Overfitting (sur-apprentissage)

Jérémie Cabessa

Laboratoire DAVID, UVSQ

Formulation du problème (apprentissage supervisé)

- Soient $X_1, \ldots, X_p$ et Y des variables aléatoires.
- $X_1, \ldots, X_p$ sont appelées **variables d'inputs**, **variables indépendantes**, **variables explicatives**, **prédicteurs**, **(features)**.
- Y est appelée **variable d'output**, **variable dépendante**, **réponse**, **(response, target)**.

Formulation du problème (apprentissage supervisé)

- Soient $X_1, \ldots, X_p$ et Y des variables aléatoires.
- $X_1, \ldots, X_p$ sont appelées **variables d'inputs**, **variables indépendantes**, **variables explicatives**, **prédicteurs**, **(features)**.
- Y est appelée **variable d'output**, **variable dépendante**, **réponse**, **(response, target)**.

Formulation du problème (apprentissage supervisé)

- Soient $X_1, \ldots, X_p$ et Y des variables aléatoires.
- $X_1, \ldots, X_p$ sont appelées **variables d'inputs**, **variables indépendantes**, **variables explicatives**, **prédicteurs**, **(features)**.
- Y est appelée **variable d'output**, **variable dépendante**, **réponse**, **(response, target)**.

Formulation du problème (apprentissage supervisé)

- On suppose qu'il existe une (vraie) **relation** f entre $X_1, \ldots, X_p$ et Y de la forme

$$Y = f(X_1, \ldots, X_p) + \epsilon$$

où f est une fonction inconnue et ϵ est une variable aléatoire indépendante de $X_1, \ldots, X_p$ et de moyenne 0, le bruit.

- On aimerait apprendre une (bonne) **estimation** $\hat{f}$ de f. On aura alors

$$\hat{Y} = \hat{f}(X_1, \ldots, X_p)$$

où $\hat{f}$ est l'**estimation** de f et $\hat{Y}$ est la **prediction** de Y.

Formulation du problème (apprentissage supervisé)

- On suppose qu'il existe une (vraie) **relation** f entre $X_1, \ldots, X_p$ et Y de la forme

$$Y = f(X_1, \ldots, X_p) + \epsilon$$

où f est une fonction inconnue et ϵ est une variable aléatoire indépendante de $X_1, \ldots, X_p$ et de moyenne 0, le bruit.

- On aimerait apprendre une (bonne) **estimation** $\hat{f}$ de f. On aura alors

$$\hat{Y} = \hat{f}(X_1, \ldots, X_p)$$

où $\hat{f}$ est l'**estimation** de f et $\hat{Y}$ est la **prediction** de Y.

Formulation du problème (apprentissage supervisé)

- Pour apprendre l'estimation $\hat{f}$ de f, on dispose de données (data)

$$S_{\text{train}} = \{(\boldsymbol{x_1}, y_1), (\boldsymbol{x_2}, y_2), \ldots, (\boldsymbol{x_n}, y_n)\}$$

où $\boldsymbol{x_i} = (x_{i1}, \ldots, x_{i_p})$ pour tout $i = 1, \ldots, n$.

- Ces données constituent le "**training set**" (training data).

Formulation du problème (apprentissage supervisé)

- Pour apprendre l'estimation $\hat{f}$ de f, on dispose de données (data)

$$S_{\text{train}} = \{(\boldsymbol{x_1}, y_1), (\boldsymbol{x_2}, y_2), \ldots, (\boldsymbol{x_n}, y_n)\}$$

  où $\boldsymbol{x_i} = (x_{i1}, \ldots, x_{i_p})$ pour tout $i = 1, \ldots, n$.
- Ces données constituent le "**training set**" (training data).

Formulation du problème (apprentissage supervisé)

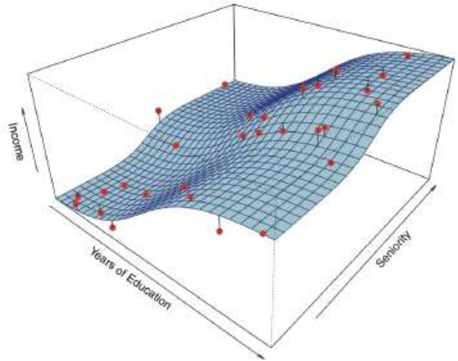

Figure taken from [James et al., 2013]

Erreur réductible et irréductible

- On a donc

$$\begin{aligned} Y &= f(X_1, \ldots, X_p) + \epsilon & \text{vraie relation} \\ \hat{Y} &= \hat{f}(X_1, \ldots, X_p) & \text{estimation} \end{aligned}$$

- On peut facilement montrer que

$$\begin{aligned} \mathrm{E}[Y - \hat{Y}]^2 &= \mathrm{E}[f(X_1, \ldots, X_p) + \epsilon - \hat{f}(X_1, \ldots, X_p)]^2 \\ &= \mathrm{E}[f(\boldsymbol{X}) + \epsilon - \hat{f}(\boldsymbol{X})]^2 \\ &= \underbrace{\left(f(\boldsymbol{X}) - \hat{f}(\boldsymbol{X})\right)^2}_{\text{erreur réductible}} + \underbrace{\mathrm{Var}(\epsilon)}_{\text{erreur irréductible}} \end{aligned}$$

où $\boldsymbol{X} = (X_1, \ldots, X_p)$.

Erreur réductible et irréductible

- On a donc

$$\begin{array}{rcll} Y & = & f(X_1,\ldots,X_p)+\epsilon & \text{vraie relation} \\ \hat{Y} & = & \hat{f}(X_1,\ldots,X_p) & \text{estimation} \end{array}$$

- On peut facilement montrer que

$$\begin{aligned} \mathrm{E}[Y-\hat{Y}]^2 &= \mathrm{E}[f(X_1,\ldots,X_p)+\epsilon-\hat{f}(X_1,\ldots,X_p)]^2 \\ &= \mathrm{E}[f(\boldsymbol{X})+\epsilon-\hat{f}(\boldsymbol{X})]^2 \\ &= \underbrace{\left(f(\boldsymbol{X})-\hat{f}(\boldsymbol{X})\right)^2}_{\text{erreur réductible}} + \underbrace{\mathrm{Var}(\epsilon)}_{\text{erreur irréductible}} \end{aligned}$$

où $\boldsymbol{X}=(X_1,\ldots,X_p)$.

Erreur réductible et irréductible

- **Erreur réductible (reducible error):** peut être réduite; plus $\hat{f}$ est une bonne estimation de f, plus cette erreur sera faible.
- **Erreur irréductible (irreducible error):** ne peut être réduite; par le bais de notre estimation $\hat{f}$, nous n'avons aucune prise sur le "bruit" réel intrinsèque au modèle.

Erreur réductible et irréductible

- **Erreur réductible (reducible error):** peut être réduite; plus $\hat{f}$ est une bonne estimation de f, plus cette erreur sera faible.
- **Erreur irréductible (irreducible error):** ne peut être réduite; par le bais de notre estimation $\hat{f}$, nous n'avons aucune prise sur le "bruit" réel intrinsèque au modèle.

Erreur réductible et irréductible

- Exemple de deux estimations $\hat{f}$. La deuxième estimation est meilleure car elle est associée à une erreur réductible plus faible.

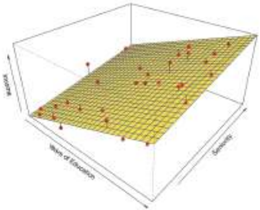

FIGURE 2.4. *A linear model fit by least squares to the* Income *data from Figure 2.3. The observations are shown in red, and the yellow plane indicates the least squares fit to the data.*

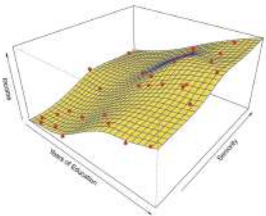

FIGURE 2.5. *A smooth thin-plate spline fit to the* Income *data from Figure 2.3 is shown in yellow; the observations are displayed in red. Splines are discussed in Chapter 7.*

Figure taken from [James et al., 2013]

Fonction de coût (cost function)

- Comment mesurer la qualité d'un modèle $\hat{f}$?
- On utilise une **fonction de coût (cost or loss function)**.
- La plus célèbre est **l'erreur des moindre carrés (mean squared error) MSE**. Étant donné un training set

$$S_{\text{train}} = \{(\boldsymbol{x_1}, y_1), (\boldsymbol{x_2}, y_2), \ldots, (\boldsymbol{x_n}, y_n)\}$$

on définit

$$\text{MSE}_{\text{train}} = \frac{1}{n} \sum_{i=1}^{n} \left(y_i - \hat{f}(\boldsymbol{x_i})\right)^2$$

Fonction de coût (cost function)

- Comment mesurer la qualité d'un modèle $\hat{f}$?
- On utilise une **fonction de coût (cost or loss function)**.
- La plus célèbre est **l'erreur des moindre carrés (mean squared error) MSE**. Étant donné un training set

$$S_{\text{train}} = \left\{(\boldsymbol{x_1}, y_1), (\boldsymbol{x_2}, y_2), \ldots, (\boldsymbol{x_n}, y_n)\right\}$$

on définit

$$\text{MSE}_{\text{train}} = \frac{1}{n}\sum_{i=1}^{n}\left(y_i - \hat{f}(\boldsymbol{x_i})\right)^2$$

Fonction de coût (cost function)

- Comment mesurer la qualité d'un modèle $\hat{f}$?
- On utilise une **fonction de coût (cost or loss function)**.
- La plus célèbre est **l'erreur des moindre carrés (mean squared error) MSE**. Étant donné un training set

$$S_{\text{train}} = \left\{ (\boldsymbol{x_1}, y_1), (\boldsymbol{x_2}, y_2), \ldots, (\boldsymbol{x_n}, y_n) \right\}$$

on définit

$$\text{MSE}_{\text{train}} = \frac{1}{n} \sum_{i=1}^{n} \left(y_i - \hat{f}(\boldsymbol{x_i}) \right)^2$$

Overfitting

- **Problème:** Le modèle $\hat{f}$ a été construit sur la base du training set S_{train}. Ainsi, il peut être très performant lorsqu'il est évalué sur le training set, mais bien moins performant lorsqu'il est évalué sur des données qu'il n'a jamais vues.
- Ainsi, il convient d'utiliser un **training set** S_{train} pour construire le modèle $\hat{f}$, et un autre ensemble de données disjoint, appelé **test set** S_{test}, pour évaluer la performance du modèle $\hat{f}$.
- Lorsque le modèle est performant sur le **training set** ($\text{MSE}_{\text{train}}$ basse), mais qu'il est bien moins performant sur sur le **test set** (MSE_{test} plus élevée), on est dans une situation d'**overfitting**.

Overfitting

- **Problème:** Le modèle $\hat{f}$ a été construit sur la base du training set S_{train}. Ainsi, il peut être très performant lorsqu'il est évalué sur le training set, mais bien moins performant lorsqu'il est évalué sur des données qu'il n'a jamais vues.
- Ainsi, il convient d'utiliser un **training set** S_{train} pour construire le modèle $\hat{f}$, et un autre ensemble de données disjoint, appelé **test set** S_{test}, pour évaluer la performance du modèle $\hat{f}$.
- Lorsque le modèle est performant sur le **training set** ($\text{MSE}_{\text{train}}$ basse), mais qu'il est bien moins performant sur sur le **test set** (MSE_{test} plus élevée), on est dans une situation d'**overfitting**.

Overfitting

- **Problème:** Le modèle $\hat{f}$ a été construit sur la base du training set S_{train}. Ainsi, il peut être très performant lorsqu'il est évalué sur le training set, mais bien moins performant lorsqu'il est évalué sur des données qu'il n'a jamais vues.
- Ainsi, il convient d'utiliser un **training set** S_{train} pour construire le modèle $\hat{f}$, et un autre ensemble de données disjoint, appelé **test set** S_{test}, pour évaluer la performance du modèle $\hat{f}$.
- Lorsque le modèle est performant sur le **training set** ($\text{MSE}_{\text{train}}$ basse), mais qu'il est bien moins performant sur sur le **test set** (MSE_{test} plus élevée), on est dans une situation d'**overfitting**.

Overfitting

- **Overfitting:** Le modèle est très performant sur le **training set** (i.e., $\mathrm{MSE}_{\mathrm{train}}$ basse) alors qu'il est bien moins performant sur sur le **test set** (i.e., $\mathrm{MSE}_{\mathrm{train}}$ élevée).
- Le modèle a donc "sur-appris" (overfit) les données d'apprentissage (train set), de sorte que ses performances ne se généralisent pas bien sur des données inconnues (test set).
- En fait, le modèle a appris le bruit des données d'apprentissage, au lieu de l'ignorer.

Overfitting

- **Overfitting:** Le modèle est très performant sur le **training set** (i.e., $\mathrm{MSE}_{\mathrm{train}}$ basse) alors qu'il est bien moins performant sur sur le **test set** (i.e., $\mathrm{MSE}_{\mathrm{train}}$ élevée).
- Le modèle a donc "sur-appris" (overfit) les données d'apprentissage (train set), de sorte que ses performances ne se généralisent pas bien sur des données inconnues (test set).
- En fait, le modèle a appris le bruit des données d'apprentissage, au lieu de l'ignorer.

Overfitting

- **Overfitting:** Le modèle est très performant sur le **training set** (i.e., $\mathrm{MSE}_{\mathrm{train}}$ basse) alors qu'il est bien moins performant sur sur le **test set** (i.e., $\mathrm{MSE}_{\mathrm{train}}$ élevée).
- Le modèle a donc "sur-appris" (overfit) les données d'apprentissage (train set), de sorte que ses performances ne se généralisent pas bien sur des données inconnues (test set).
- En fait, le modèle a appris le bruit des données d'apprentissage, au lieu de l'ignorer.

Overfitting

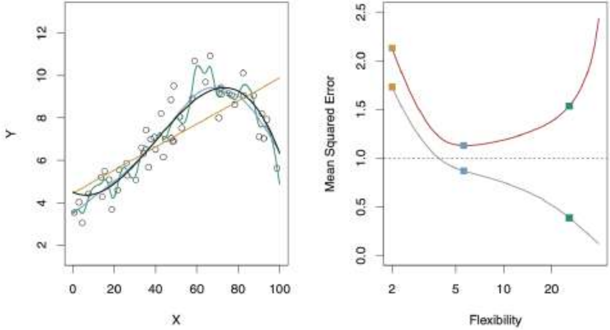

FIGURE 2.9. Left: *Data simulated from f, shown in black. Three estimates of f are shown: the linear regression line (orange curve), and two smoothing spline fits (blue and green curves).* Right: *Training MSE (grey curve), test MSE (red curve), and minimum possible test MSE over all methods (dashed line). Squares represent the training and test MSEs for the three fits shown in the left-hand panel.*

Figure taken from [James et al., 2013]

Overfitting

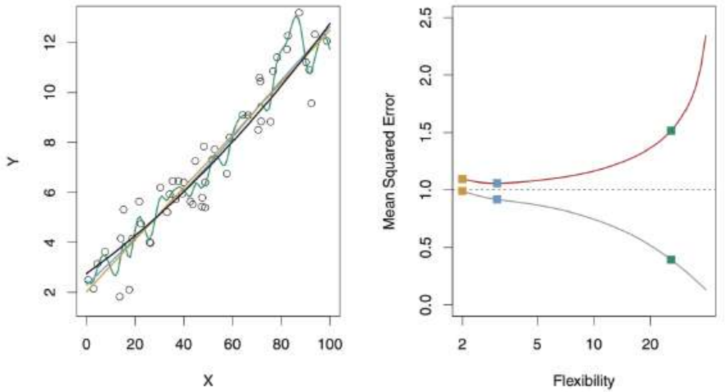

FIGURE 2.10. *Details are as in Figure 2.9, using a different true f that is much closer to linear. In this setting, linear regression provides a very good fit to the data.*

Figure taken from [James et al., 2013]

Overfitting

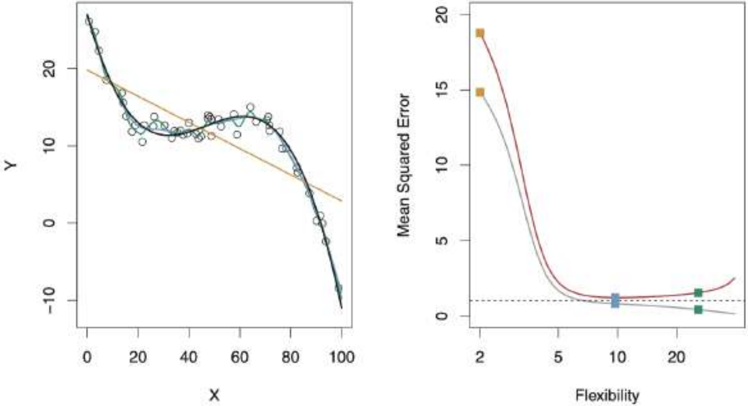

FIGURE 2.11. *Details are as in Figure 2.9, using a different f that is far from linear. In this setting, linear regression provides a very poor fit to the data.*

Figure taken from [James et al., 2013]

Bibliographie

James, G., Witten, D., Hastie, T., and Tibshirani, R. (2013).
An Introduction to Statistical Learning: with Applications in R, volume 103 of *Springer Texts in Statistics*.
Springer, New York.