Biais/Variance

Cette section est basée sur James et al. (2021), chapitre 2.

Quel est notre objectif ?

Nous souhaitons modéliser la relation entre une variable réponse Y, pouvant être quantitative, qualitative ou de nature différente, et un ensemble de p variables explicatives $X = (X_1, \dots, X_p)$, elles aussi de (potentiellement) différents types. L'idée centrale est qu'il existe un relation entre Y et les variables explicatives X. De manière générales, nous modélisons cette relation par le modèle :

$$Y = f(X) + \varepsilon. \tag{1}$$

Ici, f est une fonction déterministe (non-aléatoire) représentant l'information systématique que les variables explicatives $X_1, \dots, X_p$ apportent sur Y, et ε est un terme d'erreur aléatoire, modélisant les variations de Y non expliquées par X. Dans le cadre de ce cours, nous ferons les hypothèses suivantes : la variable aléatoire ε est indépendente de des variables explicatives X, $\mathbb{E}[\varepsilon] = 0$ et $\text{Var}(\varepsilon) = \sigma^2$. Le modèle Équation 1 est général. Il sert de cadre pour l'ensemble des méthodes que nous allons étudier, même lorsque la forme explicite de f n'est pas connue. La Figure 1 illustre les différents éléments du modèle : les données observées (X_i, Y_i), la fonction f (en bleu) et les écarts aléatoires ε_i représentés par des lignes pointillées.

```
# Load required package
library(ggplot2)

# Set seed for reproducibility
set.seed(42)

# 1. Generate data
n <- 100
x <- sort(runif(n, 0, 2 * pi))
f_x <- sin(x)
```

```r
epsilon <- rnorm(n, mean = 0, sd = 0.3)
y <- f_x + epsilon

# 2. Create data frame
data <- data.frame(x = x, y = y, f_x = f_x)

# 3. Plot
ggplot(data, aes(x, y)) +
  # Vertical lines: error = Y - f(X)
  geom_segment(
    aes(x = x, xend = x, y = f_x, yend = y),
    color = "gray60", linetype = "dashed"
  ) +
  # Observed points
  geom_point(color = "black", alpha = 0.7, size = 2) +
  # True function
  geom_line(aes(x = x, y = f_x), color = "blue", size = 1.2) +
  labs(
    x = "X", y = "Y"
  ) +
  theme_minimal()
```

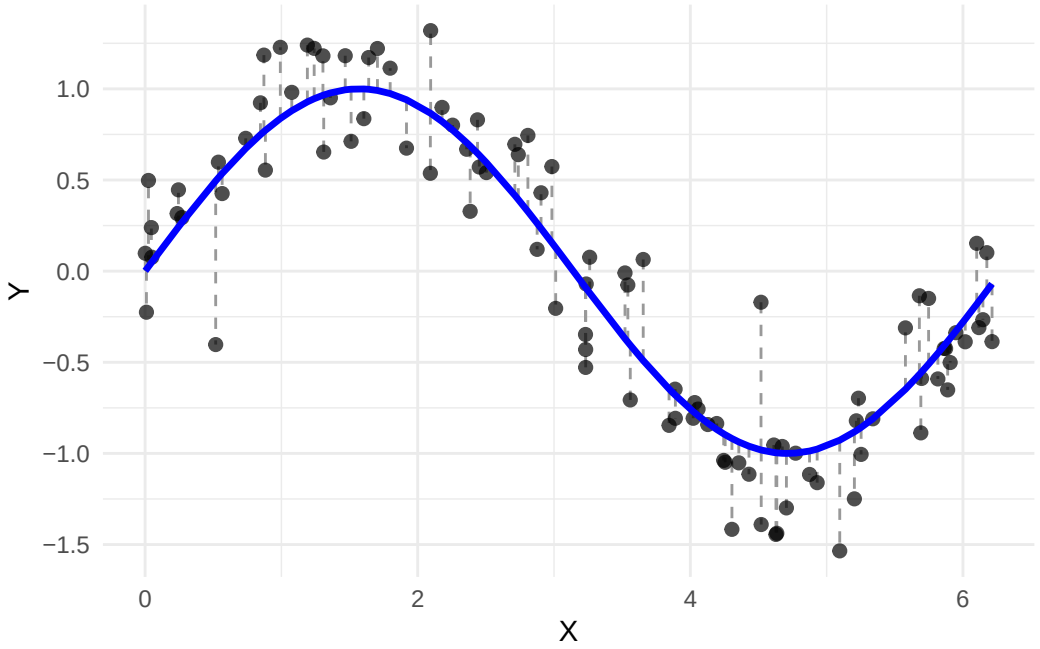

FIGURE 1 – Les différents éléments du modèle. Les points représentent les données observées (X_i, Y_i). La courbe bleue représente la fonction f et les lignes pointillées représentent l'erreur associée à chaque observation.

```
data = FileAttachment("../../include/data/data.csv").csv({ typed: true })

viewof noise = Inputs.range(
  [0.1, 0.5],
  {value: 0.3, step: 0.2, label: tex`\sigma^2`}
)

filtered = data.filter(function(df) {
  return df.noise == noise;
})

true_curve = data.filter(function(df) {
  return df.noise == 0;
})

Plot.plot({
  marks: [
    Plot.dot(filtered, {x: "x", y: "y"}),
    Plot.line(true_curve, {x: "x", y: "y"})
```

```
  ]
})
```

Dans la suite du cours, nous verrons différentes méthodes permettant d'estimer la fonction f à partir de données. Cependant avant d'étudier comment contruire un estimateur $\hat{f}$ de f, nous allons nous interroger sur la qualité d'un tel estimateur : que signifie "bien estimer" f ? Et comment évaluer la qualité de l'estimation ?

> Exemple : Régression linéaire simple
>
> Dans ce cadre très simple, nous faisons l'hypothèse que la fonction f est de la forme : $f(x) = ax + b$. Dans ce cas, l'estimation de la fonction f se résume à l'estimation des coefficients a et b.

> Remarque : Compromis entre exactitude et interprétabilité
>
> Dépendant de l'objectif de l'étude, nous devons généralement faire un choix entre l'exactitude de nos prédictions et l'interprétabilité de notre modèle. Un modèle simple, comme la régression linéaire, sera facile à interpréter mais capturera mal des relations complexes. À l'inverse, un modèle plus flexible, comme une forêt aléatoire, aura de meilleur prédiction, mais sera plus difficilement interprétable. Le choix dépend donc de l'objectif de l'analyse : compréhension ou performance prédictive ?

> Remarque : *No free lunch in statistics*
>
> Pourquoi ne pas simplement utiliser le modèle "ultime", celui qui serait toujours optimal quelque soit le jeu de données ? Parce qu'un tel modèle n'existe pas ! Il n'y a pas de méthode universellement meilleure pour tous les jeux de données et tous les objectifs. Une méthode performante dans un contexte donné peut échouer ailleurs. Il faut donc toujours adapter l'approche au problème (explication, prédiction, classification, …).

Comment mesurer la qualité d'un estimateur ?

Une fois que nous disposons d'un estimateur $\hat{f}$ de la fonction f, obtenu à partir de n observations $(y_1, x_1), \dots, (y_n, x_n)$, nous cherchons à évaluer la précision des prédictions $\widehat{Y} = \hat{f}(X)$. L'idée est de vérifier dans quelle mesure $\widehat{Y}$ est proche de la vraie valeur de Y.

Définition : Erreur quadratique moyenne

Lorsque Y est une variable quantitative, une mesure classique de la qualité de $\hat{f}$ est l'**erreur quadratique moyenne** (*mean square error*, MSE) :

$$MSE(Y, \widehat{Y}) = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 = \frac{1}{n}\sum_{i=1}^{n}\left(y_i - \hat{f}(x_i)\right)^2,$$

où $\hat{y}_i = \hat{f}(x_i)$ est la prédiction que $\hat{f}$ donne pour l'observation x_i.

Une MSE faible indique que les prédictions sont proches des observations. Nous pouvons aussi l'interpréter comme la distance moyenne entre les valeurs observées et les valeurs prédites. Nous cherchons donc à avoir une distance moyenne faible.

Dans le cas où Y est une variable qualitative, e.g. une classe ou un label, on utilise une autre mesure : le taux d'erreur.

Définition : Taux d'erreur

Lorsque Y est une variable qualitative, une mesure classique de la qualité de $\hat{f}$ est le **taux d'erreur** (*error rate*, ER) :

$$ER(Y, \widehat{Y}) = \frac{1}{n}\sum_{i=1}^{n}\mathbb{1}(y_i \neq \hat{y}_i) = \frac{1}{n}\sum_{i=1}^{n}\mathbb{1}(y_i \neq \hat{f}(x_i)).$$

où $\hat{y}_i = \hat{f}(x_i)$ est la prédiction que $\hat{f}$ donne pour l'observation x_i.

Le taux d'erreur mesure la proportion de mauvaises prédictions. Il s'agit, là encore, d'une mesure de la distance moyenne entre Y et $\widehat{Y}$, adaptée aux variables qualititatives.

Le compromis biais/variance

Notre objectif est souvent de minimiser l'erreur de prédiction, non seulement sur les données observées, mais surtout sur de nouvelles données (récupérée après avoir estimer le modèle). Pour cela, nous nous intéressons à l'erreur de prédiction :

$$\mathbb{E}\left[\left(Y - \widehat{Y}\right)^2\right] = \mathbb{E}\left[\left(Y - \hat{f}(X)\right)^2\right].$$

Cette erreur peut se décomposer en trois composantes :

— Le biais : l'erreur due à une approximation systématique, e.g. si on impose un modèle linéaire alors que la relation est non linéaire.

— La variance : la sensibilité de l'estimateur aux fluctuations de l'échantillon d'apprentissage.
— L'erreur irréductible : la variance intrinsèque du bruit ε, notée σ^2.

Décomposition biais/variance

On a :

$$\mathbb{E}\left[(Y-\widehat{Y})^2\right] = \mathbb{E}\left[(Y-\hat{f}(X))^2\right] = \text{Biais}(\hat{f}(X))^2 + \text{Var}(\hat{f}(X)) + \sigma^2.$$

Preuve

Tout d'abord, montrons que l'espérance de l'erreur de l'estimateur se décompose en une partie réductible et en une partie irréductible.

$$\begin{aligned}
\mathbb{E}\left[\left(Y-\widehat{Y}\right)^2\right] &= \mathbb{E}\left[\left(Y-\hat{f}(X)\right)^2\right] \\
&= \mathbb{E}\left[\left(f(X)+\varepsilon-\hat{f}(X)\right)^2\right] \\
&= \mathbb{E}\left[\left(f(X)-\hat{f}(X)\right)^2\right] + 2\mathbb{E}\left[\left(f(X)-\hat{f}(X)\right)\varepsilon\right] + \mathbb{E}[\varepsilon^2] \\
&= \mathbb{E}\left[\left(f(X)-\hat{f}(X)\right)^2\right] + 2\mathbb{E}\left[\left(f(X)-\hat{f}(X)\right)\right]\underbrace{\mathbb{E}[\varepsilon]}_{=0} + \sigma^2 \\
&= \underbrace{\mathbb{E}\left[\left(f(X)-\hat{f}(X)\right)^2\right]}_{\text{réductible}} + \underbrace{\sigma^2}_{\text{irréductible}}.
\end{aligned}$$

On utilise la linéarité de l'espérance et le fait que X et ε soient indépendants. On s'intéresse maintenant à la partie "réductible". L'astuce est de faire apparaître $\mathbb{E}\left[\hat{f}(X)\right]$.

$$\begin{aligned}
\mathbb{E}\left[\left(f(X)-\hat{f}(X)\right)^2\right] &= \mathbb{E}\left[\left(f(X)-\mathbb{E}\left[\hat{f}(X)\right]+\mathbb{E}\left[\hat{f}(X)\right]-\hat{f}(X)\right)^2\right] \\
&= \underbrace{\mathbb{E}\left[\left(f(X)-\mathbb{E}\left[\hat{f}(X)\right]\right)^2\right]}_{\text{A}} \\
&\quad - 2\underbrace{\mathbb{E}\left[\left(f(X)-\mathbb{E}\left[\hat{f}(X)\right]\right)\left(\hat{f}(X)-\mathbb{E}\left[\hat{f}(X)\right]\right)\right]}_{\text{B}} \\
&\quad + \underbrace{\mathbb{E}\left[\left(\hat{f}(X)-\mathbb{E}\left[\hat{f}(X)\right]\right)^2\right]}_{\text{C}}.
\end{aligned}$$

A. La fonction $f(X)$ n'étant pas aléatoire, on a $\mathbb{E}[f(X)] = f(X)$ et donc

$$\begin{aligned}\mathbb{E}\left[\left(f(X)-\mathbb{E}\left[\hat{f}(X)\right]\right)^2\right] &= \mathbb{E}\left[\left(\mathbb{E}\left[f(X)-\hat{f}(X)\right]\right)^2\right] \\ &= \mathbb{E}\left[f(X)-\hat{f}(X)\right]^2 \\ &= \text{Biais}(\hat{f}(X))^2.\end{aligned}$$

B. En développant l'expression et en utilisant l'indépendance des variables, on trouve que $B = 0$.
C. En utilisant la définition de la variance,

$$\mathbb{E}\left[\left(\hat{f}(X)-\mathbb{E}\left[\hat{f}(X)\right]\right)^2\right] = \text{Var}(\hat{f}).$$

Finalement, on a

$$\mathbb{E}\left[\left(f(X)-\hat{f}(X)\right)^2\right] = \text{Biais}(\hat{f}(X))^2 + \text{Var}(\hat{f}(X)).$$

D'où le résultat.

Cette décomposition met en avant un compromis fondamental en analyse de données :

- Si on choisit un modèle peu flexible, le biais sera élevé, mais la variance sera faible.
- Si on choisit un modèle flexible, le biais sera faible, mais la variance peut être très élevée.

Notre objectif est donc de trouver un juste équilibre entre biais et variance, i.e. un modèle qui prédit correctement, tout en étant généralisable à de nouvelles données. La Figure 2 présente un jeu de données et différents estimateurs $\hat{f}$. En faisant varier le paramètre λ, on obtient des modèles plus ou moins flexible (lorsque $\lambda = 0.15$, le modèle est flexible et lorsque $\lambda = 1$, le modèle est rigide). La Figure 3 montre la valeur du biais, de la variance et de la MSE pour les modèles estimés pour la Figure 2. On remarque que plus λ est petit, plus la variance est grande, mais le biais est petit (le modèle est flexible). Inversement, plus λ est grand, plus le biais est grand et la variance petite (le modèle est rigide). La courbe de MSE en fonction du paramètre est une courbe en U. Comme on cherche à minimiser la MSE, i.e. à faire un compromis entre le biais et la variance, on peut prendre $\lambda = 0.5$.

```r
# Load packages
library(ggplot2)
library(dplyr)
library(tidyr)

set.seed(42)

# 1. Simulate a single dataset
```

```r
n <- 100
sigma <- 0.3
x <- sort(runif(n, 0, 1))
y <- 4 * x * (1 - x) * log(x) + 2 + rnorm(n, 0, sigma)
df <- data.frame(x = x, y = y)

# 2. Define grid and spans to compare
x_grid <- seq(0, 1, length.out = 300)
spans_to_plot <- c(0.15, 0.3, 0.5, 0.75, 1.0)

# 3. Compute loess fits for each span
fits <- lapply(spans_to_plot, function(s) {
  loess_model <- loess(y ~ x, data = df, span = s)
  y_hat <- predict(loess_model, newdata = data.frame(x = x_grid))
  data.frame(x = x_grid, y_hat = y_hat, span = paste0("  = ", s))
})

fit_df <- bind_rows(fits)

# 4. Plot
ggplot() +
  geom_point(data = df, aes(x, y), color = "black", alpha = 0.5, size = 2) +
  geom_line(data = fit_df, aes(x, y_hat, color = span), size = 1.1) +
  scale_color_viridis_d(option = "C") +
  labs(
    x = "X", y = "Y",
    color = "Paramètre"
  ) +
  theme_minimal(base_size = 14)
```

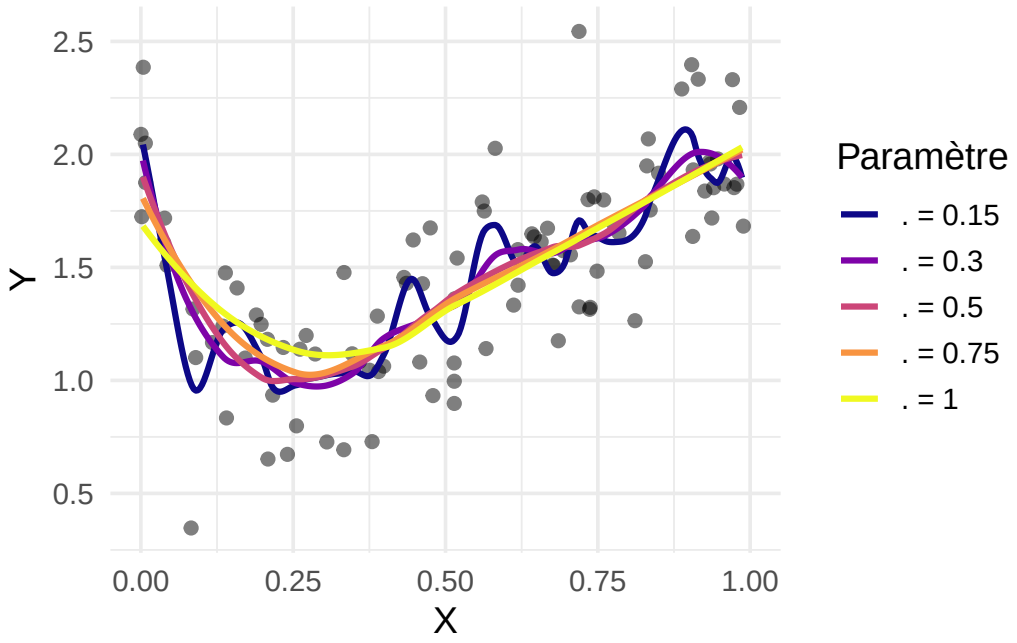

FIGURE 2 – Différents estimateurs de la fonction f.

```r
# Load packages
library(ggplot2)
library(dplyr)
library(tidyr)

set.seed(42)

# Parameters
n <- 100               # number of observations per dataset
n_sim <- 100           # number of simulated datasets
spans <- seq(0.1, 1, length.out = 15)  # LOESS smoothing parameters
sigma <- 0.1           # noise standard deviation
x_grid <- seq(0.1, 1, length.out = 200)
f_true <- 4 * x_grid * (1 - x_grid) * log(x_grid) + 2

# Storage for predictions
results <- list()

for (s in spans) {
  pred_matrix <- matrix(NA, nrow = length(x_grid), ncol = n_sim)
```

```r
  for (sim in 1:n_sim) {
    x <- sort(runif(n, 0.01, 1.1))
    y <- 4 * x * (1 - x) * log(x) + 2 + rnorm(n, 0, sigma)
    df <- data.frame(x = x, y = y)

    # Fit loess model with span = s
    model <- loess(y ~ x, data = df, span = s, degree = 2)
    pred <- predict(model, newdata = data.frame(x = x_grid), )

    pred_matrix[, sim] <- pred
  }

  # For each point in x_grid, compute bias², variance, MSE
  mean_pred <- rowMeans(pred_matrix, na.rm = TRUE)
  bias2 <- (mean_pred - f_true)^2
  var_pred <- apply(pred_matrix, 1, var, na.rm = TRUE)
  mse <- bias2 + var_pred

  results[[as.character(s)]] <- data.frame(
    span = s,
    Biais2 = mean(bias2),
    Variance = mean(var_pred),
    MSE = mean(mse)
  )
}

# Combine and reshape results
results_df <- bind_rows(results)
results_long <- pivot_longer(
  results_df,
  cols = c("Biais2", "Variance", "MSE"),
  names_to = "component", values_to = "value"
)

# Plot
ggplot(results_long, aes(x = span, y = value, color = component)) +
  geom_line(size = 1.2) +
  geom_point(size = 2) +
  scale_color_manual(
    values = c("Biais2" = "#0D0887FF", "Variance" = "#9C179EFF", "MSE" = "#ED7953FF")
  ) +
  labs(
```

```
    x = "Paramètre de lissage  " ,
    y = "",
    color = ""
  ) +
  theme_minimal(base_size = 14)
```

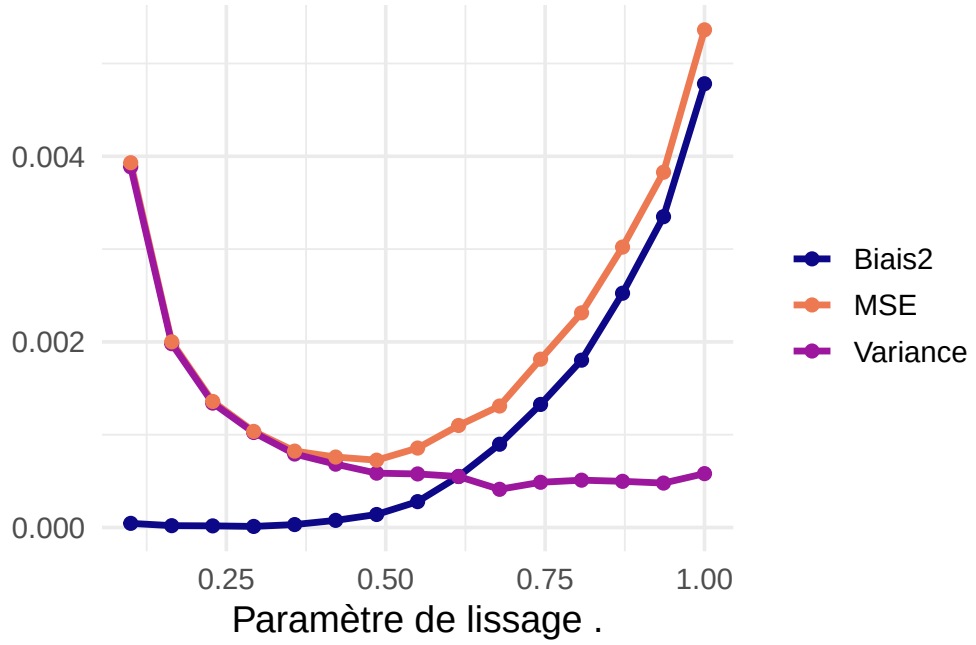

FIGURE 3 – Compromis biais/variance.

De manière générale, lorsque la flexibilité augmente, la diminution du biais est plus importante que l'augmentation de la variance, ce qui fait décroître l'erreur de prédiction. Cependant, à partir d'un certain niveau de flexibilité, le biais devient négligeable, et toute baisse supplémentaire est compensée par l'augmentation rapide de la variance. L'erreur de prédiction commence donc à croître. Il en résulte une courbe en U de l'erreur de prédiction en fonction de la flexibilité du modèle : un modèle trop rigide engendre un fort biais, tandis qu'un modèle trop flexible conduit à une trop grande variance.

Remarque : Pourquoi un compromis ?

Il est toujours possible de construire un modèle très flexible avec un biais nul, e.g. un modèle qui passe par tous les points d'observations, mais qui aura une variance énorme. À l'opposé, un modèle trop rigide, e.g. une constante, aura un biais très important mais une variance presque nulle. Le compromis biais/variance consiste à choisir un modèle qui

contrôle ces deux quantités.

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.