# Chapitre IV
# Validation et sélection
## Cycle pluridisciplinaire d'études supérieures
## Université Paris sciences et lettres

On introduit ci-après différents objets qui seront utilisés dans l'ensemble du chapitre. Soit $\mathcal{X}$ et $\mathcal{Y}$ des ensembles d'entrées et de sorties respectivement. Soit $n \geqslant 1$ et $S = (x_i, y_i)_{i\in[n]} \in \mathcal{S}(\mathcal{X}, \mathcal{Y})$ un échantillon quelconque.

Notation. — On note $S = S_1 \sqcup S_2$ s'il existe $I_1$ et $I_2$ sous-ensembles de $[n]$ tels que :

$$S_1 = (x_i, y_i)_{i\in I_1}, \quad S_2 = (x_i, y_i)_{i\in I_2} \quad \text{et} \quad [n] = I_1 \sqcup I_2,$$

où $\sqcup$ désigne l'union disjointe.

Soit $\ell : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}$ une fonction de perte et P une distribution de probabilité sur $\mathcal{X} \times \mathcal{Y}$. Pour tout prédicteur $f \in \mathcal{F}(\mathcal{X}, \mathcal{Y})$, on rappelle que son *risque* est défini par :

$$R(f) = \mathbb{E}_{(X,Y)\sim P}\left[\ell(Y, f(X))\right].$$

Notation. — Pour $S_0 = (x_i, y_i)_{i\in[n_0]} \in \mathcal{S}(\mathcal{X}, \mathcal{Y})$ échantillon quelconque, on note $R_{S_0}(f)$ le *risque empirique* de $f$ sur $S_0$ :

$$R_{S_0}(f) = \frac{1}{n_0}\sum_{i=1}^{n_0} \ell(y_i, f(x_i)).$$

## 1. VALIDATION SIMPLE

On présente dans ce paragraphe la procédure de validation simple qui permet de *construire un prédicteur* à l'aide d'un algorithme, puis d'*estimer le risque du prédicteur obtenu*. Elle consiste à séparer l'échantillon initial en deux échantillons distincts, l'un appelé *échantillon d'apprentissage* et l'autre *échantillon de validation* [1], puis à utiliser l'algorithme avec l'échantillon d'apprentissage pour obtenir un prédicteur, et enfin à calculer le risque empirique du prédicteur sur l'échantillon de validation (aussi appelé erreur de validation), ce risque empirique étant alors un estimateur du véritable risque. On pourra également calculer le risque empirique sur l'échantillon d'apprentissage (aussi appelé erreur d'apprentissage), dont la comparaison avec l'erreur de validation permettra (voir plus bas) de détecter les situations de sur- ou sous-apprentissage. On formalise ci-après cette démarche.

Soit $A \in \mathcal{A}(\mathcal{X}, \mathcal{Y})$ un algorithme d'apprentissage et $S = S_{\text{train}} \sqcup S_{\text{valid}}$ une partition de l'échantillon initial où $S_{\text{train}}$ et $S_{\text{valid}}$ désignent respectivement l'échantillon d'apprentissage et de validation. La procédure de *validation simple* s'écrit alors comme suit.

1) Calculer le prédicteur $\widehat{f} = A(S_{\text{train}})$.
2) Calculer les erreurs d'apprentissage et de validation :

$$\varepsilon_{\text{train}} = R_{S_{\text{train}}}(\widehat{f}), \qquad \varepsilon_{\text{valid}} = R_{S_{\text{valid}}}(\widehat{f}).$$

REMARQUE. — Si on suppose que les exemples de l'échantillon initial $S = (x_i, y_i)_{i\in[n]}$ sont tirés de façon i.i.d. selon la distribution P, alors l'erreur de validation $\varepsilon_{\text{valid}}$ est effectivement un estimateur du risque $R(\widehat{f})$.

## 2. SÉLECTION DE PRÉDICTEUR PAR VALIDATION

En s'appuyant sur la procédure de validation simple décrite ci-dessus, on peut élaborer une démarche qui permet de choisir, parmi plusieurs prédicteurs obtenus à l'aide d'algorithmes différents, celui qui semble être le meilleur. On partitionne cette fois-ci l'échantillon initial en trois sous-échantillons. Le premier est utilisé pour construire les prédicteurs à l'aide des différents algorithmes, le

1. Dans ce contexte, *validation* est synonyme de *test*.

second permet de sélectionner, parmi les prédicteurs obtenus, celui qui semble être le meilleur, et enfin le troisième est utilisé pour estimer le risque du prédicteur sélectionné. La démarche est formalisée ci-après.

Soit $M \geqslant 2$ un entier et $A^{(1)}, A^{(2)}, \dots, A^{(M)} \in \mathscr{A}(\mathscr{X}, \mathscr{Y})$ des algorithmes d'apprentissage. Soit $S = S_{train} \sqcup S_{valid} \sqcup S_{test}$ une partition de l'échantillon initial en trois échantillons, respectivement dits d'apprentissage, de validation et de test. La procédure de sélection par validation s'écrit alors comme suit. Pour chaque $m \in [M]$,

1) calculer le prédicteur $\widehat{f}^{(m)} = A^{(m)}(S_{train})$ ;
2) calculer les erreurs d'apprentissage et de validation :

$$\varepsilon_{train}^{(m)} = R_{S_{train}}(\widehat{f}^{(m)}), \qquad \varepsilon_{valid}^{(m)} = R_{S_{valid}}(\widehat{f}^{(m)}).$$

Ensuite,

3) sélectionner $m_* = \operatorname*{arg\,min}_{m \in [M]} \varepsilon_{valid}^{(m)}$ (on sélectionne le prédicteur $\widehat{f}^{(m_*)}$, qui est par définition de $m_*$ celui dont l'erreur de validation est la plus faible) ;
4) calculer l'erreur de test $\varepsilon_{test} = R_{S_{test}}(\widehat{f}^{(m_*)})$.

Remarque. — Le calcul de l'erreur de test sert à estimer le risque du prédicteur sélectionné, mais il n'est pas indispensable.

Remarque. — Il est important que le calcul de l'erreur de test ne se fasse pas sur l'échantillon de validation, mais bien sur un échantillon de test indépendant. En effet, l'erreur de validation du prédicteur sélectionné aura tendance à sous-estimer le risque, puisque le prédicteur sélectionné aura pu avoir l'erreur de validation la plus faible grâce des exemples de l'échantillon de validation qui lui sont particulièrement favorables.

## 3. Validation croisée

On présente dans ce paragraphe une alternative à la validation simple, qui permet de mieux estimer le risque des prédicteurs construits à l'aide d'un (unique) algorithme donné. La validation simple se donne une unique partition de l'échantillon initial (en deux sous-échantillons), tandis que la *validation croisée* (ou *cross validation*, souvent abrégée en CV) effectue plusieurs validations simples en faisant varier les échantillons d'apprentissage et de validation, et considère ensuite

la moyenne des erreurs de validation obtenues. On utilise ainsi plusieurs fois les mêmes données, mais de façon différente. La validation croisée permet une meilleure estimation du risque, surtout lorsque l'échantillon initial est petit.

Soit $2 \leqslant k \leqslant n$ un entier, $A \in \mathscr{A}(\mathscr{X}, \mathscr{Y})$ un algorithme d'apprentissage et

$$S = S_1 \sqcup S_2 \sqcup \cdots \sqcup S_k$$

une partition de l'échantillon initial S en $k$ sous-échantillons (typiquement, on choisira des sous-échantillons $(S_j)_{j\in[k]}$ de tailles similaires). Alors, la procédure de $k$-validation croisée associée s'écrit comme suit. Pour tout $j \in [k]$,

1) considérer l'échantillon $S'_j = \bigcup_{j'\in[k]\setminus\{j\}} S_{j'}$ (ce qui implique $S = S_j \sqcup S'_j$) ;
2) calculer le prédicteur $\hat{f}_j = A(S'_j)$ ;
3) calculer les erreurs d'apprentissage et de validation de $\hat{f}_j$ :

$$\varepsilon_{\text{train},j} = R_{S'_j}(\hat{f}_j), \qquad \varepsilon_{\text{valid},j} = R_{S_j}(\hat{f}_j).$$

Autrement dit, pour chaque $j \in [k]$, on construit un prédicteur $\hat{f}_j$ en utilisant $S'_j$ comme échantillon d'apprentissage et on estime son risque en utilisant $S_j$ comme échantillon de validation. Ces échantillons sont représentés sur la Figure 1. Puis,

4) calculer les erreurs moyennes d'apprentissage et de validation :

$$\varepsilon_{\text{train,CV}} = \frac{1}{k}\sum_{j=1}^{k} \varepsilon_{\text{train},j}, \qquad \varepsilon_{\text{valid,CV}} = \frac{1}{k}\sum_{j=1}^{k} \varepsilon_{\text{valid},j}.$$

Remarque. — Si le paramètre $k$ est choisi trop petit, on perd le bénéfice de la validation croisée et on se rapproche de la validation simple. À l'inverse, si le paramètre $k$ est trop grand, la taille des échantillons de validation devient petite, ce qui dégrade la qualité d'estimation de l'erreur de validation. Dans la pratique, on choisira des valeurs de $k$ comprises entre 5 et 10.

## 4. Sélection d'algorithme par validation croisée

En s'appuyant sur la validation croisée, on peut définir une procédure de sélection d'algorithme similaire à la sélection par validation présentée plus haut.

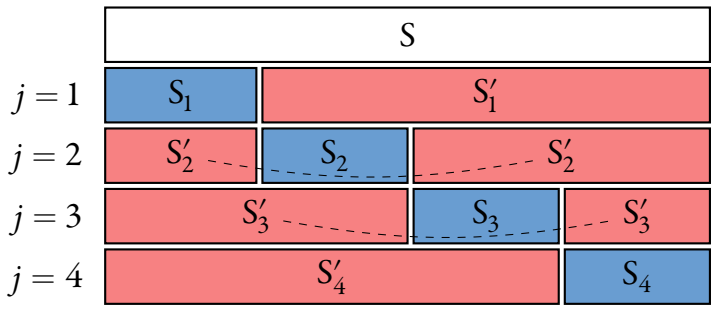



FIGURE 1. — Illustration pour $k = 4$ des différents échantillons d'apprentissage et de validation utilisés lors de la validation croisée. Sont représentés en rose les échantillons d'apprentissage et en bleu les échantillons de validation. Les arcs en pointillés relient deux parties d'un même échantillon.

Soit $M \geqslant 2$ un entier et $A^{(1)}, A^{(2)}, \dots, A^{(M)} \in \mathscr{A}(\mathscr{X}, \mathscr{Y})$ des algorithmes d'apprentissage. Soit une partition de l'échantillon initial en deux échantillons :

$$S = S_{\text{train}} \sqcup S_{\text{test}}.$$

Soit $2 \leqslant k \leqslant n$ un entier. La procédure de sélection par $k$-validation croisée s'écrit comme suit.

1) Pour chaque $m \in [M]$, calculer les erreurs d'apprentissage et de validation (notées respectivement $\varepsilon^{(m)}_{\text{train,CV}}$ et $\varepsilon^{(m)}_{\text{valid,CV}}$) de l'algorithme $A^{(m)}$ données par la $k$-validation croisée opérée sur l'échantillon $S_{\text{train}}$.
2) Sélectionner $m_* = \underset{m \in [M]}{\arg\min}\, \varepsilon^{(m)}_{\text{valid,CV}}$.
3) Calculer le prédicteur final $\widehat{f}_{\text{final}} = A^{(m_*)}(S_{\text{train}})$.
4) Calculer l'erreur de test $\varepsilon_{test} = R_{S_{\text{test}}}(\widehat{f}_{\text{final}})$.

## 5. COMPROMIS ENTRE BIAIS ET COMPLEXITÉ

Il est très fréquent que les algorithmes d'apprentissage comportent un ou plusieurs *hyperparamètres* à choisir [2]. Par exemple, l'entier $k$ de l'algorithme des

2. Les paramètres désignent des quantités qui caractérisent les prédicteurs d'une classe donnée : par exemple, la classe $\mathscr{L}_d = \{g_{w,b}\}_{\substack{w \in \mathbb{R}^d \\ b \in \mathbb{R}}}$ est paramétrisée par les *paramètres* $w \in \mathbb{R}^d$ et $b \in \mathbb{R}$. Les *hyperparamètres* désignent des quantités dont dépendent les *algorithmes*.

$k$ plus proches voisins est un hyperparamètre ; ou encore en régression polynomiale, le degré maximal des polynômes considérés est un hyperparamètre. Les hyperparamètres influent souvent sur le *biais* et la *complexité* des prédicteurs obtenus. On donne ci-après une présentation informelle de ces deux notions. Le paragraphe suivant présentera la démarche à adopter pour choisir la valeur d'hyperparamètre correspondant au meilleur compromis entre biais et complexité.

NOTATION. — Dans le reste du chapitre, $\varepsilon_{\text{train}}$ et $\varepsilon_{\text{valid}}$ désigneront des erreurs d'apprentissage et de validation dans un contexte de validation simple ou croisée.

Le *biais* est la tendance d'un algorithme à donner des prédicteurs qui prédisent mal sur les données d'apprentissage et de validation (ce qui correspond à $\varepsilon_{\text{train}}$ et $\varepsilon_{\text{valid}}$ élevés). La *complexité*[3] est la tendance d'un algorithme à donner des prédicteurs dont les prédictions dépendent de façon complexe des entrées (voir exemple plus bas). On peut alors noter les phénomènes suivants.

- Un trop grand biais amène l'algorithme à ne pas saisir les informations importantes permettant une bonne prédiction. On parle alors de *sous-apprentissage* (underfitting).
- Une trop grande complexité amène l'algorithme à donner des prédicteurs qui suivent de trop près les particularités des exemples d'apprentissage ($\varepsilon_{\text{train}}$ petit), et donc à mal prédire sur de nouveaux exemples ($\varepsilon_{\text{valid}}$ grand). On parle alors de *sur-apprentissage* (overfitting).
- Un faible biais entraîne une grande complexité et vice-versa.
- Augmenter la taille de l'échantillon d'apprentissage diminue le sur-apprentissage.

EXEMPLE. — *Régression polynomiale de* $\mathbb{R}$ *dans* $\mathbb{R}$. — Soit $n \geqslant 1$. On considère $\mathcal{X} = \mathcal{Y} = \mathbb{R}$ et pour classe de prédicteurs $\mathcal{F}$ l'ensemble des fonctions polynomiales de $\mathbb{R} \to \mathbb{R}$ de degré au plus $n$. Soit $S = (x_i, y_i)_{i\in[n]} \in \mathcal{S}(\mathbb{R}, \mathbb{R})$ un échantillon d'apprentissage. Un algorithme possible est la régression aux moindres de carrés :

$$\hat{f} = \operatorname*{arg\,min}_{f\in\mathcal{F}} \left\{ \frac{1}{n} \sum_{i=1}^{n} (y_i - f(x_i))^2 \right\}.$$

L'entier $n$ est donc un hyperparamètre à choisir. Plus $n$ est grand, plus la complexité est élevée (les polynômes de haut degré sont des prédicteurs *complexes*), et plus le biais est fiable (car les polynômes de haut degré ont une grande capacité d'interpolation). À l'inverse, plus $n$ est petit, moins la complexité est élevée, et plus le biais est grand.

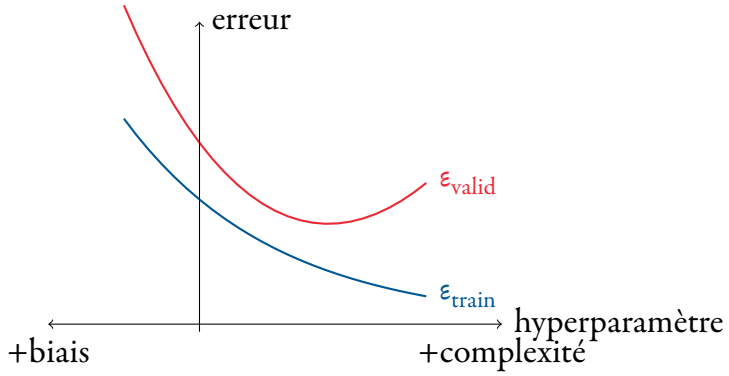
3. Cela ne désigne pas ici la complexité *algorithmique*.

La problématique du choix de l'hyperparamètre correspond donc souvent à trouver le meilleur compromis entre biais et complexité, pour lequel l'erreur de validation $\varepsilon_{\text{valid}}$ est minimal.

## 6. COURBES DE VALIDATION

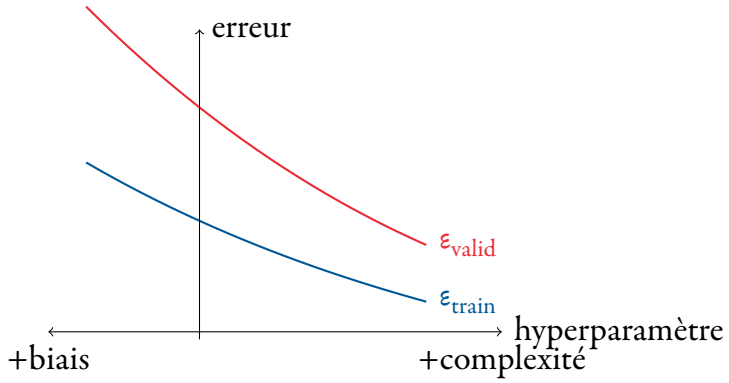
Lorsqu'on souhaite choisir parmi plusieurs valeurs pour un hyperparamètre contrôlant le compromis entre biais et complexité, il convient de tracer les courbes dites de validation : on représente $\varepsilon_{\text{train}}$ et $\varepsilon_{\text{valid}}$ en fonction de l'hyperparamètre. On peut distinguer trois cas de figure.

— Cas 1 (satisfaisant) :



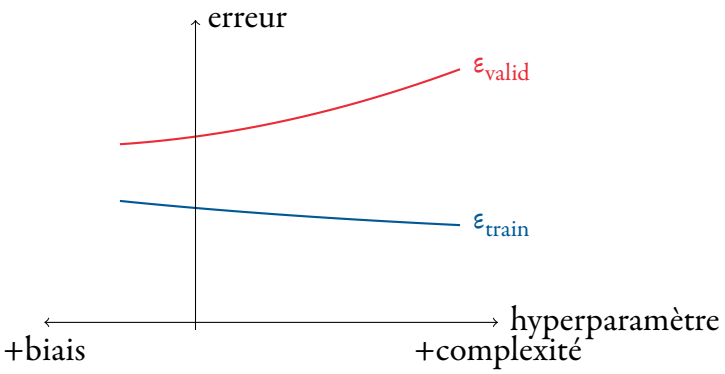

On peut observer une valeur de l'hyperparamètre qui semble minimiser l'erreur de validation. Il convient donc de choisir celle-ci.

— Cas 2 :




Cela correspond à une situation de sous-apprentissage. Il convient alors d'essayer des valeurs de l'hyperparamètre correspondant à une plus grande complexité.

— Cas 3 :

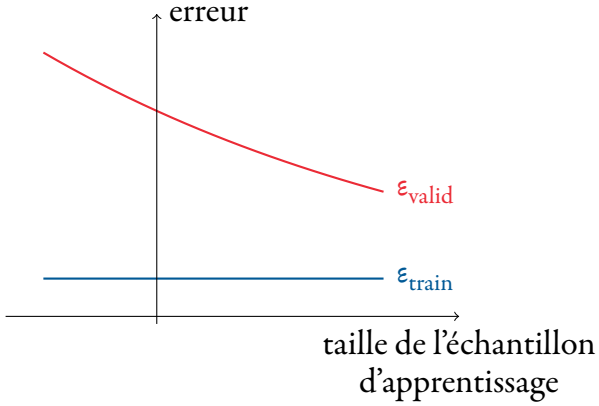



Cela correspond à une situation de sur-apprentissage. Il convient d'essayer des valeurs de l'hyperparamètre correspondant à un biais plus important.

## 7. COURBES D'APPRENTISSAGE

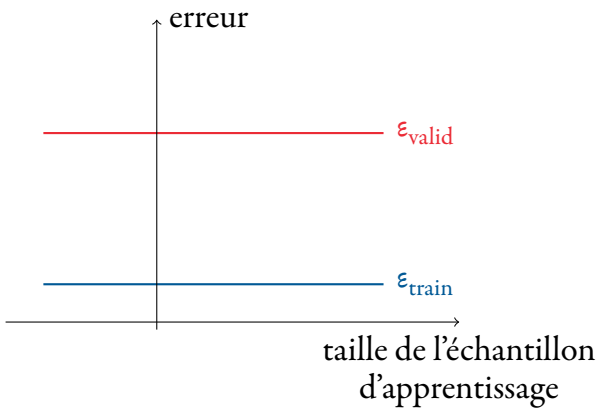
Pour savoir s'il est intéressant d'utiliser des échantillons d'apprentissage de plus grande taille, on peut tracer les *courbes d'apprentissage* : on représente $\varepsilon_{train}$ et $\varepsilon_{valid}$ en fonction de la taille de l'échantillon d'apprentissage. On distingue deux cas de figure.

— Cas 1 :




On est en situation de sur-apprentissage. Il est alors intéressant d'augmenter la taille de l'échantillon d'apprentissage.

— Cas 2 :




On est en situation de sous-apprentissage. Il est inutile d'augmenter la taille de l'échantillon d'apprentissage.