Introduction to Modelling and to Statistical Learning

Mathias Bourel

IMERL - Facultad de Ingeniería, Universidad de la República, Uruguay

September 25, 2019

Plan

Introduction

- Data Mining is the process of discovering patterns and relationships in data, with an emphasis on large observational databases.
- Special interest now because of:
  - Explosive growth of data in a great variety of fields revolution in biology, ecology, genomic, internet, network, images, multimedia.
  - Increasing of the computer power, storage devices with higher capacity
  - Faster communications, better database management systems

Extract information from a data set in such a way it can be understandable and usable.

¿For what?

Descriptive and predictive methods.

Descriptive methods

Objective: detect patterns on data by grouping units, attributes or both.
Data is usually unlabeled so we use non supervised approaches. Some descriptive techniques are:

- Clustering : find existing groups on data
- Segmentation : create groups by partitioning
- Factorial Analysis : find factors, i.e. groups of variables or groups of observations.
- Association rule : look for associations of variables
- Dimensional Reduction: Principal Component Analysis, Multidimensional Scalling, ISOMAP, etc.

Examples :

- Clustering electrical load curves
- Segmentation of clients for oriented marketing
- Look for set of items usually sold together on a supermarket.

Predictive methods

Objective : construct a mapping using available instance that can be used to predict new instances.
Data is labeled so we use supervised approaches. Some predictive techniques are:

- Regression Analysis
- Time Series Analysis
- Classification And Regression Trees (CART)
- Support Vector Machines (SVM)
- k-Nearest Neighbours (kNN)

Examples :

- Credit scoring
- Anticipate the electricity demand for tomorrow
- Estimate the probability of a disease for a patient

The Role of the Statistician

Statistics machine learning plays a central role in data mining.

- provide theoretical foundations for learning algorithms
- give useful tools to analyze an algorithms statistical properties and performance guarantee
- help researchers gain deeper understanding of the approaches, design better algorithms, and select appropriate methods for a given problem.
- help to take a better decision.

Machine Learning

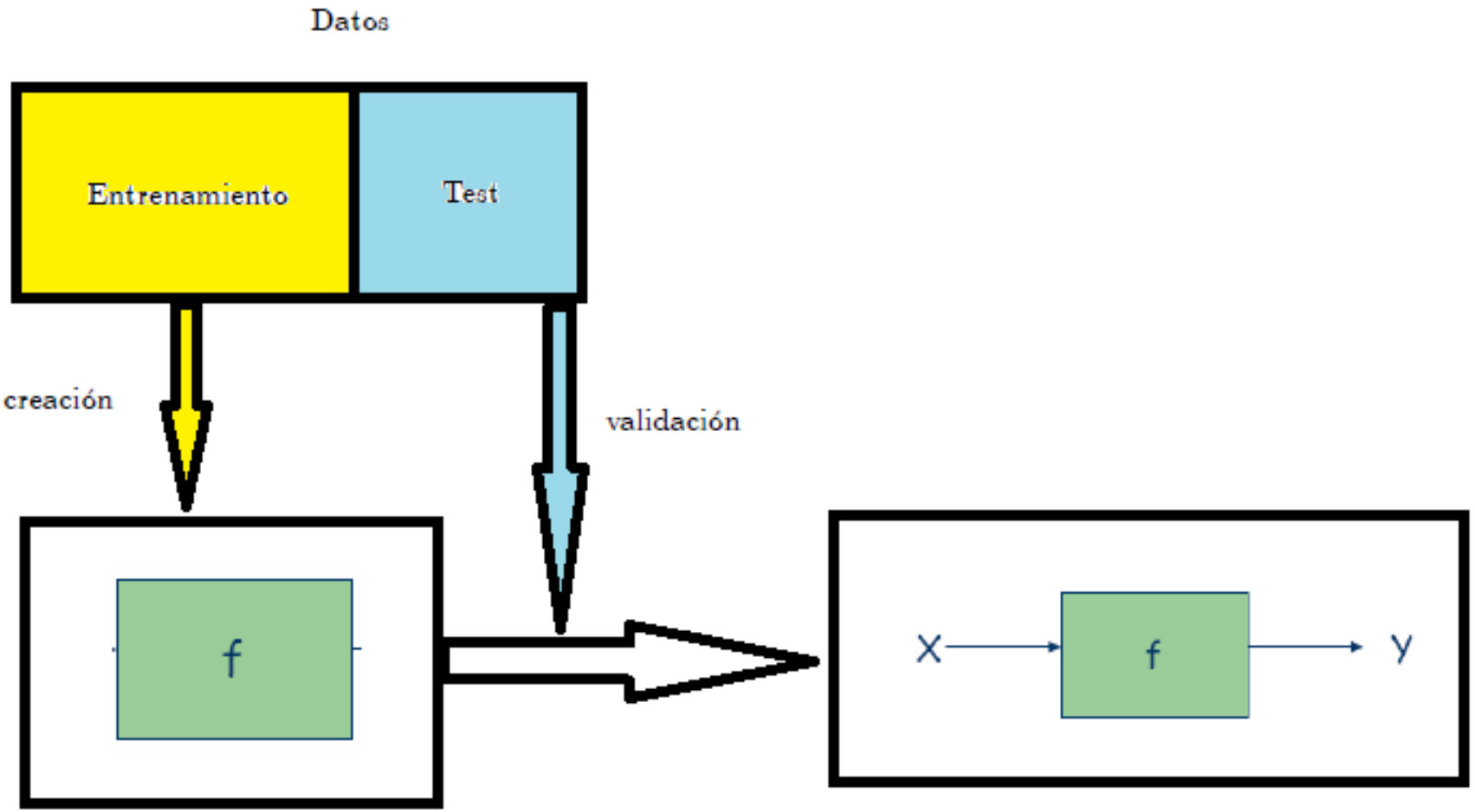
Machine Learning is about predictive methods.

- Another denominations: machine learning, statistical learning, artificial intelligence
- The techniques of Statistical Learning can help solve the problems that frequently arise when modeling an ecological problem, economic phenomenon, medical situation, climatic situation, etc..
- Idea: from a (training) data set, build and train a mathematical model f that will allow, given a new observation, to predict the category to which it belongs or some relevant output value. Predictor f is construct generally without any assumption on distribution or on nature of the dataset.

Examples

- Predict whether an email is spam or not spam.
- Predict whether a patient is prone to heart disease.
- Estimate the ozone rate in a city taking into account climatic variables.
- Predict the absence or presence of a species in a given environment.
- Predicting customer leaks for a financial institution.
- Identify handwritten figures of postcards in envelopes.
- Split a population into several subgroups.

Statistical Learning



Framework of Machine Learning

General framework:
$\mathcal{L}$ a data basis.

Framework of Machine Learning

General framework:
$\mathcal{L}$ a data basis. We search about $f : \mathcal{X} \to \mathcal{Y}$ a good predictor or a good explainer.

- Supervised Learning: $\mathcal{L} = \{(x_1, y_1), \ldots, (x_n, y_n)\} \subset \mathcal{X} \times \mathcal{Y} \subset \mathbb{R}^d \times \mathbb{R}$

  X: input variable, independent variable, explanatory (real o multidimensional), continuous, categorical, binary, ordinal.

  Y: output variable, dependent variable, real o categorical.

  - Classification: $y \in \{-1, 1\}$ (binary) or $y \in \{1, \ldots, K\}$ (multiclass).
  - Regression: $y \in \mathbb{R}$.

- Unsupervised Learning $\mathcal{L} = \{x_1, \ldots, x_n\} \subset \mathcal{X} \subset \mathbb{R}^d$

  - Clustering
  - Density estimation

  In all cases, the sample $\mathcal{L}$ is a collection of n independents realization of a multivariate random variable (X, Y) or X

Supervised and Unsupervised

1. *Supervised.* Data bases are of the type

$$X|Y$$

with $X \in \mathcal{M}_{n\times p}$ and $Y \in \mathcal{M}_{n\times 1}$ (categorical label -qualitative, classification problem- or continuous -quantitative, problem of regression-).

We will use this type of database to make inference and construct a predictor f that given a new observation can predict a category or a value having learned from the observations of the data base.

Supervised and Unsupervised

1. *Supervised.* Data bases are of the type

$$X|Y$$

with $X \in \mathcal{M}_{n\times p}$ and $Y \in \mathcal{M}_{n\times 1}$ (categorical label -qualitative, classification problem- or continuous -quantitative, problem of regression-).

We will use this type of database to make inference and construct a predictor f that given a new observation can predict a category or a value having learned from the observations of the data base.

Example: Linear models, Discriminant Analysis, Supervised learning techniques: CART, SVM, kNN, Aggregating Methods.

Supervised and Unsupervised

1. *Supervised.* Data bases are of the type

$$X|Y$$

with $X \in \mathcal{M}_{n\times p}$ and $Y \in \mathcal{M}_{n\times 1}$ (categorical label -qualitative, classification problem- or continuous -quantitative, problem of regression-).

We will use this type of database to make inference and construct a predictor f that given a new observation can predict a category or a value having learned from the observations of the data base.

Example: Linear models, Discriminant Analysis, Supervised learning techniques: CART, SVM, kNN, Aggregating Methods.

2. *Unsupervised.* Data bases are of the type

$$X$$

with $X \in \mathcal{M}_{n\times p}$

We will use this type of database to reduce the number of variables considered, find certain patterns, group, ... We do not have a label that "supervises" our analysis.

Supervised and Unsupervised

1. *Supervised.* Data bases are of the type

$$X|Y$$

with $X \in \mathcal{M}_{n\times p}$ and $Y \in \mathcal{M}_{n\times 1}$ (categorical label -qualitative, classification problem- or continuous -quantitative, problem of regression-).

We will use this type of database to make inference and construct a predictor f that given a new observation can predict a category or a value having learned from the observations of the data base.

Example: Linear models, Discriminant Analysis, Supervised learning techniques: CART, SVM, kNN, Aggregating Methods.

2. *Unsupervised.* Data bases are of the type

$$X$$

with $X \in \mathcal{M}_{n\times p}$

We will use this type of database to reduce the number of variables considered, find certain patterns, group, ... We do not have a label that "supervises" our analysis. Example: Principal Component Analysis, Multidimensional Scalling, Correspondance Analysis, Clustering, Density Estimation.

Supervised and Unsupervised

1. *Supervised.* Data bases are of the type

$$X|Y$$

with $X \in \mathcal{M}_{n\times p}$ and $Y \in \mathcal{M}_{n\times 1}$ (categorical label -qualitative, classification problem- or continuous -quantitative, problem of regression-).

We will use this type of database to make inference and construct a predictor f that given a new observation can predict a category or a value having learned from the observations of the data base.

Example: Linear models, Discriminant Analysis, Supervised learning techniques: CART, SVM, kNN, Aggregating Methods.

2. *Unsupervised.* Data bases are of the type

$$X$$

with $X \in \mathcal{M}_{n\times p}$

We will use this type of database to reduce the number of variables considered, find certain patterns, group, ... We do not have a label that "supervises" our analysis. Example: Principal Component Analysis, Multidimensional Scalling, Correspondance Analysis, Clustering, Density Estimation.

Observation:

- There may be bridges between the two approaches, there is also *semi-supervised learning*,...

Supervised and Unsupervised

1. *Supervised.* Data bases are of the type

$$X|Y$$

with $X \in \mathcal{M}_{n\times p}$ and $Y \in \mathcal{M}_{n\times 1}$ (categorical label -qualitative, classification problem- or continuous -quantitative, problem of regression-).

We will use this type of database to make inference and construct a predictor f that given a new observation can predict a category or a value having learned from the observations of the data base.

Example: Linear models, Discriminant Analysis, Supervised learning techniques: CART, SVM, kNN, Aggregating Methods.

2. *Unsupervised.* Data bases are of the type

$$X$$

with $X \in \mathcal{M}_{n\times p}$

We will use this type of database to reduce the number of variables considered, find certain patterns, group, ... We do not have a label that "supervises" our analysis. Example: Principal Component Analysis, Multidimensional Scalling, Correspondance Analysis, Clustering, Density Estimation.

Observation:

- There may be bridges between the two approaches, there is also *semi-supervised learning*,...
- The method of learning used in the case of supervised learning clearly depends on the nature of the response (whether it is qualitative or quantitative).

Supervised and Unsupervised

1. *Supervised.* Data bases are of the type

$$X|Y$$

with $X \in \mathcal{M}_{n\times p}$ and $Y \in \mathcal{M}_{n\times 1}$ (categorical label -qualitative, classification problem- or continuous -quantitative, problem of regression-).

We will use this type of database to make inference and construct a predictor f that given a new observation can predict a category or a value having learned from the observations of the data base.

Example: Linear models, Discriminant Analysis, Supervised learning techniques: CART, SVM, kNN, Aggregating Methods.

2. *Unsupervised.* Data bases are of the type

$$X$$

with $X \in \mathcal{M}_{n\times p}$

We will use this type of database to reduce the number of variables considered, find certain patterns, group, ... We do not have a label that "supervises" our analysis. Example: Principal Component Analysis, Multidimensional Scalling, Correspondance Analysis, Clustering, Density Estimation.

Observation:

- There may be bridges between the two approaches, there is also *semi-supervised learning*,...
- The method of learning used in the case of supervised learning clearly depends on the nature of the response (whether it is qualitative or quantitative).
- **There is no better method than all the rest on all data sets.**

Example

Dataset Advertising:

```
> datos=read.csv("Advertising.csv",header=T,sep=",")
> datos[,-1]
        TV Radio Newspaper Sales
1    230.1  37.8      69.2  22.1
2     44.5  39.3      45.1  10.4
3     17.2  45.9      69.3   9.3
4    151.5  41.3      58.5  18.5
5    180.8  10.8      58.4  12.9
```

In this case, each row of the dataset is an independent realization of the random multivariate variable (X, Y) where:

- $X = (X_1, X_2, X_3)$ is the *input* vector:
  - X_1 budget allocated to advertising by television (TV)
  - X_2 budget allocated to advertising by radio (Radio)
  - X_3 budget allocated to advertising by newspaper (Newspaper)
- Y (Sales) is the amount of sales made and is the output variable (response), dependent variable.

In general we will want models of the general form:

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

where $X_1, X_2, \ldots, X_p$ are predictor variables e Y is the response variable, ϵ is the error term, independent of X and with mean 0.

Data matrix

Two ways to consider the data matrix
$\mathbf{X} = ((x_{ij}))_{i=1,\dots,n}^{j=1,\dots,p} \in \mathcal{M}_{n\times p}$ (n observations with p variables).
By rows (observations):

$$\mathbf{X} = \begin{pmatrix} x_{11} & x_{12} & \dots & x_{1p} \\ x_{21} & x_{22} & \dots & x_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n1} & x_{n2} & \dots & x_{np} \end{pmatrix} = \begin{pmatrix} \text{obs 1} \\ \text{obs 2} \\ \vdots \\ \text{obs n} \end{pmatrix} = \begin{pmatrix} \mathbf{x_1} \\ \mathbf{x_2} \\ \vdots \\ \mathbf{x_n} \end{pmatrix}$$

By columns (variables):

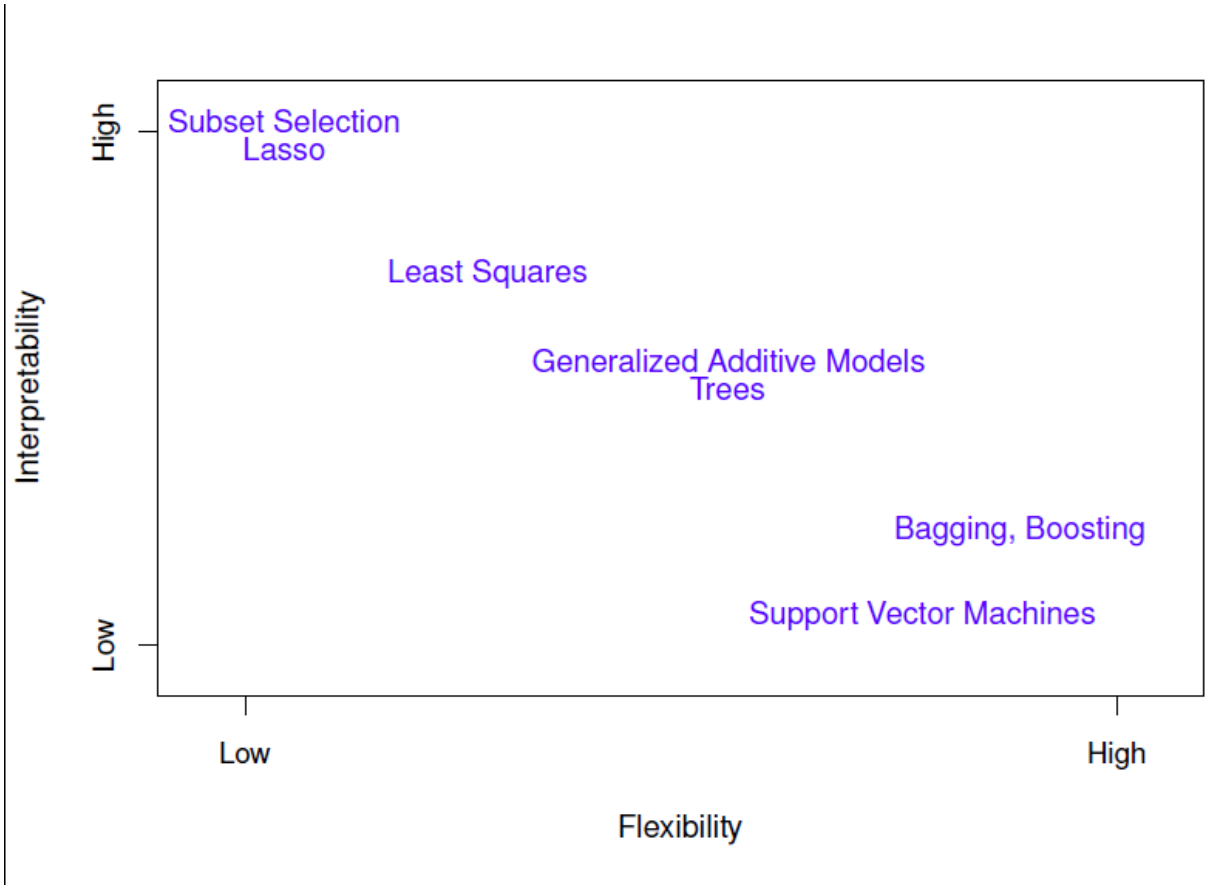
$$\mathbf{X} = \begin{pmatrix} x_{11} & x_{12} & \dots & x_{1p} \\ x_{21} & x_{22} & \dots & x_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n1} & x_{n2} & \dots & x_{np} \end{pmatrix} = \begin{pmatrix} v & v & & v \\ a & a & & a \\ r & r & & r \\ i & i & \dots & i \\ a & a & & a \\ b & b & & b \\ l & l & & l \\ e & e & & e \\ 1 & 2 & & p \end{pmatrix} = \begin{pmatrix} x_1 & x_2 & \dots & x_p \end{pmatrix}$$

Let y_i the response of observation i. Our data set is :

$$\mathcal{L} = \{(\mathbf{x_1}, y_1), (\mathbf{x_2}, y_2), \dots, (\mathbf{x_n}, y_n)\}$$

where $(\mathbf{x_1}, y_1), (\mathbf{x_2}, y_2), \dots, (\mathbf{x_n}, y_n)$ are independent realizations of variable (X, Y) where Y is dependent of X.

Performance vs Interpretability



Evaluation of the model

1. In regression quality of the fitting of a predictor can be evaluated by the *mean squared error MSE*:

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

It will be small if the predictions are close to the true response values and large if for some observations the prediction and the label are very different.

Evaluation of the model

1. In regression quality of the fitting of a predictor can be evaluated by the *mean squared error* *MSE*:

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

It will be small if the predictions are close to the true response values and large if for some observations the prediction and the label are very different.

However, evaluating the performance of the model on the data with which it has been trained, is not very interesting, or at least it is not as interesting as evaluating it on fresh data, which were not used for the estimation of $\widehat{f}$.

Evaluation of the model

1. In regression quality of the fitting of a predictor can be evaluated by the *mean squared error* *MSE*:

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

It will be small if the predictions are close to the true response values and large if for some observations the prediction and the label are very different.

However, evaluating the performance of the model on the data with which it has been trained, is not very interesting, or at least it is not as interesting as evaluating it on fresh data, which were not used for the estimation of $\widehat{f}$.

The performance of $\widehat{f}$ (construct over $\mathcal{L}$) is evaluated on a *testing set* $\mathcal{T} = \{(\mathbf{z_1}, u_1), (\mathbf{z_2}, u_2), \ldots, (\mathbf{z_s}, u_s)\}$ computing the *test*-MSE (generalization error):

$$\frac{1}{s}\sum_{i=1}^{s}\left(u_i - \widehat{f}(z_i)\right)^2$$

Evaluation of the model

1. In regression quality of the fitting of a predictor can be evaluated by the *mean squared error* *MSE*:

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

It will be small if the predictions are close to the true response values and large if for some observations the prediction and the label are very different.

However, evaluating the performance of the model on the data with which it has been trained, is not very interesting, or at least it is not as interesting as evaluating it on fresh data, which were not used for the estimation of $\widehat{f}$.

The performance of $\widehat{f}$ (construct over $\mathcal{L}$) is evaluated on a *testing set* $\mathcal{T} = \{(\mathbf{z_1}, u_1), (\mathbf{z_2}, u_2), \ldots, (\mathbf{z_s}, u_s)\}$ computing the *test*-MSE (generalization error):

$$\frac{1}{s}\sum_{i=1}^{s}\left(u_i - \widehat{f}(z_i)\right)^2$$

In practice, original data set is divided in two parts: the first, $\mathcal{L}$, usually $2/3$, to train the model, and the remaining $1/3$, $\mathcal{T}$, to test it.

Evaluation of the model

1. In regression quality of the fitting of a predictor can be evaluated by the *mean squared error MSE*:

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

It will be small if the predictions are close to the true response values and large if for some observations the prediction and the label are very different.

However, evaluating the performance of the model on the data with which it has been trained, is not very interesting, or at least it is not as interesting as evaluating it on fresh data, which were not used for the estimation of $\widehat{f}$.

The performance of $\widehat{f}$ (construct over $\mathcal{L}$) is evaluated on a *testing set* $\mathcal{T} = \{(\mathbf{z_1}, u_1), (\mathbf{z_2}, u_2), \ldots, (\mathbf{z_s}, u_s)\}$ computing the *test*-MSE (generalization error):

$$\frac{1}{s}\sum_{i=1}^{s}\left(u_i - \widehat{f}(z_i)\right)^2$$

In practice, original data set is divided in two parts: the first, $\mathcal{L}$, usually $2/3$, to train the model, and the remaining $1/3$, $\mathcal{T}$, to test it.

Also in this way the overfitting is avoided

2. In classification the error is measured with the misclassified rate:

$$\frac{1}{n}\sum_{i=1}^{n}\mathbb{1}_{\{y_i \neq \widehat{y}_i\}}$$

Bias-variance trade-off

If we assume that $y = f(x) + \epsilon$, it is possible to prove that the expected value of the MSE for a fixed test value x_0, can be decomposed as:

$$\mathbb{E}\left(y_0 - \widehat{f}(x_0)\right)^2 = \text{Var}\left(\widehat{f}(x_0)\right) + \left[\text{Sesgo}\left(\widehat{f}(x_0)\right)\right]^2 + \text{Var}(\epsilon)$$

Bias-variance trade-off

If we assume that $y = f(x) + \epsilon$, it is possible to prove that the expected value of the MSE for a fixed test value x_0, can be decomposed as:

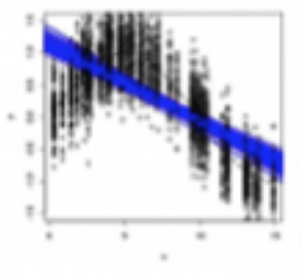
$$\mathbb{E}\left(y_0 - \widehat{f}(x_0)\right)^2 = \text{Var}\left(\widehat{f}(x_0)\right) + \left[\text{Sesgo}\left(\widehat{f}(x_0)\right)\right]^2 + \text{Var}(\epsilon)$$

- As $\text{Var}\left(\widehat{f}(x_0)\right)$ and $\left[\text{Sesgo}\left(\widehat{f}(x_0)\right)\right]^2$ are non negatives, it follows that $\mathbb{E}\left(y_0 - \widehat{f}(x_0)\right)^2$ has as lower bound $\text{Var}(\epsilon)$.
- We call *variance* to the amount that varies $\widehat{f}$ if we change the training set (different set of workouts produce different $\widehat{f}$). Under ideal conditions, the estimate of f does not change much if we change the training sets. In general, very flexible statistical models (with many parameters) have high variance. For example in the case of simple linear regression, when we change an element of the data set, the estimator does not vary so much. On the other hand if the model is very adjusted, changing a point produces a significant change in the estimation.
- *Bias* refers to the modelling error: explaining a real and complicated problem by a simpler mathematical model. For example, linear models assume that there is a linear relationship between Y and explanatory variables $X_1, \ldots, X_p$ which clearly has little chance of happening, so the bias will be important. In general, flexible statistical methods have a little bias.

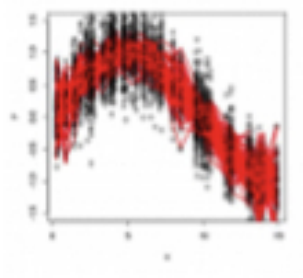
Bias-variance trade-off

Lineal (g1)

Baja Varianza
Gran sesgo



Polinomio g15



Alta Varianza
Bajo sesgo

$$\mathbf{Var}[X] = \mathbf{E}[X^2] - \mathbf{E}[X]^2$$
$$y = f + \epsilon$$

$$\mathbf{Bias}[\hat{f}(x)] = \mathbf{E}[\hat{f}(x) - f(x)]$$
$$\mathbf{Var}[\hat{f}(x)] = \mathbf{E}[\hat{f}(x)^2] - \mathbf{E}[\hat{f}(x)]^2$$

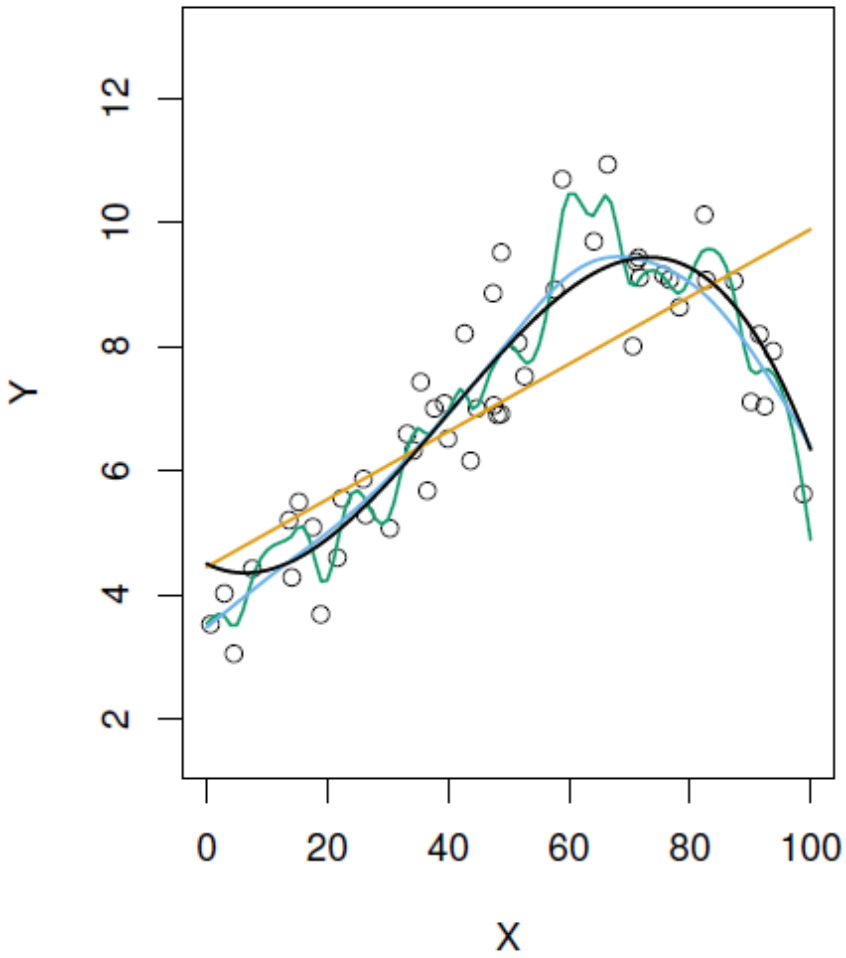
$$\begin{aligned}
\mathbf{E}[(y - \hat{f})^2] &= \mathbf{E}[y^2 + \hat{f}^2 - 2y\hat{f}] \\
&= \mathbf{E}[y^2] + \mathbf{E}[\hat{f}^2] - \mathbf{E}[2y\hat{f}] \\
&= \mathbf{Var}[y] + \mathbf{E}[y]^2 + \mathbf{Var}[\hat{f}] + \mathbf{E}[\hat{f}]^2 - 2f\mathbf{E}[\hat{f}] \\
&= \mathbf{Var}[y] + \mathbf{Var}[\hat{f}] + (f^2 - 2f\mathbf{E}[\hat{f}] + \mathbf{E}[\hat{f}]^2) \\
&= \mathbf{Var}[y] + \mathbf{Var}[\hat{f}] + (f - \mathbf{E}[\hat{f}])^2 \\
&= \mathbf{Var}[y] + \mathbf{Var}[\hat{f}] + \mathbf{E}[f - \hat{f}]^2 \\
&= \sigma^2 + \mathbf{Var}[\hat{f}] + \mathbf{Bias}[\hat{f}]^2
\end{aligned}$$

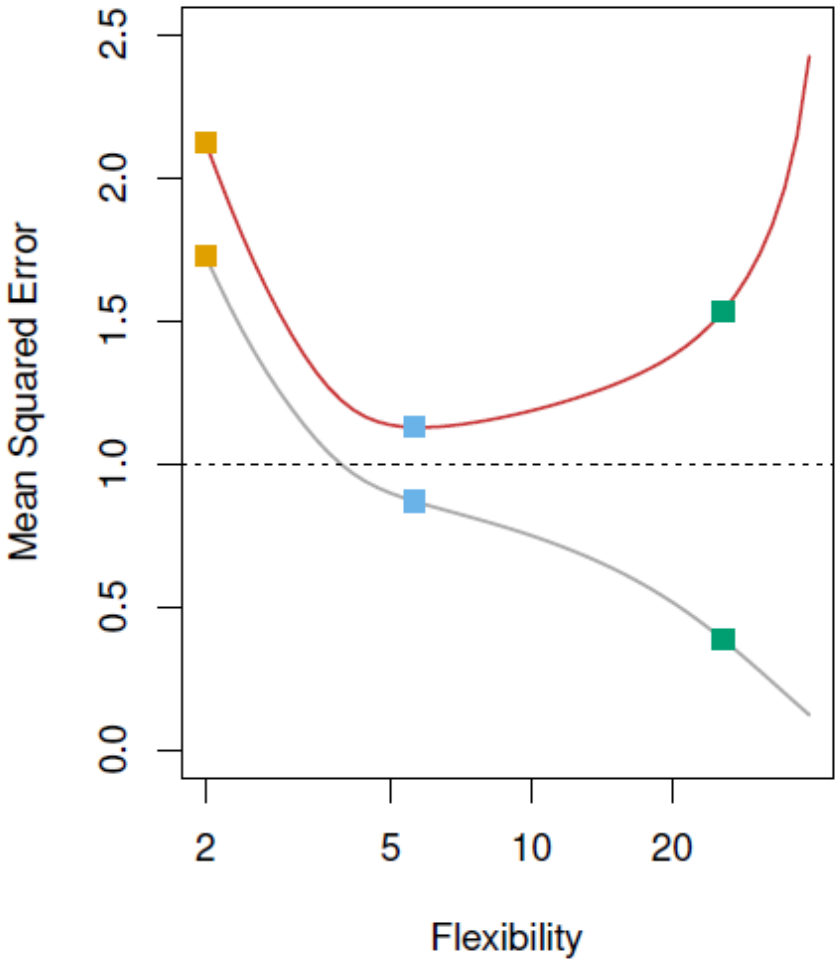
$$= error\ irreducible + varianza(\hat{f}) + Sesgo^2\hat{f}$$

Bias-variance trade-off. Example

Several estimators (smoothing splines) are considered for different data sets (example extracted of James, Witten, Hastie and Tibshirani book).

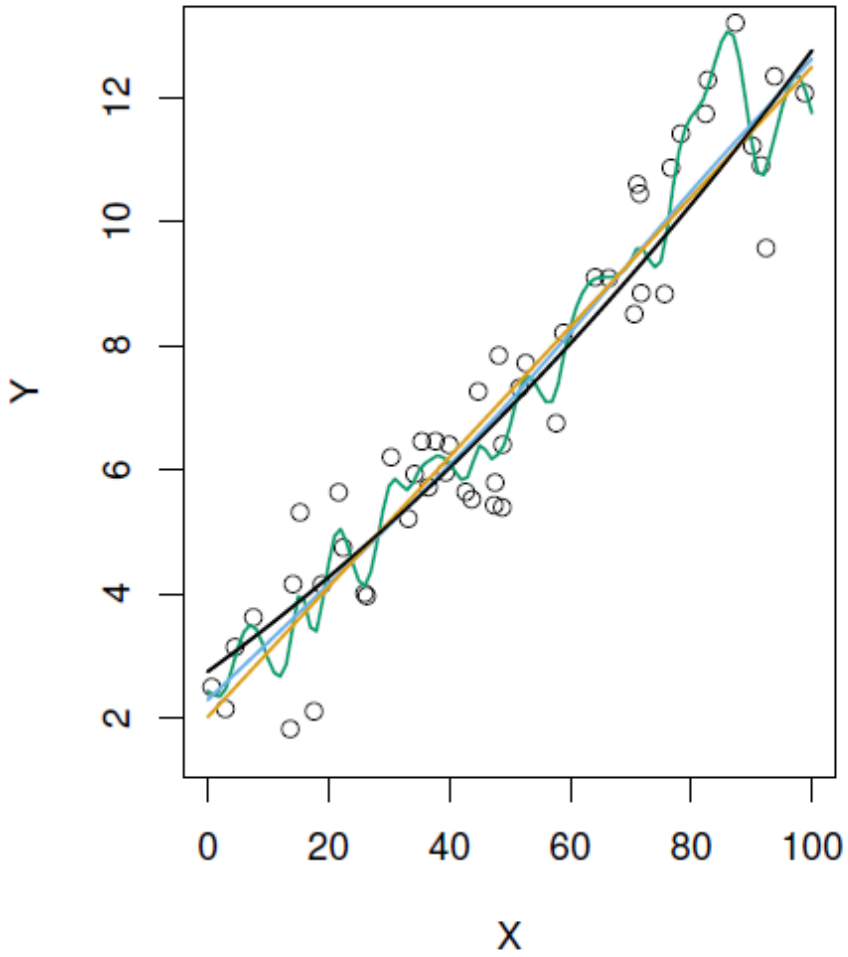
Example 1. On the left hand three estimators with different flexibility adjusting the same data points and on the right hand the MSE curve of the flexibility on the training set (grey) and on a generalization set (red).

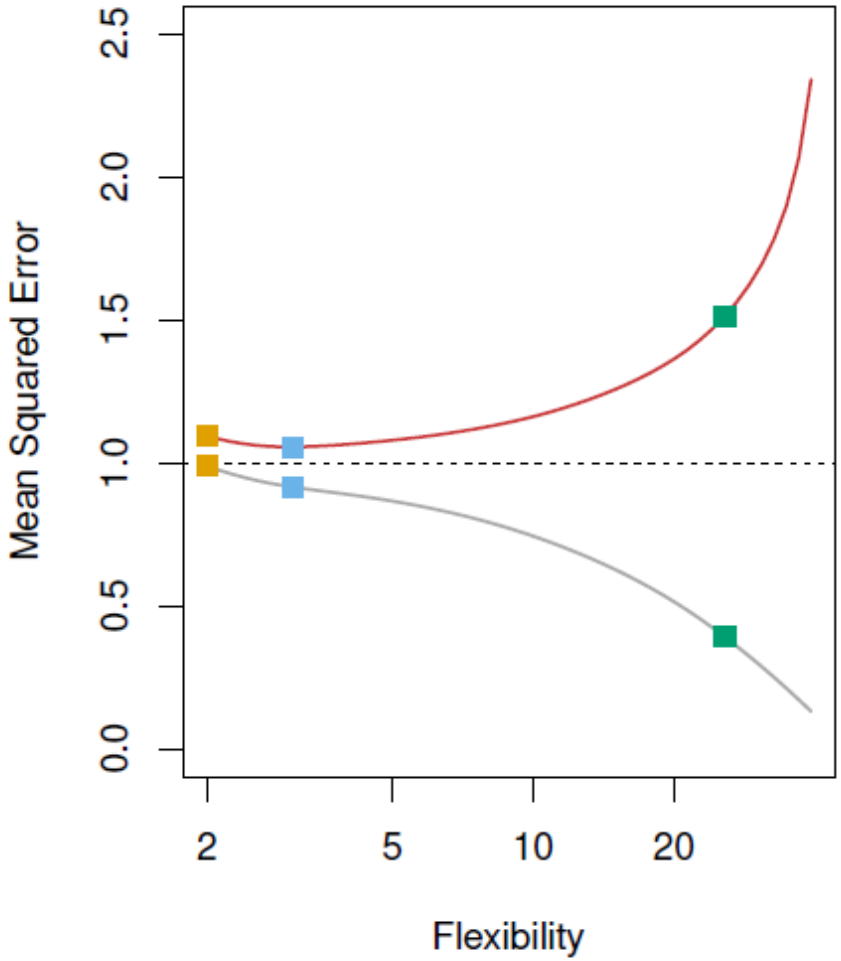




Bias-variance trade-off. Example

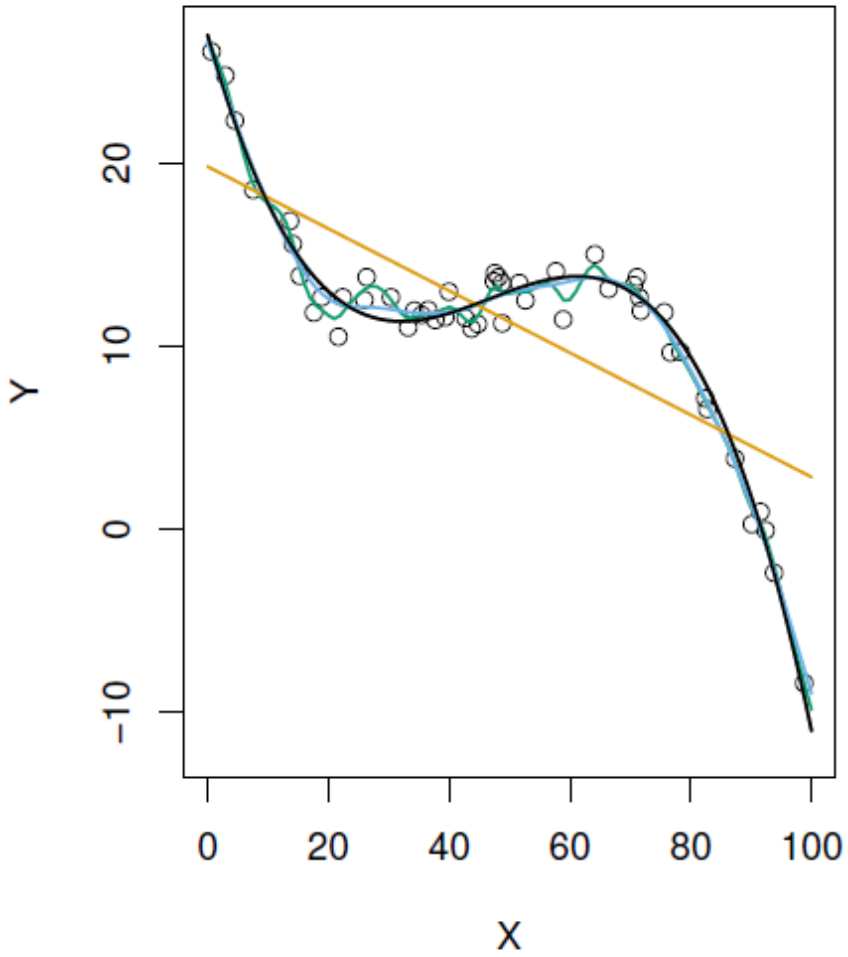
Example 2. On the left hand three estimators with different flexibility adjusting the same data points and on the right hand the MSE curve of the flexibility on the training set (grey) and on a generalization set (red).

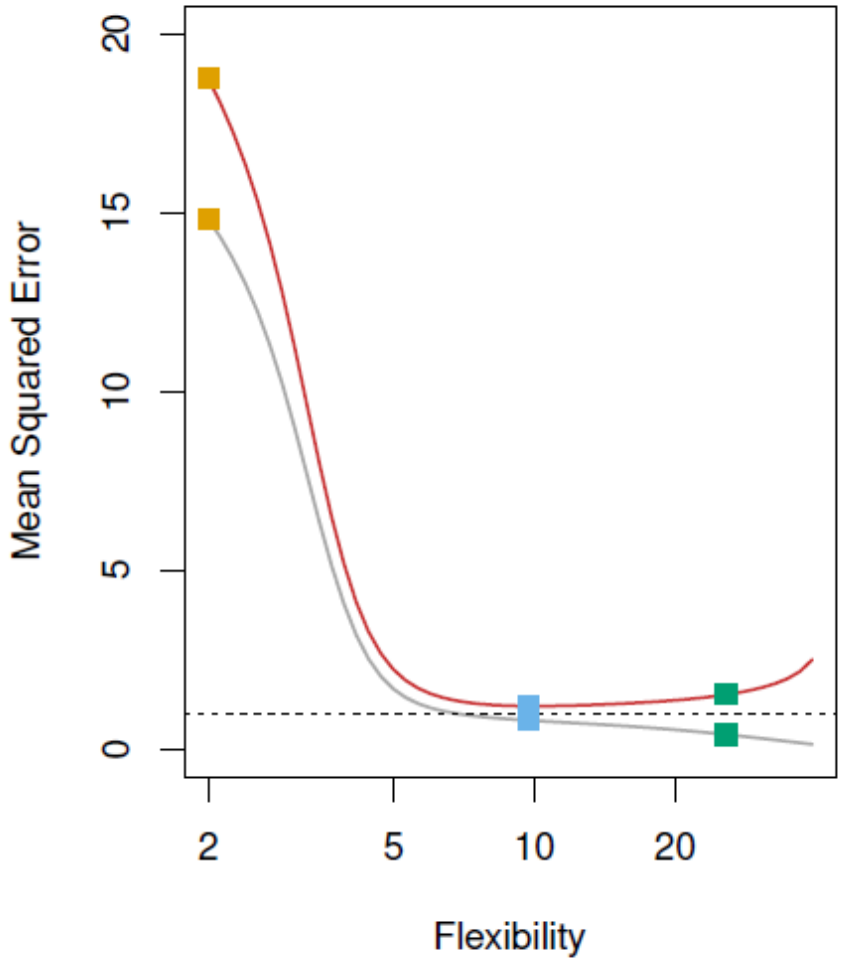




Bias-variance trade-off. Example

Example 3. On the left hand three estimators with different flexibility adjusting the same data points and on the right hand the MSE curve of the flexibility on the training set (grey) and on a generalization set (red).





Bias-variance trade-off. Example

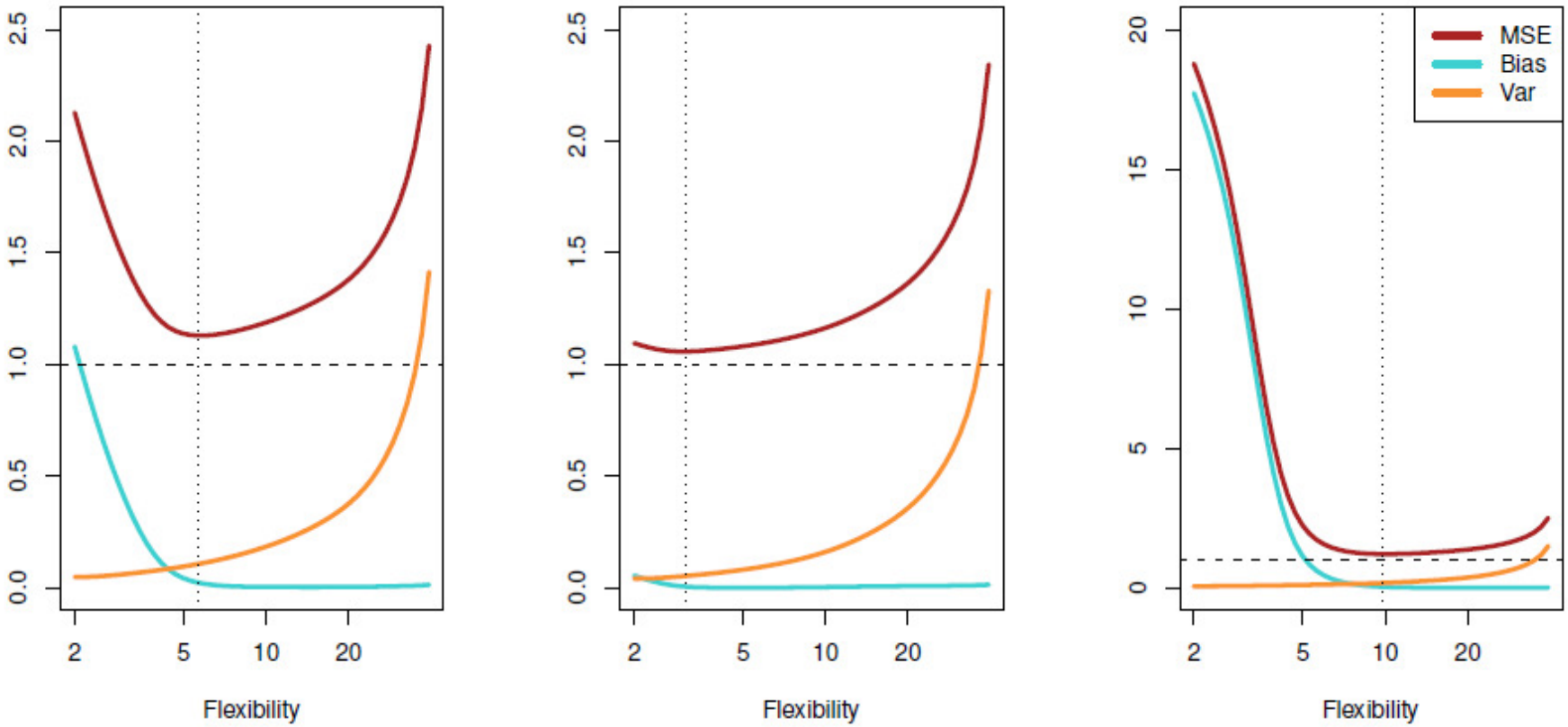


Figure: The three graphs refer to the MSE, bias and variance curves of three previous examples

Bias-variance trade-off. Example

The choice of the model will also be important to consider it a classification problem:

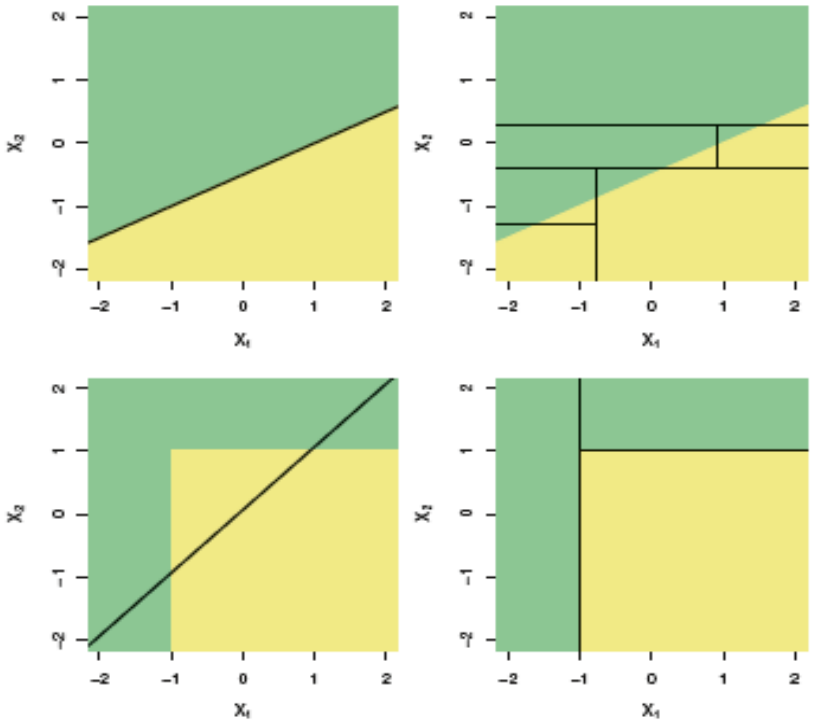


FIGURE 8.7. Top Row: *A two-dimensional classification example in which the true decision boundary is linear, and is indicated by the shaded regions. A classical approach that assumes a linear boundary (left) will outperform a decision tree that performs splits parallel to the axes (right).* Bottom Row: *Here the true decision boundary is non-linear. Here a linear model is unable to capture the true decision boundary (left), whereas a decision tree is successful (right).*

Referencias

- D. Peña, *Análisis de Datos Multivariantes*, Mac Graw Hill, 2002.
- A. I. Izenman, *Modern Multivariate Statistical Techniques*, Springer, 2008.
- James, G., Witten, D., Hastie, T., Tibshirani, R. An Introduction to Statistical Learning with Applications in R. Springer Texts in Statistics, 2013
- Devroye,L., Györfi,L. and Lugosi,G. A Probability Theory of Pattern Recognition. Springer, 1996
- Vapnik,V. Statistical Learning Theory, Wiley, 1998
- Breiman, L. Bagging predictors. Machine Learning, 24(2):123–140, 1996
- Freund, Y. y Schapire, E; A decision-theoretic generalization of on-line learning and application to boosting, Journal of Computer and System Sciences, 55(1): p 119-13, 1997