

UE - Architectures, modèles et langages de données

Big data project : Reinforcement learning

Hotelling's model of spatial competition

Moritz Müller, Professeur en Statistique & Econométrie
et Chercheur affilié au Bureau d'Economie Théorique et Appliquée
(BETA) à l'Université de Strasbourg

KHERMOUCHE Chafi
JANATI Zachary

SOMMAIRE :

1. Introduction.....	3
1.1 Contexte et Importance.....	3
1.2 Objectifs de l'étude.....	3
1.3 Structure du Projet.....	3
2. Basic model.....	4
3. Extension/Our version.....	5
3.1 Le principe.....	5
3.2 La solution/L'équilibre du modèle.....	5
3.3 La situation optimale.....	6
4. Implementation.....	7
4.1 La structure.....	7
4.2 Processus de Simulation (Initialisation, Processus de Tournoi) :.....	8
4.3 Algorithme d'Apprentissage par Renforcement.....	8
4.4 Paramètres, Processus d'Entraînement :.....	8
4.5 L'introduction de la paresse des consommateurs.....	8
5. Analysis.....	9
6. Limitations and Conclusion.....	11

1. Introduction

La compétition spatiale occupe une place cruciale dans le domaine de l'économie et de la théorie des jeux. Le modèle de Hotelling, développé par Harold Hotelling dans les années 1920, offre un cadre théorique permettant d'analyser les dynamiques concurrentielles entre entreprises situées le long d'une ligne spatiale unidimensionnelle (généralement entre 0 et 1). Ce modèle s'est avéré être fondamental pour appréhender les stratégies de localisation des entreprises et les décisions des consommateurs influencées par la proximité géographique.

1.1 Contexte et Importance

Le contexte économique contemporain met en lumière l'importance croissante de la localisation stratégique des entreprises, que ce soit dans le secteur du commerce de détail, de la restauration, ou d'autres industries physiquement enracinées. Il est essentiel de comprendre comment les entreprises positionnent leurs établissements et comment les consommateurs réagissent à ces choix pour prédire les tendances du marché, optimiser les stratégies de marketing, et éclairer les décisions de gestion.

1.2 Objectifs de l'étude

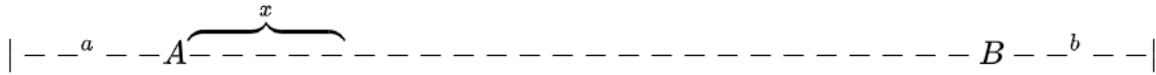
Dans le cadre de ce projet, l'objectif est d'implémenter le modèle de Hotelling en utilisant des agents d'apprentissage par renforcement en Python. Nous explorerons les principaux concepts sous-jacents, les hypothèses du modèle, ainsi que ses applications pratiques.

1.3 Structure du Projet

La présente étude sera organisée de manière à fournir une compréhension approfondie du modèle de Hotelling. Dans la section suivante, nous examinerons les fondements théoriques du modèle, en mettant en évidence ses principaux éléments. Par la suite, nous détaillerons les extensions du modèle que nous avons choisis d'incorporer dans notre projet. On pourra ensuite passer à la partie implémentation dans laquelle on détaillera la structure du modèle, le processus de simulation, l'algorithme d'apprentissage de renforcement etc.... Enfin, nous fourniront notre analyse des résultats et nous conclurons en résumant les principales limites.

2. Basic model

Le modèle général de Hotelling est un modèle de duopole avec deux entreprises dans un contexte de compétition spatiale. En effet, les deux entreprises se situent sur une route, qu'on représente par une droite graduée de 0 à 1, et l'objectif est de maximiser les profits, en tenant compte de la réaction de l'autre entreprise :



Initialement, la première entreprise se situe à une distance (a) du début de la route et la seconde se situe à une distance (b) de la fin.

Hotelling suppose que le bien est homogène (Il n'y a pas de différenciation par la qualité du bien) et que les consommateurs sont répartis sur le marché (la route) de manière uniforme. Au niveau des consommateurs, on considère qu'il achète qu'une seule unité de bien et que ce choix se fera selon les critères suivants : le prix de vente et le coût de transport liée à la distance entre le consommateur et l'entreprise.

Au final, il y a bel et bien une différenciation des produits, non pas par la qualité, mais plutôt par les coûts de transport associée à l'achat du bien.

Dans le modèle général, on suppose que les prix sont différents. Cela signifie que les parts de marché sont déterminées par les prix fixés par A et B. Cependant, si cela n'avait pas été le cas, la part de marché de A serait les consommateurs à sa gauche (a) et la moitié des consommateurs entre A et B (voir image plus haut).

Voici la solution que Hotelling donne :

$$p_A = u + c\left[L + \frac{1}{3}(a - b)\right]$$

$$p_B = u + c\left[L - \frac{1}{3}(a - b)\right]$$

Avec :

- u = prix de revient (= 0 pour Hotelling)
- L = Longueur de la route (est égale à 1 dans notre cas)

Hotelling constate que le profit de A augmente s'il se rapproche de B (c'est-à-dire lorsque A augmente). Les deux entreprises ont alors intérêt à se rapprocher. C'est pour cela que le modèle de Hotelling est aussi appelé le principe de différenciation minimale. En effet, le principe affirme que sur la plupart des marchés, la concurrence conduit les producteurs à réduire la différence entre leurs produits (par la réduction de la distance). Par contre, si nous admettons que les coûts de transports sont désormais du second degré, alors on observerait une tendance inverse (différenciation).

Dans notre cas, on reste sur l'hypothèse suivante : les coûts de transport sont fixes et égale à c . Il faut tout de même faire attention car, si les deux entreprises sont trop proches, alors, le prix d'équilibre sera égal au coût unitaire et le profit s'annule, ce qui ne profite pas aux entreprises.

3. Extension/Our version

Nous venons de vous annoncer le modèle général d'Hotelling de compétition spatiale. Dans le cadre de notre projet, nous avons décidé d'implémenter le modèle sous un format un peu plus simplifié. Il s'agit d'un exemple célèbre de la théorie des jeux : **Le problème des marchands de glace**

3.1 Le principe

Le principe est le suivant : Deux vendeurs de crème glacée doivent sélectionner un emplacement sur une plage où les clients sont répartis uniformément.

On suppose que les prix et les produits des deux vendeurs sont identiques (la distinction se fait uniquement sur l'emplacement des vendeurs, c'est-à-dire que les produits diffèrent uniquement par les coûts de transport), de sorte que chaque client se dirige systématiquement vers le vendeur le plus proche.

La problématique comporte deux aspects. D'une part, il s'agit de définir la position d'équilibre de ce jeu, c'est-à-dire comment les vendeurs vont se positionner sur la plage, en supposant que chacun cherche à maximiser ses profits. D'autre part, il s'agit d'évaluer l'optimalité de cet équilibre, tant du point de vue des vendeurs que des clients.

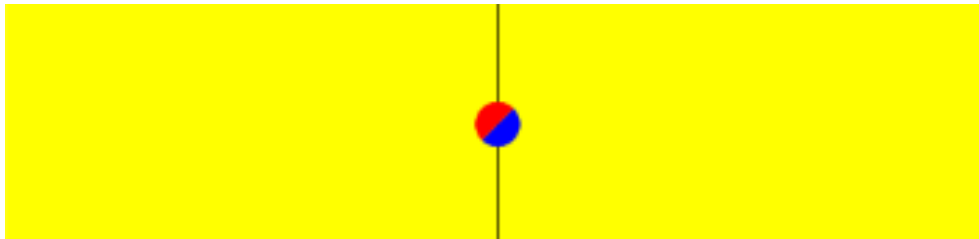
3.2 La solution/L'équilibre du modèle

Une fois que les deux vendeurs sont installés, ils divisent naturellement la plage en deux zones : la zone d'un vendeur consiste en l'ensemble des points de la plage qui sont plus proches de lui que de l'autre vendeur.

Si l'une des deux zones des vendeurs est plus petite que l'autre (ce qui se produit si un vendeur est plus éloigné du centre de la plage), il peut étendre sa zone en se déplaçant. Par conséquent, il n'y a pas d'équilibre. Ainsi, l'équilibre ne peut être atteint que si les deux zones ont la même taille, c'est-à-dire si les vendeurs se situent de part et d'autre du milieu de la plage, à une distance égale.

Cependant, si l'un des vendeurs se rapproche du milieu de la plage, il augmentera sa zone aux dépens de l'autre, qui devra également se rapprocher du milieu de la plage pour préserver sa moitié de plage. Par conséquent, les deux vendeurs se déplacent spontanément vers le milieu de la plage, jusqu'à ce qu'ils soient tous deux présents.

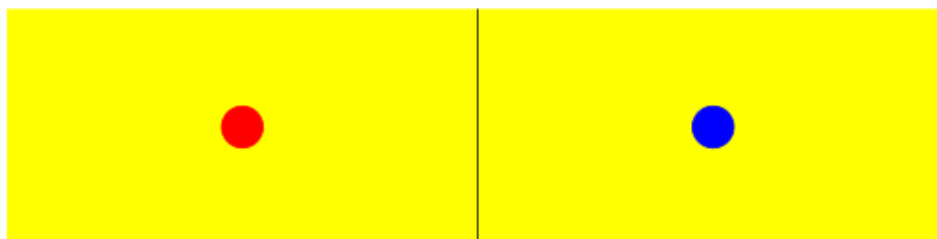
À ce stade, un équilibre est atteint : chaque vendeur occupe la moitié de la plage, et tout déplacement léger d'un côté ou de l'autre entraîne une diminution de sa zone au profit de son concurrent. C'est l'équilibre de Nash dans ce jeu (voir l'image ci-dessous).



3.3 La situation optimale

Ce jeu engendre des externalités, car les clients ne sont pas indifférents à la position des marchands, ce qui impacte leur marche. En particulier, la position d'équilibre, avec les deux marchands au centre de la plage, n'est pas idéale car les clients qui se situent à l'extrémité de la droite doivent parcourir la moitié de la plage pour acheter leur glace.

Une répartition plus favorable des vendeurs (du point de vue de l'optimum social) serait d'avoir un vendeur au milieu de chaque moitié de la plage (voir l'image ci-dessous).



Dans ce cas, chaque vendeur occuperait toujours une zone égale à la moitié de la plage, et les clients n'auraient à traverser au plus que le quart de la plage pour acheter leur glace. Cependant, cela ne constitue pas un équilibre car il est possible pour les deux vendeurs de se déplacer tout en augmentant leur profit.

Ainsi, du point de vue des clients, l'équilibre de ce jeu n'est pas optimal. Il est possible de rendre cet équilibre non-optimal également pour les marchands en supposant qu'un client préfère renoncer à sa glace plutôt que de traverser plus d'un tiers de la plage. Dans ce cas, l'équilibre pour les marchands serait de se positionner chacun à un tiers de la plage, car si un marchand souhaite se rapprocher du milieu pour attirer un client de son concurrent, il devra alors en perdre deux qui viendraient de l'extrémité de la plage (en moyenne ou en supposant une répartition homogène des clients sur la plage). Ce nouvel équilibre améliore ainsi la condition des clients sans toutefois être encore optimal pour eux.

4. Implementation

Pour cette quatrième partie, nous allons présenter la partie codage de notre projet et la manière dont nous avons implémenter notre modèle dans Python. Le code que nous avons utilisé respecte bien évidemment la structure que vous avez demandé, c'est-à-dire comportant une classe Buffer, Policy, Trainer ainsi qu'un environnement (inclu dans le trainer) pour notre modèle.

Notre code se divise en deux partie : la première correspond au modèle sans inclure d'hypothèse par rapport à la paresse des consommateur et dans la seconde partie, on inclut la paresse dans le sens où les individus ne vont pas acheter le biens si la distance est élevé quant bien même le vendeurs est le plus proche. Nous verrons que cela à un impact plutôt satisfaisant sur l'équilibre.

4.1 La structure

Comme je l'ai dit précédemment, L'architecture du modèle repose sur trois classes fondamentales : Buffer, Policy, et Trainer, chacune dédiée à des aspects spécifiques de la modélisation.

La classe Buffer est instanciée avec la taille de l'espace (qu'on a fixé à 250) et le nombre de consommateurs aléatoires/consommateurs (qu'on a fixé à 5000). Des emplacements aléatoires sont générés pour les consommateurs.

La classe Policy est initiée avec la taille de l'espace (250). Elle crée des tables Q pour deux agents. On a utilisé la méthode "update_q_table" qui met à jour la table Q en fonction de l'état actuel, de l'état suivant, de la récompense, du taux d'apprentissage (qu'on a fixé à 0.85) et du facteur de réduction (qu'on a fixé à 0.9)

Le Trainer est instancié avec des paramètres tels que le nombre d'agents/vendeurs (qu'on a fixé à 2), la taille de l'espace (250), le taux d'apprentissage (0.85), le facteur de réduction (0.9), la décroissance de l'exploration (0.995), et le nombre de consommateurs aléatoires (5000). Il initialise également le tampon et la politique.

On a utilisé la méthode "consumer_preference" afin de calculer les préférences des consommateurs en fonction des distances entre les consommateurs et les agents.

4.2 Processus de Simulation (Initialisation, Processus de Tournoi) :

Le processus de simulation débute par l'initialisation de l'objet Trainer, suivi de l'exécution d'une boucle de simulation. Durant chaque épisode, les positions des agents sont actualisées, les récompenses sont calculées selon les préférences des consommateurs, et les tables Q de la politique sont mises à jour. Une interaction de type tournoi émerge alors que les agents rivalisent pour l'attention des consommateurs en fonction de leurs positions spatiales.

4.3 Algorithme d'Apprentissage par Renforcement

L'algorithme d'apprentissage par renforcement repose sur la méthode Q-learning. La méthode "update_q_table" de la classe Policy applique la règle de mise à jour Q, prenant en compte l'état actuel, l'état suivant, la récompense, le taux d'apprentissage et le facteur de réduction. Des stratégies d'exploration et d'exploitation guident la mise à jour des positions des agents dans l'environnement.

4.4 Paramètres, Processus d'Entraînement :

La classe Trainer est initialisée avec divers paramètres, dont le nombre d'agents (qu'on a fixé à 2), la taille de l'espace, le taux d'apprentissage, le facteur de réduction (0.9), la probabilité d'exploration (0.85), la décroissance de l'exploration (0.995), et le nombre maximal d'épisodes (qu'on a fixé à 8000)

L'entraînement se déroule à travers une boucle de simulation pour un nombre défini d'épisodes. À chaque épisode, les positions des agents sont actualisées en fonction de l'exploration ou de l'exploitation. Les récompenses sont calculées selon les préférences des consommateurs, et les Q-Table sont mises à jour grâce à l'algorithme Q-learning. La simulation inclut une représentation graphique des positions des consommateurs et des agents. La probabilité d'exploration décroît progressivement au fil des épisodes.

Pour résumer, Cette implémentation modélise le scénario de compétition spatiale de Hotelling, où les agents positionnent stratégiquement pour attirer les consommateurs dans un environnement spatial. La simulation offre des perspectives sur la convergence des agents vers le centre de l'espace au fil du temps.

4.5 L'introduction de la paresse des consommateurs

La prise en compte de la paresse fait qu'il y ait un nouveau paramètre ("laziness_threshold") qui a été introduit pour la classe Trainer. Elle représente le seuil de paresse des consommateurs, une considération supplémentaire dans le calcul des préférences des consommateurs.

Le deuxième code inclut une vérification dans la méthode "consumer_preference" pour exclure les consommateurs dont la distance à tous les agents est supérieure au seuil de paresse (qu'on a fixé à 50). Ces consommateurs sont ignorés dans le calcul des récompenses.

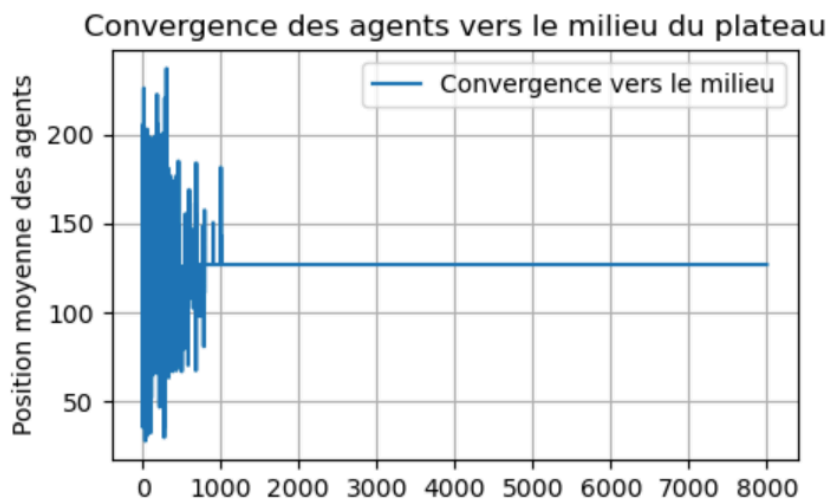
5. Analysis

Dans cette partie, nous allons mettre en avant nos résultats suite au lancement du modèle. Comme je l'ai dit plutôt, nous avons impliqué le problème des marchands de glace avec et sans paresse des consommateurs.

Pour ce qui est du modèle sans paresse, nous obtenons une convergence des positions au milieu de la droite qui correspond à l'équilibre de Nash, comme le prédit le modèle (voir l'image ci-dessous)

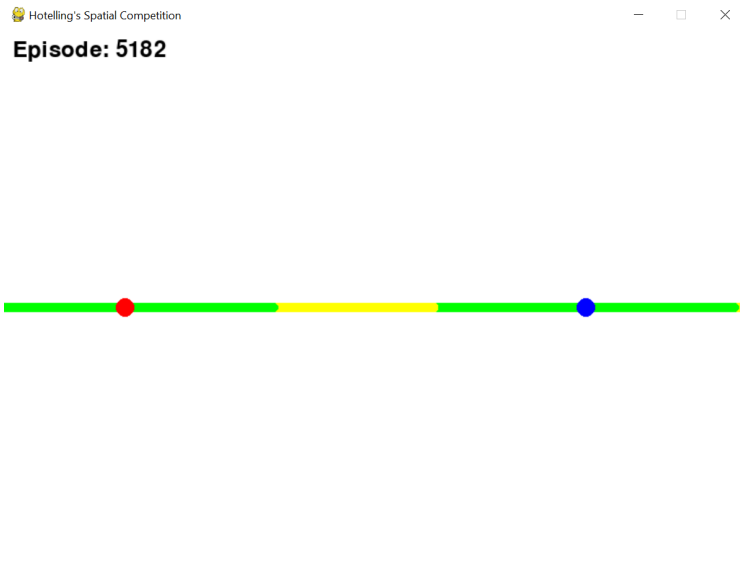
Hotelling's Spatial Competition

Épisode: 6052

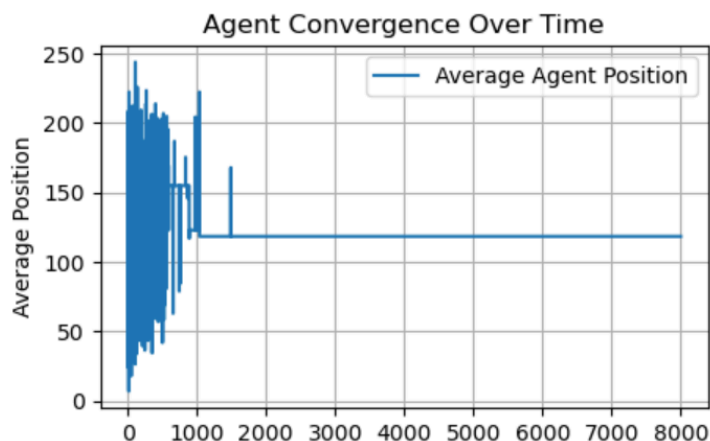


Ce graphique nous montre que, à partir du 1000ème épisode, les positions des vendeurs convergent vers l'équilibre de Nash : A et B se positionnent en 125. La moyenne est donc de 125.

Cependant, il est nécessaire de se rappeler que, dans la section 3.3, nous avons vu que l'équilibre n'était pas synonyme d'optimum social. En effet, l'équilibre ne bénéficie pas aux consommateurs présents aux extrémités de la plage. La solution à cela est justement d'introduire la paresse des individus qui vont se retirer du marché si la distance avec le vendeur le plus proche est supérieur à un certain seuil (qu'on a fixé à 50). Ainsi, nous retrouvons l'optimum social dans lequel les vendeurs seraient au milieu de chaque moitié de la plage (voir image ci dessous).



Les consommateurs représenté en jaune sont les consommateurs paresseux, qui ne vont pas acheter le biens.



Ce graphique nous montre que, à partir du 1500ème épisode, les positions des vendeurs convergent vers l'optimum social : A se situe en 62.5 et B se situe en 187,5. La moyenne est donc de 125 $((62.5 + 187.5)/2)$

Mais ce n'est pas tout, la paresse nous permet deux autres convergences (qui ne me semble pas être des équilibres) différents.

Les deux autres sont des convergences où les deux firmes ignorent totalement le côté opposé, gauche ou droite, ce qui correspond aux deux cas constatés. Cela peut s'expliquer par la méthode de calcul utilisée dans l'idée de paresse du consommateur, il aurait probablement fallu inclure l'idée de décroissance de la paresse du consommateur. Dans ce cas précis, il n'y aurait pas eu les 3 convergences constatées.

Une autre amélioration possible est l'idée de faire un nombre important d'essais des paramètres, de manière automatique, afin de tester les "meilleures" valeurs que le modèle pourrait considérer.

6. Limitations and Conclusion

Le modèle implémenté capture efficacement les dynamiques de la concurrence spatiale entre les agents dans un scénario d'allocation de ressources. L'utilisation du Q-learning et des aspects spatiaux des préférences des consommateurs contribue à une simulation réaliste. L'intégration d'un seuil de paresse pour les consommateurs ajoute une couche supplémentaire de complexité, reflétant un scénario du monde réel plus nuancé.

Malgré les forces du modèle, certaines limitations subsistent. L'implémentation actuelle suppose un environnement spatial discret et une distribution uniforme des consommateurs, ce qui pourrait ne pas refléter pleinement la complexité des scénarios réels. De plus, la simplicité du modèle de concurrence spatiale peut ne pas tenir compte de comportements de marché plus complexes.

Plusieurs améliorations pourraient être envisagées. La transition vers un environnement spatial continu pourrait offrir une représentation plus réaliste. Des améliorations dans la modélisation des préférences des consommateurs, en intégrant des comportements et des préférences diversifiés, pourraient enrichir la simulation. Affiner les paramètres d'apprentissage et explorer d'autres algorithmes d'apprentissage par renforcement pourrait optimiser les performances du modèle.

Pour finir, le projet a fourni des perspectives précieuses sur les dynamiques de la concurrence spatiale et l'application de l'apprentissage par renforcement pour simuler des scénarios économiques.

7. Sources

Hotelling, Harold. "Stability in Competition." *Economic Journal* 39.153 (1929): 41-57.
Wikipédia | Wikiwand

"La règle de Hotelling." *The Other Economy*. Lien

d'Autume, Antoine, and Katheline Schubert. "Le modèle de Hotelling avec coûts de transport et concurrence imparfaite." *HAL Archives Ouvertes* (1991). Lien vers le PDF