Introduction to Machine Learning

Session 1b: General Introduction

Reto Wüest

Department of Political Science and International Relations
University of Geneva

Outline

What is Machine Learning?

Machine Learning

Learning

The process of converting experience into expertise or knowledge.

Machine Learning

Machine learning is automated learning. We program computers so that they can learn and improve based on input available to them.

- The input to a learning algorithm is training data, representing experience.
- The output of a learning algorithm is some expertise, usually taking the form of ..... .
- A successful learning algorithm should be able to progress from individual examples to broader generalization.

(Shalev-Shwartz and Ben-David 2014, 19f.)

When Do We Need Machine Learning?

When do we rely on machine learning rather than directly programing computers to carry out the task at hand?

- **Complex tasks:** Tasks that we do not understand well enough to extract a well-defined program from our expertise (e.g., analysis of large and complex data, driving).
- **Tasks that change over time:** Machine learning tools are, by nature, adaptive to the changes in the environment they interact with (e.g., spam detection, speech recognition).

Supervised Versus Unsupervised Learning

Supervised Learning

- Data: for every observation $i = 1, \ldots, n$, we observe a vector of inputs $\mathbf{x}_i$ and a response y_i.
- Goal: fit a model that relates response y_i to $\mathbf{x}_i$ in order to accurately predict the response for future observations.
- If Y is quantitative, then this problem is a regression problem; if Y is categorical, then it is a classification problem.

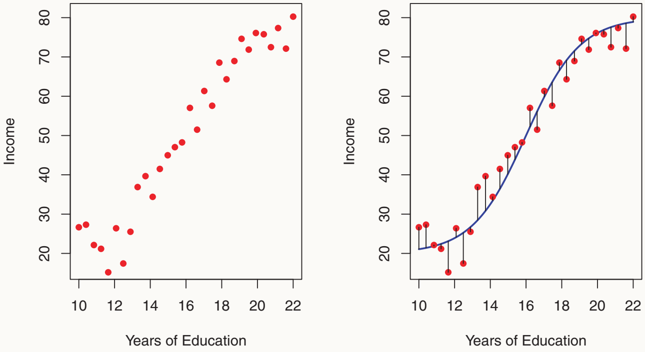
Unsupervised Learning

- Data: for every observation $i = 1, \ldots, n$, we observe a vector of inputs $\mathbf{x}_i$ but no associated response y_i.
- Goal: learning about relationships between the inputs or between the observations.

Supervised Learning

Fundamental Problem

Suppose $Y = f(X) + \varepsilon$, where $E[\varepsilon] = 0$. Goal is to estimate f based on observed data (X, Y).



(Source: James et al. 2013, 17)

Fundamental Problem

- Given estimate $\hat{f}$ and inputs X, we can predict $\hat{Y} = \hat{f}(X)$.
- How accurate is $\hat{Y}$ as a prediction for Y?
- For fixed $\hat{f}$ and X,

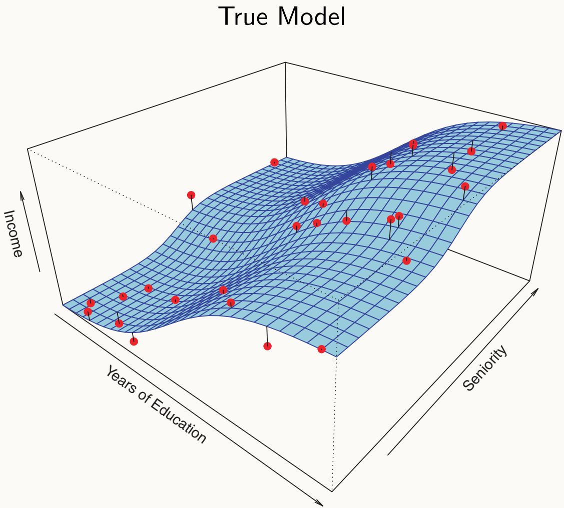
$$\begin{aligned} E\left[(Y - \hat{Y})^2\right] &= E\left[\left(f(X) + \varepsilon - \hat{f}(X)\right)^2\right] \\ &= \underbrace{\left[f(X) - \hat{f}(X)\right]^2}_{\text{reducible}} + \underbrace{Var\left[\varepsilon\right]}_{\text{irreducible}} \end{aligned} \tag{1}$$

- Our goal is to estimate f so as to minimize the reducible error.

How Do We Estimate f?

- Our goal is to apply a machine learning method to training data in order to estimate the unknown f.
- Training data consist of $\{(\mathbf{x}_i, y_i)\}_{i=1,\ldots,n}$, where $\mathbf{x}_i = (x_{i1}, x_{i2}, \ldots, x_{ip})^T$.
- There are a range of methods for estimating f, some more and some less flexible with regard to the functional form of f.
- Flexible methods can fit a wider range of possible functional forms for f, but this comes at the cost of a greater potential for overfitting.

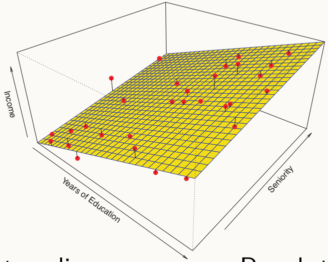
Example: f Estimated by Methods with Different Flexibility



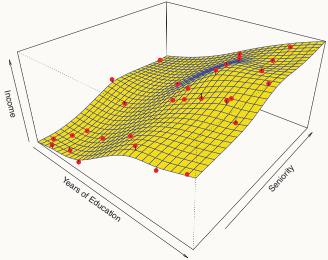
(Source: James et al. 2013, 18)

Example: f Estimated by Methods with Different Flexibility

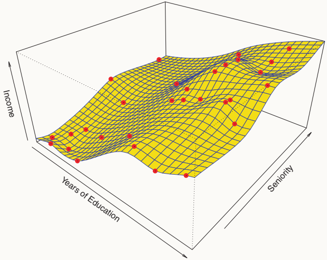
Linear model fit by least squares



Smooth thin-plate spline



Rough thin-plate spline



(Source: James et al. 2013, 22ff.)