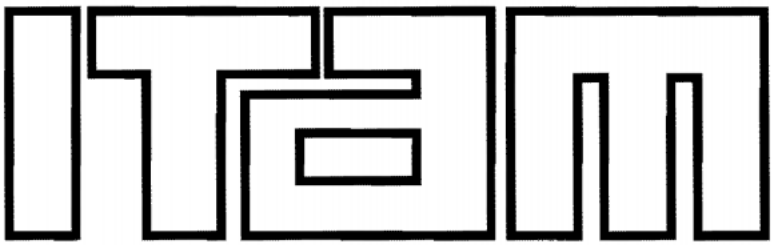
INSTITUTO TECNOLÓGICO AUTÓNOMO DE MÉXICO



Aprendizaje Reforzado para el Juego de la Distribución de Cerveza

T E S I S

QUE PARA OBTENER EL TÍTULO DE

MAESTRA EN CIENCIA DE DATOS

P R E S E N T A

MARÍA FERNANDA ALCALÁ DURAND

ASESOR: DR. ADOLFO JAVIER DE UNÁNUE TISCAREÑO

MÉXICO, D.F 2016

Con fundamento en los artículos 21 y 27 de la Ley Federal del Derecho de Autor y como titular de los derechos moral y patrimonial de la obra titulada "**Aprendizaje Reforzado para el Juego de Distribución de Cerveza**", otorgo de manera gratuita y permanente al Instituto Tecnológico Autónomo de México y a la Biblioteca Raúl Baillères Jr., autorización para que fijen la obra en cualquier medio, incluido el electrónico, y la divulguen entre sus usuarios, profesores, estudiantes o terceras personas, sin que pueda percibir por tal divulgación una contraprestación.

María Fernanda Alcalá Durand

_____

FECHA

_____

FIRMA

Agradecimientos

En este momento, le agradezco a Drake por hacer música tan espantosamente repetitiva: mi cerebro lo toma como ruido blanco y puedo concentrarme muy bien.

Índice general

Capítulo 1

Introducción

Necesito una cita cool para empezar mi tesis.

Fleo

Uno de las principales dificultades de las cadenas de suministro es que los agentes encargados de optimizar las estrategias solamente pueden tomar decisiones "dentro"del eslabón en el que se encuentran, y no tienen información más allá de los eslabones inmediatemente conectados. Así, la información acerca de la demanda del consumidor se va diluyendo en cada nivel, además de que las decisiones tomadas tienen repercusiones más allá del futuro inmediato.

Los agentes optimizadores deben tratar de inferir el patrón global por medio de información local bastante restringida. Sin embargo, los datos que reciben obedecen al tiempo real y no tienen la oportunidad de repetir experimentos.

Un modelo computacional que se comporte suficientemente parecido al mundo real, en el que todos los demás eslabones tomen estrategias que también maximizarían sus beneficios podría dar una opción: el experimento es replicable tantas veces como sea necesario y cada eslabón puede conocer una extrategia óptima para una gran cantidad de demandas de consumidor posibles.

En este trabajo se modelará el Problema de Distribución de Cerveza, *The Beer Distribution Game*, planteado por primera vez en la Escuela de Administración y Dirección de Empresas Sloan del MIT en los años 60[1],

Este documento tiene un formato simple y una estructura de propuesta de proyecto a propósito, dado que se pretende continuar trabajando en este proyecto hasta concretar una Tesis para obtener el grado de Maestría en Ciencia de Datos.

[1] En este momento no cuento con la fuente original.

1.1. Reinforcement Learning: conceptos

Su principio se basa en la psicología conductista: un *agente* busca ser recompensado por un premio, el cual obtiene cuando realiza una secuencia de *acciones* que lo llevan a concluir una tarea exitosamente. Además, para maximizar la cantidad de *recompensa* que recibe - o, alternativamente, minimizar el tiempo que espera entre un premio y el siguiente - comienza a optimizar su política (π) para llegar a la meta satisfactoriamente.

La principal característica del agente es que tiene la capacidad de tomar decisiones sobre sus acciones, las cuales son su forma de interactuar con el *mundo*, llevándolo de un *estado* a otro. El agente no tiene acceso a todas las consecuencias de sus acciones; de hecho, ni siquiera conoce todo el mundo.

El agente toma una acción en el tiempo t, la cual depende del estado s_t del mundo. En $t+1$, el mundo reaccionó ya a la interacción del agente con él, así que el agente recibe una recompensa r_{t+1} y toma una nueva acción dependiendo del estado s_{t+1} del mundo. Sin embargo, no es óptimo seleccionar acciones solamente con base en la recompensa r_{t+1}, pues la naturaleza temporal del problema lo convierte en un problema a largo plazo, y el agente estaría considerando solamente consecuencias en el corto plazo.

Así, el agente debe aprender que existe un *retraso* entre cada acción que toma y el premio. Supongamos que, en una cuadrícula, el premio se encuentra en la casilla (x,y). El agente solamente puede llegar a esa casilla meta desde las adyacentes, pero si no se encuentra en una de estas, primero debe acercárseles. Así, cuando el agente comienza su exploración, irá aprendiendo que, lejos de que la recompensa sea inmediata, debe tomar una secuencia de acciones para llegar a ella.

Podemos entonces definir la *función de valor* asociada a la política como el valor esperado de la recompensa al tiempo t dado que el agente se encuentra en el estado s.

También es necesario que el agente ajuste su comportamiento mientras transcurre el tiempo: al

principio debe explorar para conocer la mayor cantidad de consecuencias a sus acciones posibles, pero debe mantener el conicimiento de cuáles acciones le han reportado buenas acciones y tomar esas decisiones más seguido. A esta estrategia de exploración se le llama $\epsilon - greedy$.

Definamos $p_r t$ y $p_t t$ como las probabilidades al tiempo t de exploración y explotación, respectivamente. Entonces:

$$p_r t = 1 - \epsilon(t)$$
$$p_t t = 1 - p_r t \qquad \forall t$$

La función ϵ suele ser implementada como decreciente de forma lineal para el aprendizaje, de tal forma que mientras pasa el tiempo, el agente escoge las acciones conocidas que le reportan mayor utilidad más seguido; junto con un parámetro aleatorio para asegurar que siempre existe una probabilidad positiva de explorar.

Generalmente se supone este tipo de problemas como Procesos de Decisión de Markov (MDP), cuya principal característica es que cumplen con la famosa propiedad de Markov: a grandes rasgos, el futuro solamente depende del presente, no del pasado.

Cuando la política a tomar es difícil de aprender porque no tenemos ejemplos, o el mundo / conjunto de acciones / conjunto de consecuencias es demasiado grande, es apropiado utilizar Aprendizaje Reforzado en lugar de Aprendizaje de Máquina regular.

1.2. Aprendizaje Reforzado

1.2.1. Q-Learning: conceptos

$$V(s) = \max_a Q(s, a)$$

$$Q(s, a) = R(s, a) + \gamma * \max_a Q(s^{'}, a^{*})$$

Donde s' es el siguiente estado, y a^* representa todas las acciones posibles. Al estimar la función Q para cada par de estado con acción, es posible encontrar la mejor acción para cada estado y, así, obtener una política óptima.

Algoritmo

1. Asignar $Q(s, a) = 0$ para todos los estados y acciones.
2. Posicionarse en un estado s
3. Seleccionar acción a^* y ejecutar
4. Recibir recompensa r
5. Observar estado nuevo $s^{'}$
6. Actualizar $\hat{Q}(s, a) = r(s, a) + \lambda \, \text{máx}_{a'} \, \hat{Q}(s^{'}, a^{'})$
7. Asignar nuevo estado $s \leftarrow s^{'}$
8. Volver a 2 hasta convergencia

1.3. Modelo Multiagente

En este trabajo, consideraremos a cada eslabón de la cadena de suministro como un agente.

Con las siguientes definiciones:

Cada agente solamente puede comunicarse con los niveles inmediatamente vecinos; es decir, las únicas interacciones que puede tener con el mundo son el número de órdenes que recibe del nivel inferior y el inventario que pide al nivel superior. Sin embargo, como hemos definido una penalización por mantener cerveza en el inventario (el costo del almacén), la decisión concerniente a la petición del nivel inferior queda determinada: venderá todo lo que pueda, pues cada venta le reporta una ganancia, y no llenar la orden completa cuando tiene suficiente inventario lo haría incurrir en un costo innecesario.

Esto quiere decir que, para cada agente, el conjunto de **acciones** que puede tomar es solamente el número de cervezas que pedirá al nivel inmediatamente superior en cada tiempo t. Esta acción está declarada por:

Por lo tanto, lo que tendrá guardado en la bodega en el tiempo t estará constituido por el número de cervezas que tenía en el tiempo anterior $t-1$, menos el número de cervezas vendidas, más el número de cervezas que recibe del nivel inmediatamente superior por el pedido de reaprovisionamiento.[2]

Restringido a que cada agente solamente cubrirá la orden del nivel inferior si tiene suficiente inventario para hacerlo (es por esto que en la definición de clase no se utilizan las variables explícitas, sino $orders\_in$ y $orders\_out$.

Su recompensa está dada por:

El objetivo de cada agente es maximizar su recompensa. Sin embargo, este es un problema ligeramente diferente a los comunes de *Q-learning*, en los cuales el valor de la recompensa es conocido y, una vez encontrado, se buscan las acciones óptimas "de atrás hacia adelante" (como el ejemplo típico de una cuadrícula).

Su política está definida con base en la función Q, una vez que el proceso de aprendizaje fue finalizado, de esta manera, puede realizar una búsqueda sobre todas las posibles acciones en los estados y sencillamente escoger la mejor, lo cual converge a la política (cuasi)óptima. Tal política se puede definir como:

Es importante destacar que este sistema toma solamente una de las ramas que existen en la industria de cualquier producto (existe más de un minorista, etc.), e incluso, toma solamente un producto. Aún así, es un sistema complejo bastante robusto y sensible a cambios pequeños.

[2]Se agregan algunas indentaciones y saltos de línea al código para facilitar claridad de lectura.

Capítulo 2

El Problema: Juego de Distribución de la Cerveza

La estructura se puede observar en la figura 2.1.[1]

Figura 2.1: Distribución de Cerveza

Las variables que tienen efecto en este problema son:

- Demanda del Consumidor
- Tiempo de Ajuste de Inventario
- Tiempo de Envío
- Tiempo de Producción

Para cada uno de los agentes: tiendas minorista, mayorista y de distribución, y fábrica.

[1] Imagen tomada de la página de Wikipedia *The Beer Distribution Game*, bajo la licencia Creative Commons Attribution-Share Alike 3.0 Unported

Este problema se ha estudiado antes en [2] por medio de Algoritmos Genéticos y en [1] por medio de $Q - learning$.

El aporte de este trabajo será agregar un componente de estacionalidad en el proveedor de la fábrica: el campo.

2.0.1. *Efecto Látigo*

Imaginemos el siguiente escenario: el comprador, que generalmente compra 6 cervezas, ahora quiere 10, pero la tienda minorista solamente cuenta con 6. El minorista le venderá todo su inventario, pues es la acción que maximiza su ganancia, pero ¿qué pasa después?.

El minorista debe decidir si volverá a tener un inventario de 6 o si debe pedir un número mayor de cervezas, atendiendo la aparentemente creciente demanda. Supongamos que decide pedir 9 cervezas al siguiente nivel, la tienda de mayoreo.

Imaginemos que el mayorista cuenta con 17 cervezas, y decide llenar el pedido del minorista de 9 cervezas y quedarse con 8 cervezas en su inventario, sin hacer una orden al siguiente nivel, la tienda de distribución.

En este nivel, la tienda de distribución no tuvo ninguna información acerca del repentino crecimiento en la demanda del comprador. Si este comportamiento se mantiene durante algunos periodos más, recibiría la noticia (por medio de un incremento en las órdenes regulares) con un retraso considerable.

El *Efecto Látigo* se refiere precisamente a este fenómeno: mientras más arriba en la cadena de suministro se encuentre un agente (es decir, más lejos del contacto directo con el comprador), más distorsionada es la información que tiene acerca de la verdadera demanda del consumidor.

Bibliografía

[1] S. K. Chaharsooghi, J. Heydari, y S. H. Zegordi. A reinforcement learning model for supply chain ordering management: An application to the beer game. *Decision Support Systems*, 45(4):949–959, 2008.

[2] F. Strozzi, J. Bosch, y J.M. Zaldívar. Beer game order policy optimization under changing customer demand. *Decision Support Systems*, 42(4):2153–2163, 2007.