# Apprentissage automatique (Machine learning)

Évaluation et validation

Visseho Adjiwanou, PhD.

13 June 2024

## Introduction

La section précédente nous a présenté une variété de méthodes, chacune ayant ses avantages et ses inconvénients, et aucune méthode n'est garantie de surpasser les autres pour un problème donné. Cette section se concentre sur les méthodes d'évaluation, avec trois objectifs principaux.

1. Sélection du modèle : Comment sélectionner une méthode à utiliser à l'avenir ? Quels paramètres devons-nous choisir pour cette méthode ?
2. Estimation des performances : Comment estimer la performance de notre modèle une fois qu'il est déployé et appliqué à de nouvelles données ?
3. Compréhension : Une compréhension plus approfondie des types de modèles qui fonctionnent bien, et de ceux qui ne fonctionnent pas, peut indiquer l'efficacité et l'applicabilité des méthodes existantes et fournir une meilleure compréhension de la structure des données et du problème que nous abordons. Cette section couvrira les méthodologies d'évaluation ainsi que les métriques couramment utilisées.

## Méthodologie

### Évaluation sur échantillon d'entraînement (In-sample evaluation)

En tant que scientifiques sociaux, nous évaluons déjà les méthodes en fonction de leur performance sur l'échantillon d'entraînement (sur l'ensemble de données pour lequel le modèle a été entraîné). Comme nous l'avons mentionné précédemment dans le chapitre, le but des méthodes d'apprentissage automatique est de généraliser à de nouvelles données, et valider les modèles sur l'échantillon d'entraînement ne nous permet pas de le faire. Nous nous concentrons ici sur les méthodologies d'évaluation qui nous permettent d'optimiser (autant que possible) les performances de généralisation. Les méthodes sont illustrées à la Figure 7.7.

### Échantillon hors-échantillon et ensemble de validation (Out-of-sample and hold-out set)

La manière la plus simple de se concentrer sur la généralisation est de faire semblant de généraliser à de nouvelles données (invisibles). Une façon de le faire est de prendre les données originales et de les diviser aléatoirement en deux ensembles : un ensemble d'entraînement et un ensemble de test (parfois aussi appelé ensemble de validation). Nous pouvons décider de la proportion de chaque ensemble (généralement les divisions vont de 50-50 à 80-20, selon la taille de l'ensemble de données). Nous entraînons ensuite nos modèles sur l'ensemble d'entraînement et classifions les données dans l'ensemble de test, ce qui nous permet d'obtenir une estimation de la performance relative des méthodes.

Un inconvénient de cette approche est que nous pouvons être extrêmement chanceux ou malchanceux avec notre division aléatoire. Une façon de contourner ce problème est de créer plusieurs ensembles d'entraînement et de test de manière répétée. Nous pouvons alors entraîner sur $TR_1$ et tester sur $TE_1$, entraîner sur $TR_2$ et tester sur $TE_2$, et ainsi de suite. Les mesures de performance sur chaque ensemble de test peuvent ensuite nous donner une estimation de la performance des différentes méthodes et de la variation de cette performance à travers différents ensembles aléatoires.

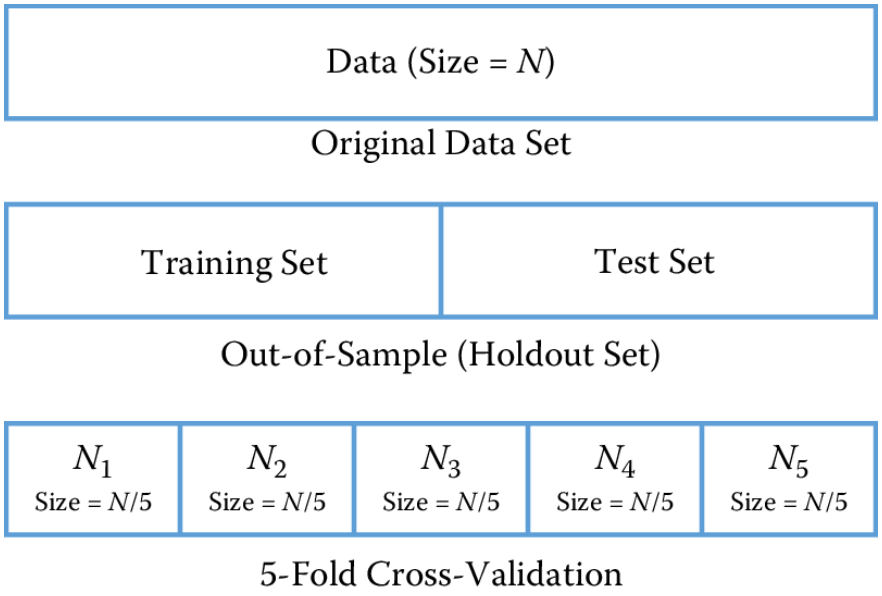


Figure 1: Validation methodologies: holdout set and cross-validation

### Validation croisée (Cross-validation)

La validation croisée est une procédure d'entraînement et de test par échantillon de validation plus sophistiquée qui corrige certains des inconvénients de l'approche par échantillon de validation. La validation croisée commence par diviser un ensemble de données étiqueté en $k$ partitions (appelées plis). En général, $k$ est fixé à 5 ou 10. La validation croisée procède

ensuite par itérations $k$ fois. À chaque itération, l'un des $k$ plis est mis de côté comme ensemble de test, tandis que les $k-1$ autres plis sont combinés et utilisés pour entraîner le modèle. Une propriété intéressante de la validation croisée est que chaque exemple est utilisé dans un ensemble de test pour tester le modèle. Chaque itération de la validation croisée nous donne une estimation de la performance qui peut ensuite être agrégée (généralement moyennée) pour générer l'estimation globale.

Un cas extrême de la validation croisée est appelé validation croisée par "leave-one-out", où, pour un ensemble de données de taille $N$, nous créons $N$ plis. Cela signifie itérer sur chaque point de données, le mettant de côté comme ensemble de test, et s'entraîner sur les $N-1$ exemples restants. Cela illustre l'avantage de la validation croisée en nous donnant de bonnes estimations de généralisation (en s'entraînant sur la plus grande partie possible de l'ensemble de données) et en s'assurant que le modèle est testé sur chaque point de données.

### Validation temporelle (Temporal validation)

Les approches de validation croisée et d'échantillon de validation décrites ci-dessus supposent que les données n'ont pas de dépendances temporelles et que la distribution est stationnaire dans le temps. Cette hypothèse est presque toujours violée en pratique et affecte les estimations de performance d'un modèle.

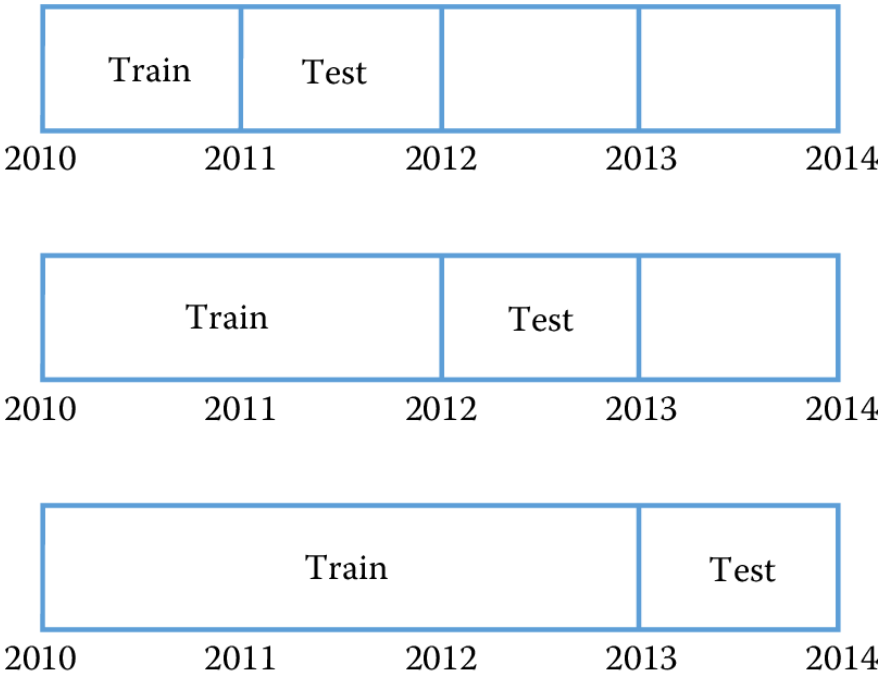


Figure 2: Temporal validation

Dans la plupart des problèmes pratiques, nous voulons utiliser une stratégie de validation qui émule la manière dont nos modèles seront utilisés et fournit une estimation précise de la performance. Nous appellerons cela validation temporelle. Pour un point donné dans le

temps $t_i$, nous entraînons nos modèles uniquement sur les informations disponibles avant $t_i$ pour éviter de s'entraîner sur des données provenant du "futur". Nous prédisons ensuite et évaluons sur les données de $t_i$ à $t_i + d$ et nous itérons, en élargissant la fenêtre d'entraînement tout en gardant la taille de la fenêtre de test constante à $d$.

La Figure (**ref?**)(fig) montre ce processus de validation avec $t_i = 2010$ et $d = 1$ an. La fenêtre de l'ensemble de test $d$ dépend de plusieurs facteurs liés à la manière dont le modèle sera déployé pour émuler au mieux la réalité :

1. À quel moment dans le futur les prédictions doivent-elles être faites ? Par exemple, si l'ensemble des étudiants qui doivent être ciblés pour des interventions doit être finalisé au début de l'année scolaire pour l'année entière, alors $d = 1$ an.
2. À quelle fréquence le modèle sera-t-il mis à jour ? Si le modèle est mis à jour quotidiennement, nous pouvons déplacer la fenêtre d'un jour à la fois pour refléter le scénario de déploiement.
3. À quelle fréquence le système recevra-t-il de nouvelles données ? Si nous recevons fréquemment de nouvelles données, nous pouvons faire des prédictions plus fréquemment.

La validation temporelle est similaire à la manière dont les modèles de séries temporelles sont évalués (également connue sous le nom de rétrovalidation) et devrait être l'approche de validation utilisée pour la plupart des problèmes pratiques.

## Métriques (Metrics)

Le sous-section précédent était axé sur les méthodologies de validation en supposant que nous avons une métrique d'évaluation en tête. Cette section va passer en revue les métriques d'évaluation couramment utilisées. Vous êtes probablement familier avec l'utilisation du coefficient de détermination $R^2$, l'analyse des résidus et l'erreur quadratique moyenne (MSE) pour évaluer la qualité des modèles de régression. Pour les problèmes de régression, le MSE calcule les différences quadratiques moyennes entre les prédictions $\hat{y}_i$ et les valeurs réelles $y_i$. Lorsque les modèles de prédiction ont un MSE plus petit, ils sont meilleurs. Cependant, le MSE lui-même est difficile à interpréter car il mesure les différences quadratiques. À la place, l'erreur quadratique moyenne racine (RMSE) est plus intuitive car elle mesure les différences moyennes à l'échelle d'origine de la variable réponse. Une autre alternative est l'erreur moyenne absolue (MAE), qui mesure les distances moyennes absolues entre les prédictions et les valeurs réelles.

Nous allons maintenant décrire quelques autres métriques d'évaluation couramment utilisées en apprentissage automatique pour la classification. Avant d'aborder les métriques, il est important de souligner que les modèles d'apprentissage automatique pour la classification ne prédisent généralement pas directement des valeurs 0/1. Les SVM, les forêts aléatoires et la régression logistique produisent tous un score (parfois une probabilité) qui est ensuite

transformé en 0 ou 1 en fonction d'un seuil spécifique à l'utilisateur. Vous pourriez constater que certains outils (comme scikit-learn[1]) utilisent une valeur par défaut pour ce seuil (souvent 0,5), mais il est important de savoir que c'est un seuil arbitraire et vous devriez choisir le seuil en fonction des données, du modèle et du problème que vous résolvez. Nous aborderons cela un peu plus tard dans cette section.

Une fois que nous avons transformé les prédictions en valeurs de classification 0/1 réelles, nous pouvons maintenant créer une matrice de confusion à partir de ces prédictions, comme illustré dans la Figure (**ref?**)(fig). Chaque point de données appartient soit à la classe positive soit à la classe négative, et pour chaque point de données, la prédiction du classificateur est correcte ou incorrecte. C'est ce que représentent les quatre cellules de la matrice de confusion. Nous pouvons utiliser la matrice de confusion pour décrire plusieurs métriques d'évaluation couramment utilisées.

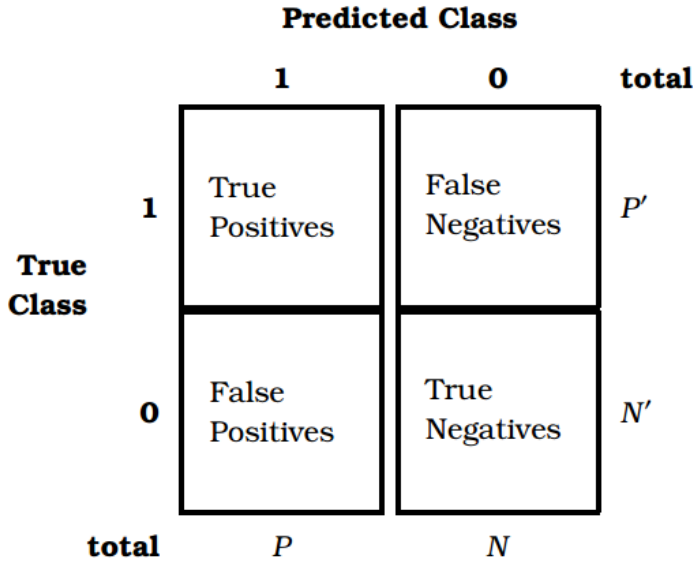


Figure 3: A *confusion matrix* created from real-valued predictions

La précision est le rapport des prédictions correctes (à la fois positives et négatives) à l'ensemble des prédictions :

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} = \frac{TP + TN}{P + N} = \frac{TP + TN}{P' + N'},$$

où $TP$ désigne les vrais positifs, $TN$ les vrais négatifs, $FP$ les faux positifs, $FN$ les faux négatifs, et les autres symboles représentent les totaux par ligne ou par colonne. La précision est la métrique d'évaluation la plus couramment décrite pour la classification, mais elle est étonnamment la moins utile dans les situations pratiques (du moins par elle-même). Un problème avec la précision est qu'elle ne nous donne pas une idée de l'amélioration par

[1]Vous ne devriez jamais utiliser la fonction de prédiction dans scikit-learn car elle suppose un seuil de 0,5.

rapport à la base de référence. Par exemple, dans un problème de classification où 95 % des données sont positives et 5 % sont négatives, un classificateur avec 85 % de précision se comporte moins bien qu'un classificateur stupide qui prédit toujours positif (et aurait une précision de 95 %).

Deux métriques supplémentaires souvent utilisées sont la précision (precision) et le rappel (recall), définies comme suit :

$$\text{Precision} = \frac{TP}{TP + FP} = \frac{TP}{P},$$
$$\text{Recall} = \frac{TP}{TP + FN} = \frac{TP}{P'}$$

La précision mesure l'exactitude du classificateur lorsqu'il prédit qu'un exemple est positif. C'est le rapport des exemples positifs correctement prédits ($TP$) sur tous les exemples prédits comme positifs ($TP + FP$). Cette mesure est également appelée valeur prédictive positive dans d'autres domaines. Le rappel mesure la capacité du classificateur à trouver des exemples positifs. C'est le rapport de tous les exemples positifs correctement prédits ($TP$) sur tous les exemples positifs dans les données ($TP + FN$). On l'appelle aussi sensibilité dans d'autres domaines.

Vous avez peut-être rencontré une autre mesure appelée spécificité dans d'autres domaines. Cette mesure est le taux de vrais négatifs : la proportion de négatifs correctement identifiés.

Une autre mesure utilisée est le score $F_1$, qui est la moyenne harmonique de la précision et du rappel :

$$F_1 = \frac{2 * \text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}}$$

C'est souvent utilisé lorsque vous voulez équilibrer à la fois la précision et le rappel.

Il y a souvent un compromis entre la précision et le rappel. En sélectionnant différents seuils de classification, nous pouvons faire varier et ajuster la précision et le rappel d'un classifieur donné. Un classifieur très conservateur qui prédit seulement un 1 lorsqu'il est absolument certain (par exemple, un seuil de 0,9999) sera le plus souvent correct lorsqu'il prédit un 1 (haute précision) mais manquera la plupart des 1 (faible rappel). À l'autre extrême, un classifieur qui prédit un 1 pour chaque point de données (un seuil de 0,0001) aura un rappel parfait mais une précision faible. La figure (**ref?**)(fig) montre une courbe précision-rappel qui est souvent utilisée pour représenter la performance d'un classifieur donné.

Si nous nous soucions d'optimiser l'ensemble de l'espace précision-rappel, une métrique utile est l'aire sous la courbe (AUC-PR), qui est l'aire sous la courbe précision-rappel. L'AUC-PR ne doit pas être confondue avec l'AUC-ROC, qui est l'aire sous la courbe caractéristique de fonctionnement du récepteur (ROC) associée. La courbe ROC est créée en traçant le rappel par rapport à (1 – spécificité). Les deux AUC peuvent être des métriques utiles pour comparer la performance de différentes méthodes et la valeur maximale que l'AUC peut atteindre est 1. Cependant, si nous nous intéressons à une partie spécifique de la courbe précision-rappel, nous devons examiner des métriques plus fines.

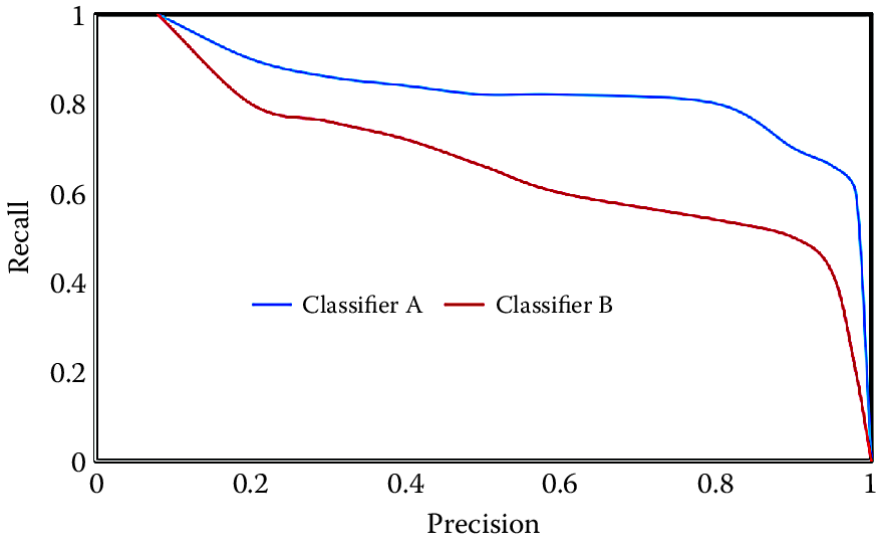


Figure 4: Precision–recall curve

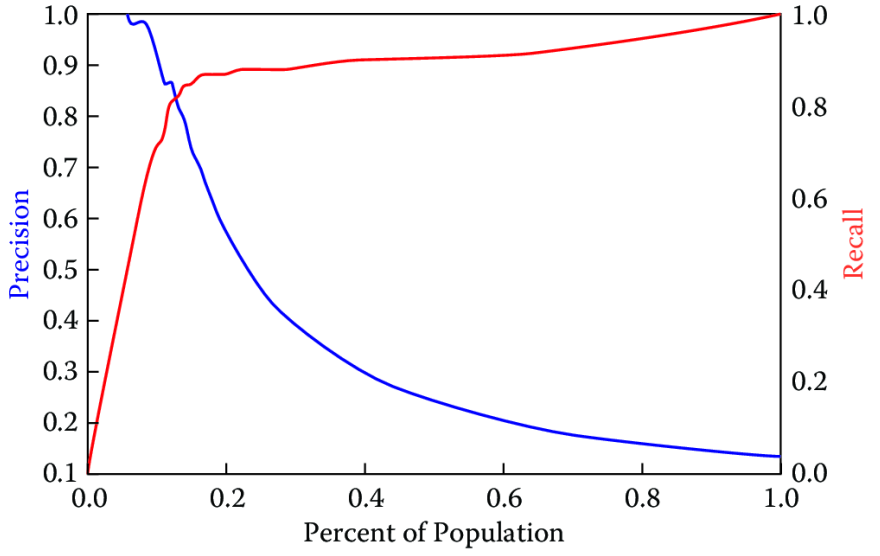


Figure 5: Precision or recall at different thresholds

Prenons un exemple dans le domaine de la santé publique. La plupart des agences de santé publique effectuent des inspections de divers types pour détecter les violations des normes sanitaires (par exemple, les dangers du plomb). Le nombre de lieux potentiels (domiciles ou entreprises) à inspecter dépasse largement les ressources d'inspection généralement disponibles. Supposons en outre qu'ils ne peuvent inspecter que 5 % de tous les lieux possibles ; ils souhaiteraient clairement prioriser l'inspection des endroits les plus susceptibles de contenir le danger. Dans ce cas, le modèle attribuera des scores et classera tous les lieux d'inspection possibles en fonction du risque de danger. Nous voudrions alors savoir quel pourcentage des 5 % supérieurs (ceux qui seront inspectés) sont susceptibles de présenter un danger, ce qui correspond à la précision dans les 5 % supérieurs des prédictions les plus confiantes — la précision à 5 %, comme on l'appelle couramment (voir Figure (**ref?**)(fig )). La précision au top k pourcentage est une classe courante de métriques largement utilisée dans la littérature sur la recherche d'information et les moteurs de recherche, où l'objectif est de s'assurer que les résultats récupérés en tête des résultats de recherche sont précis. Plus généralement, cette métrique est souvent utilisée dans les problèmes où la distribution des classes est biaisée et où seul un petit pourcentage des exemples sera examiné manuellement (inspections, enquêtes pour fraude, etc.). La littérature offre de nombreuses études de cas de telles applications (**Kumar2010?**; **Lakkaraju2015?**; **Potash2015?**).

Une dernière métrique que nous voulons mentionner est une classe de métriques sensibles au coût où différents coûts (ou bénéfices) peuvent être associés aux différentes cellules de la matrice de confusion. Jusqu'à présent, nous avons implicitement supposé que chaque prédiction correcte et chaque erreur, que ce soit pour la classe positive ou négative, ont des coûts et des bénéfices égaux. Dans de nombreux problèmes pratiques, ce n'est pas le cas. Par exemple, nous pouvons vouloir prédire si un patient dans une salle d'urgence d'hôpital est susceptible de faire un arrêt cardiaque au cours des six prochaines heures. Le coût d'un faux positif dans ce cas est le coût de l'intervention (qui peut être quelques minutes supplémentaires du temps d'un médecin), tandis que le coût d'un faux négatif pourrait être la mort. Ce type d'analyse nous permet de calculer la valeur attendue des prédictions d'un classifieur et de sélectionner le modèle qui optimise cette métrique sensible au coût.