién es una variante gramatical de Quienes y
Opina es una variante semántica de opinar y
el análisis del Tema arroja estos Descriptores entonces
Rastreo la Web buscando todos los Autores de documentos
con esos Descriptores.

Figura 1.6: Agente Web Semántica

Agentes, Ontologías y la Web Semántica

Los ejemplos anteriores se referieren todos a descripciones de agentes en ciertos roles muy puntuales. Pareciera que para tener un agente completo y funcional se requiere de un esfuerzo mucho mayor. Lo cierto es que esto último, incluso para efectos de prestación de servicios, no necesariamente es el caso.

Lo que queremos es mostrar a continuación son dos ejemplos de modelos de agentes que implementan otras dos nociones que se han vuelto fundamentales en la Web: Los **metadatos** y el **razonamiento no monótono**.

Una de las primeras aplicaciones naturales de la tecnología de la **Web Semántica** es la búsqueda de documentos. Con recursos como repositorios **RDF** y **OWL** en la Web, uno puede imaginar un agente que nos ayude a ubicar documentos a partir del tipo de conocimiento que contienen (y que buscamos). Por ejemplo, para responder preguntas como *¿Quienes han escrito sobre el legado epistolar del Libertador de Venezuela?*, uno puede pensar en un agente con una META⁷ como la que se muestra en la figura 1.6.

Esta meta requerirá, desde luego, mecanismos de soporte (que podrían

⁷Agradecemos a Icaro Alzuru por motivar este ejemplo con su trabajo de maestría en Alejandría Inteligente: Un experimento en la Web Semántica con un sistema de gestión de documentos <http://webdelprofesor.ula.ve/ingenieria/jacinto/tesis/2007-feb-msc-icaro-alzuru.pdf>

convertirse en creencias del agente) para realizar el análisis del Tema que produce los Descriptores y el Rastreo de la Web.

En la tarea de **análisis del tema**, sería muy útil contar con un mecanismo que le permita al agente, en el caso particular de nuestra pregunta, traducir *legado epistolar* en todos sus **sinónimos** e **hipónimos**. Así el agente sabría que quien quiera que declare haber escrito sobre *las cartas del Libertador* es un autor a revisar. Algo similar habría que hacer con *Libertador de Venezuela* para que la máquina entienda que se trata de Simón Bolívar. Esto último requeriría, desde luego, un mecanismo adecuado para el caso *venezolano-colombiano-ecuatoriano-peruano-boliviano*.

Esta clase de relaciones de **sinonimía**, tanto universales como locales, son posibles con sistemas como **WordNET**[], una especie de diccionario y tesauro automático. WordNET codifica una **ontología** que le permite relacionar los términos. Desafortunadamente, esa ontología no está escrita en ninguno de los lenguajes ontológicos de la Web Semántica, pero aún así puede ser útil.

La segunda tarea es también muy interesante. Requiere que nuestro agente disponga de medios para identificar entre la enorme cantidad de documentos contenidos en la Web, los autores y los descriptores de esos documentos. Esa información que refiere el contenido y otros atributos de los datos es conocida como **metadata** y es fundamental para que las máquinas puedan contribuir a la gestión de los **repositorios de conocimiento**.

Hay muchos esfuerzos en la dirección de proveernos de un standard de metadatos en la forma de una ontología que explique que son y que contienen los documentos formales. El Dublin Core[] es quizás el más avanzado de esos esfuerzos.

Podemos ver, en ese primer ejemplo de agente para la Web Semántica, como se incorporaría el manejo de significado en un agente que atienda consultas en la Web.

Hay, sin embargo, un siguiente nivel de complejidad en el manejo de significados para atender consultas. Algo que es bien conocido en Bases de Datos desde hace años: El manejo de datos temporales y el razonamiento no monótono.

Se llama **razonamiento no monótono** a aquel que cambia de opinión. Típicamente, un agente con cierto cúmulo de creencias alcanza ciertas conclusiones, pero si esas creencias cambian (debido quizás, a un cambio en el mundo), las conclusiones también podrían cambiar.

si me solicitan los Propietarios de terrenos en un Área entonces reportar Propietarios del Área.

Para reportar Propietarios del Área haga transforme la descripción del Área en un Área Geográfica y Los Propietarios son los sujetos para quienes se cumple (ahora) que poseen propiedades contenidas en el Área Geográfica.

Figura 1.7: Gea: El Agente Catastral

Los lectores del libro del Profesor Kowalski podrán asociar esa forma de razonar con esos lenguajes lógicos que estudiaron en el capítulo sobre el cambiante mundo (**lógicas modales**, el **cálculo de situaciones** y el **cálculo de eventos**). Permítanme, sin embargo, un último ejemplo con un agente que implementa una forma de razonamiento no monótono indispensable para la aplicación a su cargo:

Imaginen un agente que maneja un repositorio de información catastral elemental⁸. Entre muchos otros tipos de consulta, este agente debe poder explicar quienes son los propietarios de los terrenos en un área dada. La meta de mantenimiento y creencia correspondiente para este agente es simple se muestra en fig 1.7.

La última sub-meta (*Los Propietarios..*) es una consulta que se podría responder con una versión extendida del cálculo de situaciones, o el de eventos, configurada para lidiar con propiedades y áreas geográficas. Lo importante en este caso es que si a este agente se le hace la consulta en un momento dado, su respuesta bien puede variar respecto a otro momento si han ocurrido eventos que cambien las relaciones propiedad-propietario en ese área geográfica.

Desde luego, este agente tendría que operar con conocimiento que le permita asociar cualquier descripción de un área (por ejemplo, en términos políticos: parroquia, municipio, país, etc) con un conjunto de coordenadas standard. Más importante aún, el agente requeriría de un

⁸Agradecemos a Nelcy Patricia Piña por motivar este ejemplo con su trabajo de maestría en Una Ontología para Manejo de Información Catastral <http://webdelprofesor.ula.ve/ingenieria/jacinto/tesis/2006-mayo-msc-nelcy-pina.pdf>

registro sistemático de los eventos de compra y venta de propiedades que incluya los detalles de cada propietario y propiedad (incluyendo la ubicación de cada una, desde luego). Todo este conocimiento bien puede estar almacenado en un repositorio asociado con una ontología de registro catastral (por ejemplo, un archivo, con marcaciones en **OWL** específicas para el problema).

Esos dos ejemplos, informales y muy simples, pueden servir para ilustrar la extraordinaria riqueza de posibilidades en este encuentro entre la Web, la lógica y los agentes.

Kally: Agente para atención a usuarios y enseñanza isocéntrica

Los agentes Web Semántica y Gea Catastral dejan ver someramente la importancia del procesamiento lenguaje natural en la Inteligencia Artificial. Es, todavía, una de las fronteras más activas de investigación y promete grandes avances, siempre que logremos contener el entusiasmo que produce la posibilidad de interactuar con el computador en los mismos términos (lenguaje) que usamos con otros humanos.

Kally⁹ es un pequeño agente diseñado para asistir a los humanos en el uso de una herramienta Ofimática, OpenOffice, interactuan en una forma controlada de lenguaje natural. En fig 1.8, 1.9 y 1.10 se muestra el código de Kally en una versión preliminar de Gloria (ver capítulo ??).

Kally tiene sólo un par de reglas simples, pero tiene también una gramática interconstruída (el predicado **pregunta**) que le permite entender ciertas preguntas simples en Español y establecer “su significado” (es decir, lo que debe hacer al respecto). La gramática está formalizada usando un lenguaje conocido como DCG, *Definite Clause Grammar*.

El Cálculo de Eventos y El disparo de Yale

El sentido común se ha constituido en uno de los mayores desafíos en la IA, al punto que algunos problemas referenciales al tema se han vuelto muy populares.

En la historia del Disparo de Yale, una persona es asesinada de un disparo. Para una posible formalización uno considera tres acciones:

⁹<http://kally.sourceforge.net>

```

si me_pregunta(Preg),
    significa(Preg, Sig),
    not(Sig=[no, entiendo|_]) entonces responde_a(Sig).
si me_pregunta(Preg),
    significa(Preg, Sig),
    Sig=[no, entiendo|_] entonces disculpas(Sig).

para responde_a(S) haga
rastrear(S),
mostrar(S)

```

Figura 1.8: Kally

```

% -----
% Reglas gramaticales
% -----
% que es X
pregunta([V|S]) -->
    pro_interrog, v_atributivo(V), s_atributivo(S).
% como puedo instalar Y
pregunta([V|S]) -->
    adv_interrog, s_verbal(V), s_atributivo(S).
% puedo instalar Y
pregunta([V|S]) --> s_verbal(V), s_atributivo(S).
% esta es la via de escape cuando no entiende
pregunta([no, entiendo, tu, pregunta, sobre|S]) -->
    atributo(S).
%
s_verbal(V) --> v_modal, v_infinitivo(V).
s_verbal(V) --> v_infinitivo(V).
s_verbal(V) --> v_conjugado(V).
s_atributivo(S) --> especificador, atributo(S).
s_atributivo(S) --> atributo(S).

```

Figura 1.9: La Gramática Española de Kally

```
% -----
% Reglas de inserción léxica
% -----
pro_interrog --> ['qué'] ; [que] ; [cuál] ; [cuales] ;
    ['cuáles'] ; [cuantos] ; [cuantas] .
especificador --> [el] ; [la] ; [lo] ; [los] ; [las] ; [un] ;
    [una] ; [unos] ; [unas] ; [mi] ; [mis] .
v_atributivo(es) --> [es] ; [son] ; [significa] .
adv_interrog --> ['cómo'] ; [como] ; [cuando] ;
    [donde] ; [por, que] ; [por, 'qué'] .
v_modal --> [puedo] ; [puede] ; [podemos] .
v_infinitivo(utilizar) --> [utilizar] .
v_infinitivo(abrir) --> [abrir] .
v_infinitivo(salvar) --> [guardar] .
v_infinitivo(salvar) --> [salvar] .
v_infinitivo(crear) --> [crear] .
v_infinitivo(instalar) --> [instalar] .
v_infinitivo(instalar) --> [reinstalar] .
v_infinitivo(definir) --> [definir] .
v_infinitivo(realizar) --> [realizar] .
v_infinitivo(agregar) --> [agregar] .
v_infinitivo(agregar) --> [anadir] .
v_infinitivo(exportar) --> [exportar] .
v_infinitivo(insertar) --> [insertar] .
v_conjugado(guardar) --> [guarda] .
v_conjugado(desinstalar) --> [desinstalo] .
v_conjugado(ubicar) --> [existe] .
v_conjugado(usr) --> [uso] .
v_conjugado(cambiar) --> [cambio] .
v_conjugado(instalar) --> [instalo] .
v_conjugado(crear) --> [creo] .
v_conjugado(estar) --> [esta] .
v_conjugado(actualizar) --> [actualizo] .
%
prep --> [a] ; [como] ; [con] ; [de] ; [desde] ;
    [durante] ; [en] ; [entre] ; [hacia] ; [mediante] ;
    [para] ; [por] ; [sin] ; [sobre] .
atributo(T,T,_).
```

Figura 1.10: El Léxico Español de Kally

cargar (el arma), esperar y disparar; y dos propiedades: *vive* y *cargada*. La acción cargar coloca una bala en el arma. La víctima muere después del disparo, siempre que el arma esté cargada en ese instante. Se asume que la víctima vive al principio y también que el arma está descargada entonces.

Considere el axioma central del Cálculo de Eventos:

- 0) Se cumple un Hecho en un Momento si un Evento ocurrió antes y ese Evento inició el Hecho y no hay Otro evento que ocurra luego del Evento iniciador, antes del Momento y que termine el Hecho.

Suponga que nos piden completar este axioma con las reglas y hecho necesarios para describir el problema del Disparo de Yale. Con todas esas reglas, debemos probar (con un argumento formal) que: no se cumple que (la víctima) vive en en último momento del cuento.

La historia se puede describir, con algo de formalización, así:

- 1) La (presunta) víctima nace en el momento 0.
- 2) No es cierto que el arma está cargada en el momento 0.
- 3) El (presunto) asesino carga el arma en el momento 1.
- 4) El asesino espera entre el momento 1 y el momento 2.
- 5) El asesino dispara en el momento 2.

Con la misma terminología podemos escribir las reglas complementarias específicas a esta situación:

- 6) El evento **Un agente (le) dispara** en el momento T termina el hecho **la víctima vive** en T si se cumple que el arma está cargada antes de T.
 - 7) El evento **Un agente (le) dispara** en el momento T termina el hecho **el arma esta cargada** en T si se cumple que el arma está cargada antes de T.
 - 8) El evento **Un agente carga el arma** en el momento T inicia el hecho **el arma está cargada** en el momento T.
-

- 9)** El evento **Un agente nace** en el momento T inicia el hecho **el agente vive** en el momento T.

Con esas reglas razonamos hacia atrás a partir del hecho de que no se cumple que la víctima vive en el último momento del cuento. Digamos que ese momento es momento 3.

- 10)** no se cumple que la víctima vive en el momento 3.

de 10) y 0) se obtiene:

- 11)** no (es cierto que) un evento ocurrió antes del momento 3 y ese evento inició (el hecho de) víctima vive y no hay otro evento que ocurra luego de ese evento iniciador y antes del momento 3 y que termine el hecho de que la víctima vive.

Pero 11) se puede reescribir (por equivalencia lógica $\neg(a \wedge b) = (\neg a) \vee (\neg b)$), así:

- 11')** no es cierto que un evento ocurrió antes del momento 3, ó no inició ese evento víctima vive, ó hay otro evento que ocurra luego de ese evento iniciador y antes del momento 3 y que termine el hecho.

11') se transforma en 12) al considerar a 1),

- 12)** hay otro evento que ocurra luego del evento iniciador **victima nace** en momento 0 y antes del momento 3 y que termine el hecho de que **la víctima vive**.

Pero, por 5), sabemos que hay un candidato a posible terminador, con lo que 12) se reduce a 14), luego de considerar a 13)

- 13)** **el asesino dispara** en momento 2 termina el hecho de que **la víctima vive**.

Así, por 6), pasamos a preguntarnos si 14)

- 14)** el arma está cargada en el momento 2..

y gracias a 0), 3) y 2) (y que la acción de esperar no cambia nada), podemos probar a 14 con un par de pasos similares a los anteriores. Fin de la prueba.

Visto así parece increíble que un computador lo pueda resolver. Pero lo resuelve.

```
si inspiracion_sistemica entonces
crear_obra_tipo(holistica, Obra).

para crear_obra_tipo(holistica, Obra) haga
    seleccionar_elementos(holistica, Elementos),
    plasmar(Elementos, Obra).

para seleccionar_elementos(Tipo, [Elemento]) haga
    tomar_elemento(Tipo, Elemento).

para seleccionar_elementos(Tipo, [R|Resto]) haga
    tomar_elemento(Tipo, R),
    seleccionar_elementos(Tipo, Resto).

para tomar_elemento(holistica, nota(do)) haga true.
para tomar_elemento(holistica, color(azul)) haga true.
para tomar_elemento(holistica, textura(suave)) haga true.
para plasmar([nota(do)], sonido_bajo) haga true.
para plasmar([nota(do), color(azul)], azul_profundo)
haga true.
para plasmar([nota(do), color(azul), textura(suave)],
extasis ) haga true.
```

Figura 1.11: El Artista

El Agente Artista

Considere las reglas en la figura artista que son parte de un agente artista holístico.

Suponiendo que este agente observa `inspiracion_sistemica`, vemos como, paso a paso, este agente artista produciría un plan para la creación de una obra con la mayor cantidad de elementos posibles. Asegúrese de explicar en que consiste la `Obra` planeada.

Razonando hacia adelante con la primer regla, el agente produce su primer meta de logro:

1) `crear_obra_tipo(holistica, Obra)`

noten que la escribimos en notación Prolog para hacer más fácil

la representación subsiguiente. Razonando hacia atrás a partir de 1) usando segunda regla obtenemos:

2) `seleccionar_elementos(holistica, Elementos)` y
`plasmar(Elementos, Obra).`

Para resolver la primer submeta de 2) tenemos dos reglas (la tercera y cuarta arriba), por lo que 2) se convierte en 3)

3) `tomar_elemento(holistica, [Elemento])` y
`plasmar([Elementos], Obra), o`
`tomar_elemento(holistica, R)` y
`seleccionar_elementos(holistica, Resto)` y
`plasmar([RResto], Obra).`

Uno podría, si no es cuidadoso con lo que se nos pide, terminar la prueba con la primera opción (antes del ,o). Pero eso produciría una obra con un solo elemento. Eso es justamente lo que NO se nos pide. Por esta razón, un agente inteligente optaría por la segunda alternativa y, luego de sucesivos refinamientos obtendría algo como 4)

4) `tomar_elemento(holistica, [Elemento])` y
`plasmar([Elementos], Obra), o`
`tomar_elemento(holistica, nota(do))` y
`tomar_elemento(holistica, color(azul))` y
`tomar_elemento(holistica, textura(suave))` y
`plasmar([nota(do), color(azul), textura(suave)], Obra)`

el cual, al ser ejecutado, implicaría *Obra = extasis* que consiste en esta lista de elementos: `[nota(do), color(azul), textura(suave)]`. El truco acá era observar que una reducción hacia atrás, izquierda a derecha (tal como lo hace Prolog) no nos conduce a la obra pretendida (que debe tener el máximo numero de elementos posibles). Fin de la prueba.

Como lidiar con la inflación

Presentamos un último ejemplo en este capítulo para ilustrar un caso mucho más complejo (y humano) del modelado lógico de un agente. El Prof. Carlos Domingo se plantea el siguiente conjunto de reglas para enfrentar el problema de la inflación que afectará sus ingresos como académico.

Meta: Evitar las pérdidas en mis ahorros causadas por la inflación de este año.

Creencias:

1. Durante el 2008, los precios se esperan que aumenten en un 30 por ciento en artículos como libros, del hogar, medicinas y ropas. En los artículos básicos (comida y transporte nacional), el incremento puede ser menor al 10 por ciento debido al control de precios y, además, es probable que sea compensado con un aumento salarial. Pero mis ahorros anteriores podrían ser afectados por la inflación.
2. Las tasas de interés estimadas para este año en un 13 por ciento no compensarán la inflación. Por lo tanto, depositar el dinero en el banco no es una buena estrategia para proteger mis ahorros. Otras alternativas deben ser consideradas.

Restricciones:

1. No debería tomarme mucho tiempo para encontrar una solución. Tengo trabajo pendiente.
2. No debo comprometerme a trabajos posteriores a este año.
3. Cualquiera que sea la decisión no deben disminuir mi ingreso total anual.

Tomando eso en consideración, Carlos Domingo decide que sus metas inmediatas son:

1. Conseguir un ingreso adicional que no dependa de sus ahorros para compensar por la inflación. Para ello puedo:
 - a) Participar en proyectos universitarios extra que le den un ingreso extra (obvenciones). Esto podría chocar con la restricción 1 porque es difícil conseguir proyectos en mi área de trabajo.
 - b) Ocuparme en negocios privados que siempre están disponibles. Esto choca con todas las restricciones.

2. Invertir mis ahorros (total o parcialmente) para obtener beneficios equivalentes a las pérdidas esperadas. Para ello puedo:
 - a) Comprar acciones (pero esto requiere tiempo y puede contrariar a 1 y probablemente a 2).
 - b) Comprar bienes “durables” para vender en el futuro. Tendría que tener cuidado con los precios futuros y la obsolescencia.
 - c) Mudar mis ahorros a moneda extranjera. Difícil debido al control cambiario y muy riesgoso debido a una posible devaluación de esa moneda extranjera.

El Prof. Carlos ha concluído que una combinación de 1.a y 2.b parece ser la mejor decisión posible (una combinación minimizaría el riesgo de fallar completamente). Suponga que Carlos posee una cierta cantidad en AHORROS al INICIO del año y monto total por su SALARIO en todo el año. Ayudemos a Carlos a proponer una justificación formal de su decisión.

Un análisis de utilidad completo supondría estimar las utilidades esperadas de cada opción. Esto puede consumir mucho tiempo y, en algunos casos, aún así no se mejoraría la evaluación. Así que, entendiendo las conclusiones de Carlos, uno se puede concentrar en las dos alternativas de acción que le parecen las mejores decisiones.

Carlos fallaría si sus ahorros son consumidos por la inflación. Es decir, si al final del año sus AHORROS no satisfacen:

$$AHORROS_{AHORROS_INICIALES} * 1,3 \quad (1.1)$$

Es decir, ese es el mínimo valor razonable para la utilidad esperada global de las acciones que se plantea Carlos.

Las acciones que se plantea Carlos son:

1. ganar obvenciones por proyectos extras y
2. invertir en durables.

La probabilidad de ganar obvenciones por proyectos extras, $ProbObv$, es menor al 0.5. Así que la utilidad estimada de la acción 1 es:

$$utilidad_1 = ProbObv * OBVENCIONES \quad (1.2)$$

Invertir en durables significa: 1) Adquirirlos a buen precio y, desde luego, con un costo no superior a los ahorros en ese momento y 2)

venderlos al precio justo. La probabilidad de adquirirlos a buen precio es $ProbAdDur$. La probabilidad de venderlos es $ProbVenta$. Así, la utilidad de acción 2 puede ser aproximada por:

$$\begin{aligned} utilidad_2 = & \quad ProbAdDur * ProbVenta \\ & * (PRECIO JUSTO - \\ & AHORROS_ESPERADOS) \end{aligned} \quad (1.3)$$

donde $AHORROS_ESPERADOS$ es una variable que depende del $SALARIO$, un probable aumento, la inflación y los gastos de cada tipo en que incurra Carlos. Por ejemplo:

$$\begin{aligned} AHORROS_ESPERADOS &= SALARIO \\ &+ ProbAumento * AUMENTO \\ &- GASTOS_BASICOS * 1,10 \\ &- OTROS_GASTOS * 1,3 \end{aligned} \quad (1.4)$$

Así que los escenarios que satisfacen la siguiente fórmula son los que justificarían la decisión de Carlos:

$$\begin{aligned} & ProbObv * OBVENCIONES + \\ & ProbAdDur * ProbVenta * (SALARIO \\ & + ProbAumento * AUMENTO \\ & - GASTOS_BASICOS * 1,10 \\ & - OTROS_GASTOS * 1,3) > \\ & AHORROS_INICIALES * 1,3 \end{aligned} \quad (1.5)$$

Hay todavía varios otros ejemplos que discutir. En particular, hemos reservado espacio separado para los agentes aprendices (en el capítulo ??) y para los motores de minería de datos (en el capítulo ??).