Évaluation de modèles

Cette section est basée sur James et al. (2021), chapitre 5.

1 Évaluer la performance d'un modèle prédictif

Dans la section précédente, nous avons introduit des outils pour mesurer la qualité d'un estimateur : l'erreur quadratique moyenne (MSE) pour les variables quantitatives et le taux d'erreur (ER) pour les variables qualitatives. Ces mesures comparent les valeurs prédites $\widehat{Y} = \widehat{f}(X)$ aux valeurs observées Y. Cependant, si l'on calcule ces erreurs uniquement à partir des données qui ont servi à entraîner le modèle, on risque de **sous-estimer** la véritable erreur de prédiction. Pourquoi ? Parce que l'estimateur $\widehat{f}$ a été ajusté pour minimiser l'erreur sur ces mêmes données. Il s'y adapte donc bien, et généralement, trop bien ! Cela peut conduire à l'illlusion que notre modèle est performant. En effet, un modèle très flexible peut avoir une erreur faible sur les données d'entraînement simplement parce qu'il capture le bruit plutôt que le signal. Mais si le modèle s'adapte trop aux données d'entraînement, il risque de mal généraliser à de nouvelles données, i.e. des données qu'il n'a jamais vues. Ce phénomène s'appelle le **sur-ajustement** (*overfitting*).

> Remarque : Sur-ajustement et sous-ajustement
>
> Un modèle trop flexible peut s'adapter parfaitement aux données d'entraînement, y compris au bruit aléatoire. Il aura une erreur faible sur ces données mais une erreur élevée sur de nouvelles observations. On dira qu'il y a **sur-ajustement** (*overfitting*) du modèle. À l'inverse, un modèle trop rigide (par exemple, une droite constante) ne pourra pas capturer la structure des données, même sur l'ensemble d'entraînement. On dira qu'il y a **sous-ajustement** (*underfitting*) du modèle.
> L'objectif est de trouver le bon compromis entre flexibilité et capacité de généralisation.

Pour évaluer objectivement un modèle, l'idéal serait de le tester sur des données complètement indépendantes de celles utilisé pour l'apprentissage. On distingue donc deux ensembles : un **jeu d'entraînement**, utilisé pour ajuster le modèle et un **jeu de test**, utilisé pour évaluer la performance prédictive du modèle. En practique, nous n'avons généralement pas accès à un jeu

de test pour faire cette évaluation. Dans cette section, nous allons deux approches permettant de contourner ce problème.

2 Jeu de données de validation

Quand on ne dispose que d'un seul jeu de données, une solution simple consiste à le diviser **aléatoirement** en deux sous-ensembles : un **jeu d'entraînement** pour ajuster le modèle et un **jeu de validation** pour estimer l'erreur de prédiction. On parle alors d'approche par jeu de validation. La Figure 1 présente un schéma de cette approche et la Figure 2 et la Figure 3 présentent un exemple de jeu d'entraînement et de validation, ainsi que la MSE associée.

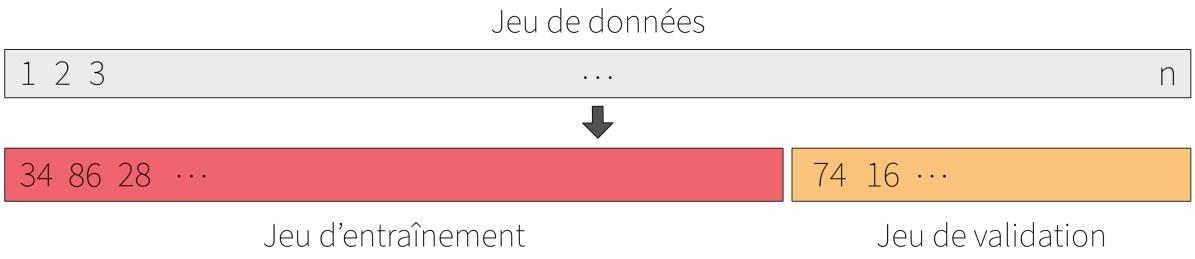

FIGURE 1 : Schéma de l'approche par jeu de validation.

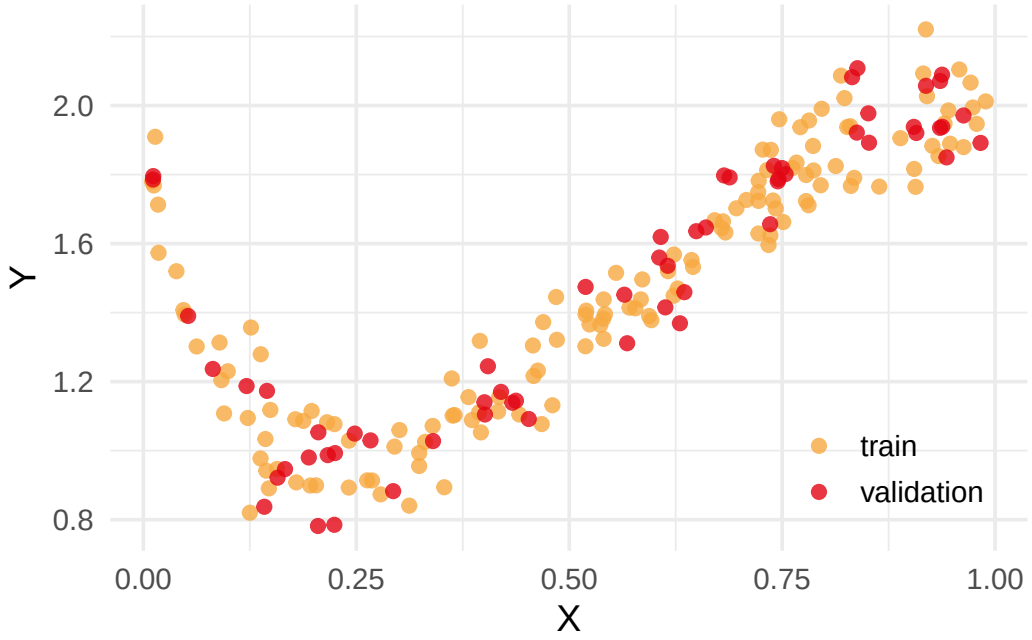

FIGURE 2 : Exemple de jeu d'entraînement et de validation.

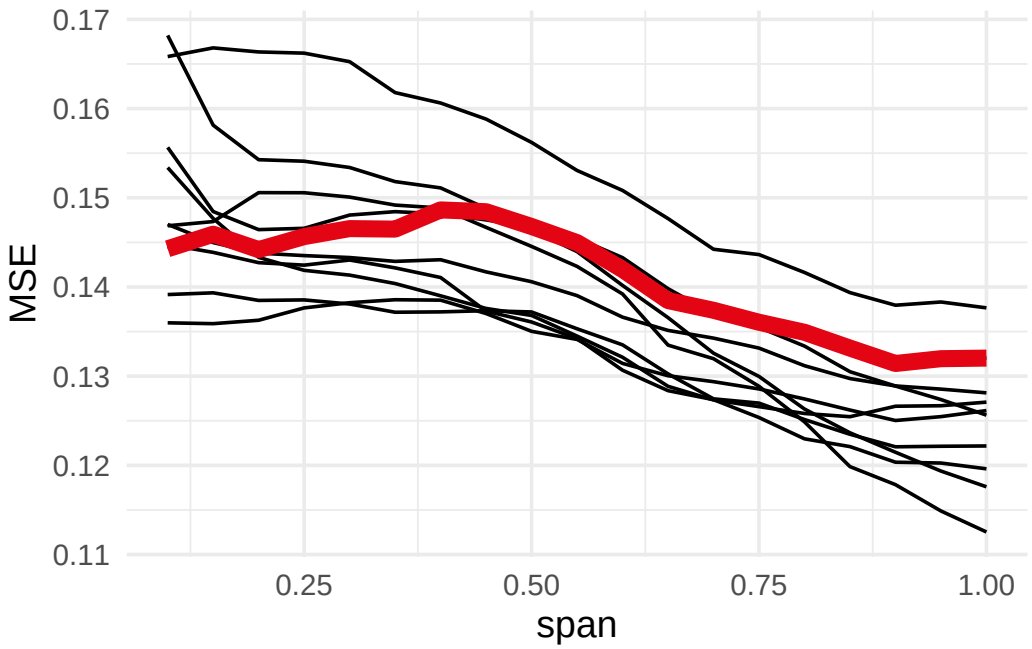

FIGURE 3 : MSE pour le jeu d'entraînement / validation précédent (en rouge). Les courbes grises sont les MSE pour d'autres découpages.

Comment choisir la taille des sous-ensembles ?

En général :

- Si l'on dispose d'un grand nombre d'observations (disons plusieurs milliers), on peut faire une division $50 - 50$.
- Si l'on dispose de moins d'observations, on préférera garder plus d'observations pour l'entraînement. On peut, par exemple, faire une division $70 - 30$ ou $80 - 20$.

Cependant, il n'existe pas de règle universelle. Le bon choix dépend du contexte, de la complexité du modèle et de la quantité de données disponibles.

La méthode a cependant deux inconvénients. Le premier est que l'estimation de l'erreur est instable. En effet, la valeur de l'erreur de prédiction dépend des observations qui sont dans le jeu de validation. Un autre jeu de validation peut donner un résultat différent. Le deuxième est qu'il y a moins de données pour ajuster le modèle. Comme une partie des données est réservée à la validation, le modèle est appris sur un ensemble plus petit, et cela peut donc surestimer son erreur réelle par rapport à s'il avait été appris sur l'ensemble de données complet.

3 Validation croisée

Pour contourner les limites de l'approche précédente, on utilise souvent la **validation croisée** (*cross-validation*). Cette méthode est plus robuste et plus stable. Le principe est de répéter l'approche par jeu de validation plusieurs fois sur différents sous-ensembles du jeu de données.

L'approche consiste à découper aléatoirement l'ensemble des observations en K sous-ensembles de taille équivalentes (appelés *folds*). Le premier *fold* est utilisé comme jeu de données de validation et le modèle est ajusté sur les $K-1$ *folds* restant. L'erreur de prédiction est calculé sur le premier *fold*. Cette procédure est répété K fois ; à chaque fois, un différent *fold* est utilisé comme jeu de données de validation. À la fin, on a donc K valeurs pour l'erreur de prédiction. On calcule enfin la moyenne des K valeurs de prédiction. La Figure 4 présente un schéma de cette approche.

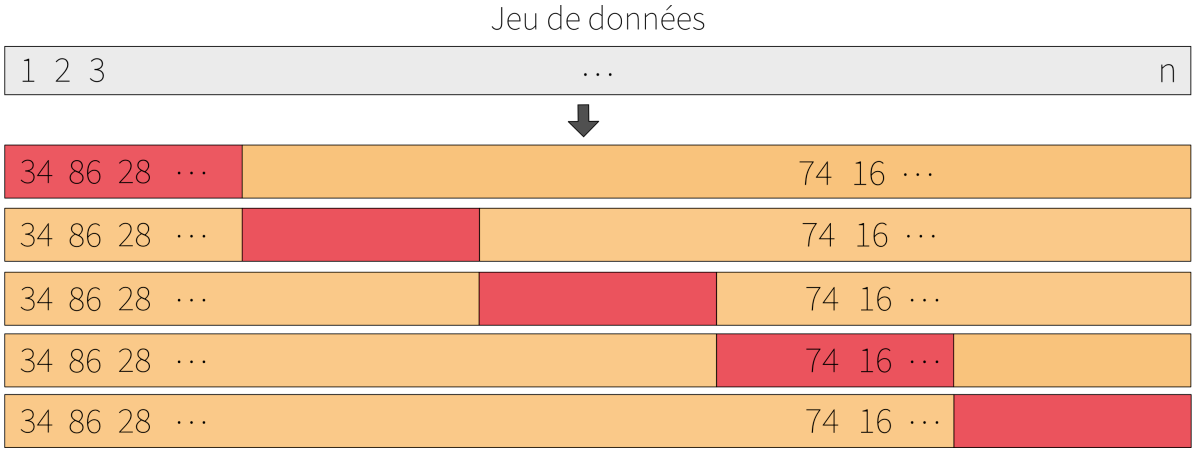

FIGURE 4 : Schéma de l'approche par validation croisée

Comment choisir le nombre de sous-ensembles K ?

Le choix du nombre de sous-ensembles K a un impact sur la qualité de l'estimation de l'erreur de prédiction, ainsi que sur le coût computationnel de la procédure. En pratique, on utilise souvent $K = 5$ ou $K = 10$. Ce choix repose sur un compromis entre la précision de l'estimation de l'erreur et le temps de calcul nécessaire. En effet, pour chaque valeur de K, le modèle est ajusté K fois. Par conséquent, plus K est grand, plus le coût computationnel augmente.
Dans le cas limite où $K = n$, i.e. K est égal au nombre d'observations dans le jeu de données, on parle de validation croisée *leave-one-out* (LOOCV). Dans ce cas, chaque observation sert une fois de validation et le modèle est entraîner sur les $n-1$ autres observations.
Ce choix de $K = n$ minimise le biais dans l'estimation de l'erreur de prédiction, car

à chaque itération, le modèle est ajusté sur presque toutes les observations du jeu de données. Cependant, cela se fait au prix d'une forte variance. En effet, comme les ensembles d'entraînement sont presque identiques, les erreurs de prédiction sont très corrélées entre elles, ce qui rend l'estimation globale de l'erreur instable.
Inversement, des valeurs plus faibles de K introduisent un léger biais dans l'estimation de l'erreur (car les modèles sont ajustés sur des ensembles contenant moins d'observations), mais réduisent la variance de cette estimation. Ce compromis biais/variance, couplé à une réduction significative du temps de calcul, explique pourquo $K = 5$ ou $K = 10$ sont des choix standards en pratique.

Pour finir, la validation croisée est une méthode générale qui peut être appliquée avec la plupart des modèles.

Références

James, Gareth, Daniela Witten, Trevor Hastie, et Robert Tibshirani. 2021. *An Introduction to Statistical Learning : With Applications in R*. Springer Texts in Statistics. New York, NY : Springer US. https://doi.org/10.1007/978-1-0716-1418-1.