Prédiction des récessions aux États-Unis : variables financières en tant qu'indicateurs avancés

Econométrie appliquée des séries temporelles II

UNIVERSITÉ PARIS 1
PANTHÉON SORBONNE

Elodie Hutin
Hugo Lemonnier
Camille Loegel-Orts

Année universitaire 2022 - 2023

SOMMAIRE

INTRODUCTION

La prévision précise des points d'inflexion du cycle économique, en particulier des récessions économiques imminentes, est cruciale pour les ménages, les entreprises, les investisseurs et les décideurs politiques. Les recherches antérieures ont montré que diverses relations économiques et financières contiennent des informations prédictives sur les futures récessions aux États-Unis. Parmi celles-ci l'inversion de la courbe des taux, mise en évidence par les travaux d'Estrella et Mishkin (1998)[1], s'avère être un indicateur prédictif solide.

Dans cette étude, nous réexaminons la prévisibilité des récessions américaines en utilisant un large éventail de variables d'indicateurs avancés, qui ont été considérées comme des facteurs de risque pour la stabilité des prix. Nous avons sélectionné les variables d'indicateurs avancés qui ont été étudiées dans la littérature académique et pratique, et nous utilisons les informations fournies par le National Bureau of Economic Research (NBER) comme série de référence des points d'inflexion du cycle économique.

Cependant, alors que l'indicateur de récession du NBER est une variable binaire, la plupart des indicateurs avancés ont des distributions continues. Par conséquent, pour mettre en correspondance les changements des variables prédictives avec les prévisions de récession, une grande partie de la littérature empirique a utilisé le modèle probit non linéaire, que nous suivons également dans cette étude.

Tout d'abord, nous avons effectué une revue de la littérature afin de comprendre les notions clés. Cette première partie nous a permis de mieux appréhender les différentes variables économiques et financières qui ont été considérées comme des indicateurs avancés de récession, ainsi que les méthodes empiriques utilisées pour évaluer leur prédictibilité. Ensuite, dans la deuxième partie, nous reproduirons l'étude et l'analyse d'Estrella A. et Mishkin F. S. Enfin, dans la troisième partie, nous avons formulé des suggestions d'amélioration pour les futures études sur la prévision des récessions économiques.

[1]Estrella, A., and F. S. Mishkin (1998): "Predicting U.S. recessions: Financial variables as leading indicators," The Review of Economics and Statistics, 80(1), 45–61.

1 Revue de littérature

1.1 Base de données FRED-QD

La qualité des données utilisées peut avoir une incidence significative sur les résultats de l'étude. Pour cette raison, il est essentiel de bien comprendre les sources, la nature et la qualité des données avant de commencer une analyse économique.

D'après le papier[2], McCracken et Ng ont développé deux bases de données macroéconomiques pour la recherche universitaire en macroéconomie. La première base de données, FRED-QD, est une grande base de données macroéconomiques trimestrielles, mise à jour en temps réel à l'aide de la base de données FRED. Elle contient 248 séries trimestrielles classées en 14 groupes, tels que NIPA, Production industrielle, Emploi et chômage, Logement, Inventaires, commandes et ventes, Prix, Revenus et productivité, Taux d'intérêt, Monnaie et crédit, Bilans des ménages, Taux de change, Autres marchés boursiers et Bilans des non-ménages. Cette base de données est conçue pour émuler la base de données utilisée dans l'article de Stock et Watson (2012a)[3], tout en incluant plusieurs séries supplémentaires.

La deuxième base de données, FRED-MD, est une base de données de variables macroéconomiques mensuelles qui contient 128 séries standard de macroéconomie américaine. Elle est également mise à jour en temps réel à l'aide de la base de données FRED. Le but de FRED-MD est de fournir une source de données macroéconomiques volumineuses disponible publiquement pour la recherche universitaire. Finalement, les deux bases de données ont été développées par McCracken et Ng pour faciliter l'accès à un environnement riche en données standardisées qui peut être utilisé pour la recherche académique en macroéconomie.

Les données fournies ne sont pas toutes stables en niveaux, et doivent donc être transformées en utilisant des techniques telles que les logarithmes ou les différences pour devenir stationnaires. Selon le document de référence de l'étude, il est important de prendre en compte ce point afin de garantir la validité des résultats et leur interprétation correcte.

[2]McCracken, M.W., Ng, S., 2020; FRED-QD: A Quarterly Database for Macroeconomic Research, Federal Reserve Bank of St. Louis Working Paper 2020-005

[3]Stock, J.H. and Watson, M. "Disentangling the Channels of the 2007-2009 Recession."Brookings Papers on Economic Activity, 2012a, Spring, pp. 81-156.

En effet, les auteurs ont évalué l'utilité des facteurs extraits de la base de données FRED-QD pour la prévision des agrégats macroéconomiques en se concentrant sur la question de savoir si les codes de transformation impliquant des racines unitaires ont un impact sur la précision des prévisions. Ils ont examiné les performances des modèles basés sur les facteurs pour prédire une gamme de séries macroéconomiques, en se concentrant sur les séries réelles, financières et nominales. Ils ont constaté que, pour les séries réelles et financières, les facteurs estimés à l'aide des codes de transformation basés sur les racines unitaires peuvent fournir un contenu prédictif supplémentaire, mais sont souvent dominés par ceux utilisant les codes de transformation originaux. En revanche, ils ont constaté que, pour les séries de prix nominaux, l'exactitude des prévisions est généralement meilleure lorsqu'on utilise des facteurs estimés à l'aide des codes de transformation basés sur les racines unitaires.

1.2 Estrella, A., and F. S. Mishkin (1998)

Le but principal de cet article consiste à déterminer si des variables financières simples peuvent être des indicateurs utiles pour prédire les futures récessions. Les variables examinées comprennent les taux d'intérêt, les écarts de taux d'intérêt, les indices boursiers, et les agrégats monétaires, qu'ils soient nominaux ou réels. Pour déterminer leur utilité, ces résultats sont comparés avec des modèles basés sur des indicateurs macroéconomiques traditionnels tels que l'indice des indicateurs économiques avancés du Département du commerce et plusieurs de ses séries composantes.

Les indicateurs macroéconomiques sont généralement connus pour leur capacité à prévoir l'activité économique réelle, mais leur performance n'est pas toujours soumise à des tests comparatifs. De plus, les retards dans la disponibilité des données pour certaines variables peuvent poser un problème. Afin de résoudre ce problème, seules les observations effectivement disponibles à la fin d'un trimestre donné sont utilisées pour garantir une comparabilité entre toutes les séries. La variable de récession est construite à partir des dates du NBER (comme expliqué plus haut).

L'objectif final est d'examiner différentes variables qui pourraient avoir un pouvoir prédictif potentiel pour les récessions, en considérant des horizons prédictifs allant de 1 à 8 trimestres à l'avance. Étant donné que le volume de sortie généré par cette analyse est important, il est crucial de pouvoir résumer les résultats de manière significative. Pour ce faire, l'auteur introduit plusieurs mesures sommaires du pouvoir prédictif d'une variable à un horizon donné.

La mesure principale est le pseudo R^2 développé par Estrella (1995), qui est une mesure sim-

ple de la qualité de l'ajustement, correspondant intuitivement au coefficient de détermination largement utilisé, ou R^2, dans une régression linéaire standard. Cette mesure prend des valeurs comprises entre 0 et 1 pour les résultats en échantillon, où une valeur proche de 0 indique que la ou les variables du modèle ont peu de pouvoir explicatif, une valeur proche de 1 indique un bon ajustement, et des valeurs intermédiaires peuvent être utilisées pour classer les modèles en termes de pouvoir prédictif.

$$pseudoR^2 = 1 - \left(\frac{\log L_u}{\log L_c}\right)^{-\frac{2}{n}logL_c} \tag{1}$$

Où L_u est la vraisemblance du modèle estimé et L_c est la valeur d'un modèle contenant que la constante α_0. Pour les résultats hors échantillon, il n'y a aucune garantie que la valeur du pseudo R^2 se situera entre 0 et 1, mais cette mesure reste utile comme mesure simple de l'ajustement et est rapportée dans la section appropriée.

Comme dans le cas de la régression linéaire, le R^2 est associé à un test statistique valide, mais il n'est pas suffisant à lui seul pour tester des hypothèses statistiques. Par conséquent, l'auteur présente deux mesures supplémentaires qui sont associées à des tests statistiques valides. L'une est analogue à la t-statistique pour une variable individuelle dans une régression linéaire et peut être interprétée de manière similaire. En général, une valeur absolue de 2 ou plus tend à indiquer une signification statistique. Enfin, l'auteur indique également si un test statistique formel indique qu'une variable est significative aux niveaux de 5 % et de 1 %.

Dans le but d'évaluer la capacité prédictive des variables étudiées par rapport aux futures récessions, nous appliquons une méthode de régression statistique. Le choix du modèle spécifique, à savoir l'équation probit, est motivé par le fait que la variable prédite ne peut prendre que deux valeurs possibles. Plus précisément, la variable dépendante du modèle est représentée par la variable R, qui prend la valeur de 1 lorsque l'économie est en récession au trimestre t, et 0 dans le cas contraire. Ainsi, l'équation du probit a cette forme :

$$P(R_{t+k} = 1) = F(\alpha_0 + \alpha_1 X_{1t} + \alpha_2 X_{2t}..) \tag{2}$$

Les coefficients alpha sont estimés statistiquement et F représente la fonction de distribution cumulative normale. En utilisant une somme pondérée de une ou plusieurs variables explicatives (X) observées à la fin du trimestre t, nous pouvons prédire si R sera égal à 1 ou 0 k trimestres plus tard. L'application de la fonction F à cette somme pondérée permet de convertir le résultat en une probabilité qu'une récession se produise au trimestre 1+k. Une probabilité proche de 1 indique une forte prédiction d'une récession, tandis qu'une probabilité proche de 0 indique le contraire.

Avant de suggérer des améliorations, il est nécessaire de mettre à l'épreuve la robustesse et la pertinence du modèle proposé par les auteurs une fois que nous avons acquis une compréhension approfondie de sa théorie et de ses aspects économiques.

2. Application Estrella, A., and F. S. Mishkin

Dans cet article, rappelons que les auteurs se penchent sur l'efficacité de différentes variables financières pour prédire si l'économie américaine entrera en récession dans les 1 à 8 trimestres à venir. Cependant, l'application de transformations aux variables est nécessaire car les données financières peuvent présenter des caractéristiques spécifiques qui peuvent biaiser les résultats si elles ne sont pas prises en compte. Voici les différentes transformations appliquées en fonction de la valeur de la deuxième ligne pour une variable donnée :

- Si le code de transformation est égale à 1, aucune transformation n'est appliquée à cette variable. Elle reste inchangée.

- Si la valeur est égale à 2, la différence première est calculée pour cette variable entre chaque valeur et sa valeur précédente est calculée pour cette variable. Cela permet de calculer la variation absolue entre les périodes.

- Si la valeur est égale à 3, la différence seconde est alors calculée entre chaque valeur et sa valeur précédente est calculée en sautant une période supplémentaire. Cela permet de calculer la variation absolue sur une période plus longue.

- Si la valeur est égale à 4, le logarithme naturel de chaque valeur est pris pour cette variable. Cela permet de transformer les données en une échelle logarithmique.

- Si la valeur est égale à 5, le logarithme naturel de chaque valeur est pris, suivi du calcul de la différence entre chaque valeur et sa valeur précédente. Cela permet de calculer la variation relative sur une échelle logarithmique.

- Si la valeur est égale à 6, le logarithme naturel de chaque valeur est pris, suivi du calcul de la différence entre chaque valeur et sa valeur précédente, en sautant une période supplémentaire. Cela permet de calculer la variation relative sur une échelle logarithmique pour une période plus longue.

- Si la valeur est égale à 7, la variation en pourcentage entre chaque valeur et sa valeur précédente est calculée, suivi du calcul de la différence entre chaque valeur et sa valeur précédente. Cela permet de calculer la variation relative en pourcentage.

Cela donne finalement :

```
def transfo(dataset):

    mask=dataset.index>=2

    for i in dataset.columns:
        if dataset[i][1]==1:
            dataset[i]=dataset[i]

        elif dataset[i][1]==2:
            dataset[i]=dataset[i].diff()

        elif dataset[i][1]==3:
            dataset[i]=dataset[i][mask].diff(periods=2)

        elif dataset[i][1]==4:
            dataset[i]=np.log(dataset[i][mask])

        elif dataset[i][1]==5:
            dataset[i]=np.log(dataset[i][mask]).diff()

        elif dataset[i][1]==6:
            dataset[i]=np.log(dataset[i][mask]).diff(periods=2)

        elif dataset[i][1]==7:
            dataset[i]=dataset[i][mask].pct_change().diff()

    return(dataset)
```

Une considération importante abordée dans cet article concerne l'utilisation de retards sur les variables dans le cadre de la prévision des récessions. Cette approche présente plusieurs motivations et avantages significatifs. Tout d'abord, il est observé que les récessions économiques sont souvent caractérisées par des réactions retardées des variables économiques. Ainsi, en appliquant des retards sur ces variables, il devient possible de capturer ces relations temporelles et de prendre en compte l'inertie économique associée. De cette manière, le modèle est en mesure de saisir les effets retardés des conditions économiques sur la probabilité d'occurrence d'une récession.

En outre, l'utilisation de retards présente un avantage pratique dans les cas où les données économiques nécessaires à la prévision des récessions ne sont pas immédiatement disponibles. Par exemple, les données relatives au PIB peuvent être soumises à un certain délai avant leur publication officielle. En appliquant des retards sur ces variables, on peut utiliser les données disponibles avec un certain décalage pour effectuer des prévisions en temps réel, ce qui permet de réduire les contraintes liées à la disponibilité des données.

Une autre justification importante concerne la corrélation retardée entre certaines variables économiques, comme le spread, et les récessions. Par exemple, l'écart de la courbe des taux peut refléter les anticipations des investisseurs concernant l'évolution future de l'économie. En utilisant des retards, il devient possible de capturer cette relation retardée entre ces variables et les récessions, ce qui peut renforcer la capacité prédictive du modèle.

En résumé, l'utilisation de retards sur des variables telles que le spread ou le PIB dans un modèle Probit pour la prévision des récessions offre l'avantage de tenir compte des relations temporelles et d'exploiter les informations retardées pour améliorer la qualité des prévisions. Cette approche permet d'obtenir une meilleure compréhension des dynamiques économiques et d'améliorer la capacité du modèle à identifier les périodes de récession de manière plus précise et fiable.

Une fois que nous avons compris ces deux notions importantes, nous pouvons sélectionner les variables les plus pertinentes, en particulier celles qui sont liées à celles choisies dans le document référencé.

Les taux d'intérêt sont des mesures clés dans l'économie qui influencent les décisions financières. Certains taux d'intérêt importants sont suivis de près, tels que le taux sur les bons du Trésor à court terme (TB3MS ou BILL) et à long terme (GS10 ou BOND), le taux des fonds fédéraux (FEDFUNDS) et le taux hypothécaire fixe à 30 ans (MORTGAGE30US).

Les écarts de taux d'intérêt fournissent des informations sur les différences entre différents instruments financiers. Par exemple, l'écart entre les bons du Trésor à long terme et à court terme (GS10TB3Mx ou SPREAD) mesure la différence entre les bons du Trésor à 10 ans et à 3 mois. Il existe également des écarts entre les taux d'intérêt des obligations d'État et des obligations d'entreprise.

Les indices boursiers, tels que le S&P 500 et le NASDAQ Composite, sont utilisés pour évaluer la performance globale du marché boursier.

Les agrégats monétaires, tels que BOGMBASEREALx, M1REAL et M2REAL, permettent d'analyser la quantité de monnaie en circulation, en prenant en compte des facteurs tels que l'inflation.

Enfin, certains indicateurs macroéconomiques individuels sont également importants, tels que les permis de construire (PERMIT) ou la croissance du PIB réel (GDPC1).

En ce qui concerne la gestion des valeurs manquantes, nous commençons par examiner le nombre de valeurs manquantes par variable à l'aide de la commande suivante.

```
manquant = df_estrella2.isna().sum()
for i in range(0,len(manquant)) :
    if manquant[i] >= 1 :
        print(manquant.index[i],manquant[i])
```

Nous identifions ainsi que les variables NASDAQCOM et MORTGAGE30US présentent des valeurs manquantes. Avant de décider de les supprimer ou de les imputer, nous vérifions si elles sont corrélées à d'autres variables du dataframe. Pour ce faire, le code calcule la matrice de corrélation entre nos variables. Ensuite, il identifie les paires de variables ayant une corrélation supérieure à 0.75 (en valeur absolue) et les affiche dans un DataFrame. Les variables du DataFrame sont triées en fonction de leur corrélation, avec les valeurs les plus élevées en tête. Les corrélations entre une variable et elle-même sont exclues.

```
corr_estrella=df_estrella2.corr()
print(corr_estrella)
high_corr=corr_estrella[(corr_estrella.abs()>0.75)].stack().reset_index()
high_corr.columns=['Variable 1', 'Variable 2', 'Corrélation']
high_corr=high_corr.sort_values(by=['Corrélation'],ascending=False)

#Afficher les variables avec des corrélations supérieures à 0.8 (en valeur absolue)
high_corr=high_corr.drop(high_corr[high_corr['Variable 1']==high_corr['Variable 2']].index)
print(high_corr)
```

Nous constatons que MORTGAGE30US et NASDAQCOM sont fortement corrélées à une variable respective. En raison de cette corrélation et des valeurs manquantes, nous décidons de supprimer ces deux variables de notre dataframe. De plus, les variables TB3MS et FEDFUNDS présentent également une corrélation élevée. Par conséquent, nous décidons de supprimer la variable FEDFUNDS.

	Variable 1	Variable 2	Corrélation
10	TB3MS	FEDFUNDS	0.907880
14	FEDFUNDS	TB3MS	0.907880
7	S&P 500	NASDAQCOM	0.869246
6	NASDAQCOM	S&P 500	0.869246
12	GS10	MORTGAGE30US	0.767797
16	MORTGAGE30US	GS10	0.767797

2.1 Le cas in sample

2.1.1 Variable & Constante : $P(Y_{t+k}) = \alpha_0 + \alpha_1 X_t$

Les résultats en échantillon sont basés sur des équations estimées sur toute la période d'échantillonnage. Leurs prévisions ou valeurs ajustées sont ensuite comparées aux dates réelles de récession. Étant donné que l'article porte sur la prédiction hors échantillon, seuls quelques résultats sélectionnés dans l'échantillon sont présentés dans cette section.

Une fois que les transformations, la gestion des valeurs manquantes et la compréhension des retards sur nos variables, la partie théorique de l'article est mise en pratique à l'aide de la fonction R2_insample_1. Cette fonction calcule les pseudo R^2 et les t statistiques pour chaque variable dans un modèle Probit avec l'ajout de retards.

La fonction "R2_insample_1" prend deux DataFrames en entrée, "df_1" et "df_2". Elle effectue une boucle sur les colonnes de "df_1" et pour chaque colonne, elle calcule le pseudo R^2 et la t statistique correspondants. Pour chaque colonne de "df_1", la méthode effectue une boucle sur une plage de retard allant de 1 à 8.

Ensuite, la méthode utilise la régression Probit pour estimer le modèle avec la variable et le modèle avec la constante uniquement. Elle calcule la log-vraisemblance pour chaque modèle. En utilisant ces valeurs de log-vraisemblance, la méthode calcule le pseudo R^2 pour chaque retard en utilisant la formule correspondante. Le pseudo R^2 est arrondi à trois décimales et stocké dans une liste. La t statistique est également calculée en utilisant le modèle avec la variable, et elle est arrondie à trois décimales. Cette t statistique est stockée dans une autre liste.

La méthode effectue également un test de significativité en comparant la valeur du p associée à la t statistique avec des seuils prédéfinis (0.01 et 0.05). Si la valeur de p est inférieure à 0.01, la t statistique est marquée avec "**", si elle est inférieure à 0.05 mais supérieure ou égale à 0.01, elle est marquée avec "*", sinon elle est laissée inchangée.

En résumé, la fonction R2_insample_1 permet de calculer les pseudo R^2 et les t statistiques pour chaque variable dans un modèle Probit avec l'ajout de retards. Cela permet d'évaluer la performance prédictive de chaque variable en tenant compte de la temporalité et des effets retardés dans les données économiques.

```
def R2_insample_1 (df_1, df_2):
    R2=[] #liste de liste contenant tous les R2
    tstat=[] #liste de liste contenant toutes les t-stat
    for j in df_1.columns:
        liste_R2=[] #liste de l'ensemble des R2 d'une variable
        liste_stat=[] #liste de l'ensemble des t-stat d'une variable
        for retard in range (1,9): #applique les 8 retards
            X=df_1[j].iloc[:-retard,].reset_index(drop=True)
            X=sm.add_constant(X)
            y=df_2.iloc[retard:,].reset_index(drop=True)
        #Variable
            model=sm.Probit(y,X).fit(disp=0)
            log_var=model.llf #log-vraisemblance du modèle avec variable
            log_const=model.llnull #log-vraisemblance du modèle avec constante
        #R²
            pseudo_R2=round(1-(log_var/log_const)**((-2/len(X))*log_const),3)
            liste_R2.append(pseudo_R2)
        #T-stat
            t_stat=model.tvalues[1]
            t_stat=round(t_stat,3)
            liste_stat.append(t_stat)
        #p value
            p_value=model.pvalues[1]
            liste_stat[retard-1]= str(t_stat)+'**' if p_value < 0.01 else str(t_stat)+'*' if p_value < 0.05 else t_stat
        liste_R2.insert(0,j)
        liste_R2.insert(1,'Pseudo R2')
        liste_stat.insert(0,'')
        liste_stat.insert(1,'t stat')
        R2.append(liste_R2)
        tstat.append(liste_stat)
        data=[x for i in zip(R2, tstat) for x in i] #pour afficher en alternance les R2 et t-stat de chaque var
    df=pd.DataFrame(data,columns=["Variable","R2","1","2","3","4","5","6","7","8"]) #création d'un df avec résultat
    return(df)
```

Plus précisement, la fonction "R2_insample_1" utilise la régression Probit pour estimer des modèles avec une variable explicative et une constante afin de calculer le pseudo R^2 et la t statistique associés. Pour l'estimation du probit :

Pour chaque colonne j dans df_1 et pour chaque retard de 1 à 8 : on définit la variable X comme une série temporelle de $df_1[j]$ sans les $retard$ dernières valeurs. Cela correspond à décaler la série temporelle vers le passé. On estime le modèle Probit avec y comme variable dépendante et X comme variable indépendante, en utilisant la fonction $sm.Probit(y, X).fit(disp = 0)$. On estime la log-vraisemblance du modèle avec la variable, log_var. On estime également la log-vraisemblance du modèle avec la constante en utilisant $llnull$. On enregistre cette même vraisemblance dans log_const.

Calcul du pseudo R^2 : pour chaque retard de 1 à 8 : le pseudo R^2 est calculé en utilisant la formule :

$$1 - \left(\frac{\log L_{var}}{\log L_{cons}}\right)^{-\frac{2}{len(X)}\log L_{cons}} \tag{3}$$

où log_var est la log-vraisemblance du modèle avec la variable et log_const est la log-vraisemblance du modèle avec la constante uniquement. Le pseudo R^2 est arrondi à trois décimales et enregistré dans une liste, $liste\_R2$.

Calcul de la t statistique et du marquage de significativité : pour chaque retard de 1 à 8 : la t statistique est extraite du modèle Probit avec la variable en utilisant $model.tvalues[1]$. La t statistique est arrondie à trois décimales. La valeur de p associée à la statistique t est extraite en utilisant $model.pvalues[1]$. Si la valeur de p est inférieure à 0.01, la t statistique est marquée avec "**". Si la valeur de p est inférieure à 0.05 mais supérieure ou égale à 0.01, la t statistique est marquée avec "*". Sinon, la t statistique est laissée inchangée. La t statistique est enregistrée dans une liste, $liste\_stat$.

Finalement cela donne :

	Variable	R2	1	2	3	4	5	6	7	8
0	GDPC1	Pseudo R2	0.063	0.029	0.003	0.0	0.0	0.004	0.012	0.003
1		t stat	-3.979**	-2.746**	-0.869	-0.342	-0.151	0.948	1.65	0.823
2	PERMIT	Pseudo R2	0.109	0.11	0.061	0.034	0.016	0.012	0.006	0.001
3		t stat	-4.867**	-4.913**	-3.752**	-2.871**	-1.988*	-1.743	-1.219	-0.38
4	M1REAL	Pseudo R2	0.029	0.105	0.113	0.065	0.06	0.06	0.042	0.038
5		t stat	-2.534*	-4.48**	-4.574**	-3.615**	-3.438**	-3.422**	-2.889**	-2.743**
6	M2REAL	Pseudo R2	0.036	0.092	0.074	0.031	0.022	0.021	0.013	0.008
7		t stat	-2.879**	-4.395**	-3.957**	-2.614**	-2.21*	-2.153*	-1.706	-1.296
8	BOGMBASEREALx	Pseudo R2	0.001	0.033	0.073	0.074	0.037	0.034	0.016	0.009
9		t stat	0.609	-2.581**	-3.815**	-3.768**	-2.658**	-2.523*	-1.678	-1.242
10	S&P 500	Pseudo R2	0.133	0.055	0.024	0.003	0.0	0.003	0.015	0.008
11		t stat	-5.13**	-3.68**	-2.451*	-0.937	0.144	0.776	1.846	1.367
12	TB3MS	Pseudo R2	0.007	0.0	0.006	0.028	0.02	0.035	0.034	0.008
13		t stat	-1.354	-0.002	1.224	2.423*	2.143*	2.772**	2.71**	1.429
14	GS10	Pseudo R2	0.008	0.007	0.01	0.014	0.006	0.025	0.038	0.021
15		t stat	1.354	1.348	1.512	1.796	1.226	2.39*	2.92**	2.21*
16	GS10TB3Mx	Pseudo R2	0.105	0.185	0.228	0.22	0.161	0.108	0.068	0.046
17		t stat	-4.666**	-5.667**	-5.943**	-5.749**	-5.264**	-4.594**	-3.809**	-3.208**

2.1.2 Variable & Constante & SPREAD : $P(Y_{t+k}) = \alpha_0 + \alpha_1 X_t + \alpha_2 SPREAD_t$

La méthode "R2_insample_2" effectue des estimations de modèles Probit en utilisant une variable explicative et une constante. Elle calcule ensuite le pseudo R^2 et les t statistiques associées. Cette méthode présente quelques différences par rapport à la méthode précédente.

Étapes d'estimation du Probit : Une colonne spécifique j de df_1 est sélectionnée à chaque itération. Pour chaque retard de 1 à 8 : Une série X est créée à partir de df_1 en utilisant les colonnes $GS10TB3Mx$ et j, en excluant les $retard$ dernières valeurs. Une constante est ensuite ajoutée à X. Une série y est créée à partir de df_2 en excluant les $retard$ premières valeurs. Le modèle Probit est estimé en utilisant y comme variable dépendante et X comme variable indépendante, en utilisant la fonction $sm.Probit(y, X).fit(disp = 0)$. La log-vraisemblance du

modèle est crée avec la variable log_var. Le modèle est également estimé en utilisant uniquement la constante de X en tant que variable indépendante, et la log-vraisemblance du modèle avec la constante, log_const, est enregistrée $\log_vraisemblancedumodèlecontenantjustelaconstante$.

Le calcul du pseudo R^2 reste le même que durant la précédente étape pour les retards allant de 1 à 8. Cette fois-ci *log_var* est la log-vraisemblance du modèle avec les variables. Calcul des t statistiques et marquage de significativité: Pour chaque retard de 1 à 8 : la t statistique pour la variable j est extraite du modèle Probit en utilisant $model.tvalues[1]$ et arrondie à trois décimales. La t statistique pour la variable $GS10TB3Mx$ (spread) est extraite en utilisant $model.tvalues[2]$ et également arrondie à trois décimales. La valeur de p associée à la t statistique est extraite en utilisant $model.pvalues[1]$ pour la variable j et $model.pvalues[2]$ pour le spread. Si la valeur de p est inférieure à 0.01, la t statistique est marquée avec "**". Si elle est inférieure à 0.05 mais supérieure ou égale à 0.01, la t statistique est marquée avec "*". Sinon, la t statistique est laissée inchangée.

```
def R2_insample_2 (df_1, df_2):
    #Création de la liste qui accueillera les résultats des tests pour ensuite créer le df (plus rapide en terme de puissance)
    R2=[]
    tstat=[]
    tstatsp=[]
    new_df=df_1.drop(columns="GS10TB3Mx") #on enlève la colonne SPREAD pour pas qu'elle
    for j in new_df.columns:
        liste_R2=[]
        liste_stat=[]
        liste_stat2=[]
        for retard in range (1,9):
            X=df_1[["GS10TB3Mx",j]].iloc[:-retard,].reset_index(drop=True)
            X=sm.add_constant(X) #ajout d'une constante
            y=df_2.iloc[retard:,].reset_index(drop=True)
        #Variable
            model_var=sm.Probit(y,X).fit(disp=0)
            log_var=model_var.llf #log-vraisemblance du modèle avec variable
        #Constante
            model_const=sm.Probit(y,X['const']).fit(disp=0)
            log_const=model_const.llf #log-vraisemblance du modèle avec constante
        #R²
            pseudo_R2=round(1-(log_var/log_const)**((-2/len(X))*log_const),3)
            liste_R2.append(pseudo_R2)
        #T-stat
            t_stat=round(model_var.tvalues[1],3) #t-stat
            t_stat_spread=round(model_var.tvalues[2],3) # t stat du spread
            liste_stat.append(t_stat)
            liste_stat2.append(t_stat_spread)
        #p value
            p_value=model_var.pvalues[1]
            p_value_sp=model_var.pvalues[2]
            liste_stat[retard-1]= str(t_stat)+'**' if p_value < 0.01 else str(t_stat)+'*' if p_value < 0.05 else t_stat
            liste_stat2[retard-1]= str(t_stat_spread)+'**' if p_value_sp < 0.01 else str(t_stat_spread)+'*' if p_value_sp < 0.05 else t_stat_spread
        liste_R2.insert(0,j+" / SPREAD")
        liste_R2.insert(1,'Pseudo R2')
        liste_stat.insert(0,'')
        liste_stat.insert(1,'t stat')
        liste_stat2.insert(0,'')
        liste_stat2.insert(1,'t stat spread')
        R2.append(liste_R2)
        tstat.append(liste_stat)
        tstatsp.append(liste_stat2)
        data=[x for i in zip(R2, tstat,tstatsp) for x in i]  #pour afficher en alternance les R2 et t-stat de chaque var
df=pd.DataFrame(data,columns=["Variable"," ","1","2","3","4","5","6","7","8"])
    return(df)
```

Finalement cela donne :

	Variable		1	2	3	4	5	6	7	8
0	GDPC1 / SPREAD	Pseudo R2	0.165	0.212	0.229	0.22	0.161	0.114	0.084	0.051
1		t stat	-4.512**	-5.586**	-5.928**	-5.741**	-5.261**	-4.616**	-3.892**	-3.263**
2		t stat spread	-3.764**	-2.536*	-0.605	0.053	0.192	1.172	1.89	1.041
3	PERMIT / SPREAD	Pseudo R2	0.17	0.235	0.239	0.22	0.162	0.108	0.068	0.049
4		t stat	-3.58**	-4.744**	-5.324**	-5.255**	-4.917**	-4.288**	-3.612**	-3.248**
5		t stat spread	-3.752**	-3.287**	-1.546	-0.16	0.379	0.051	0.221	0.822
6	M1REAL / SPREAD	Pseudo R2	0.108	0.208	0.249	0.224	0.167	0.119	0.078	0.058
7		t stat	-4.259**	-4.438**	-4.837**	-5.238**	-4.44**	-3.512**	-2.814**	-2.147*
8		t stat spread	-0.483	-2.053*	-1.89	-0.55	-0.78	-1.347	-1.217	-1.394
9	M2REAL / SPREAD	Pseudo R2	0.113	0.219	0.241	0.221	0.162	0.11	0.07	0.047
10		t stat	-3.994**	-4.721**	-4.999**	-5.297**	-4.851**	-4.149**	-3.464**	-2.962**
11		t stat spread	-1.299	-2.679**	-1.573	-0.237	-0.334	-0.692	-0.658	-0.507
12	BOGMBASEREALx / SPREAD	Pseudo R2	0.119	0.194	0.264	0.253	0.171	0.118	0.072	0.048
13		t stat	-4.862**	-5.381**	-5.535**	-5.307**	-4.888**	-4.131**	-3.5**	-2.978**
14		t stat spread	1.807	-1.317	-2.66**	-2.513*	-1.31	-1.369	-0.81	-0.6
15	S&P 500 / SPREAD	Pseudo R2	0.253	0.243	0.247	0.22	0.166	0.117	0.096	0.06
16		t stat	-4.674**	-5.618**	-5.892**	-5.65**	-5.231**	-4.642**	-4.03**	-3.362**
17		t stat spread	-5.148**	-3.692**	-2.107*	-0.027	1.046	1.469	2.446*	1.767
18	TB3MS / SPREAD	Pseudo R2	0.156	0.216	0.245	0.22	0.161	0.116	0.079	0.047
19		t stat	-5.24**	-5.801**	-5.699**	-5.346**	-4.952**	-4.045**	-3.167**	-2.936**
20		t stat spread	-3.376**	-2.639**	-1.998*	-0.109	0.213	1.391	1.617	0.42
21	GS10 / SPREAD	Pseudo R2	0.105	0.186	0.231	0.221	0.162	0.116	0.09	0.057
22		t stat	-4.443**	-5.427**	-5.619**	-5.433**	-5.086**	-4.238**	-3.346**	-2.85**
23		t stat spread	-0.052	-0.535	-0.824	-0.229	-0.366	1.359	2.229*	1.62

2.2 Le cas out of sample

Rappelons que l'échantillon "out-of-sample" fait référence à l'utilisation de nouvelles données qui n'ont pas été utilisées pour ajuster le modèle. Ces données sont utilisées pour évaluer les performances du modèle sur des observations inédites. L'objectif est d'estimer comment le modèle se comporterait lorsqu'il est appliqué à de nouvelles données, qui n'ont pas été utilisées dans le processus d'apprentissage. Cela permet d'évaluer la capacité de généralisation du modèle et d'obtenir une estimation plus réaliste de ses performances réelles.

2.2.1 Variable & Constante : $P(Y_{t+k}) = \alpha_0 + \alpha_1 X_t$

De manière analogue à la partie in sample et à la fonction définit plus haut, une autre boucle itère sur les valeurs de retard allant de 1 à 8. À l'intérieur de cette boucle, les données X sont extraites en utilisant un décalage correspondant au retard actuel. Une constante est ensuite ajoutée à X en utilisant la fonction sm.add_constant(). Les données y sont également extraites du DataFrame en utilisant un décalage approprié. Ensuite, une boucle itère sur une plage allant de 0 au nombre d'observations de X moins 100. Cette boucle est utilisée pour

diviser les données en ensembles d'entraînement et de test, où les premières 100+i observations sont utilisées pour l'entraînement et la 100+i-retard observation est utilisée pour les tests. À l'intérieur de cette boucle, un modèle Probit est ajusté en utilisant les données d'entraînement à l'aide de la fonction sm.Probit(). Les prédictions sont ensuite effectuées sur les données de test à l'aide de la méthode predict(). Les log-vraisemblances du modèle avec la variable (log_var) et du modèle avec seulement une constante (log_cons) sont calculées. Les sommes des log-vraisemblances sont mises à jour à chaque itération. Enfin, le pseudo-R2 est calculé en utilisant les sommes des log-vraisemblances. Une fois que toutes les variables et leurs pseudo-R2 ont été calculés, le résultat est stocké dans un DataFrame avec des colonnes correspondant aux retards de 1 à 8.

Ce qui donne comme résultat :

	Variable		1	2	3	4	5	6	7	8
0	GDPC1	Pseudo R2	1.000000	1.000000	0.796259	0.153547	0.248970	0.628043	0.945489	0.326566
1	PERMIT	Pseudo R2	1.000000	1.000000	0.999998	0.993641	0.818078	0.695070	0.440678	0.105815
2	M1REAL	Pseudo R2	0.999999	1.000000	1.000000	0.999475	0.998800	0.997695	0.960755	0.947547
3	M2REAL	Pseudo R2	1.000000	1.000000	1.000000	0.998793	0.973374	0.966336	0.789761	0.500973
4	BOGMBASEREALx	Pseudo R2	0.982717	0.999973	1.000000	1.000000	0.996880	0.996664	0.875634	0.648384
5	S&P 500	Pseudo R2	1.000000	0.999999	0.996698	0.284616	0.095839	0.487115	0.979613	0.696707
6	TB3MS	Pseudo R2	0.441706	0.517277	0.946208	0.999888	0.989912	0.998833	0.996254	0.573233
7	GS10	Pseudo R2	0.992669	0.992222	0.992911	0.997414	0.813699	0.988438	0.997417	0.922728
8	GS10TB3Mx	Pseudo R2	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	0.999847	0.994531

2.2.1 Variable & Constante & SPREAD : $P(Y_{t+k}) = \alpha_0 + \alpha_1 X_t + \alpha_2 SPREAD_t$

Quant au modèle avec la variable spread, la seule différence dans le code de celui précédemment expliqué réside dans la manière dont les variables sont utilisées pour ajuster le modèle. Dans le code précédent, chaque variable de new_df était utilisée individuellement pour ajuster le modèle, tandis que dans ce code, "GS10TB3Mx" est ajoutée à chaque variable lors de l'ajustement du modèle. Cela signifie que chaque modèle est ajusté en utilisant à la fois la variable "GS10TB3Mx" et une autre variable spécifique à chaque itération.

Ce qui donne comme résultat :

	Variable		1	2	3	4	5	6	7	8
0	GDPC1 / SPREAD	Pseudo R2	1.0	1.0	1.0	1.0	1.0	1.0	0.999999	0.998780
1	PERMIT / SPREAD	Pseudo R2	1.0	1.0	1.0	1.0	1.0	1.0	0.999947	0.998848
2	M1REAL / SPREAD	Pseudo R2	1.0	1.0	1.0	1.0	1.0	1.0	0.999949	0.998358
3	M2REAL / SPREAD	Pseudo R2	1.0	1.0	1.0	1.0	1.0	1.0	0.999941	0.997495
4	BOGMBASEREALx / SPREAD	Pseudo R2	1.0	1.0	1.0	1.0	1.0	1.0	0.999960	0.997467
5	S&P 500 / SPREAD	Pseudo R2	1.0	1.0	1.0	1.0	1.0	1.0	1.000000	0.999712
6	TB3MS / SPREAD	Pseudo R2	1.0	1.0	1.0	1.0	1.0	1.0	0.999994	0.997181
7	GS10 / SPREAD	Pseudo R2	1.0	1.0	1.0	1.0	1.0	1.0	0.999998	0.999260

Il faut remarquer que les résultats obtenus avec le pseudo R^2 sont illogiques et empêche toute interprétation par celui-ci. Nous avons donc tenté de réaliser une analyse out-sample du R^2 normal. Voici donc les résultats de celle-ci : mettre ici screen résultat out-sample var+spread R^2 Possible aussi de mettre le graphique

2.3 Interprétation

Lors de notre analyse, nous avons identifié les variables qui sont significatives sur la plupart de leurs retards (lags). Pour évaluer leur importance, nous avons utilisé le critère du pseudo R^2 et celui de R^2 en ne considérant que les variables dont le coefficient est supérieur à 0.1.

Dans le cas de notre modèle in-sample, nous avons observé que seule la variable GS10TB3Mx (SPREAD), qui mesure l'écart de la courbe des taux, présente un pseudo R^2 supérieur à 0.1 pour la plupart des lags (1 à 6). Cela confirme ainsi son utilisation par Estrella et Mishkin dans leurs travaux. De plus, la variable M1REAL présente également un pseudo R^2 supérieur à 0.1 pour les retards -2 et -3. Le PERMIT et le S&P500 se révèlent également intéressants en retard t-1.

Pour confirmer nos hypothèses, nous avons constaté lorsque l'écart de la courbe des taux est combiné avec d'autres variables, nous observons des résultats intéressants. En effet, l'importance de la courbe des taux ne diminue pratiquement pas au-delà des 2 ou 3 premiers trimestres pour presque toutes nos variables.

Les résultats du modèle qui combine la courbe des taux avec les cours boursiers, tels que le S&P500, suggèrent que ces deux variables financières, facilement disponibles et continuellement mises à jour, forment une combinaison solide, notamment jusqu'au retard t-6 et surtout dès le premier trimestre.

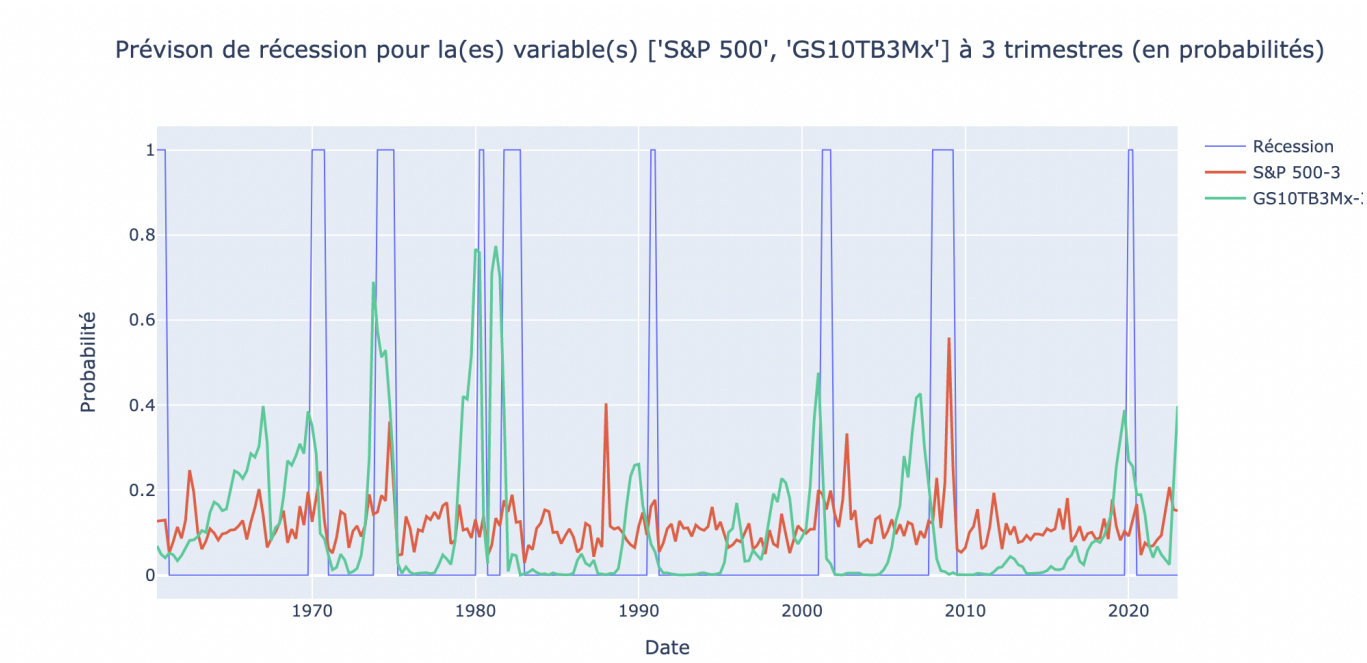
Dans le cas out of sample la combinaison du SP500 et du SPREAD dans le modèle $P(Y_{t+k}) = \alpha_0 + \alpha_1 X_t + \alpha_2 SPREAD_t$ présente, selon le R^2, une association significative assez forte. De plus, dans le cadre du modèle sans spread, certaines variables continuent de démontrer leur pertinence, notamment le SP500 qui reste un facteur influent dès le trimestre t-2. Ces constatations renforcent les résultats obtenus dans l'échantillon, où l'écart de la courbe de rendement joue un rôle prédominant.

En combinant le SPREAD et le S&P 500, nous associons donc deux variables qui captent différentes dimensions économiques. Le SPREAD reflète les anticipations du marché sur les taux d'intérêt, tandis que le S&P 500 reflète la confiance des investisseurs dans les marchés financiers. En les combinant, nous obtenons une vision plus complète de l'environnement

économique et financier, ce qui peut améliorer la précision de nos prévisions. La présence d'un pseudo R^2 élevé dès le premier trimestre, sans nécessité de retard, peut s'expliquer par le fait que ces deux variables réagissent rapidement aux changements économiques. Par conséquent, les signaux fournis par le SPREAD et le S&P 500 peuvent être plus immédiats et moins sujets à des décalages temporels par rapport aux autres variables.

Mais aussi, le SPREAD est étroitement lié à la performance des bons du Trésor à long terme. En effet, une augmentation significative du SPREAD indique une attente de taux d'intérêt à court terme plus faibles que ceux à long terme. Cette situation peut être associée à une anticipation de ralentissement économique, car les investisseurs exigent une prime de risque plus élevée pour détenir des titres à plus long terme. Ainsi, le SPREAD, en tant que mesure de l'écart entre les bons du Trésor à court terme et à long terme, capture les anticipations du marché concernant l'évolution de l'économie.

Dans ce qui suit, nous allons analyser un ensemble de quatre fonctions du code, qui permettent d'effectuer des prévisions de récession à l'aide d'une régression Probit et de représenter graphiquement les résultats obtenus. Les quatre fonctions du code ont pour objectif d'effectuer des prévisions de récession et de représenter graphiquement les résultats. La première fonction, prev_insample1(df, df2, trimestre), réalise une régression Probit en utilisant les données d'échantillon afin de prédire la probabilité de récession. La deuxième fonction, prev_insample2(df, df2, trimestre), est similaire à la première mais elle inclut une variable supplémentaire ("GS10TB3Mx") dans l'analyse. Elle effectue une régression Probit en prenant en compte à la fois les variables dans df et la variable "GS10TB3Mx". Les troisième et quatrième fonctions, prev_outsample1(df, df2, trimestre) et prev_outsample2(df, df2, trimestre), suivent une approche similaire mais utilisent une division train-test des données. Elles divisent les données dans df et df2 en ensembles d'entraînement et de test, puis réalisent une régression Probit sur l'ensemble d'entraînement pour prédire la probabilité de récession. La fonction prev_outsample2 inclut également la variable supplémentaire "GS10TB3Mx" dans l'analyse et utilise la fonction prev_outsample1 pour obtenir les prévisions de probabilité de récession pour cette variable.



2.4 Conclusion

Les résultats obtenus sont prometteurs et suggèrent que ces mesures peuvent jouer un rôle utile dans les prévisions macroéconomiques. Cependant, il est important de souligner que ces indicateurs ne doivent pas remplacer les modèles macroéconomiques. Au lieu de cela, ils peuvent compléter ces modèles et prévisions plus élaborées, en fournissant une vérification rapide et fiable.

Dans le processus d'évaluation du pouvoir prédictif des variables financières, plusieurs principes importants ont été mis en évidence. Premièrement, le sur-ajustement constitue un problème sérieux dans les prédictions macroéconomiques. Même en utilisant seulement quelques variables, l'ajout d'une seule variable supplémentaire ou d'un autre retard dans une variable peut diminuer la capacité prédictive d'un modèle parcimonieux. Et deuxièmement, les performances à in sample et out sample peuvent différer considérablement.

En outre, pour améliorer la situation, il est possible d'adopter une approche différente pour la sélection des variables, telle que l'utilisation du Lasso. Ces méthodes de sélection de variables visent à identifier les variables les plus pertinentes et à éliminer celles qui ont peu d'influence sur les prévisions, ainsi qu'améliorer la précision des prévisions en identifiant les variables les plus informatives. Cela permet de réduire le bruit et de concentrer l'attention sur les facteurs les plus pertinents pour la prédiction des récessions économiques.

3. Améliorations

3.1 Gestion des valeurs manquantes

L'option de suppression des valeurs manquantes est une autre approche couramment utilisée. Elle peut être réalisée de différentes manières. Une approche simple consiste à supprimer toutes les observations contenant au moins une valeur manquante. Cela garantit que seules les observations complètes sont utilisées pour l'analyse, mais cela peut entraîner une perte de données considérable. Une autre approche consiste à effectuer une suppression sélective des valeurs manquantes en fonction de critères spécifiques. Par exemple, on peut décider de supprimer une observation uniquement si plus de la moitié des variables sont manquantes. Cela permet de conserver un plus grand nombre d'observations tout en limitant l'inclusion de données potentiellement moins fiables. Il est important de noter que la suppression des valeurs manquantes peut avoir des implications sur les analyses et les résultats. L'échantillon réduit peut entraîner une perte de puissance statistique et une diminution de la représentativité des données. De plus, la suppression sélective des valeurs manquantes peut introduire un biais dans les estimations si les valeurs manquantes ne sont pas totalement aléatoires.

En résumé, le code calcule ci dessous le nombre de valeurs manquantes pour chaque colonne du dataframe df_1_transfo et identifie les colonnes qui contiennent des valeurs manquantes. Les noms de ces colonnes sont ensuite stockés dans la liste col_na. Ces informations vont être utilisées pour prendre des mesures appropriées.

```
manquant=df_1_transfo.isna().sum()
for i in range(0,len(manquant)) :
    if manquant[i]>=1 :
        print(manquant.index[i],manquant[i])
col_na=[]
for i in df_1_transfo.columns :
    if df_1_transfo[i].isna().sum() > 0 :
        col_na+=[i]
```

Ensuite, ce code sépare les colonnes ayant des valeurs manquantes en deux listes en fonction de leur pourcentage de valeurs manquantes : manq_sup5 pour les colonnes ayant plus de 5 % de valeurs manquantes et manq_inf5 pour les colonnes ayant 5 % ou moins de valeurs manquantes. Ensuite, un nouveau dataframe df_1nonnull est créé en supprimant les colonnes de df_1_transfo qui sont présentes dans la liste manq_sup5.

```
manq_sup5=[]
manq_inf5=[]
for i in col_na :
    mask=df_1_transfo[i].isna()
    if (len(df_1_transfo[mask])/len(df_1_transfo[i])*100)>5:
        manq_sup5+=[i]
    if ((len(df_1_transfo[mask])/len(df_1_transfo[i])*100)<=5):
        manq_inf5+=[i]
print(manq_sup5)
print(manq_inf5)
df_1nonnull=df_1_transfo.drop(columns=manq_sup5)
```

Finalement, ici les manq_sup5 sont : ['OUTMS', 'TCU', 'LNS13023621', 'LNS13023557', 'LNS13023705', 'LNS13023569', 'HOAMS', 'AWHNONAG', 'ACOGNOx', 'ANDENOx', 'INVCQRMTSPL', 'WPU0531', 'AHETPIx', 'COMPRMS', 'OPHMFG', 'ULCMFG', 'MORTGAGE30US', 'MORTG10YRx', 'REVOLSLx', 'DRIWCIL', 'VIXCLSx', 'USSTHPI', 'SPCS10RSA', 'SPCS20RSA', 'TWEXAFEGSMTHx', 'EXUSEU', 'USEPUINDXM', 'GFDEGDQ188S', 'GFDEBTNx', 'NASDAQCOM', 'CUSR0000SEHC'], ce qui correspond à l'ensemble de nos variables présentant des données manquantes. Nous n'avons pas besoin ici de réaliser d'imputation.

3.2 Sélection de variables

La méthode Lasso (Least Absolute Shrinkage and Selection Operator) se révèle être une approche puissante pour la sélection de variables dans le contexte des variables macroéconomiques. En présence d'un grand nombre de variables potentielles, le Lasso permet d'identifier les variables les plus importantes pour expliquer une variable cible tout en excluant celles qui ont une contribution négligeable. Cette technique présente plusieurs avantages clés pour la recherche en économie.

Tout d'abord, le Lasso permet de réduire la dimensionnalité des modèles en identifiant les variables les plus pertinentes. Cela facilite l'interprétation des résultats et permet de se concentrer sur les facteurs économiques les plus significatifs. Cette réduction de dimensionnalité est particulièrement cruciale dans le contexte macroéconomique où un grand nombre de variables peut entraîner une complexité excessive et une augmentation du risque de surajustement.

Deuxièmement, la méthode Lasso contribue à éviter le sur-ajustement des modèles. En pénalisant les coefficients des variables moins importantes et en les réduisant à zéro, le Lasso favorise des modèles plus parcimonieux et plus généraux. Cela permet d'obtenir des résultats plus robustes et une meilleure capacité de généralisation, ce qui est essentiel pour l'application des modèles à de nouvelles données et pour la prise de décision éclairée.

Enfin, le Lasso améliore l'interprétabilité des modèles économiques en identifiant les variables les plus significatives. Les coefficients non nuls sélectionnés par le Lasso peuvent être considérés comme les déterminants économiques clés, fournissant ainsi des informations précieuses pour les décideurs et les chercheurs. Cette capacité à extraire les facteurs économiques essentiels permet de mieux comprendre les mécanismes économiques sous-jacents et de formuler des politiques plus efficaces.

Finalement pour appliquer cette méthode on utilise la bibliothèque scikit-learn. On commence par une division des données en un ensemble d'entraînement et un ensemble de test à l'aide de la fonction train_test_split. Les données d'entraînement représentent 70 % des données initiales, tandis que les données de test représentent 30 %. Ensuite une initialisation d'un Lasso avec un hyperparamètre alpha fixé à 0.1. Puis, un entraînement du modèle de régression linéaire Lasso en utilisant les données d'entraînement (X_train et y_train) à l'aide de la méthode fit. Et finalement on utilise le modèle entraîné pour effectuer des prédictions (y_pred) sur les données de test (X_test) à l'aide de la méthode predict. En résumé, ce code effectue une régression linéaire Lasso pour prédire la variable cible USRECQ à partir des données du dataframe df_1nonnull. Il sélectionne les variables significatives à l'aide du Lasso et les stocke dans df_1final pour une analyse ultérieure.

La régression logistique avec pénalisation Ridge peut également être utilisée pour sélectionner les variables importantes dans notre modèle. La pénalisation consiste à réduire le coefficients des variables les moins importantes vers 0. Les variables avec un fort impact auront un coefficient de régression élevé, et cela permet alors de sélectionner les variables. De la même manière que pour le LASSO, nous divisons nos données en deux ensembles, et entraînons le modèle de régression logistique afin d'obtenir les coefficients. Nous remarquons que les variables sélectionnées sont, à une variable près, les mêmes que celles retenues par le critère du LASSO. Cela confirme notre choix de sélection, et nous continuons donc avec cet ensemble de variables.

Ensuite, nous examinons la corrélation entre les variables sélectionnées.

	Variable 1	Variable 2	Corrélation
6	TNWMVBSNNCBBDIx	TNWBSNNBBDIx	0.901548
8	TNWBSNNBBDIx	TNWMVBSNNCBBDIx	0.901548
4	AAAFFM	GS10TB3Mx	0.890753
10	GS10TB3Mx	AAAFFM	0.890753

Nous allons alors supprimer les variables AAAFFM (Obligations d'entreprises Moody's Seasoned Aaa - taux des fonds fédéraux) et TNWMVBSNNCBBDIx (Valeur nette du secteur des entreprises non financières p/au revenu disponible des entreprises)

D'une manière plus théorique, il est important de rappeler que le modèle de régression linéaire Lasso est donné par :

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \ldots + \beta_n x_n + \epsilon$$

où y est la variable cible (dans ce cas, `USRECQ`), $\beta_0, \beta_1, \ldots, \beta_n$ sont les coefficients du modèle, $x_1, x_2, \ldots, x_n$ sont les variables prédictives (les colonnes de `df_1nonnull` sélectionnées) et ϵ est le terme d'erreur.

L'objectif du Lasso est de trouver les valeurs des coefficients $\beta_0, \beta_1, \ldots, \beta_n$ qui minimisent la somme des moindres carrés régularisée par la norme L1 de ces coefficients. Cela peut être formulé comme suit :

$$\min_{\beta_0, \beta_1, \ldots, \beta_n} \left\{ \frac{1}{2m} \sum_{i=1}^{m} (y_i - (\beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \ldots + \beta_n x_{in}))^2 + \alpha \sum_{j=1}^{n} |\beta_j| \right\}$$

où m est le nombre d'échantillons dans les données d'entraînement, y_i est la valeur observée de la variable cible pour l'échantillon i, x_{ij} est la valeur de la variable prédictive j pour l'échantillon i, $\beta_0, \beta_1, \ldots, \beta_n$ sont les coefficients du modèle à estimer et α est le paramètre d'hyperparamètre qui contrôle la force de la régularisation.

La fonction `fit` du modèle Lasso ajuste les coefficients $\beta_0, \beta_1, \ldots, \beta_n$ en utilisant l'algorithme d'optimisation des moindres carrés avec contrainte L1. Elle trouve les valeurs qui minimisent l'expression précédente.

La fonction `predict` du modèle Lasso utilise les coefficients ajustés pour faire des prédictions sur de nouvelles données. Les prédictions sont obtenues en calculant y à partir des variables prédictives $x_1, x_2, \ldots, x_n$ en utilisant les coefficients ajustés.

Ce qui finalement donne comme résultat pour le in sample :

	Variable	R2	1	2	3	4	5	6	7	8
0	HWIx	Pseudo R2	0.0	0.002	0.011	0.018	0.021	0.022	0.021	0.017
1		t stat	-0.349	0.743	1.679	2.12*	2.289*	2.338*	2.255*	2.004*
2	NWPIx	Pseudo R2	0.005	0.001	0.0	0.001	0.002	0.002	0.002	0.001
3		t stat	-1.038	-0.407	0.13	0.487	0.759	0.785	0.713	0.546
4	UMCSENTx	Pseudo R2	0.073	0.041	0.017	0.002	0.0	0.0	0.001	0.0
5		t stat	-4.14**	-3.134**	-2.028*	-0.649	-0.304	0.05	0.356	0.199
6	TLBSNNBBDIx	Pseudo R2	0.022	0.024	0.029	0.023	0.015	0.009	0.003	0.005
7		t stat	2.349*	2.483*	2.71**	2.422*	1.97*	1.549	0.864	1.08
8	TNWBSNNBBDIx	Pseudo R2	0.0	0.002	0.001	0.003	0.002	0.008	0.0	0.004
9		t stat	-0.327	-0.667	0.466	0.805	0.686	1.391	0.216	0.939
10	GS10TB3Mx	Pseudo R2	0.105	0.185	0.228	0.22	0.161	0.108	0.068	0.046
11		t stat	-4.666**	-5.667**	-5.943**	-5.749**	-5.264**	-4.594**	-3.809**	-3.208**

	Variable		1	2	3	4	5	6	7	8
0	HWIx / SPREAD	Pseudo R2	0.113	0.185	0.23	0.225	0.167	0.115	0.075	0.052
1		t stat	-4.808**	-5.666**	-5.884**	-5.647**	-5.081**	-4.314**	-3.444**	-2.842**
2		t stat spread	-1.28	-0.296	0.657	1.093	1.202	1.307	1.322	1.191
3	NWPIx / SPREAD	Pseudo R2	0.109	0.185	0.23	0.224	0.164	0.11	0.069	0.047
4		t stat	-4.642**	-5.619**	-5.877**	-5.688**	-5.237**	-4.575**	-3.783**	-3.184**
5		t stat spread	-0.894	-0.004	0.659	0.916	0.836	0.662	0.467	0.297
6	UMCSENTx / SPREAD	Pseudo R2	0.189	0.236	0.246	0.221	0.161	0.108	0.069	0.046
7		t stat	-4.769**	-5.727**	-5.942**	-5.642**	-5.189**	-4.563**	-3.8**	-3.204**
8		t stat spread	-4.302**	-3.364**	-2.033*	-0.205	0.032	0.22	0.434	0.237
9	TLBSNNBBDIx / SPREAD	Pseudo R2	0.127	0.213	0.263	0.245	0.173	0.113	0.069	0.049
10		t stat	-4.623**	-5.637**	-5.905**	-5.706**	-5.206**	-4.519**	-3.758**	-3.135**
11		t stat spread	2.291*	2.556*	2.791**	2.367*	1.657	1.139	0.452	0.782
12	TNWBSNNBBDIx / SPREAD	Pseudo R2	0.11	0.195	0.229	0.22	0.161	0.111	0.068	0.048
13		t stat	-4.731**	-5.687**	-5.882**	-5.702**	-5.226**	-4.496**	-3.808**	-3.141**
14		t stat spread	-1.136	-1.562	-0.539	0.019	0.063	0.823	-0.2	0.626

Et pour out of sample : dans l'application du travail d'Estrella, les résultats hors échantillons du modèle probit ont été difficilement interprétables, car leurs pseudo R2 sont très proches de 1, et ne devraient pas atteindre de telles valeurs. Ici, dans le but d'identifier des améliorations de leur travail, nous allons utiliser la méthode des résultats hors échantillons comme elle est généralement employée. Nous divisons nos données en un ensemble d'entraînement sur lequel on applique le modèle probit et un ensemble de test, sur lequel on applique des prédictions. De plus, nous prenons en compte ici le coefficient R2 qui se calcule grâce à la fonction $r2_score$

	Variable		1	2	3	4	5	6	7	8
0	GDPC1	R^2	-	0.000967	-	-	-	0.000755	-	-
1	PERMIT	R^2	-	0.037797	0.020483	0.051266	0.011528	0.00718	-	-
2	M1REAL	R^2	-	0.069837	0.120988	0.085909	0.10059	0.049166	0.009043	0.018846
3	M2REAL	R^2	-	0.015325	0.057254	0.044726	0.037382	0.011482	-	0.004334
4	BOGMBASEREALx	R^2	-	0.020257	-	0.109086	-	-	-	-
5	S&P 500	R^2	0.099983	0.021506	0.027582	0.000616	-	-	-	-
6	TB3MS	R^2	-	-	0.001814	0.01374	0.023211	0.036431	-	-
7	GS10	R^2	-	-	-	-	-	-	-	0.024345
8	GS10TB3Mx	R^2	-	0.375387	0.372241	0.219184	0.090136	0.115314	-	0.038784

	Variable		1	2	3	4	5	6	7	8
0	GDPC1 / SPREAD	R^2	0.022374	0.385478	0.375353	0.216152	0.086868	0.129450	-	0.027229
1	PERMIT / SPREAD	R^2	-	0.331017	0.319536	0.201107	0.062669	0.113970	-	0.037093
2	M1REAL / SPREAD	R^2	-	0.385234	0.402792	0.226254	0.098967	0.123977	0.001076	0.043713
3	M2REAL / SPREAD	R^2	-	0.350324	0.358027	0.106101	0.079067	0.115242	-	0.037238
4	BOGMBASEREALx / SPREAD	R^2	-	0.407698	0.387873	0.290300	0.085757	0.048103	-	0.013011
5	S&P 500 / SPREAD	R^2	0.316506	0.470219	0.418379	0.199122	0.103635	0.109574	0.006121	0.058368
6	TB3MS / SPREAD	R^2	-	0.412354	0.404308	0.220350	0.088565	0.127963	-	0.036117
7	GS10 / SPREAD	R^2	-	0.379175	0.368139	0.218620	0.086975	0.123023	0.025618	0.061399

Rappelons que : HWIx fait référence à l'indice de demande de main-d'œuvre aux États-Unis. UMCSENTx fait référence à l'indice de sentiment des consommateurs de l'Université du Michigan. NWPIx fait référence à l'indice des prix à la production pour les industries non-agricoles. TLBSNNBBDIx fait référence au pourcentage de dettes des secteurs d'activité non financiers non constitués en société par rapport au revenu disponible des entreprises. TNWBSNNBBDIx fait référence au pourcentage de la valeur nette du secteur des entreprises non financières

non-corporatives par rapport au revenu disponible des entreprises non-corporatives. Et nous retrouvons SPREAD. (NB : nous retrouvons les mêmes variables via une régression ridge, ce qui confirme les résultats de notre méthode)

Lorsqu'on examine les résultats en échantillon, on constate que seul le spread (écart de la courbe des taux) présente une performance supérieure à 0.1 dès le premier trimestre. En revanche, la variable UMCSENTx (indice de sentiment des consommateurs) montre une diminution de sa force prédictive avec les retards. Cependant, lorsqu'elles sont combinées avec le spread, toutes les variables présentent une performance solide dès le premier trimestre, ce qui confirme l'importance du Spread dans la prévision des récessions économiques.

Dans le cas out-of-sample, on observe une combinaison significative entre la variable TLBSNNBBDIx (pourcentage de dettes des secteurs d'activité non financiers non constitués en société par rapport au revenu disponible des entreprises) et le spread avec un retard de trois trimestres. Cette combinaison s'avère particulièrement pertinente pour prendre en compte leur impact dans la prévision des récessions économiques

En associant le niveau d'endettement des entreprises (mesuré par TLBSNNBBDIx) avec les anticipations de marché (mesurées par le spread), on peut capturer des informations complémentaires sur la santé financière des entreprises et les perspectives économiques globales. Une hausse du niveau d'endettement des entreprises combinée à un spread élevé peut indiquer une situation économique précaire et augmenter les risques de récession.

Cette combinaison permet ainsi de prendre en compte à la fois des aspects financiers spécifiques aux entreprises (endettement) et des indicateurs économiques plus larges (spread) dans la prévision des récessions. Elle permet d'obtenir une vision plus complète et robuste de la situation économique, ce qui peut améliorer la précision des prévisions et renforcer la capacité à anticiper les périodes de contraction économique.

3.3 AUC

L'AUC (Area Under the Curve) est une mesure d'évaluation couramment utilisée pour évaluer la performance d'un modèle de classification, tel qu'un modèle Probit dans notre cas. L'AUC représente la capacité du modèle à classer correctement les observations en termes de probabilité de classification.

3.3.1 In sample

Plus spécifiquement, l'AUC mesure la capacité du modèle à distinguer entre les observations positives et négatives. Elle est calculée en traçant la courbe ROC (Receiver Operating Characteristic) qui représente le taux de vrais positifs (Sensibilité) en fonction du taux de faux positifs (1 - Spécificité) pour différentes valeurs seuil de classification. L'AUC est l'aire sous cette courbe ROC.

Une valeur d'AUC proche de 1 indique une très bonne capacité de discrimination du modèle, ce qui signifie qu'il est capable de bien classer les observations positives et négatives. Une valeur d'AUC proche de 0.5 indique une capacité de discrimination faible, où le modèle ne fait que des prédictions aléatoires.

Pour ce faire, auc_insample1(df_1, df_2), calcule l'AUC pour évaluer la performance d'un modèle de classification. Il itère sur chaque variable du dataframe et pour chaque variable, il effectue une série d'étapes. Ces étapes consistent à ajuster un modèle Probit. Ensuite, il calcule l'AUC pour la partie "in-sample" en comparant les prédictions du modèle avec les vraies valeurs cibles des données d'entraînement, et pour la partie "out-sample" en comparant les prédictions du modèle avec les vraies valeurs cibles des données de test. Les résultats d'AUC pour chaque variable sont ensuite stockés dans deux dataframes.

Le deuxième code, auc_insample2(df_1, df_2), effectue des étapes similaires au premier code, mais avec une différence. Avant de commencer les calculs, il supprime une colonne spécifique du dataframe. Ensuite, lors de l'itération sur les variables, il combine cette colonne supprimée avec chaque variable pour former une nouvelle variable d'entrée. Ensuite, il suit les mêmes étapes que le premier code pour ajuster le modèle, calculer l'AUC "in-sample" et "out-sample", et stocker les résultats dans les dataframes.

En résumé, le premier code le fait en utilisant les variables originales, tandis que le deuxième code le fait en combinant une colonne spécifique : SPREAD, avec chaque variable.

Ce qui donne pour variable & constante ainsi que variable & constante & SPREAD :

	Variable		1	2	3	4	5	6	7	8
0	HWIx	AUC	0.562240	0.484905	0.601144	0.688487	0.664145	0.644040	0.641667	0.639333
1	NWPIx	AUC	0.693405	0.682540	0.631373	0.523026	0.493421	0.482781	0.478333	0.548000
2	UMCSENTx	AUC	0.764409	0.781668	0.709150	0.543421	0.547862	0.499503	0.545667	0.529333
3	AAAFFM	AUC	0.857695	0.914255	0.905065	0.867599	0.855592	0.808940	0.740333	0.711000
4	TNWMVBSNNCBBDIx	AUC	0.490345	0.561469	0.521405	0.588158	0.588980	0.627152	0.338000	0.629667
5	TLBSNNBBDIx	AUC	0.537433	0.538126	0.555882	0.640461	0.590132	0.570861	0.543000	0.508667
6	TNWBSNNBBDIx	AUC	0.415627	0.577965	0.480719	0.585526	0.548520	0.577483	0.415833	0.593000
7	GS10TB3Mx	AUC	0.779263	0.861811	0.882190	0.881908	0.828618	0.761921	0.692000	0.683667

	Variable		1	2	3	4	5	6	7	8
0	HWIx	AUC	0.294613	0.217172	0.769360	0.643098	0.691453	0.697436	0.673504	0.631944
1	NWPIx	AUC	0.291246	0.215488	0.255892	0.503367	0.608547	0.589744	0.564103	0.414931
2	UMCSENTx	AUC	0.618687	0.447811	0.417508	0.578283	0.476068	0.509402	0.505128	0.511285
3	AAAFFM	AUC	0.648148	0.702020	0.786195	0.850168	0.805128	0.770940	0.747863	0.625000
4	TNWMVBSNNCBBDIx	AUC	0.318182	0.599327	0.589226	0.515152	0.529915	0.683761	0.370085	0.590278
5	TLBSNNBBDIx	AUC	0.831650	0.833333	0.796296	0.597643	0.630769	0.620513	0.550427	0.517361
6	TNWBSNNBBDIx	AUC	0.335017	0.616162	0.577441	0.537037	0.586325	0.658120	0.372650	0.515625
7	GS10TB3Mx	AUC	0.694444	0.778620	0.870370	0.856061	0.833333	0.809402	0.761538	0.664062

	Variable		1	2	3	4	5	6	7	8
0	HWIx	AUC	0.294613	0.217172	0.769360	0.643098	0.691453	0.697436	0.673504	0.631944
1	NWPIx	AUC	0.291246	0.215488	0.255892	0.503367	0.608547	0.589744	0.564103	0.414931
2	UMCSENTx	AUC	0.618687	0.447811	0.417508	0.578283	0.476068	0.509402	0.505128	0.511285
3	AAAFFM	AUC	0.648148	0.702020	0.786195	0.850168	0.805128	0.770940	0.747863	0.625000
4	TNWMVBSNNCBBDIx	AUC	0.318182	0.599327	0.589226	0.515152	0.529915	0.683761	0.370085	0.590278
5	TLBSNNBBDIx	AUC	0.831650	0.833333	0.796296	0.597643	0.630769	0.620513	0.550427	0.517361
6	TNWBSNNBBDIx	AUC	0.335017	0.616162	0.577441	0.537037	0.586325	0.658120	0.372650	0.515625
7	GS10TB3Mx	AUC	0.694444	0.778620	0.870370	0.856061	0.833333	0.809402	0.761538	0.664062

	Variable		1	2	3	4	5	6	7	8
0	HWIx / SPREAD	AUC	0.806892	0.863990	0.882026	0.898026	0.834211	0.764901	0.709667	0.701000
1	NWPIx / SPREAD	AUC	0.797683	0.862745	0.879412	0.891776	0.826974	0.762583	0.693000	0.687667
2	UMCSENTx / SPREAD	AUC	0.862448	0.916900	0.910131	0.880921	0.830921	0.763245	0.693000	0.681000
3	TLBSNNBBDIx / SPREAD	AUC	0.777481	0.861189	0.896078	0.905263	0.836184	0.763576	0.688667	0.687333
4	TNWBSNNBBDIx / SPREAD	AUC	0.786393	0.863679	0.879739	0.883882	0.828618	0.774503	0.695667	0.705000

3.3.2 Out of sample

Le premier code, auc_outsample1(df_1, df_2) itère sur chaque variable du dataframe et pour chaque variable, il effectue une série d'étapes. Ces étapes consistent à ajuster un modèle Probit en utilisant les données d'entrée de la variable et les données de sortie correspondantes du dataframe df_2. Ensuite, il effectue une prédiction pour chaque observation de test en utilisant le modèle ajusté. Il compare ensuite les prédictions avec les vraies valeurs cibles des données de test et calcule l'AUC correspondant.

Le deuxième code, auc_outsample2(df_1, df_2), effectue des étapes similaires au premier code, mais avec une différence. Avant de commencer les calculs, il supprime une colonne spécifique (spread) du dataframe df_1. Ensuite, il suit les mêmes étapes que le premier code pour ajuster le modèle, effectuer les prédictions, comparer avec les vraies valeurs cibles et calculer l'AUC correspondant pour chaque retard.

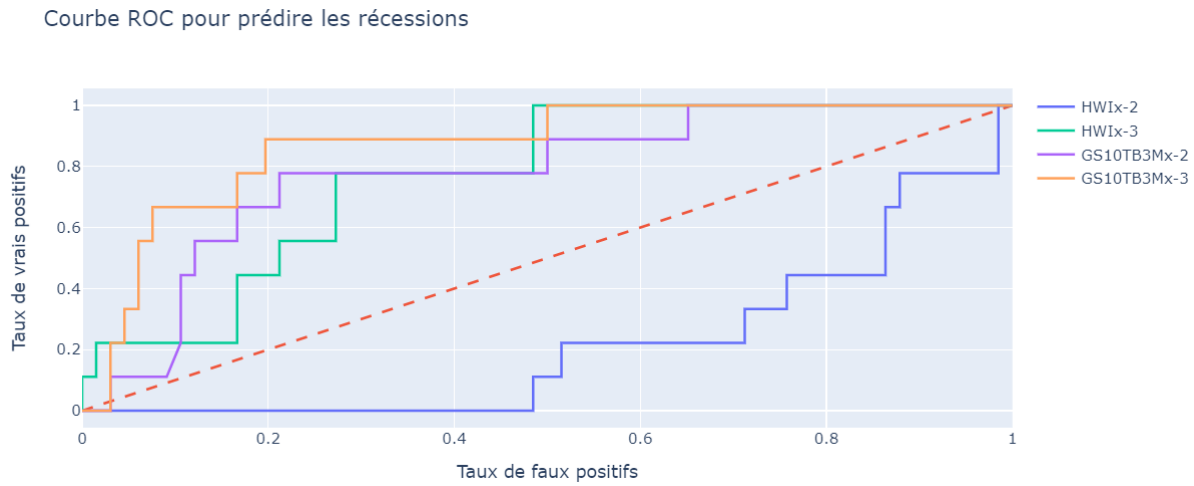
Ce qui donne pour variable & constante ainsi que variable & constante & SPREAD :

	Variable		1	2	3	4	5	6	7	8
0	HWIx	AUC	0.604716	0.633484	0.676353	0.729047	0.759977	0.768065	0.762772	0.751479
1	NWPIx	AUC	0.564290	0.727941	0.891168	0.931114	0.917872	0.895688	0.840282	0.723669
2	UMCSENTx	AUC	0.653565	0.588235	0.538462	0.526980	0.545402	0.568765	0.610100	0.581657
3	TLBSNNBBDIx	AUC	0.838293	0.880090	0.911681	0.908726	0.873916	0.833333	0.736935	0.696450
4	TNWBSNNBBDIx	AUC	0.424481	0.627828	0.596011	0.618255	0.673222	0.694056	0.672343	0.692308
5	GS10TB3Mx	AUC	0.702414	0.832014	0.892308	0.910448	0.908039	0.896853	0.889607	0.866864

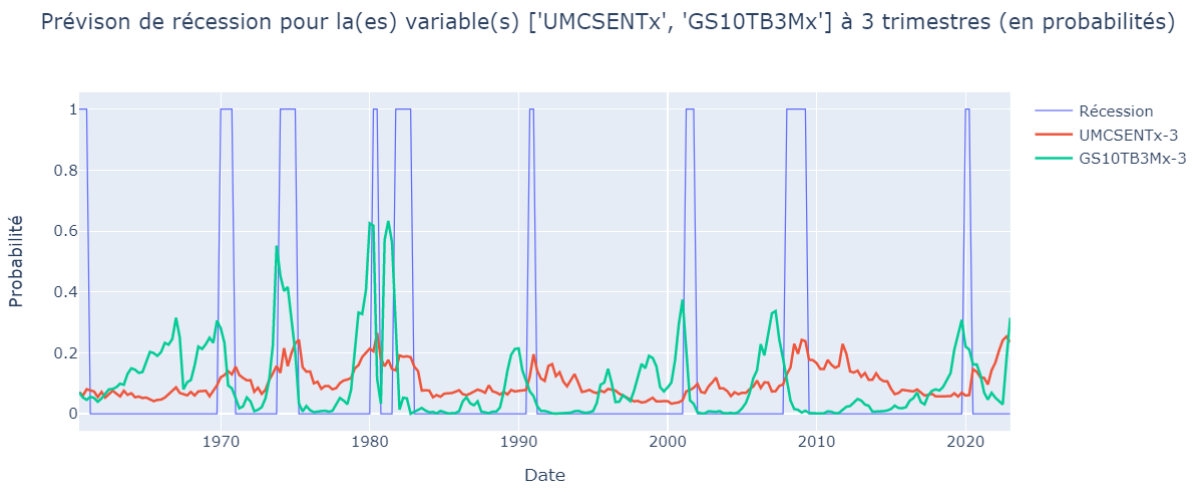
	Variable		1	2	3	4	5	6	7	8
0	HWIx / SPREAD	AUC	0.732173	0.837670	0.885470	0.913892	0.924234	0.913170	0.905461	0.865680
1	NWPIx / SPREAD	AUC	0.752386	0.882353	0.923077	0.947761	0.938693	0.907925	0.887845	0.854438
2	UMCSENTx / SPREAD	AUC	0.846154	0.909502	0.931624	0.920781	0.914980	0.893357	0.879037	0.851479
3	TLBSNNBBDIx / SPREAD	AUC	0.820887	0.939480	0.962393	0.951206	0.924234	0.903846	0.879037	0.863314
4	TNWBSNNBBDIx / SPREAD	AUC	0.727120	0.863122	0.895726	0.912744	0.908618	0.899184	0.897827	0.831953

3.4 Graphique

La fonction plot_ROC1 que nous allons explorer permet de tracer les courbes ROC (Receiver Operating Characteristic) pour évaluer les performances des modèles de prédiction des récessions en utilisant différentes variables et périodes de prévision. Cette dernière effectue une analyse de la courbe ROC pour évaluer les performances d'un modèle de prédiction des récessions. Elle prend en compte les variables explicatives, la variable cible, les variables à inclure dans l'analyse et les périodes de prévision. Elle divise les données en ensembles d'entraînement et de test, ajuste un modèle de régression Probit sur l'ensemble d'entraînement, effectue des prédictions sur l'ensemble de test, et calcule les taux de faux positifs, les taux de vrais positifs et l'aire sous la courbe ROC. Elle trace ensuite les courbes ROC pour chaque variable et période de prévision, fournissant ainsi une évaluation graphique des performances du modèle.



En utilisant les quatre fonctions qui effectuent des prévisions de récession et représentent graphiquement les résultats, les prévisions pour les variables UMCSENTX et SPREAD sur une période de trois trimestres



Selon la courbe ROC, nous pouvons observer que les droites les plus à gauche et en haut correspondent aux prévisions réalisées à l'aide du SPREAD, avec un écart de -2 trimestres et de -3 trimestres. Cette observation suggère que ces périodes de prévision ont une capacité supérieure à distinguer les récessions des périodes non récessives. En examinant le graphique, nous constatons également des pics de spread à chaque période de récession. Ces pics indiquent que les prévisions effectuées pendant ces périodes spécifiques sont plus susceptibles de détecter les récessions imminentes. Cela suggère que ces périodes de prévision plus proches de la récession fournissent des informations cruciales pour le modèle de prédiction, lui permettant d'obtenir de meilleures performances.En combinant ces observations, il est clair que l'utilisation d'une fenêtre de prévision plus large avant la période de récession réelle, telle que celle avec un écart de -2 ou -3 trimestres, améliore considérablement les performances prédictives. Ces résultats indiquent que des périodes de prévision plus éloignées fournissent des signaux précurseurs plus forts et plus fiables de la récession à venir.

De plus, il convient de noter que l'utilisation de l'indice de sentiment des consommateurs de l'Université du Michigan (UMCSENTx) en tant que variable dans les prévisions semble également produire des résultats prometteurs. Les courbes associées à cette variable démontrent une capacité distincte à prédire les récessions, avec des pics significatifs aux périodes de récession.Il est intéressant de noter que l'utilisation de l'indice de demande de main-d'œuvre aux États-Unis (HWIx) comme variable dans nos prévisions de récession présente un comportement surprenant sur la courbe ROC. En particulier, pour un écart de temps de -3 trimestres, la courbe ROC montre une performance relativement élevée, avec une position à gauche et en haut, ce qui indique une capacité à distinguer efficacement les récessions. Cependant, pour un écart de temps de -2 trimestres, la courbe ROC présente une position plus basse à droite, ce qui suggère une performance moins favorable dans la prédiction des récessions.

L'indice de demande de main-d'œuvre peut être considéré comme un indicateur avancé de l'activité économique, reflétant les perspectives d'embauche et l'expansion des entreprises. Ainsi, lorsqu'il est utilisé avec un écart de temps de -3 trimestres, il peut capter les signaux précurseurs d'une récession imminente, ce qui explique sa performance plus élevée. En revanche, avec un écart de temps de -2 trimestres, il est possible que l'indice de demande de main-d'œuvre ne capture pas encore pleinement les effets réels d'une récession en cours. Les fluctuations économiques peuvent prendre un certain temps pour se refléter dans les chiffres de la demande de main-d'œuvre, ce qui peut expliquer la baisse de performance observée dans ce cas spécifique.

3.5 Conclusion

En conclusion, l'étude réalisée en utilisant la méthode du papier de référence a permis d'obtenir des résultats intéressants dans la prédiction des récessions. Cependant, en effectuant des améliorations telles que la sélection de variables pertinentes et l'utilisation de la courbe ROC, il est possible d'améliorer encore davantage les performances du modèle. La sélection judicieuse des variables à inclure dans l'analyse permet de prendre en compte les facteurs les plus significatifs pour prédire les récessions. En identifiant les variables les plus pertinentes, on peut obtenir des modèles plus précis et éviter d'inclure des variables redondantes ou peu informatives. L'utilisation de la courbe ROC offre également une évaluation plus approfondie des performances du modèle. Elle permet de visualiser la capacité du modèle à discriminer les récessions des périodes non récessives, en tenant compte à la fois de la sensibilité et de la spécificité. Cela permet de choisir les seuils de prédiction les plus optimaux, en maximisant la capacité de prédiction du modèle. Ainsi, en combinant la méthodologie du papier de référence avec des améliorations telles que la sélection de variables pertinentes et l'utilisation de la courbe ROC, il est possible d'obtenir des résultats plus précis et fiables dans la prédiction des récessions. Ces améliorations permettent d'ajuster le modèle en fonction du contexte spécifique de l'étude et d'optimiser ses performances prédictives.

De plus, lors de nos prévisions, nous avons observé une récente augmentation du "spread" ainsi que de "TLBSNNBBDIx", qui représente le pourcentage de dettes des secteurs d'activité non financiers non constitués en société par rapport au revenu disponible des entreprises. Cette observation peut être considérée comme un signe précurseur potentiel d'une future récession. Si, dans les mois à venir, nous continuons à observer une augmentation soutenue, cela pourrait renforcer nos inquiétudes quant à une éventuelle récession économique. De plus, si la valeur du coefficient de probabilité dépasse 0,4 pendant plus d'un trimestre, cela confirmerait la récession.