



UNIVERSITÉ PARIS 1 PANTHÉON-SORBONNE

ÉCOLE D'ÉCONOMIE DE LA SORBONNE  
MASTER 1 ÉCONOMÉTRIE STATISTIQUES

ANNÉE UNIVERSITAIRE 2022-2023

---

# Prévisions de crises économiques

---

*Étudiants :*

Cécile HUANG

Yoan JSEM

Alice LIU

*Enseignant :*

Aryan

RAZAGHI

# Table des matières

<b>1</b>	<b>Introduction</b>	<b>2</b>
<b>2</b>	<b>Le modèle</b>	<b>3</b>
2.1	Modèle d'Estrella et Mishkin . . . . .	3
2.2	Le modèle sélectionné . . . . .	4
2.3	Les variables sélectionnées . . . . .	4
2.4	Les critères et métriques . . . . .	5
<b>3</b>	<b>Les résultats</b>	<b>8</b>
3.1	Les résultats d'Estrella et Mishkin . . . . .	8
3.2	Performance des variables en In Sample . . . . .	9
3.3	Performance des variables en Out Sample . . . . .	12
<b>4</b>	<b>Pour aller plus loin : les modèles de Machine Learning</b>	<b>15</b>
4.1	Le modèle avec le Random Forest . . . . .	15
4.2	Le modèle avec l'AIC . . . . .	16
4.3	Le modèle Ridge . . . . .	17
4.4	Le modèle Logit avec les pénalités $\ell_1$ et $\ell_2$ . . . . .	17
4.4.1	Pénalité $\ell_1$ . . . . .	18
4.4.2	Pénalité $\ell_2$ . . . . .	18
4.5	Le modèle Naive Bayes . . . . .	19
4.5.1	Naive Bayes Gaussien . . . . .	20
4.5.2	Naive Bayes de Bernoulli . . . . .	21
4.6	Conclusion sur le Machine Learning . . . . .	21
<b>5</b>	<b>Conclusion</b>	<b>23</b>

---

**Résumé.** Dans ce projet, nous allons nous intéresser à la prévision de crises économiques entre les années 1961 jusqu'à 2022. On essaiera de vérifier si les conclusions d'Estrella et Mishkin (1995) sont toujours d'actualité en reprenant leur méthode. Pour aller plus loin on essaiera de challenger cette méthode et de voir si l'on peut faire mieux à l'aide de modèles de Machine Learning.

---

## 1 Introduction

Une crise économique est un phénomène touchant l'économie d'un pays de façon brutale : elle dégrade sa situation et affecte son système économique, pouvant mener à une récession. Suivant sa gravité, une crise économique peut se propager plus ou moins vite dans les autres pays et perdurer plusieurs années.

Selon *l'INSEE*, la récession est une "période de recul temporaire de l'activité économique d'un pays", et on observe "un recul du Produit Intérieur Brut (PIB) sur au moins deux trimestres consécutifs".

Si l'on s'attache à décrire l'économie, le temps long de l'histoire nous démontre que les crises sont des événements rares. Pour autant, leurs impacts sur l'économie sont bien réels et pouvoir prédire les crises serait un grand atout. Le papier "*Predicting U.S. Recessions : Financial Variables as Leading Indicators*" écrit en décembre 1995 par Arturo Estrella et Frederic S. Mishkin qui sont respectivement professeur à l'Institut Polytechnique Rensselaer et professeur à l'Université de Columbia à New York. Ils ont essayé de vérifier la relation empirique selon laquelle l'inversion de la courbe des taux avait un pouvoir prédictif. Les auteurs cherchent donc à vérifier si l'on peut prédire une récession à l'aide de variables financières. Ils n'entendent pas remplacer les modèles macro-économiques existant, loin de là. En effet, dans ce papier de 1995 ils espéraient seulement proposer un indice supplémentaire pour pouvoir anticiper et prévoir les crises. A l'aide d'un modèle *probit* et de métrique telle que le pseudo  $R^2$ , les auteurs pouvaient prévoir une crise à partir de variables comme le Spread, le S&P 500, des variables financières, monétaires et indicateurs macro-économiques.

A partir d'une base de données regroupant plus de 200 variables économiques et financières, notre problématique sera donc de savoir quelles sont celles que l'on va retenir pour pouvoir prédire une récession.

*Les conclusions d'Estrella et Mishkin sont-elles encore valable à ce jour ? Est-il possible d'aller au delà ?*

Pour cela, nous allons voir dans un premier temps le modèle que l'on va utiliser, puis nous allons estimer les performances de notre modèle en in-sample et out-sample, enfin nous allons utiliser des modèles de Machine Learning afin de challenger nos résultats.

## 2 Le modèle

Dans cette première partie nous revenons sur la construction des données et du modèle par Estrella et Mishkin qui a servi d'inspiration pour notre propre approche du modèle et données. Ensuite, nous introduisons les métriques utilisées tout le long de notre projet.

### 2.1 Modèle d'Estrella et Mishkin

Dans le but de mesurer la performance prédictive des variables financières sur la récession économique, les auteurs Estrella et Mishkin procèdent par une approche de régression statistique en utilisant le modèle *probit*. En effet, la variable cible (la récession) est une variable binaire, les moindres carrés ordinaires (MCO) ne peuvent être utilisés pour effectuer la régression. L'utilisation du modèle *probit* permet d'obtenir des résultats similaires aux MCO dans ce contexte.

La variable récession binaire ( $Y$ ) se présente comme ceci :

$$\begin{cases} Y = 1 \text{ en cas de récession} \\ Y = 0 \text{ sinon} \end{cases}$$

Les auteurs ont travaillé sur une base de données temporelle trimestrielle plutôt que mensuelle, car des données mensuelles auraient entraîné des résultats plus faibles en termes de performance prédictive. Dans leur base de données, ils ont inclus des variables explicatives comme l'indice des principaux indicateurs économiques du Département du Commerce, ainsi que deux indices construits par Stock et Watson.

Une fois les données définies, les auteurs estiment les retards optimaux pour chaque variable en utilisant le critère du Pseudo  $R^2$  d'Estrella, mais aussi la significativité à 5% et 1%. Les auteurs ont déterminé que les lags les plus significatifs au sens de ces critères étaient le lag 2. Si l'on regarde les résultats plus en détails, le Spread, était significatif à tout horizon (1 à 8 trimestres au-delà de  $t$ ). L'indice de la masse monétaire réel aussi, en revanche le PIB trimestriel n'était significatif que sur 3 trimestres, au-delà il devenait obsolète. De manière générale, ils relèvent parmi les variables financières que les variables qui suivent l'indicateur du Stock Watson sont robustes en performance. Aussi des variables comme le cours d'une action et le Spread des bons du trésor suivent une tendance similaire à ces indices même s'ils ne rivalisent pas avec le Spread des taux.

Parmi les variables sélectionnées les auteurs ont mentionné le Spread qui serait l'écart de taux d'intérêt entre le Trésors à 10 ans et à 3 mois, des taux d'intérêts, ainsi que l'indice des prix (NYSE, Dow Jones industriels et *S&P500*) , et d'autres variables macroéconomiques. Dans le cadre de leur étude, les auteurs ont effectué une division de leur base de données en deux parties distinctes. Cette division a permis de créer une partie en échantillon (in sample) allant de 1959 à 1971 et une partie hors échantillon (out sample) de 1971 à 1995.

L'objectif de cette division était de mesurer la capacité prédictive des variables dans deux situations distinctes. Les données antérieures à 1970 ont été utilisées pour entraîner leur modèle et évaluer ses performances prédictives. En revanche, la partie hors échantillon a permis l'évaluation des performances prédictives des modèles face à de nouvelles données. Pour mesurer la capacité prédictive les auteurs se sont basés sur le pseudo  $R^2$  la  $t$ -statistique et des seuils de significativité à 5% et 1%.

## 2.2 Le modèle sélectionné

Comme dans le rapport d'Estrella et Mishkin, nous travaillons sur une base de données trimestrielles temporelles et une variable cible binaire. Cette variable binaire est nulle autre que la récession que nous retrouvons sous le nom de « USRECD » dans les données. Cette variable binaire viens d'une base de donnée connexe fournie par la "Federal Reserve Bank of St. Louis".

Dans notre cas, nous avons fait le choix de travailler avec un modèle *logit* plutôt que *probit* pour pouvoir avoir des coefficients interprétables.

Notre base de données a été construite de sorte à avoir la période 1959 à 2022 pour avoir une période plus étendue que celle des travaux d'Estrella et Mishkin et ainsi vérifier si les résultats des auteurs étaient toujours d'actualité. On intègre également dans nos données la variable NYSE et Dow Jones mentionnés par les auteurs grâce à une concaténation. Pour ajouter ces deux variables il fallait, comme pour la variable binaire ajouter 2 mois à chaque date.

Ensuite, nous avons décidé de segmenter nos observations tous les 10 ans afin d'obtenir pour chaque période la moyenne et ainsi de traiter les variables manquantes. Procéder ainsi permet d'éviter de trop biaiser les données. Une fois qu'il n'y a plus de valeurs manquantes nous pouvons intégrer des valeurs retardées. Nous avons imposé une limite de variable retardée de 10. Pour trouver les lags optimaux, nous avons procédé par le modèle probit qui nous fournit le Pseudo  $R^2$  de McFadden :

$$McFadden R^2 = 1 - \left( \frac{\log L_u}{\log L_c} \right)$$

Avec  $L_u$  la vraisemblance du modèle estimé, et  $L_c$  la vraisemblance du modèle avec seulement une constante.

Pour obtenir le pseudo  $R^2$  d'Estrella on doit donc ajouter à  $R^2$  de McFadden une puissance de sorte à avoir :

$$Pseudo R^2 = 1 - \left( \frac{\log L_u}{\log L_c} \right)^{-\frac{2}{n} \log L_c}$$

Tout comme les auteurs, nous avons constaté que les indices de prix NYSE et Dow Jones n'ont pas de retard optimal dans notre analyse. Cela signifie que l'ajout de retards pour ces indices n'ont pas amélioré le pseudo  $R^2$ .

Intégrer ces variables a apporté des valeurs manquantes dans la base de données qu'on a dû supprimer. On a donc maintenant une base de données allant de 1961Q3 à 2022Q3.

## 2.3 Les variables sélectionnées

En ce qui concerne la sélection de variables, nous avons utilisé une métrique de classification tel que F1-score et le pseudo  $R^2$  d'Estrella. Nous avons donc d'abord procédé à la division de notre base de données en 1995 qui est une date sans récession et sans événement majeur. La partie avant cette date est utilisée comme donnée d'entraînement et la partie après 1995 est utilisée comme données test. Ce split va permettre de faire entraîner notre modèle sur les données d'entraînement et de mesurer la performance prédictive en échantillon. La partie test va permettre de voir comment le modèle performe face à des nouvelles données hors échantillon. Nous avons par ailleurs décidé de sélectionner les 20

variables ayant les meilleurs pseudo  $R^2$  d'Estrella et par raisonnement économique nous avons sélectionné parmi ces 20 variables 10 variables significatives au sens économique pour expliquer la récession.

```
variables_30
✓ 0.0s

['FPIx',
'T5YFFM lag 3',
'CMRMTSPLx',
'USCONS',
'Y033RC1Q027SBEAx',
'GS10TB3Mx lag 4',
'USGOOD',
'WPSFD4111 lag 4',
'IPMANSICS',
'DFXARG3Q086SBEA lag 3']
```

Nous retrouvons la variable Spread retardée de 4 trimestres dans la liste. Ensuite, dans l'objectif de trouver la meilleure combinaison de variables au sens de la performance prédictive, nous avons réalisé toutes les combinaisons possibles avec les 10 variables sélectionnées. A chaque combinaison, nous avons mesuré le F1-score pour évaluer la performance.

Nous avons accordé une grande importance à la sélection des variables afin d'obtenir de meilleures prédictions à la fois sur les données en échantillon et hors échantillon. Par prolongement à la méthode proposée par les auteurs on teste une à une les variables mais aussi toutes les combinaisons possibles de variables. Nous cherchons à identifier si une combinaison est meilleure qu'un modèle avec qu'une seule variable.

La meilleure combinaison possible est :

$$USRECD = const + \beta_1 \times FPIx + \beta_2 \times CMRMTSPLx \\ + \beta_3 \times GS10TB3Mxlag4 + \beta_4 \times IPMANSICS$$

Avec un F1 Score de 0.9.

Cette combinaison optimale nous permet de construire un modèle plus robuste et fiable que celle du papier d'Estrella et Mishkin qui se sont limités à une combinaison des variables une à une avec le spread. Le F1-score est de 0.9 ce qui atteste que notre modèle a une très bonne performance prédictive.

## 2.4 Les critères et métriques

Dans notre projet nous utilisons différentes métriques à différentes étapes de notre code. Il est donc essentiel de définir ces métriques et de les comprendre.

Nous utilisons une métrique utilisée dans le papier d'Estrella et Mishkin, le Pseudo  $R^2$  :

$$Pseudo R^2 = 1 - \left( \frac{\log L_u}{\log L_c} \right)^{-\frac{2}{n} \log L_c}$$

Le pseudo  $R^2$  d'Estrella mesure la proportion expliquée par le modèle, comme le  $R^2$  que nous connaissons habituellement, mais la formule n'est pas la même :

$$R^2 = 1 - \frac{\sum_{i=1}^n (Y_i - \hat{Y}_i)^2}{\sum_{i=1}^n (Y_i - \bar{Y})^2}$$

Une valeur proche de 1 indique que le modèle explique en grande partie la variation de notre variable binaire.

Le recall est défini comme le nombre de vrais positifs divisé par la somme des vrais positifs et des faux négatifs.

$$Recall = \frac{TP}{TP + FN}$$

Avec TP, les vrais positifs et FN les faux négatifs. Le recall mesure la proportion de vrais positifs que le modèle a réussi à trouver parmi tous les vrais positifs. Une valeur élevée du recall indique donc une bonne capacité du modèle à éviter les faux négatifs, c'est-à-dire qu'il y a moins de risque que le modèle prédit à tort qu'il n'y a pas de récession alors qu'il y en a une.

La spécificité est définie comme le nombre de vrais négatifs divisé par la somme des vrais positifs et des faux négatifs :

$$Spécificité = \frac{TN}{TN + FP}$$

La spécificité mesure la probabilité de prédire la non récession dans le cas de non récession.

La Précision est définie comme le nombre de vrais positifs divisé par la somme des vrais positifs et faux positifs.

$$Précision = \frac{TP}{TP + FP}$$

Avec FP, les faux positifs. La précision mesure la proportion de vrais positifs que le modèle a réussi à trouver. Une valeur élevée de la précision indique donc une bonne capacité du modèle à éviter les faux positifs. C'est-à-dire qu'il y a peu de risque que le modèle prédit à tort qu'il y a une récession alors qu'il n'y en a pas.

L'accuracy mesure le nombre de prédiction correctes divisé par l'ensemble des prédictions :

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

L'accuracy mesure la proportion de vrais positifs et de vrais négatifs que le modèle a réussi à trouver. Une valeur élevée indique donc une bonne capacité du modèle à prédire la présence de récession et l'absence de récession.

Le Balanced Accuracy est la somme du Recall et de la spécificité divisée par 2 :

$$Balanced Accuracy = \frac{Recall + Spécificité}{2}$$

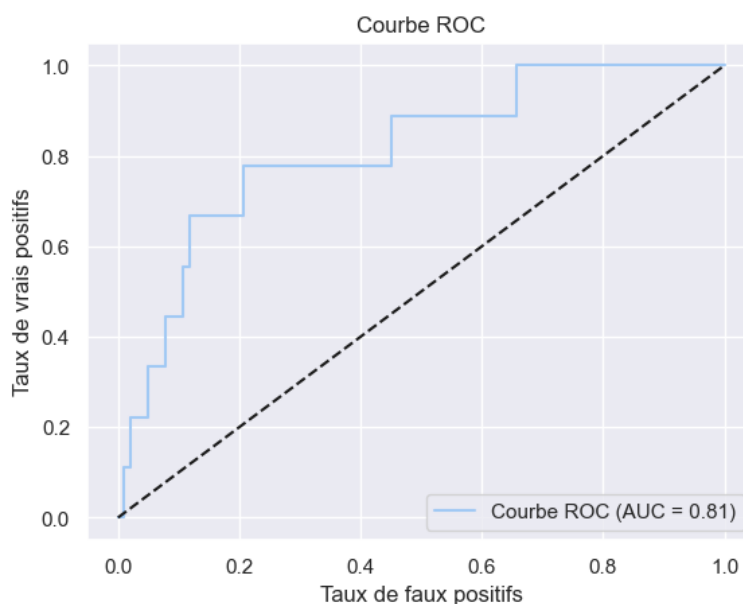
Cette métrique est très utile dans le cas où les 2 classes ne sont pas équilibrées ce qui est le cas dans notre travail on a plus de 0 (pas de récession) que de 1 (présence de récession). Lorsqu'on a ce déséquilibre, l'accuracy n'est pas adapté et donc il vaut mieux regarder le balanced accuracy qui va nous donner une proportion plus réaliste de si le modèle arrive à bien prédire la présence de récession et l'absence de récession.

Le F1-score est une combinaison du recall et de la précision :

$$F1\ score = 2 \times \frac{Recall \times Precision}{Recall + Precision}$$

On peut remarquer qu'un recall élevé permet juste de dire que le modèle ne se trompe pas lorsqu'il dit qu'il n'y a pas de récession. Mais cela peut aussi signifier qu'avec un recall élevé, le modèle prédit souvent 1 la récession alors que ce n'est pas le cas. Au contraire la précision permet de dire que le modèle ne se trompe pas lorsqu'elle prédit une récession mais donc une précision élevée peut signifier que le modèle ne prédit jamais de récession. Le F1-score est une combinaison de ces deux métriques et donc va permettre une bonne évaluation de la performance prédictive de notre modèle.

L'AUC (Area Under the ROC Curve). Pour améliorer le recall il faut renoncer à maximiser la spécificité, un bon modèle serait un compromis entre les deux c'est le rôle du ROC AUC. Le ROC est une courbe qui va évaluer la performance prédictive du modèle pour chaque seuil en tant que relation recall vs Spécificité. La courbe ROC est construite en calculant à chaque fois le recall et la sensibilité :



L'AUC est la probabilité de bonne prédiction mesuré par l'aire sous la courbe ROC. La valeur de AUC se trouve entre 50% et 100%, en dessous de 50% signifie qu'il y a une erreur dans le code. A 50% le pouvoir prédictif est totalement aléatoire. Plus l'AUC se trouve proche de 1 (100%), meilleur est le modèle pour séparer les individus en classes et améliore la prédiction.



## 3 Les résultats

Dans cette section on s'attache à décrire dans un premier temps les résultats obtenus par Estrella et Mishkin, puis on observera les résultats que nous avons obtenus en suivant la même méthode que les auteurs.

### 3.1 Les résultats d'Estrella et Mishkin

On a vu que les auteurs séparaient leur analyse en deux parties, la partie *in sample* et la partie *out sample*.

Dans la partie *in sample*, ils utilisent comme décrit précédemment une régression *probit* pour estimer les valeurs de la cible : la récession.

En regardant maintenant les variables les auteurs observent, pour les bons du trésor, un pouvoir prédictif de 1 à 2 trimestres. Cela fait sens puisque le spread des bons du trésor représente la différence entre les taux à 2 et 6 mois. Pour ce qui est de l'indice de la masse monétaire, on voit qu'il performe bien jusqu'à une année et ce de manière consistante. Cela vient prouver l'efficacité de la politique monétaire de long terme qui devient apparente avec de longs décalages. Enfin l'écart du taux de rendement qui correspond aux taux d'intérêts applicable à 3 mois et 10 ans dans le futur, devrait être un très bon indicateur de récession entre 2 et 6 trimestres dans le futur.

Pour conclure sur les résultats en *in sample*. En combinant le spread avec une autre variable nous sommes quasiment sûrs d'obtenir de grandes performances sur 2 à 3 trimestres. Deux exceptions cependant, on ne peut combiner l'écart des taux de rendement avec les bons du trésors ou la base monétaire réelle sous peine de trouver des résultats non significatifs. Si l'on prend la combinaison du spread et le cours des actions, on s'assure des performances les plus élevées avec l'avantage que ces variables sont disponibles en continues. En effet pour certaines variables comme le PIB on ne peut obtenir leur information en continu, le PIB est trop bruité lorsque l'on observe ses fluctuations par mois.

Pour ce qui est de l'*out sample*, ils commencent leurs prédictions en 1971 afin d'avoir suffisamment de données sur lesquelles apprendre et ils prévoient jusqu'au premier trimestre de 1995. Il s'agit de la date la plus récente à l'heure où l'étude est publiée. Les résultats retrouvés sont similaires à ce qu'on a pu voir précédemment, à quelques différences près. Les variables pour lesquelles on observe de bons résultats sont : la courbe de taux, les bons du trésor, la base monétaire réelle, le cours des actions, et les indices principaux. On observe à la différence du *in sample* une déperdition en terme d'accuracy (probabilité de prédire correctement) et en terme d'horizon. Lorsqu'on parle de détérioration des résultats on a des baisses substantielles comme le spread des bons du trésor qui donnent un pseudo  $R^2$  négatif. Encore une fois on observe une dominance du spread parmi les variables à fort pouvoir prédictif. Même si 7 ou 8 trimestres au delà l'effet prédictif est absent, il est bien plus juste à 2 et 3 trimestres.

Une conclusion aussi, le spread est une variable si forte qu'elle tend à faire de l'ombre à des variables comme la masse monétaire ou encore l'indice du Stock Watson. Si bien

que parfois, la combinaison de 2 variables peut s'avérer donner des résultats bien pires que lorsqu'on utilise la variable seule. En *out sample* les modèles parcimonieux sont donc à privilégier. Pour ce qui est de prédire 1,2,3 et 5 trimestres dans le futur, il vaut mieux privilégier les variables qui définissent les marchés globaux, on peut nommer le *NYSE* (stock exchange) et le *S&P500*. On peut aussi observer que le cours des actions contient de l'information que le spread ne connaît pas, c'est pourquoi il peut être intéressant de l'utiliser. Par ailleurs cette combinaison s'avère être le meilleur choix de variable possible pour prédire la récession. Enfin, la parcimonie n'est pas à sous estimer, on pourrait prendre un modèle moins économe en prenant le PIB, le spread et le *NYSE*. Mais cela ne nous avantagerait pas puisqu'à chaque horizon supplémentaire les prédictions deviennent bien pire pour toutes nos variables.

## 3.2 Performance des variables en In Sample

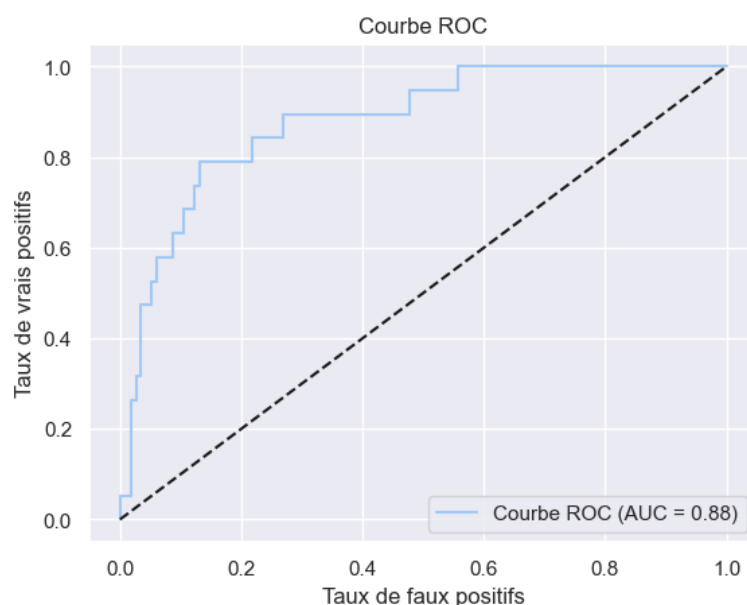
Dans cette partie nous utilisons notre base de données afin de vérifier si vous pouvons aboutir aux mêmes conclusions que les auteurs. Tout comme eux nous utilisons le pseudo  $R^2$  d'Estrella pour calculer les retards et observer la significativité. On regardera aussi la courbe ROC et l'AUC pour pouvoir comparer les résultats en *in sample* et ceux en *out sample*.

L'estimation des retards à l'aide du pseudo  $R^2$  donne des résultats similaires à ce qu'avaient retrouvé Estrella et Mishkin. On retrouve bien le spread, les bons du trésors ou encore des indices de broad market comme le S&P500. Mais on retrouve aussi et surtout, des variables macro-économiques, cela est grandement dû au fait que notre base de données est différente et que nous sommes sur une période plus grande (pour rappel notre base de données s'étale de 1961Q3 à 2022Q3). On trouve alors que l'investissement privé fixe réel (déflaté) est un très bon indicateur pour prédire les récessions et ce sans avoir besoin d'être retardé. L'investissement est un indicateur de la confiance des marchés, personne n'investirait s'il n'a pas confiance en l'avenir, le faire est donc preuve que les agents n'ont pas une vision pessimiste du futur. Par conséquent cela peut être un bon indicateur de récession, on n'investit pas lorsque l'on est en récession car l'avenir est incertain, la récession peut sans doute devenir dépression. Le bon du trésor est tout comme la conclusion des auteurs une variable significative et à fort pouvoir prédictif, au même titre que la courbe des taux. De nouvelles variables significatives macro-économiques, comme le taux d'emploi/chômage dans différents secteurs, s'avèrent significatives. De manière générales les variables relevées sont significatives à 1%. Contrairement à ce qu'ont fait les auteurs ici nous préférons prendre un lag personnalisé pour chaque variable, en effet plutôt que de trouver un seul décalage pour toutes les variables nous préférons lagger en fonction du meilleur lag au sens du pseudo  $R^2$ .

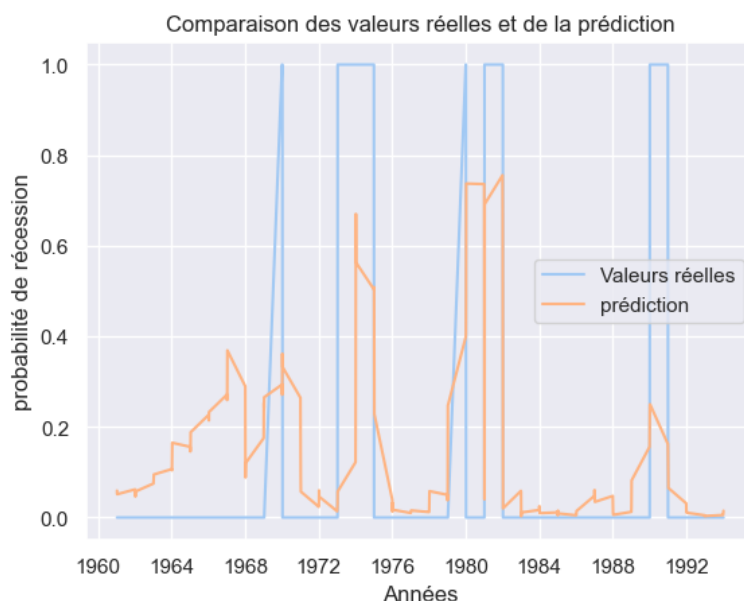
Dans l'objectif de pouvoir prédire en *out sample*, on décide d'enlever du modèles des variables dont l'information est déjà contenue dans d'autres variables. Par exemple l'investissement privé fixe réel des **résidents** est déjà compris dans la variable d'investissement réel privé fixe total. La même information est donc contenue dans ces deux variables, on choisi donc la variable avec le plus grand pseudo  $R^2$ . On peut alors réduire le nombre de variable à 10, on teste alors différentes combinaisons linéaire avec ces variables pour obtenir la meilleure combinaison de variables pour prédire la récession. Nous utilisons une

métrique de classification afin de trancher de si un modèle est bon : le F1-Score. Après avoir réalisé cela on se retrouve avec notre meilleure combinaison de variable au sens du F1, il s'agit d'un modèle avec constante composé de 4 variables : l'investissement réel privé fixe, les ventes des industries manufacturières et commerciales déflatés, l'écart du taux de rendement et la production industrielle manufacturée.

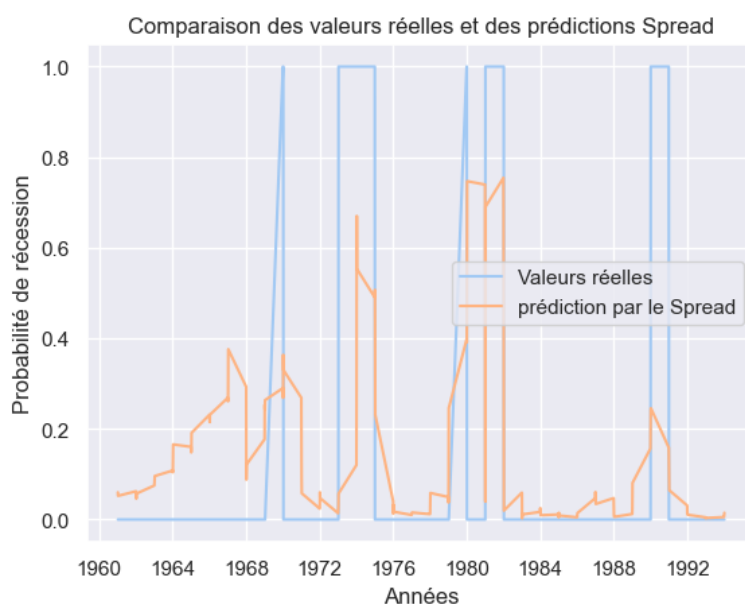
Avant de regarder les résultats il est bon de préciser que en *in sample*, avec le critère choisi on préférera toujours avoir le plus de variables possibles. Cependant, plus ne veut pas dire mieux et cela se vérifie dans la partie *out sample*. De manière générale les résultats sont meilleurs dans le *in sample*, on a une AUC de 0.88. Un résultat proche de 1 signifie une bonne performance pour le modèle, on peut donc avancer que le modèle à une bonne capacité de discrimination.



Si l'on regarde le graphique qui compare les valeurs réelles observées à celles qu'on a **fit** en appliquant notre modèle *in sample* on voit de manière générale une bonne performance, bien que avec le temps le modèle s'affaiblit. Encore une fois, nous avons privilégié un modèle plus parcimonieux, en ajoutant plus de variables nous aurions pu sans aucun doute avoir de meilleures prévisions mais, l'objectif ici n'est pas tant de se concentrer sur le *in sample*.



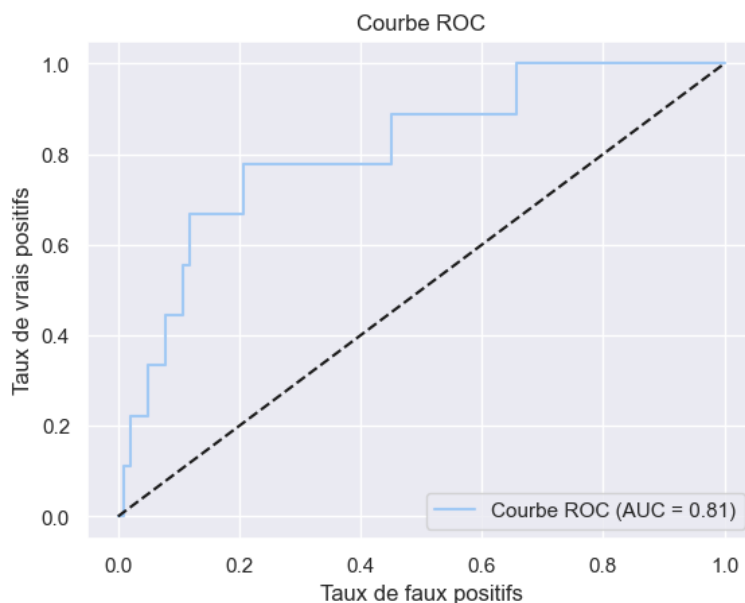
Le spread donne de bons résultats, bien que le premier pic soit discutable, la variable parvient à trouver et suivre les valeurs réelles de manière satisfaisante.



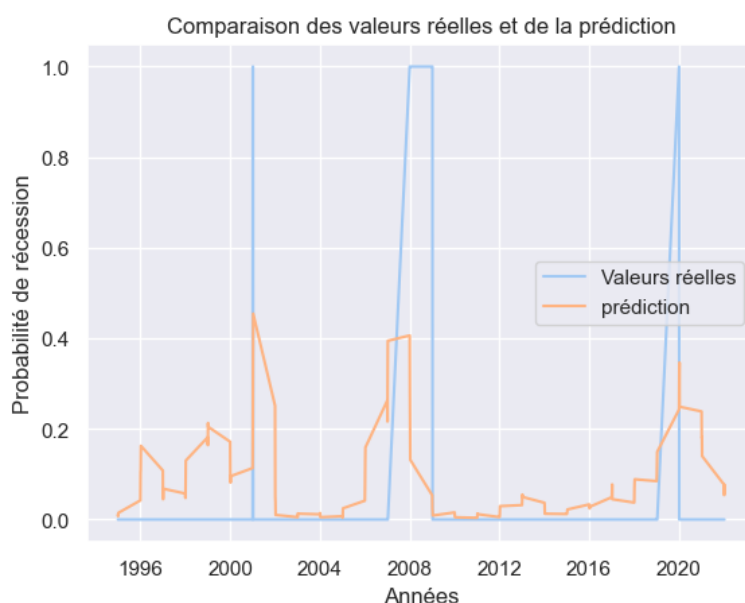
Aussi, lorsqu'on estime par logit la relation, on voit un pseudo  $R^2$  de 0.5894. Estrella et Mishkin ont utilisé le probit pour estimer leur modèle, mais dans notre cas on utilise le logit pour avoir accès aux odd ratios. Par exemple, si l'on prend la variable du spread, en sachant que sa moyenne est de 1,5 et son écart type de 1,21 on peut interpréter comme suit : lorsque le spread augmente d'un point, (une augmentation conséquente), la probabilité de ne pas être en récession par rapport à être en récession diminue de 50% ( $(e^{-0.6915} - 1) \cdot 100\%$ ). Le *logit* a modélisé la probabilité de ne pas être en récession, nos "négatifs" au sens du code sont les récessions. On peut donc aussi dire que la variation du spread, traduit une détérioration de la confiance puisqu'on anticipe une contraction de l'économie, l'écart de la courbe des taux s'agrandit, on observe une inversion de la courbe des taux. On retombe donc bien sur les conclusions attendues.

### 3.3 Performance des variables en Out Sample

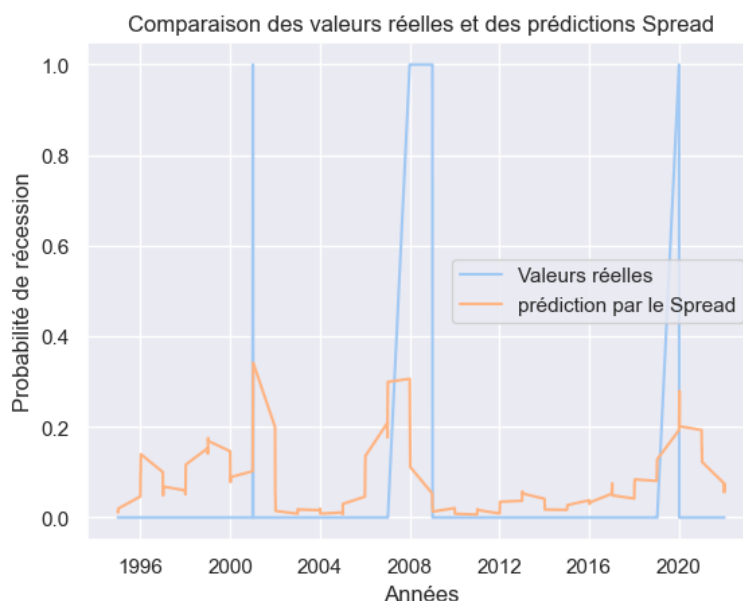
Pour ce qui est de cette partie, on utilise non plus le pseudo  $R^2$  mais le F1-Score, ainsi que la matrice de confusion, la courbe ROC et l'AUC. On étudie la période de 1995Q1 à 2023Q3. Nous avons déjà nos variables sélectionnées de sorte à maximiser le F1 (combinaison du spread et d'indices macro-économiques). On regarde maintenant la courbe ROC et l'AUC. De manière générale les performances sont moins élevées, elle restent satisfaisante néanmoins puisqu'on a une AUC de 0.81 (contre 0.88 auparavant).



Si on s'attarde maintenant à comparer nos valeurs réelles et nos valeurs prédites en *out sample*, on peut voir que le modèle est limité, il parvient à prédire nos récessions mais la probabilité associée reste faible comparativement à la section précédente.



On peut aussi regarder les performances de la variable du spread seule :



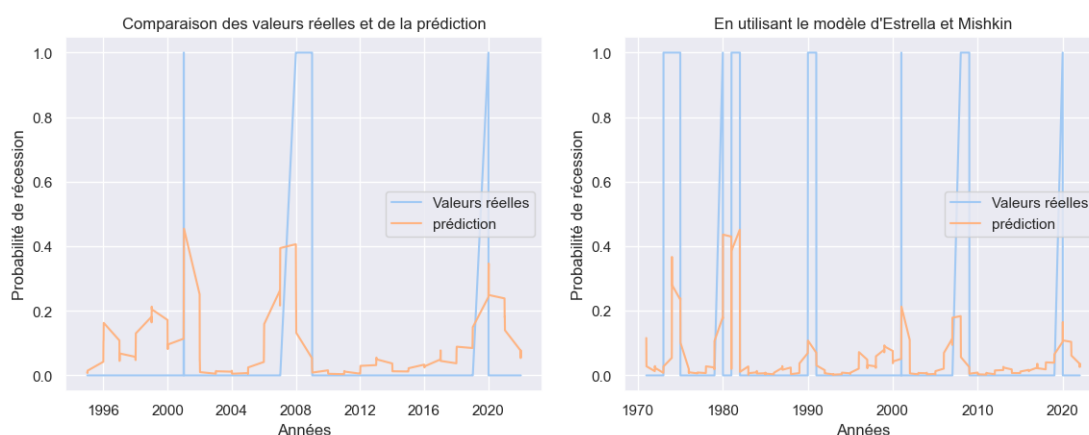
Les prédictions restent correctes, comparé à d'autres variables pour lesquelles la courbe est plate on peut considérer ces résultats comme bons. Pour finir on utilise nos prédictions et on les visualise dans une matrice de confusion. On peut observer suite à notre régression logit, les résultats suivants :

		Valeurs Prédites	
		Négatifs	Positifs
Valeurs réelles	Négatifs	100	2
	Positifs	0	9

On peut faire les remarques suivantes, le modèle a su trouver toutes les récessions. En effet, on ne compte aucun faux négatif, il n'y a aucun moment où le modèle prédit qu'il n'y aura pas de récession alors qu'il y en aura une. Par contre on rencontre 2 faux positifs, on a à certains moment prédit une récession alors qu'il n'y en avait pas. La présence de faux positifs peut être problématique, ne serait-ce que pour l'effet de panique. Il est possible que si l'on prévoit une crise alors les agents se mettent à paniquer, soient réticents à investir et que l'économie se contracte. Cela pourrait même aboutir au phénomène de prophétie auto-réalisatrice : les agents craignant la crise vont en réalité la précipiter. En essayant d'éviter la crise et en étant préoccupé on pourrait réellement arriver à une crise qui n'avait pas lieu d'être à la base. En soi, le modèle que l'on a prolongé est très bon puisqu'il ne se trompe que 2 fois sur un total de 111 prédictions. C'est donc une accuracy de 98,2 %, la probabilité de se tromper est de moins de 2%. Le recall qui qualifie notre taux de faux négatifs est de 1 et notre précision qui qualifie notre taux de faux positifs est de 0.81. En cela on peut dire que les prévisions sont concluantes, la relation entre des variables comme l'investissement, l'emploi, les échanges de biens/services et le spread ont

un réel pouvoir prédictif sur la récession.

On retombe sur les mêmes conclusions que les auteurs. L'inversion de la courbe des taux est un bon indicateur de récession, mais avec lui nous pouvons aussi utiliser des variables macro-économiques telles que l'investissement ou encore le chômage en temps réel. On remarquera aussi que nous avons estimé à partir de la méthode d'Estrella et Mishkin mais que pour prolonger leurs travaux, nous avons préféré utiliser le F1-Score pour sélectionner la combinaison de variables qui permettait la meilleure prédiction. Ce modèle n'est en aucun cas infaillible, loin de là, il faut prendre en compte que plus le temps passe plus ce modèle devra être actualisé, il reste cependant meilleur que celui des auteurs. Si l'on prend en *out sample* le modèle qu'ils ont proposé à savoir : le spread et un indicateur comme le *S&P500* avec un split en 1971, on obtient des prévisions bien qui sont plus mauvaises que celles que l'on a pu obtenir en prenant nos variables.



Pour finir cette section on a pu observer et retomber sur les résultats d'Estrella et Mishkin, mais quand bien même ces résultats sont satisfaisants nous pouvons faire beaucoup mieux en utilisant un prolongement à la méthode en implémentant le F1-score comme métrique à maximiser. Les résultats obtenus après la sélection de variable via celui-ci offre de biens meilleurs résultats même si, les deux modèles ont en commun une chose importante et qui vient prouver que les auteurs avaient bien raisons : le spread est présent dans les deux modèles. Cela prouve bien que la courbe des taux est un élément essentiel et indicateur robuste des récessions. Maintenant que nous avons suivi la méthode d'Estrella et Mishkin, peut-on faire autrement ? Peut-on faire mieux ?

## 4 Pour aller plus loin : les modèles de Machine Learning

Dans cette partie, nous allons nous intéresser aux modèles faisant recours au Machine Learning. Le but étant de pouvoir challenger les résultats obtenus dans la partie précédente : nous allons voir si on peut améliorer le modèle obtenu en in-sample et out-sample. On se demande ici, si les modèles de Machine Learning peuvent fonctionner afin d'obtenir de meilleures prédictions. On trouvera que pour la plupart ils ne fonctionnent pas car moins performant sur des séries temporelles.

Pour éviter toute confusion, nous allons définir la matrice de confusion comme suit :

		Predicted	
		Negative (N) -	Positive (P) +
Actual	Negative -	True Negative (TN)	False Positive (FP) Type I Error
	Positive +	False Negative (FN) Type II Error	True Positive (TP)

### 4.1 Le modèle avec le Random Forest

Dans cette partie, nous nous intéressons au Random Forest qui est un algorithme constitué de plusieurs arbres de décision indépendants. Chaque arbre est constitué d'un sous-échantillon aléatoire de la partie *train*, appelée *bootstrap*. L'idée est qu'à la fin de l'assemblage, la prédiction faite par le Random Forest dans le cadre d'une tâche de classification (et non pas une tâche de régression qui utilise la moyenne de tous les arbres), un vote majoritaire a lieu, c'est-à-dire que l'on prend en compte la variable qui est la plus fréquente dans chaque catégorie, et on la garde pour faire de la prédiction. Par exemple, si un échantillon est classé dans la classe A par 16 arbres, la classe B par une 10 arbres, et la classe C par 5 arbres, alors on retient qu'il s'agit de la classe A qui correspond à la prédiction de l'échantillon.

Dans notre cas, nous avons utilisé un `RandomForestClassifier` sur lequel nous avons entraîné les données `X_train[meilleure_combinaison_F1]` qui correspond à notre partie d'entraînement restreinte aux 10 variables que nous avons gardé par le  $R^2$  et après en avoir enlevé par rapport au sens économique. Ensuite, on fait un `.predict(X_test[meilleure_combinaison_F1])` qui nous permet de réaliser des prédictions et à partir desquels nous pouvons obtenir la matrice de confusion suivante :

```
array([[100,  2],  
       [ 3,  6]], dtype=int64)
```

Sur 111 récessions, 100 "pas de récessions" ont été prédites et sont réellement des "pas de récessions" (vrais négatifs), 6 récessions ont été prédites et sont réellement des récessions (vrais positifs), 2 récession a été prédite alors qu'il n'y en avait pas (faux positifs),



et 3 "pas de récessions" ont été prédites alors qu'il y en a eu (faux négatifs).

L'accuracy est 0.95, le recall de 0.55, le F1-score de 0.67 et la Balanced Accuracy de 0.78.

## 4.2 Le modèle avec l'AIC

L'Akaike Information Criterion, plus communément appelé l'AIC, est un critère d'information prenant en compte la vraisemblance d'un modèle et le nombre des variables explicatives  $X$ . Il est défini comme suit :

$$AIC = -2\log(L) + 2(p + 1)$$

avec :

- $\log(L)$  : la log-vraisemblance qui est à maximiser
- $p + 1$  : le nombre de paramètres dans le modèle avec la constante

En le minimisant, il permet de pénaliser le modèle de sorte à ce qu'on ait les meilleures prédictions pour le futur. L'AIC est un compromis entre l'ajustement des données (qualité du modèle) et la parcimonie du modèle, en réduisant le nombre de régresseurs.

Concernant notre démarche, nous avons utilisé notre dataframe *dft* qui correspond à une base de données nettoyée avec les transformations, conversions et sans valeurs manquantes. En partant de *dft*, nous voulons faire une sélection de variables. Pour cela, pour chaque colonne du dataframe *dft* (donc nos variables explicatives  $X$ ), nous avons ajouté une constante puis avons appliqué un modèle *probit* dessus. A chaque modèle, nous avons calculé l'AIC puis nous avons classé les variables du plus petit au plus grand AIC de sorte à récupérer les 10 meilleures variables au sens de l'AIC.

Ensuite avec ces variables sélectionnées qui sont : 'FPIx', 'T5YFFM lag 3', 'CPF3MTB3Mx lag 1', 'CMRMTSPLx', 'PRFIx lag 1', 'CPF3MTB3Mx', 'UMCSENTx', 'TB3SMFFM lag 1', 'USCONS', 'PRFIx', nous avons estimé le modèle par un *probit*, en utilisant `X_train_aic` auquel on a ajouté une constante, puis `sm.Probit(y_train, X_train_aic)` et le `.fit()` qui nous permet d'entraîner le modèle. Une fois cette étape réalisée, on utilise la fonction `.predict(X_test_aic)` qui nous permet de faire les prédictions sur la partie test (out-sample). Grâce à cela, on peut calculer la matrice de confusion avec `confusion_matrix(y_test, pred_aic.apply(lambda x: 1 if x>0.5 else 0))` qui nous permet de voir le nombre de vrais positifs, vrais négatifs, faux positifs, et faux négatifs trouvés avec le modèle utilisant le critère de l'AIC. Nous obtenons les résultats suivants :

```
array([[100,  2],
       [ 3,  6]], dtype=int64)
```

Sur 111 récessions, 100 "pas de récessions" ont été prédites et sont réellement des "pas de récessions" (vrais négatifs), 6 récessions ont été prédites et réellement des récessions (vrais positifs), 2 récessions ont été prédites alors qu'il n'y en avait pas (faux positifs), et 3 "pas de récessions" ont été prédites alors qu'il y en a eu (faux négatifs).

L'accuracy est de 0.95, le recall de 0.67, le F1-score de 0.71, et la Balanced Accuracy de 0.82.

### 4.3 Le modèle Ridge

Le Ridge est une méthode de Machine Learning permettant de limiter la variance des coefficients dans une régression et d'atténuer la corrélation élevée entre les variables explicatives (multicolinéarité). On voudrait minimiser la fonction de coût suivante, qui inclut la pénalité  $\ell_2$  :

$$\begin{aligned} RSS_{Ridge} &= \sum_{i=1}^n (y_i - \beta_0 - \sum_{j=1}^p \beta_j x_{ij})^2 + \lambda \sum_{j=1}^p \beta_j^2 \\ &= RSS + \lambda \sum_{j=1}^p \beta_j^2 \end{aligned} \quad (1)$$

Cette méthode permet de jouer sur les coefficients des variables explicatives et ne peut pas les mettre à 0, de sorte à toutes les garder dans le modèle.

Dans notre cas, nous avons utilisé la fonction `SelectFromModel(estimator=Ridge(), max_features=5).fit(X_train, y_train)` de la librairie `sklearn` qui nous permet de garder 5 variables explicatives sélectionnées par le Ridge qui sont 'CIVPART', 'UNRATEStx', 'HOUSTW lag 1', 'WPU0561 lag 3', 'COMPAPFF lag 3'. On entraîne ce modèle sur un *probit* afin d'en faire des prédictions et nous obtenons la matrice de confusion suivante :

```
array([[100,  2],
       [ 7,  2]], dtype=int64)
```

On voit qu'il a prédit 2 récessions qui étaient vraiment des récessions, 100 "pas de récessions" et il n'y avait vraiment pas de récessions, 2 récessions alors qu'il n'y en avait pas et 7 "pas de récessions" alors qu'il y en avait.

On obtient une accuracy de 0.92, un recall de 0.22, un F1-score de 0.31 et une Balanced Accuracy de 0.60.

On remarque donc le F1-score, qui mélange la précision et le recall, est assez mauvais. Cela est dû au fait qu'utiliser une régression Ridge n'est pas pertinent car il s'agit d'une méthode de régression, or nous sommes dans le cas d'une classification. S'ajoute à cela le fait que nous utilisons des données basées sur des séries temporelles, et le modèle Ridge n'est pas adapté pour cela, car ne prend pas en compte la dépendance temporelle. Ce modèle est donc à éviter pour faire de la prédiction sur des données temporelles.

### 4.4 Le modèle Logit avec les pénalités $\ell_1$ et $\ell_2$

Dans cette section, nous allons nous intéresser à la sélection de variables avec la régression logistique, mais en incluant la pénalité  $\ell_1$  (Lasso) ou  $\ell_2$  (Ridge). Nous avons défini ce qu'est la régression Ridge juste en haut. La régression Lasso est similaire à celle du Ridge à la différence qu'elle permet de mettre des coefficients nuls, donc d'enlever des variables explicatives, donc de réduire la dimension. Le Lasso vise donc à minimiser la fonction de

coût suivante :

$$\begin{aligned} RSS_{Lasso} &= \sum_{i=1}^n (y_i - \beta_0 - \sum_{j=1}^p \beta_j x_{ij})^2 + \lambda \sum_{j=1}^p |\beta_j| \\ &= RSS + \lambda \sum_{j=1}^p |\beta_j| \end{aligned} \quad (2)$$

#### 4.4.1 Pénalité $\ell_1$

D'une part, on veut sélectionner les variables à l'aide d'une régression logistique dans laquelle on applique la pénalité  $\ell_1$  :

```
1 logistic_regression = LogisticRegression(penalty='l1', solver='  
    liblinear')  
2 logistic_regression.fit(X_train, y_train)  
3 selector_logit_cv = SelectFromModel(logistic_regression)
```

Puis on utilise la fonction `SelectFromModel` qui va nous sélectionner les variables permettant de prédire s'il y a une récession ou non. Ensuite, après avoir récupéré les variables explicatives qui sont au nombre de 22, on les réinjecte dans une régression logistique dans laquelle nous avons ajouté une constante à `X_train` et `X_test`, puis nous avons estimé ce modèle ce qui nous donne la matrice de confusion suivante :

```
array([[102,  0],  
       [ 9,  0]], dtype=int64)
```

On obtient alors une accuracy de 0.92, un recall de 0, un F1-score de 0, et une Balanced Accuracy de 0.5. Les résultats ne sont donc pas très favorables, puisque le F1-score est nul. Aussi, le modèle n'a réussi à prédire aucune récession. Cette méthode est clairement à proscrire au vu des résultats

#### 4.4.2 Pénalité $\ell_2$

D'autre part, on veut désormais appliquer la pénalité  $\ell_2$  sur une régression logistique afin de procéder à une sélection de variables :

```
1 logistic_regression_l2 = LogisticRegression(penalty='l2', solver='  
    liblinear')  
2 logistic_regression_l2.fit(X_train, y_train)  
3 selector_logit_cv_l2 = SelectFromModel(logistic_regression_l2)
```

La matrice de confusion pour ce modèle est :

```
array([[92, 10],  
       [ 3,  6]], dtype=int64)
```

L'accuracy est de 0.88, le recall est de 0.66, le F1-score est de 0.48 et la Balanced Accuracy est de 0.78.

## 4.5 Le modèle Naive Bayes

Nous allons porter notre attention plus particulièrement sur cette partie, qui est celle sur le Naive Bayes. En effet, parmi tous les modèles que nous avons évoqué, il s'agit de celui qui s'adapte le mieux à la classification prenant en compte les séries temporelles.

Le Naive Bayes, aussi appelé classification naïve bayésienne, est une méthode appartenant à la classification supervisée basée sur le théorème de Bayes qui fait intervenir la notion de probabilité conditionnelle :

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}$$

où :

- A et B sont 2 évènements
- $P(A|B)$  est la probabilité que A s'est réalisé sachant B
- $P(B|A)$  est la probabilité que B s'est réalisé sachant A
- $P(B)$  est différent de 0

Ce classifieur suppose que les caractéristiques entre deux classes sont indépendantes, ce qui constitue une hypothèse forte et importante. Par exemple, une orange ayant pour caractéristiques "couleur orange", "ronde", "sucré", nous permettent de dire qu'il s'agit d'une orange, mais qu'il faudrait étudier leurs caractéristiques de façon totalement indépendantes (les régresseurs), d'où le terme "naïve".

Un atout non négligeable de cette méthode de classification est que grâce à son hypothèse forte d'indépendance entre les classes, nous n'avons pas besoin d'avoir un grand jeu de données pour estimer les paramètres permettant une classification.

Pour pouvoir classer notre variable explicative Y, l'algorithme regarde la probabilité de chaque possibilité de Y sachant les variables explicatives connues en utilisant le théorème de Bayes, puis retient la probabilité la plus élevée. Reprenons l'exemple des fruits, on sait que le fruit a les caractéristiques "rond", "jaune", "acide" et on veut savoir si c'est une pomme, une banane, ou un citron. Alors on calcule les probabilités conditionnelles suivantes :  $P(\text{Citron} | \text{Rond, Jaune, Acide}) = 0.85$ ,  $P(\text{Pomme} | \text{Rond, Jaune, Acide}) = 0.2$ ,  $P(\text{Banane} | \text{Rond, Jaune, Acide}) = 0.4$ . On voit que la probabilité 0.85 est la plus élevée, donc selon les caractéristiques, on peut dire qu'il s'agit d'un citron.

Dans notre cas, on pourra dire que notre Y vaut 1 s'il y a une récession, 0 s'il n'y en a pas, et les caractéristiques correspondent à nos variables explicatives X.

Il existe différentes déclinaisons du Naive Bayes avec différents modèles de probabilités conditionnelles : le modèle de Bernoulli (mieux adapté pour les régresseurs binaires) et le modèle de distribution gaussienne (mieux pour les régresseurs continus). D'autres modèles existent aussi comme par exemple le modèle catégoriel, le modèle multinomial et le modèle complémentaire mais ne s'utilisent pas dans notre cas en raison de la présence des valeurs négatives. Nous allons donc utiliser deux déclinaisons et voir laquelle sera la meilleure pour notre use-case.

Pour tester plusieurs modèles, nous allons subdiviser chacune des déclinaisons en deux afin de voir quelle méthode sera la plus robuste : une partie où on donne à l'algorithme l'ensemble de nos variables, et une autre partie où nous utilisons nos variables sélectionnées au préalable.

#### 4.5.1 Naive Bayes Gaussien

Avec toutes les variables explicatives :

Dans cette partie, on va utiliser le Naive Bayes Gaussien auquel on ne lui enlève aucune variable. On fait donc :

```
1 model_bayes = GaussianNB()
2 model_bayes.fit(X_train, y_train)
3 y_pred_bayes = model_bayes.predict(X_test)
```

Après entraînement du modèle et la prédiction, nous obtenons la matrice de confusion suivante :

```
array([[102,  0],
       [  9,  0]], dtype=int64)
```

Il n'a réussi à prédire aucune récession et qui s'est réellement produite, et a prédit qu'il n'y avait jamais de récession, qu'il y en ait eu après ou non.

On a une accuracy de 0.92, un recall de 0, un F1-score de 0 et une Balanced Accuracy de 0.5. Le Naive Bayes Gaussien sans réduction de dimension est donc assez mauvais puisque son recall et son F1-score sont nuls. Il est donc à éviter.

Avec sélection de variables :

En utilisant le Naive Bayes Gaussien appliqué à notre sélection de variables 'FPIx', 'CMRMTSPLx', 'GS10TB3Mx lag 4', 'IPMANSICS' :

```
1 model_bayes = GaussianNB()
2 model_bayes.fit(X_train[meilleure_combinaison_F1], y_train)
3 y_pred_bayes = model_bayes.predict(X_test[meilleure_combinaison_F1])
```

On obtient la matrice de confusion :

```
array([[98,  4],
       [ 0,  9]], dtype=int64)
```

Il prédit 9 vraies récessions (vrais positifs), 98 "pas de récessions" et qu'il n'y en a vraiment pas eu, et 4 faux positifs, ce qui donne une accuracy de 0.96, un recall de 1, un F1-score de 0.81 et une Balanced Accuracy de 0.98.

On voit donc que si on utilise un modèle gaussien pour notre Naive Bayes, il vaut mieux avoir réduit les dimensions donc sélectionné les variables au préalable afin d'obtenir un meilleur F1-score qui prend en compte la précision et le recall. Nous allons voir si utiliser le modèle de Bernoulli peut améliorer les résultats.

### 4.5.2 Naive Bayes de Bernoulli

Avec toutes les variables explicatives :

Pour utiliser le Naive Bayes avec le modèle de Bernoulli sans sélection de variables, on écrit :

```
1 model_bayes = BernoulliNB()
2 model_bayes.fit(X_train, y_train)
3 y_pred_bayes = model_bayes.predict(X_test)
```

Et on obtient la matrice de confusion suivante :

```
array([[97,  5],
       [ 0,  9]], dtype=int64)
```

On voit qu'il n'a pas prédit de faux négatifs, mais 5 faux positifs, et a trouvé 97 vrais négatifs et 9 vrais récessions. En termes de métriques on a alors : une accuracy de 0.95, un recall de 1, un F1-score de 0.78 et une Balanced Accuracy de 0.97. On remarque donc ces résultats sont très similaires à ceux du Naive Bayes Gaussien avec sélection de variables.

Avec sélection de variables :

On a :

```
1 model_bayes = BernoulliNB()
2 model_bayes.fit(X_train[meilleure_combinaison_F1], y_train)
3 y_pred_bayes = model_bayes.predict(X_test[meilleure_combinaison_F1])
```

Qui nous donne la matrice de confusion suivante :

```
array([[97,  5],
       [ 0,  9]], dtype=int64)
```

On remarque qu'il s'agit exactement des mêmes résultats que si on considérait tout le modèle en entier. On peut donc déduire qu'il n'y a pas d'impact sur le fait de limiter le nombre de variables lorsqu'on utilise le modèle de Bernoulli. Cela peut s'expliquer par le fait que nos variables explicatives sont continues donc qu'il n'y a pas de réel impact sur l'utilisation de cette déclinaison.

## 4.6 Conclusion sur le Machine Learning

Pour rappel, voici les valeurs des métriques que l'on obtient avec notre modèle pour prévoir une récession, à savoir avec les variables suivantes : la constante, 'FPIx', 'CMRMTS-PLx', 'GS10TB3Mx lag 4', 'IPMANSICS' :

- Accuracy : 0.98
- Recall : 1
- F1-score : 0.9
- Balanced Accuracy : 0.99

Voici un tableau récapitulatif des modèles de Machine Learning que nous avons utilisé :

Modèle utilisé	Accuracy	Recall	F1-score	Balanced Accuracy
Random Forest	0.95	0.55	0.67	0.78
AIC	0.95	0.67	0.71	0.82
Ridge	0.92	0.22	0.31	0.60
Logit + $\ell_1$	0.92	0	0	0.5
Logit + $\ell_2$	0.88	0.66	0.48	0.78
Naive Bayes Gaussien total	0.92	0	0	0.5
Naive Bayes Gaussien restreint	0.96	1	0.81	0.98
Naive Bayes Bernoulli total	0.95	1	0.78	0.97
Naive Bayes Bernoulli restreint	0.95	1	0.78	0.97

On voit donc l'accuracy est particulièrement bon pour tous les modèles, mais n'est pas une métrique pertinente pour sélectionner le meilleur modèle. L'accuracy n'est pas une métrique adaptée à ce genre de modélisation dans le sens où l'on regarde la probabilité de se tromper de diagnostic. Ce n'est pas tant le recall qui nous effraie mais plus la précision. Il est sans doute préférable de se tromper et ne pas voir la crise arriver plutôt que de la prédire alors qu'elle n'arrivera pas (effet de panique). Le recall est une métrique qui donne de l'importance à la détection de faux négatifs, on ne voit pas la récession alors qu'elle arrive. Les marchés s'affolent et n'ont pas vu la crise, en soi c'est aussi un problème, mais il peut être "moins important" puisque pour le coup la crise a vraiment lieu comparé aux faux positifs qui peuvent perturber le marché pour un faux diagnostic. Il serait donc intéressant de pouvoir compter sur le recall et la précision.

On peut donc se référer au F1-score qui prend en compte la précision et le recall en même temps (voir précédemment). Si on devait classer les modèles dans l'ordre décroissant vis-à-vis du F1-score, on peut retenir qu'il s'agit du modèle Naive Bayes Gaussien avec la sélection de variables imposées qui s'en sort le mieux. Avec un score de 0.81, c'est le modèle que l'on pourrait considérer comme le meilleur en utilisant le Machine Learning, et que l'on peut comparer face aux résultats de notre modèle que nous avons développé petit à petit dans notre démarche. De plus, sa Balanced Accuracy (0.98) est quasiment identique à notre Balanced Accuracy (0.99), ce qui confirme bien qu'il s'agit d'un candidat pertinent à notre modèle.

On peut alors dire : si on a sélectionné nos variables au préalable, et que l'on souhaite prévoir une crise économique, au lieu de faire une estimation et une prévision avec un modèle *logit*, on peut utiliser un modèle de classification supervisée qui prend en compte l'indépendance des variables explicatives continues et le fait qu'elles se sont réalisées dans le passé (probabilité conditionnelle) appelée le Naive Bayes Gaussien.

## 5 Conclusion

Pour conclure ce projet, on a dans un premier temps vérifié la véracité du modèle proposé par Estrella et Mishkin en observant si l'inversion de la courbe des taux avait bel et bien un pouvoir prédictif sur les récession. En *in sample* comme en *out sample* on a pu observer les mêmes conclusions, le spread est bel et bien un indicateur robuste. Nous avons alors actualisé son étude et son modèle afin de vérifier si ce modèle était toujours d'actualité. A l'aide du F1-Score nous avons effectué de la sélection de variable parmi les variables que les auteurs ont jugé importante. Et cela nous a permis d'obtenir des résultats qui surpassaient la méthode précédente. Nous avons donc aboutit à un modèle qui contenait encore une fois le spread, mais aussi des indicateurs macro-économiques comme le chômage, l'investissement, ou la consommation (les échanges de biens et services). Enfin dans un dernier temps nous avons essayé de challenger cette méthode en appliquants diverses méthodes telles que la sélection de variable par le critère de l'AIC ou des méthodes de Machine Learning. Il s'est avéré que ses méthodes ont des résultats qui varient grandement. Certaines méthodes ne sont pas adaptées aux séries temporelles et résultent en de faibles pouvoirs prédictifs, mais des modèles font bien mieux qu'Estrella et Mishkin dans leur modèle naïf comme le Naive Bayes qui performe bien dans notre cas. Cependant si l'on utilise les variables proposées par Mishkin couplées au prolongement que nous y avons apporté il est difficile de faire mieux que les auteurs. Le modèle est donc encore aujourd'hui compétitif et d'actualité.