Données Financières massives sous Python

Prédire les récessions

Tasnim Ben Hamed & Sarra Boudjadja

Problème :

Il s'agit d'un problème de classification binaire qui permet de prédire si on est en récession ou pas (à horizon de 2 mois), selon des variables macroéconomiques et financières. La base de données a été construite en récupérant des données du site FRED via Python. Initialement, nous avions voulu utiliser une base de données préparée par Michael W. McCracken & Serena Ng dans un papier de recherche (2015). Mais nous avons reconstruit un subset de cette base, avec moins de features pour ne pas dépasser la limite des requêtes sur FRED. La période prise en compte est **1959-01 à 2022-11** et les données sont mensuelles. Notre base est donc composée de 12 features (versus 129 pour la base originale) et 768 observations. Le pays en question est les Etats Unis.

- La variable target est une variable binaire : 'recession' ou 'no recession' . Nous avons utilisé les périodes de récession identifiées par NBER . La série de la target a été récupérée du site FRED (ticker: usrec).
- 12 variables explicatives , dont **4 financières** (taux Fed funds , yield 10 ans US , taux 3 mois US , indice SP500) et **8 macroéconomiques** (industrial production, real personal income, Non-farm payrolls, unemployement rate, Housing starts, PPI, Headline CPI, Core CPI) . Il est à noter que la variable SP500 (et la seule) n'a pas été récupérée via Python mais directement de la base originale, vu que nous n'avons pas trouvé d'historique assez long (sur Yahoo Finance, la dernière date disponible est en 1985). Travailler sur une longue période est ici primordial car les récessions ne sont pas fréquentes.

Méthodologie :

Nous avons dans un premier temps stationnarisé les variables explicatives, en appliquant les mêmes transformations de la base de données originale :

- log différencier le SP500, Industrial Production, Real Personal Income, Non-farm payrolls
- différencier le unemployement rate, le fed funds rate, le 10Y-yield, 3M-yield
- logger les housing starts
- double différencier le log du PPI, Headline CPI, Core CPI

Ensuite, nous avons laggé les variables explicatives de 2 mois par rapport à la série de la target , car nous évaluons le pouvoir de prévision de la récession en amont.

1- Analyse descriptive et sélection des variables :

La 1ère étape est de diviser l'échantillon en sous-échantillon d'entraînement et de test. C'est sur l'échantillon d'entraînement que l'analyse et la sélection seront faites.

- **Split train/test**

Notre split est 70% train et 30% test. Le split n'est pas aléatoire mais manuelle (dans l'ordre) car notre base est temporelle.

Le split train/test montre que notre base de données est **déséquilibrée** **(annexe 1)**: 14% des observations de l'échantillon train constituent des récessions et 86% sont des périodes de non récession. Nous prendrons en compte ce déséquilibre dans la partie évaluation de la performance des modèles .

- **Analyse des variables & Feature selection**

Nous avons choisi d'agréger les 2 variables yield 10 ans et yield 3 mois dans une seule variable spread 10Y-3M = yield 10 ans – yield 3 mois

La matrice de corrélation (de Pearson) montre des corrélations très faibles entre les variables **(annexe 2)** . Nous n'avons pas un problème de multicolinéarité. La seule paire présentant une corrélation relativement « forte » est le fed funds & le spread 10Y-3M (-0.73).

Vu que le target est qualitatif et que les variables explicatives sont continues , nous avons utilisé la **corrélation point-bisserial** pour évaluer la corrélation entre le target et les features **(annexe 3)**. Par ailleurs, nous avons plotté les box plots de chaque variable visualisant la distribution pour chaque classe (classe 'recession' et 'no recession') **(annexe 4)**. Ces derniers confirment les corrélations point-bisserial: les variables différencantes (celles pour lesquelles la distribution change entre la classe 'recession' et 'no recession') présentent le plus de corrélation avec la target. Nous supprimons le PPI , les Housing starts et le CPI headline qui ne sont pas du tout différencants selon les box plots et leurs corrélations (point-biserial) avec le target sont négligeables.

Les features qui présentent le plus de corrélation (relativement aux autres) avec la target sont : Non-farm payrolls, Inudstrial production, real personal income et Fed funds.

2- Modèles de ML :

Notre problème étant un problème de classification, les modèles qui vont être testés sont :

- Régression logistique
- KNN (qui requiert le scaling des variables, car c'est une méthode basée sur des distances)
- Random Forest

Le metric d'évaluation choisi est le **« Recall »** et non pas l'accuracy qui n'est pas du tout pertinent vu le **déséquilibre** des données . Le « Recall » est définie comme étant = $\frac{vrais\ positifs}{faux\ négatifs+vrais\ positifs}$. En d'autres termes , c'est le nombre d'observations correctement prédites comme étant des récessions par rapport au nombre total de récessions effectives. Notre but est de maximiser ce ratio.

En entrainant les modèles avec les paramètres de défaut, les résultats sur le testing set sont les suivants :

Avant tuning	**Accuracy**	**Confusion matrix**	**Recall**
Régression logistique	91%	Act \ Pred: 0 → 208, 1 → 0; 1 → 20, 1 → 0	0%
KNN	91.66%	Act \ Pred: 0 → 206, 1 → 2; 1 → 17, 1 → 3	15%
Random Forest	92,98%	Act \ Pred: 0 → 207, 1 → 1; 1 → 15, 1 → 5	25%

1 correspond à « recession » et 0 correspond à « no recession »

Notons que l'accuracy n'est pas représentatif de la performance du modèle : la régression logistique donne une accuracy très élevée mais ne prédit **aucune récession** ! Le random Forest arrive à prédire 25% des récessions, ce qui est très insuffisant.

- **Tuning des hyperparamètres :**

Pour optimiser nos hyperparamètres, nous avons utilisé la cross-validation avec 3 splits. L'entrainement du modèle et son test ne doivent pas se faire sur des splits sélectionnés aléatoirement , car nos données sont temporelles . La fonction **TimeSeriesSplit** de Scikitlearn réalise ce process « en séquence » et prend donc en compte le caractère temporel des séries.

- **Régression logistique :** Il n' y a pas d'hyperparamètres particuliers à optimiser . Cependant , nous nous sommes rendus compte que la pénalité par défaut est la L2. La pénalisation ici n'étant pas pertinente car nos features ne sont pas nombreux, nous avons éliminé la pénalisation. Mais ceci n'a pas eu d'impact sur la performance du modèle.
- **KNN :** L'hyperparamètre à optimiser est le nombre de voisins. Le metric d'évaluation à prioriser est le recall. **L'annexe 5** montre les 3 différents metrics (accuracy, recall , précision) en fonction du nombre des voisins . En privilégiant le recall, le nombre de voisins optimal est 1, pour lequel un peu moins de 50% des récessions sont correctement prédites. Notons bien que l'accuracy est toujours meilleure que la precision et recall, quelque soit le nombre de voisins et ceci est du au fait que la data est déséquilibrée. Cette optimisation n'a toujours pas amélioré le modèle pisque le recall sur le test set est resté le même.
- **Random Forest :** Les 2 hyperparamètres fondamentaux à optimiser sont le nombre d'arbres et le nombre de features . Rappelons que le Random Forest repète le processus de construction de l'arbre de classification plusieurs fois (nombre à optimiser), en sélectionnant à chaque fois l'échantillon par boostrapping, avec à chaque split, une sélection aléatoire d'un subset de n_features parmi le nombre global de features.
  - ➔ Le couple (n_arbres,n_features) optimal est **(9,7)**. En entraînant le modèle avec ces 2 valeurs, la performance out-of-sample s'est drastiquement améliorée: 11 récessions sur 20 ont été correctemet prédits. Cependant, ceci nous donne un **recall de 55%**. La performance du modèle n'est toujours pas bonne mais beaucoup plus acceptable après le tuning.

Les résultats sont synthétisés dans le tableau ci-dessous :

Après tuning	**Accuracy**	**Confusion matrix**	**Recall**
Régression logistique	91%	Act \ Pred: 0 → 0: 208, 1: 0 ; 1 → 0: 20, 1: 0	0%
KNN	92.1%	Act \ Pred: 0 → 0: 207, 1: 1 ; 1 → 0: 17, 1: 3	15%
Random Forest	95,17%	Act \ Pred: 0 → 0: 207, 1: 1 ; 1 → 0: 9, 1: 11	55%

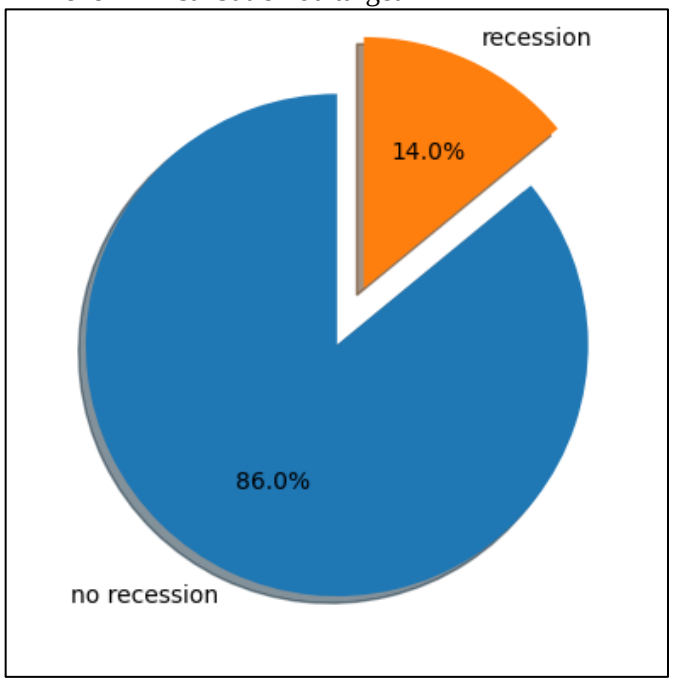
1 correspond à « recession » et 0 correspond à « no recession »

Conclusion :

Le Random Forest a « relativement » le mieux performé, mais notre projet présente plusieurs limites. La plus importante est que le nombre de features est faible. Les récessions dépendent d'une centaine de variables , mais vu la limite des requêtes à envoyer au site FRED, nous avons préféré sélectionné seulement quelques variables des 2 univers: financier et macroéconomique. La 2ème limite est que la distribution de notre target est déséquilibrée. Même si nous avons pris cela en compte dans l'évaluation, il existe des méthodes qui permettent d'équilibrer la data (sampling) qui ne s'appliquent pas aux séries temporelles. Finalement, la base de données globale n'est pas très volumineuse (768 observations) .

Annexes :

Annexe1 – Distribution du target



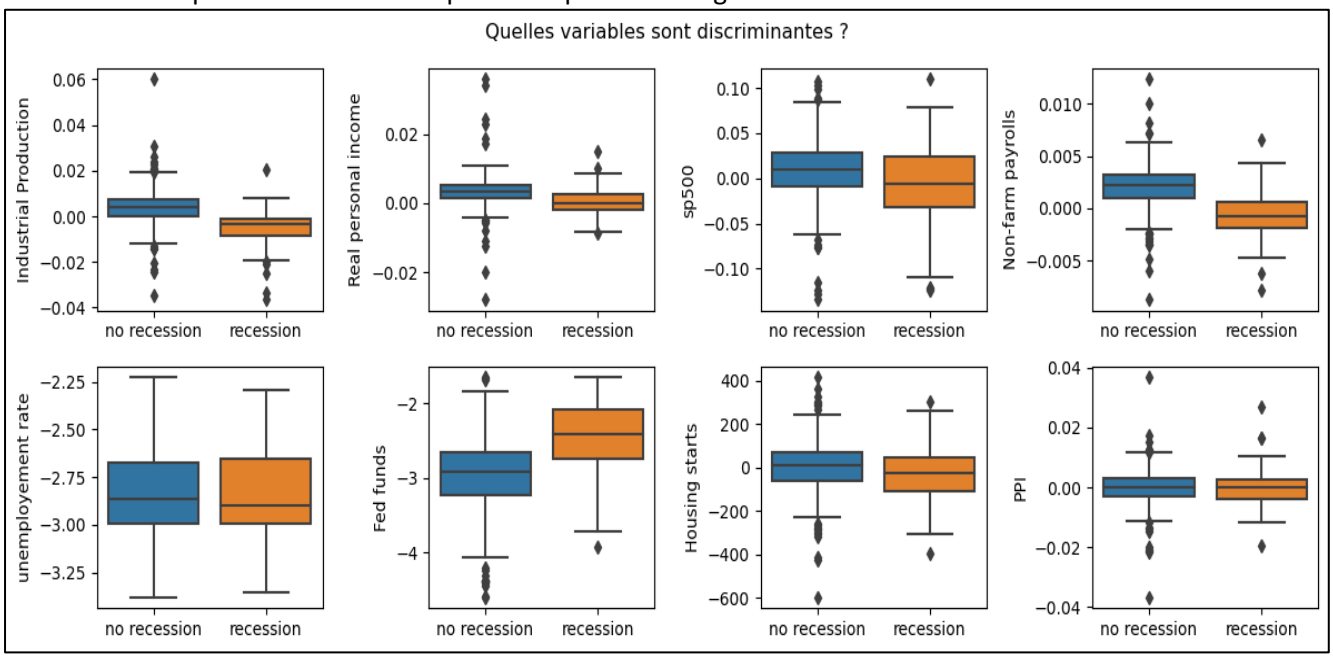
Annexe2 – Matrice de corrélation

	Industrial Production	Real personal income	sp500	Non-farm payrolls	unemployement rate	Fed funds	Housing starts	PPI	Headline CPI	Core CPI	spread 10Y-3M
Industrial Production	1.000	0.316	0.006	0.646	-0.041	-0.150	0.118	0.015	0.065	0.122	0.046
Real personal income	0.316	1.000	0.105	0.320	-0.084	-0.064	0.102	-0.036	-0.087	0.054	-0.033
sp500	0.006	0.105	1.000	0.004	0.109	-0.045	0.093	-0.023	-0.020	0.040	0.045
Non-farm payrolls	0.646	0.320	0.004	1.000	-0.175	0.000	0.083	0.020	0.074	0.110	-0.144
unemployement rate	-0.041	-0.084	0.109	-0.175	1.000	0.150	0.076	-0.002	-0.019	-0.028	0.338
Fed funds	-0.150	-0.064	-0.045	0.000	0.150	1.000	-0.084	-0.014	-0.016	0.000	-0.730
Housing starts	0.118	0.102	0.093	0.083	0.076	-0.084	1.000	0.051	-0.023	-0.011	0.106
PPI	0.015	-0.036	-0.023	0.020	-0.002	-0.014	0.051	1.000	0.389	0.000	0.021
Headline CPI	0.065	-0.087	-0.020	0.074	-0.019	-0.016	-0.023	0.389	1.000	0.351	0.011
Core CPI	0.122	0.054	0.040	0.110	-0.028	0.000	-0.011	0.000	0.351	1.000	-0.017
spread 10Y-3M	0.046	-0.033	0.045	-0.144	0.338	-0.730	0.106	0.021	0.011	-0.017	1.000

Annexe 3 – Corrélation point biserial

	correlation	pvalue
Industrial Production	-0.38115	0.00000
Real personal income	-0.23839	0.00000
sp500	-0.19350	0.00001
Non-farm payrolls	-0.42761	0.00000
unemployement rate	0.04486	0.30079
Fed funds	0.29604	0.00000
Housing starts	-0.10003	0.02079
PPI	-0.00104	0.98079
Headline CPI	-0.01108	0.79832
Core CPI	-0.03897	0.36880
spread 10Y-3M	-0.19917	0.00000

Annexe 4 – Box plots des variables explicatives pour les 2 régimes



Annexe 5 – Tuning du nombre de voisins pour le KNN (3 metrics : precision, recall, accuracy)