# Data Management

## Statistical Learning

Malka Guillot

HEC Liège | ECON2306

# Prologue

# Table of contents


1. Machine Learning: an overview
2. What is statistical learning?
3. Why estimate $f(X)$?
4. How do we estimate $f(X)$?
5. Model accuracy
6. The Bias-Variance Trade Off
7. How to choose training and test set?
8. Conclusion

# Context

## Today

- What is statistical learning?
- Statistics in social science – causality.
- Statistics in machine learning – prediction.
- Accuracy v. interpretability.
- Model accuracy.
- The bias-variance tradeoff.

## Next

- Supervised learning
  - Classification
  - Regression
- Unsupervised learning

# References


- introduction to Statistical Learning chap 1. & 2 & 5.1
- Kleinberg, Ludwig, Mullainathan, and Obermeyer (2015), "Prediction Policy Problems." American Economic Review, 105 (5), pp. 491-95.
- Mullainathan and Spiess (2017), "Machine Learning: An Applied Econometric Approach", Journal of Economic Perspectives, 31 (2), pp. 87-106,

# Machine Learning: overview and examples

## Supervised vs. unsupervised learning

# Supervised learning

Estimating functions with **known observations** and **outcome** data.

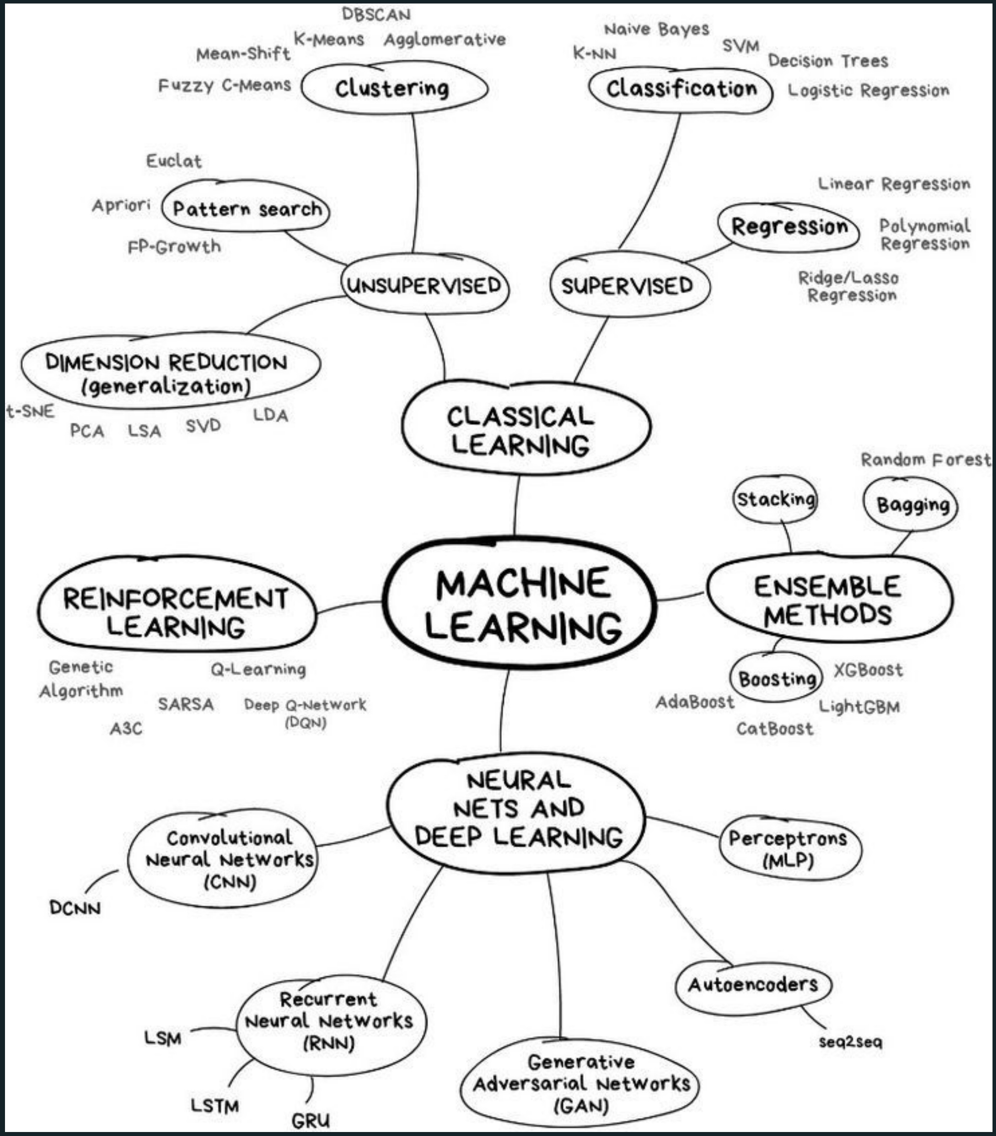
- We observe data on $Y$ and $X$
- We want to learn the mapping $\hat{Y} = \hat{f}(X)$
- **Classification** when $\hat{Y}$ discrete
- **Regression** when $\hat{Y}$ continuous

# Unsupervised learning

Estimating functions without the aid of outcome data.

- We only observe $X$ and want to learn something about its structure
- **Clustering**: Partition data into homogeneous groups based on X
- **Dimensionality reduction** (e.g. PCA)

# The Machine learning landscape



# Examples: Studies using ML for p rediction

- Glaeser, Kominers, Luca, and Naik (2016) use images from Google Street View to measure block-level income in New York City and Boston
- Jean et al. (2016) train a neural net to predict local economic outcomes from satellite data in African countries
- Chandler, Levitt, and List (2011) predict shootings among high-risk youth so that mentoring interventions can be appropriately targeted
- Kleinberg, Lakkaraju, Leskovec, Ludwig, and Mullainathan (2018) predict the crime probability of defendants released from investigative custody to improve judge decisions
- Kang, Kuznetsova, Luca, and Choi (2013) use restaurant reviews on Yelp.com to predict the outcome of hygiene inspections
- Huber and Imhof (2018) use machine learning to detect bid-rigging cartels in Switzerland
- Kogan, Levin, Routledge, Sagi, and Smith (2009) predict volatility of firms from market-risk disclosure texts (annual 10-K forms)

# What is statistical learning?

# Setting

- Input variables $\mathcal{X}$
  - AKA features, independent variables, predictors
- Output variables $\mathcal{Y}$
  - AKA dependent variables, outcomes, etc.

# Statistical learning theory

$$f : \mathcal{X} \rightarrow \mathcal{Y}$$

$$\mathcal{X} \in \mathbb{R}^{n \times p}, \mathcal{Y} \in \mathbb{R}^{p}$$

*SL= approaches for finding a function that accurately maps the inputs $\mathcal{X}$ to outputs $\mathcal{Y}$*

# Statistical model

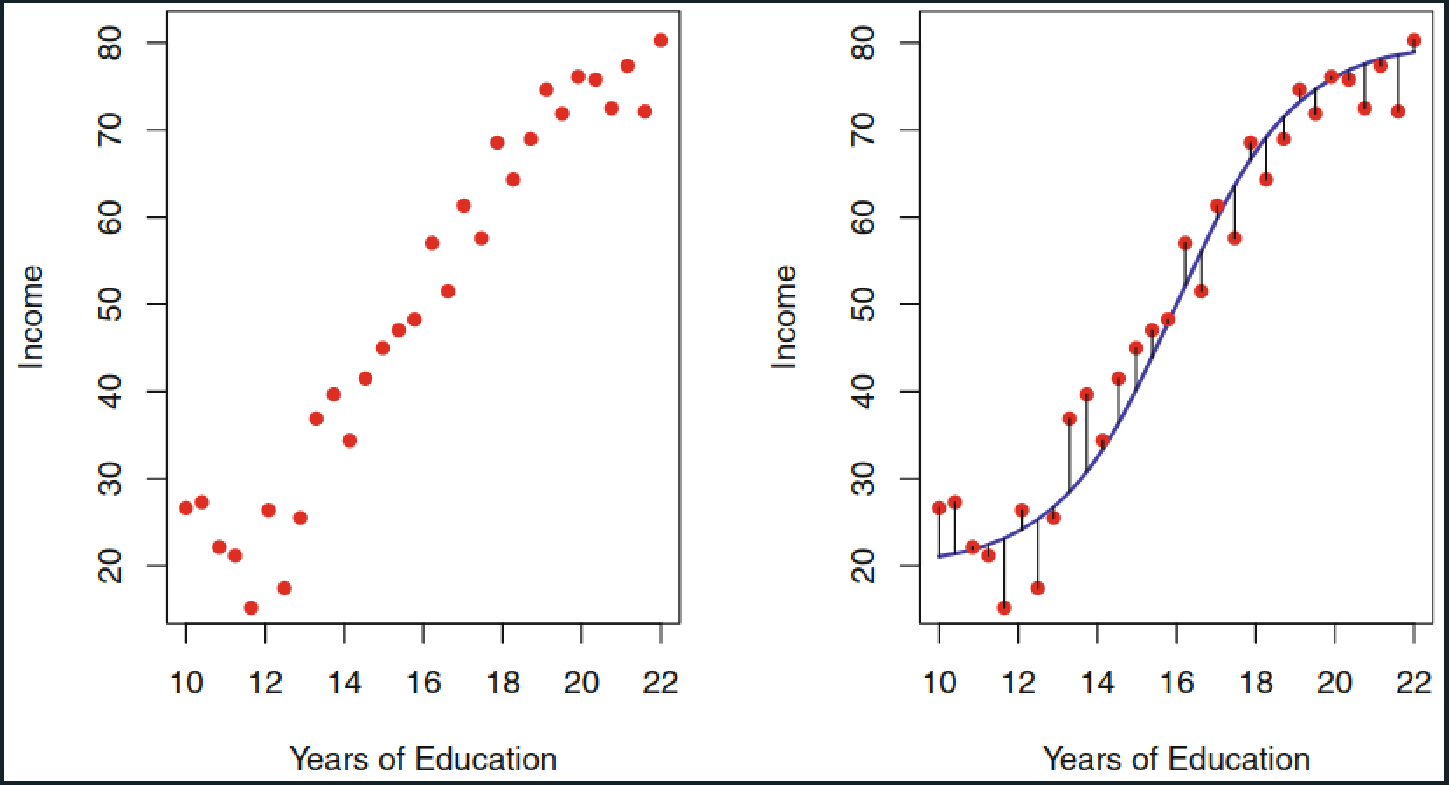
Concretely, finding $f(.)$ s.t.

$$Y = f(X) + \epsilon$$

- $f(X)$ is an unknown function of a matrix of predictors $X = (X_1, \cdots, X_p)$,
- $Y$: a scalar outcome variable
- an error term $\epsilon$ with mean zero.
- While $X$ and $Y$ are known, $f(\cdot)$ is unknown.

**Goal of statistical learning**: to utilize a set of approaches to estimate the "best" $f(\cdot)$ for the problem at hand.

# Example: income as a function of education



# Why estimate $f(X)$?

# Prediction

- Predict $Y$ by $\hat{Y} = \hat{f}(X)$
- When do we care about "pure prediction"?
  - $X$ readily available but $Y$ is not
- $\hat{f}$ can be a **black box**:
  - the only concern is accuracy of the prediction

# Inference

- Understanding the way that $Y$ is affected as $X_1, \ldots, X_p$ change
  - Which predictors are associated with the response?
  - What is the relationship between the response and each predictor?

$\Rightarrow$ $\hat{f}$ is cannot be a **black box** anymore

# Example: prediction or inference paradigm?

Two policy makers :

- one **facing a drought**:
  - must decide whether to invest in a rain dance to increase the chance of rain.
- one **seeing clouds**:
  - must deciding whether to take an umbrella to work to avoid getting wet on the way home

$\rightarrow$ Both decisions could benefit from an empirical analysis on rain

*Which one relates to a causality / prediction problem?*

# Approach in social science

- *Objective*: Understanding the way that $Y$ is affected as $X_1, \ldots, X_p$ change
- The goal not necessarily to make predictions for $Y$
- Often linear function to estimate $Y$: $f(X) = \sum_{i=1}^{p} \beta_i x_i$
- Assume $\epsilon \sim N(0, \sigma^2)$
- Parameters $\beta$ are estimated by minimizing the sum of squared errors

$$Y = \sum_{i=1}^{p} \beta_i x_i + \epsilon$$

# Approach in social science: causality

$$Y = \beta_0 + \beta_1 T + \sum_{i=1}^{p-1} \beta_i x_i + \epsilon$$

- Interested in the values of one or two parameters and whether they are **causal** or not.
- Framework to interpret statistical causality: **Rubin (1974)**
- $\beta_1$ measures the extent to which $\Delta X_t$ will affect $\Delta Y_{t+1}$

## Approach in social science: causality

- Causal inference requires that $T \perp \epsilon$ or $T|X \perp \epsilon$

$\rightarrow$ can be achieved through randomization of $T$

- This implies that we are not really all that interested in choosing an optimal $f(.)$
- (We want to estimate unbiased coefficients)

# Approach in machine learning: prediction

$$\hat{Y} = \hat{f}(X)$$

- *Objectives*:
  - find the "best" $f(\cdot)$ and the "best" set of $X$'s which give the best predictions, $\hat{Y}$
  - **Accuracy:** find the function that **minimize the difference between *predicted* and *observed* values**
  - (We want to minimize prediction error)

# Reducible and irreducible error

$\hat{f}(X) = \hat{Y}$ estimated function

$f(X) + \epsilon = \hat{Y}$ true function

- **Reducible error**: $\hat{f}$ is used to estimate f, but not perfect
  - Accuracy can be improved by adding more features
- **Irreducible error**: $\epsilon$ = all other features that can be used to predict $f$
  - Unobserved $\rightarrow$ irreducible

## Reducible and irreducible error

$$
\begin{aligned}
E(Y - \hat{Y})^2 &= E[f(X) + \epsilon - \hat{f}(X)]^2 \\
&= \underbrace{[f(X) - \hat{f}(X)]^2}_{Reducible} + \overbrace{Var(\epsilon)}^{Irreducible}
\end{aligned}
$$

$\Rightarrow$ **Objective**: estimating $f$ with the aim of minimizing the reducible error

# How do we estimate $f$?

# Context

We use observations to "teach" our ML algorithm to predict outcomes

- **Training data**: $\{(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)\}$ where $x_i = (x_{i1}, x_{i2}, \ldots, x_{ip})^T$
- *Goal*: use the training data to estimate the unknown function $f$
- 2 types of SL methods: **parameteric vs. nonparametric**

# Parametric methods

**Model-based approaches**, 2 steps:

1. Specify a ***parametric* (functional) form** for $f(X)$, e.g. linear:

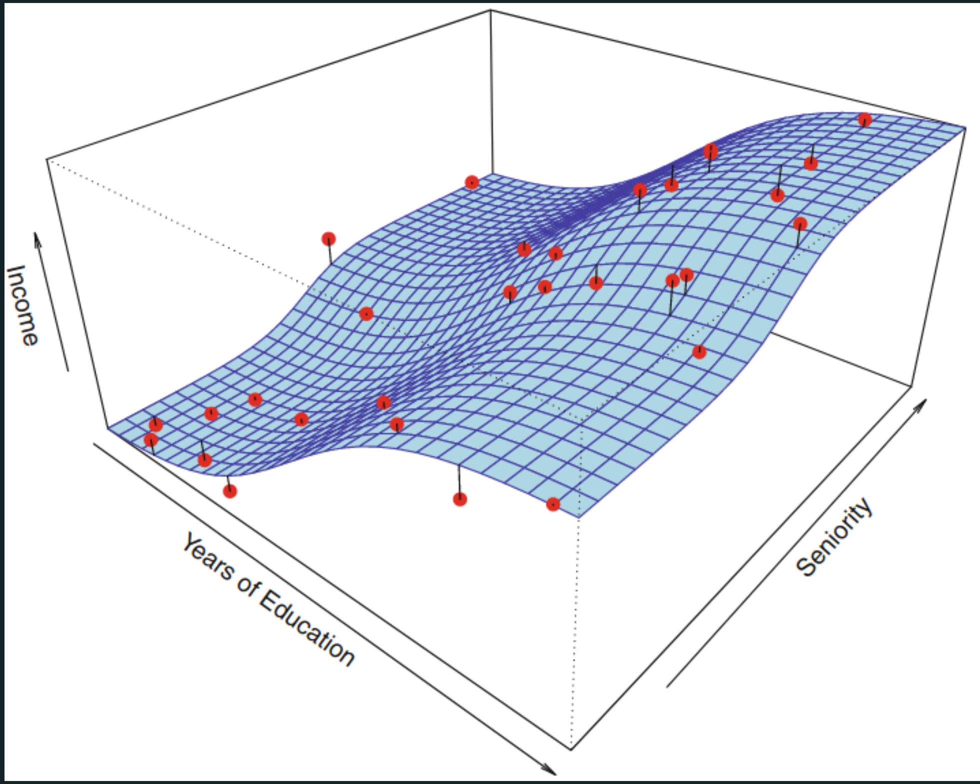
$$f(X) = \beta_0 + \beta_1 X_1 + \cdots + \beta_p X_p$$

   (Parametric means that the function depends on a finitenumber of parameters, here $p+1$).

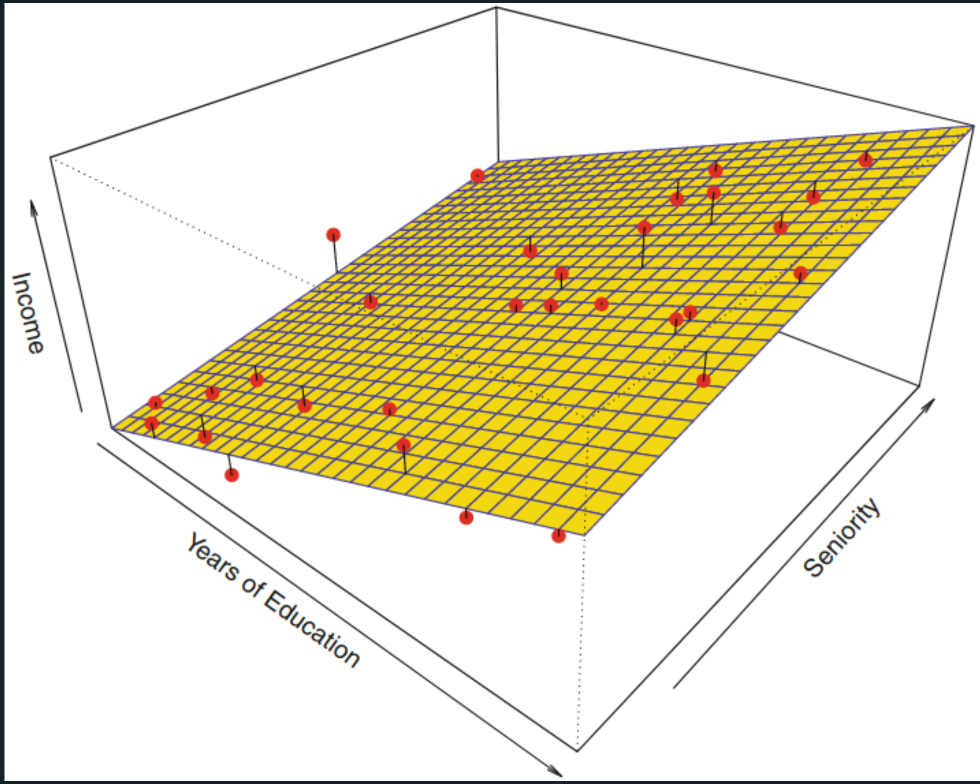
2. **Training**: Estimate the parameters by OLS and predict $Y$ by

$$\hat{Y} = \hat{f}(X) = \hat{\beta}_0 + \hat{\beta}_1 X_1 + \cdots + \hat{\beta}_p X_p$$

# True function



# Linear estimate



## Parametric methods -- issues

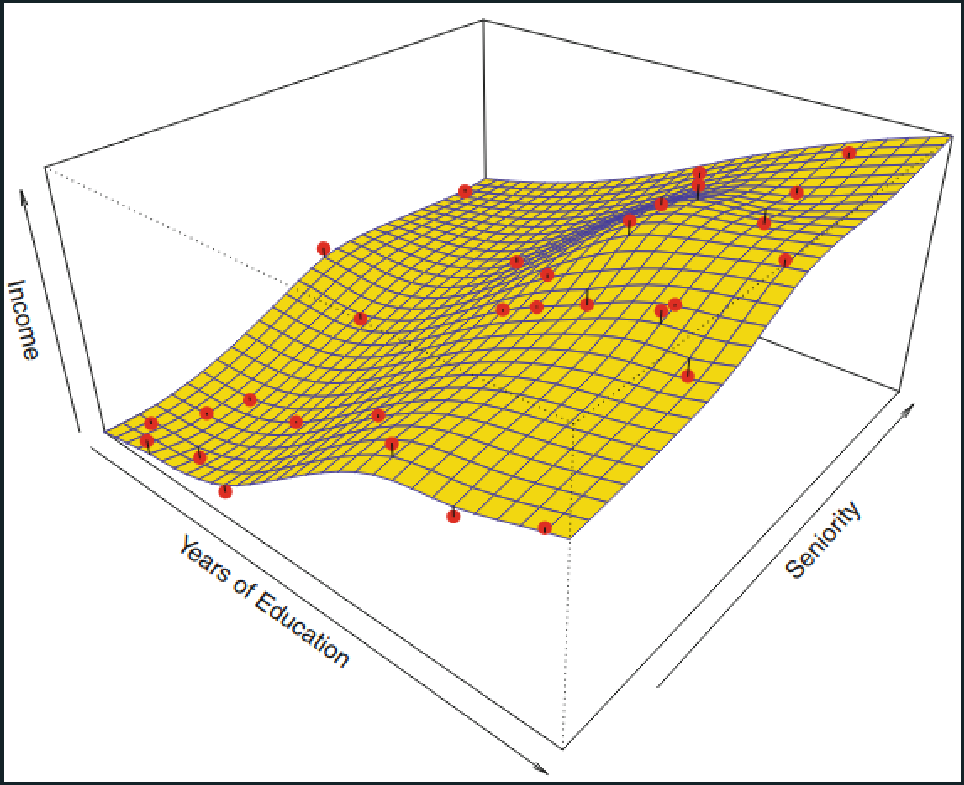
Misspecification of $f(X)$

1. Rigid models (e.g. strictly linear) may not fit the data well
2. More flexible models require more parameter estimation $\rightarrow$ **overfitting**

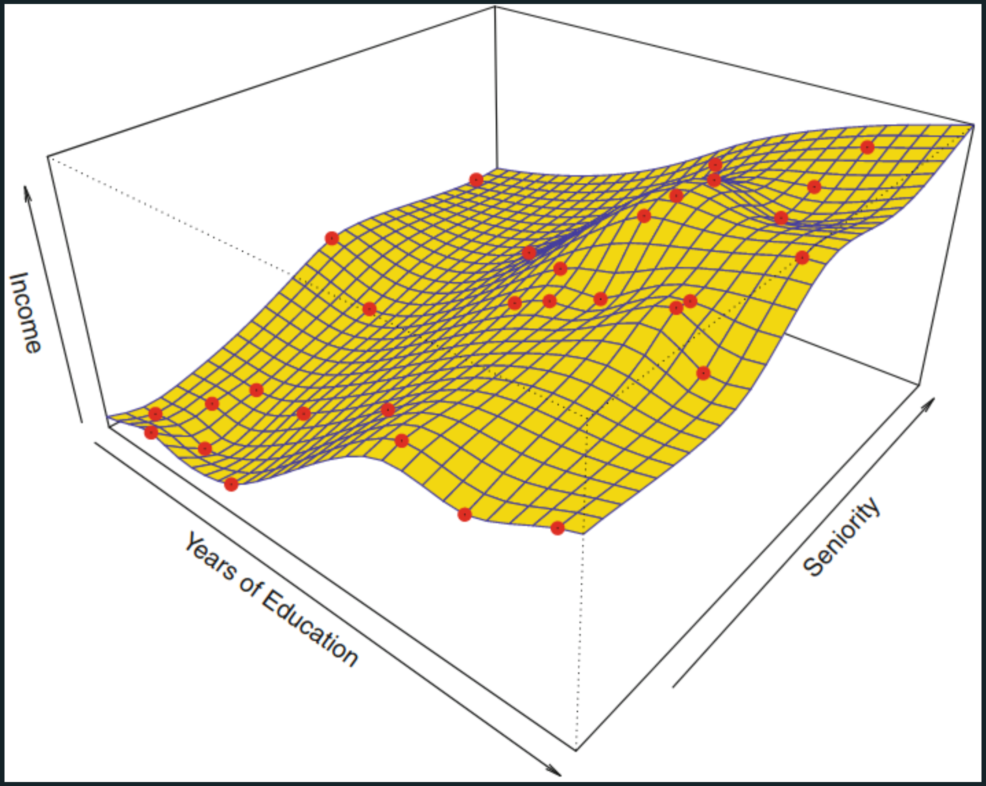
## Non-parametric methods

- **No assumptions** about the functional form of $f$
- Estimates a function only **based on the data itself**.
- **Disadvantage**: very large number of observations is required to obtain an accurate estimate of $f$

# “Smooth” nonlinear estimate



# Rough nonlinear estimate with perfect fit ⇒ overfit



# Accuracy and interpretability tradeoffs

- **More accurate** models often require estimating **more parameters** and/or having more flexible models
- Models that are better at prediction generally are **less interpretable**.

$\Rightarrow$ What we care about:

- For inference: interpretability.
- For prediction: accuracy.

# Model Accuracy

# Mean Squared Error (MSE)

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

- **Regression setting**: the **mean squared error** is a metric of how well a model fits the data.
- But it's **in-sample**.
- What we are really interested in is the **out-of-sample** fit!

# Measuring fit (1)

- We would like $(y_0 - \hat{f}(x_0))^2$ to be small for some $(y_0, x_0)$, not in our training sample $(x_i, y_i)_{i=1}^{n}$.
- Assume we had a large set of observations $(y_0, x_0)$ (a test sample),
- then we would like a low

$$Ave(y_0 - \hat{f}(x_0))^2$$

- i.e a low average squared prediction error (test MSE)

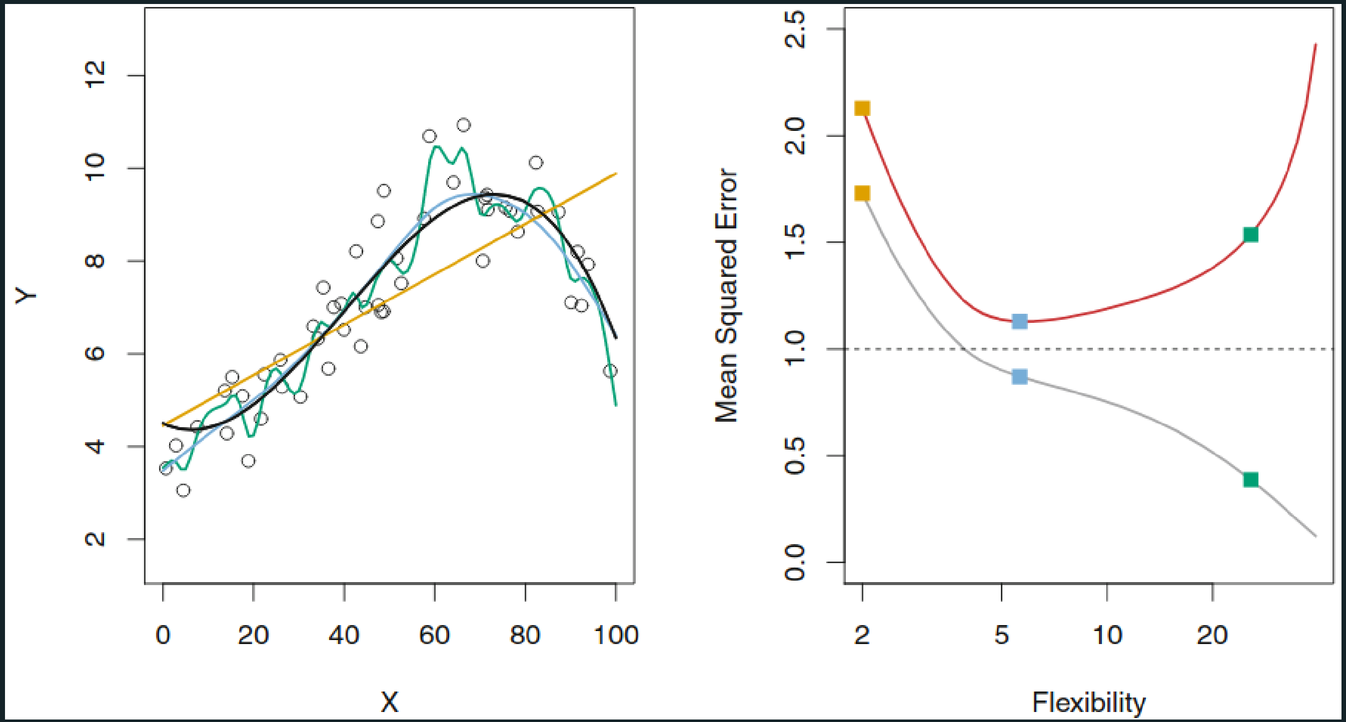
# Measuring fit (2)

To estimate model fit we need to partition the data:

1. **Training set**: data used to **fit** the model
    - **Training MSE**: how well our model fits the training data.
2. **Test set**: data used to **test the fit**
    - **Test MSE**: how well our model fits new data

We are most concerned in **minimizing test MSE**

# Training MSE, test MSE and model flexibility




Red (grey) curve is test (train) MSE

Increasing model flexibility tends to **decrease** training MSE but will eventually **increase** test MSE

# Overfitting

- As model flexibility increases, training MSE will decrease, but the test MSE may not.
- When a given method yields a small training MSE but a large test MSE, we are said to be **overfitting** the data.
- (We almost always expect the training MSE to be smaller than the test MSE)
- Estimating test MSE is important, but requires training data…

# The Bias-Variance Trade-Off

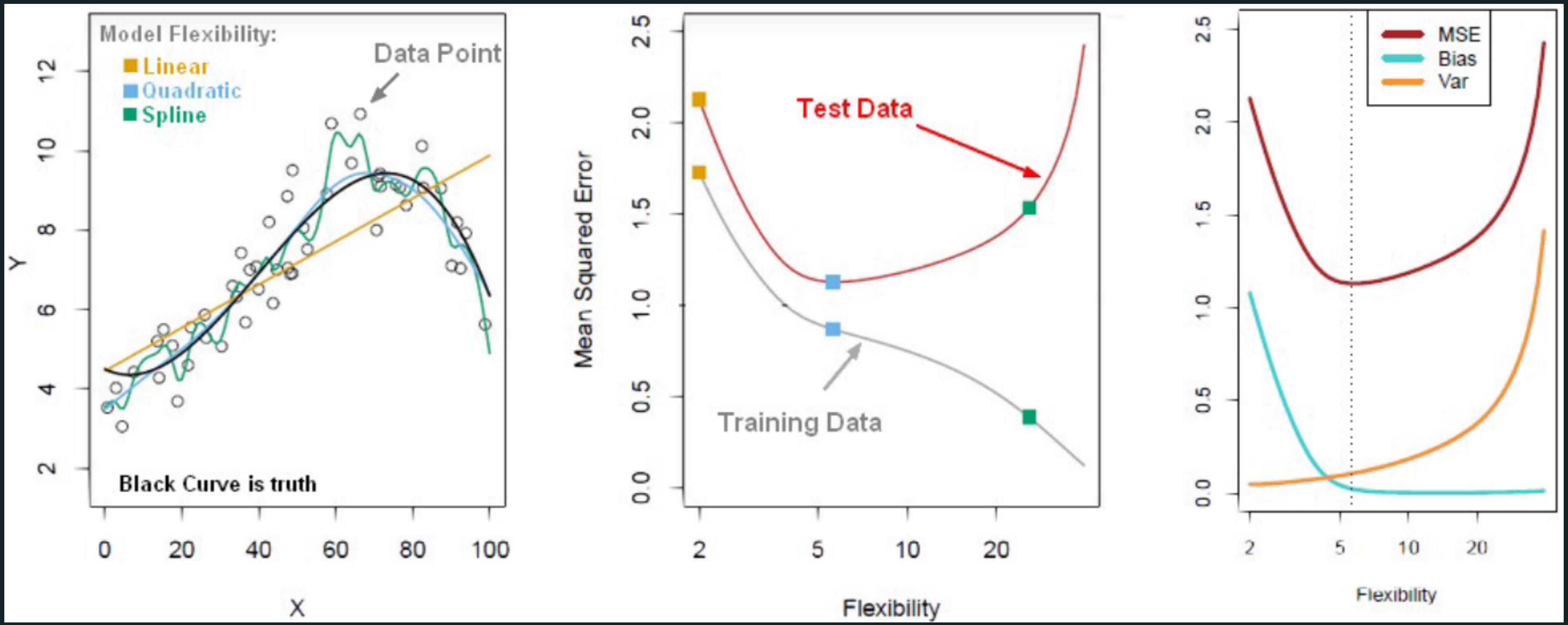
# Decomposing the expected (test) MSE

$$E(y_0 - \hat{f}(x_0))^2 = Var(\hat{f}(x_0)) + [Bias(\hat{f}(x_0))]^2 + Var(\epsilon)$$

3 components:

1. $Var(\hat{f}(x_0))$ = Variance of the predictions
    - how much would $\hat{f}$ change if we applied it to a different data set
2. $[Bias(\hat{f}(x_0))]^2$ = Bias of the predictions
    - how well does the model fit the data?
3. $Var(\epsilon)$ = variance of the error term

# The bias-variance tradeoff




- less flexibility → high bias and low variance
- more flexibility → low bias and high variance

Models that are too flexible or expressive or complex overfit!!

# Accuracy in Classifications

$$\text{(training) error rate} = \frac{1}{n}\sum_{i=1}^{n} 1(y_i \neq \hat{y}_i)$$

$$\text{(test) error rate} = Ave(1(y_0 \neq \hat{y}_0))$$

- MSE in the context of regression (continuous predictor).
- Modifications in the setting in which we're interested in prediction classes
- We are essentially interested in what % of classifications are correct.
- For cross-validation we could also use the estimated test error rate

# How to choose training and test set?

# Resamling methods

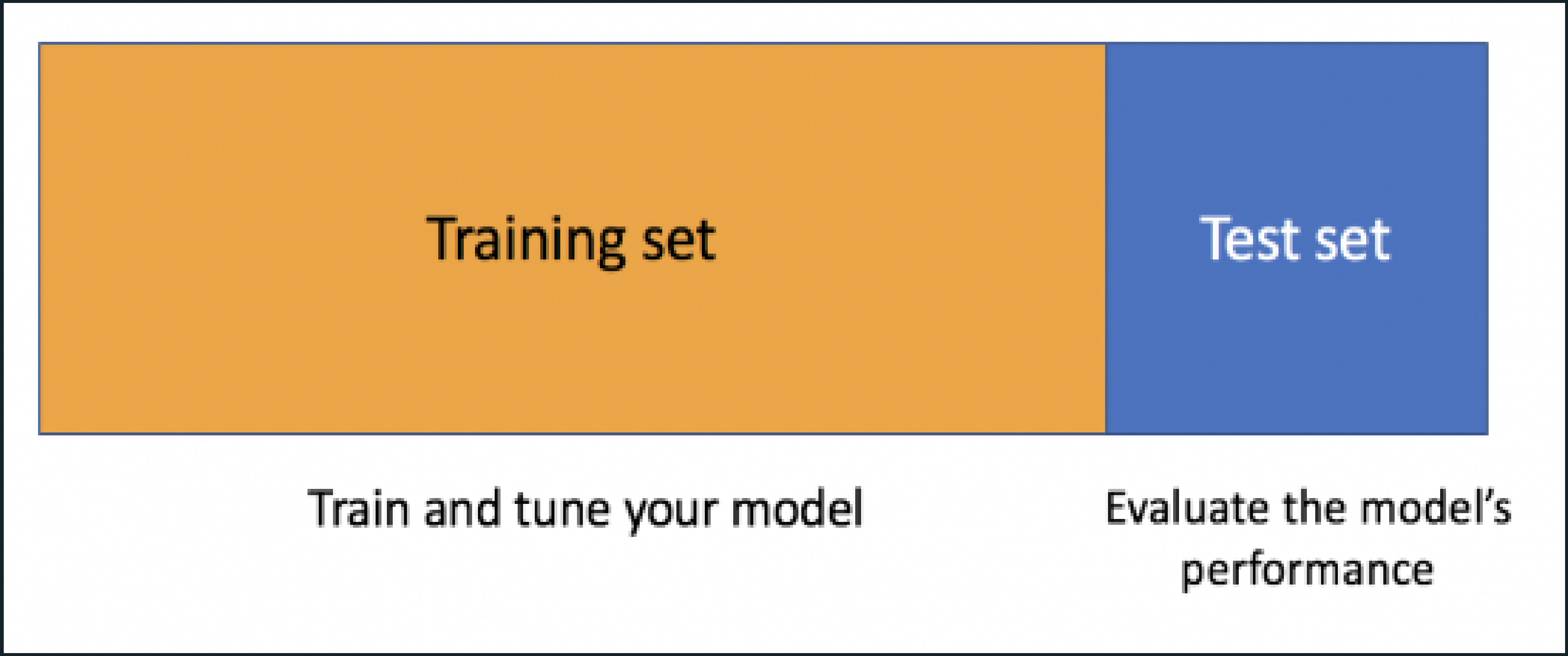
Estimate the test error rate by

**holding out** a subset of the training observations from the fitting process,

+ then **applying** the statistical learning method to those held out observations

# Validation set approach

- Labeled data **randomly** into two parts: training and test (validation) sets.



# Two concerns

- Arbitrariness of split
- Only use parts of the data for estimation

  $\rightarrow$ we tend to overestimate test MSE because our estimate of $f(x)$ is less precise

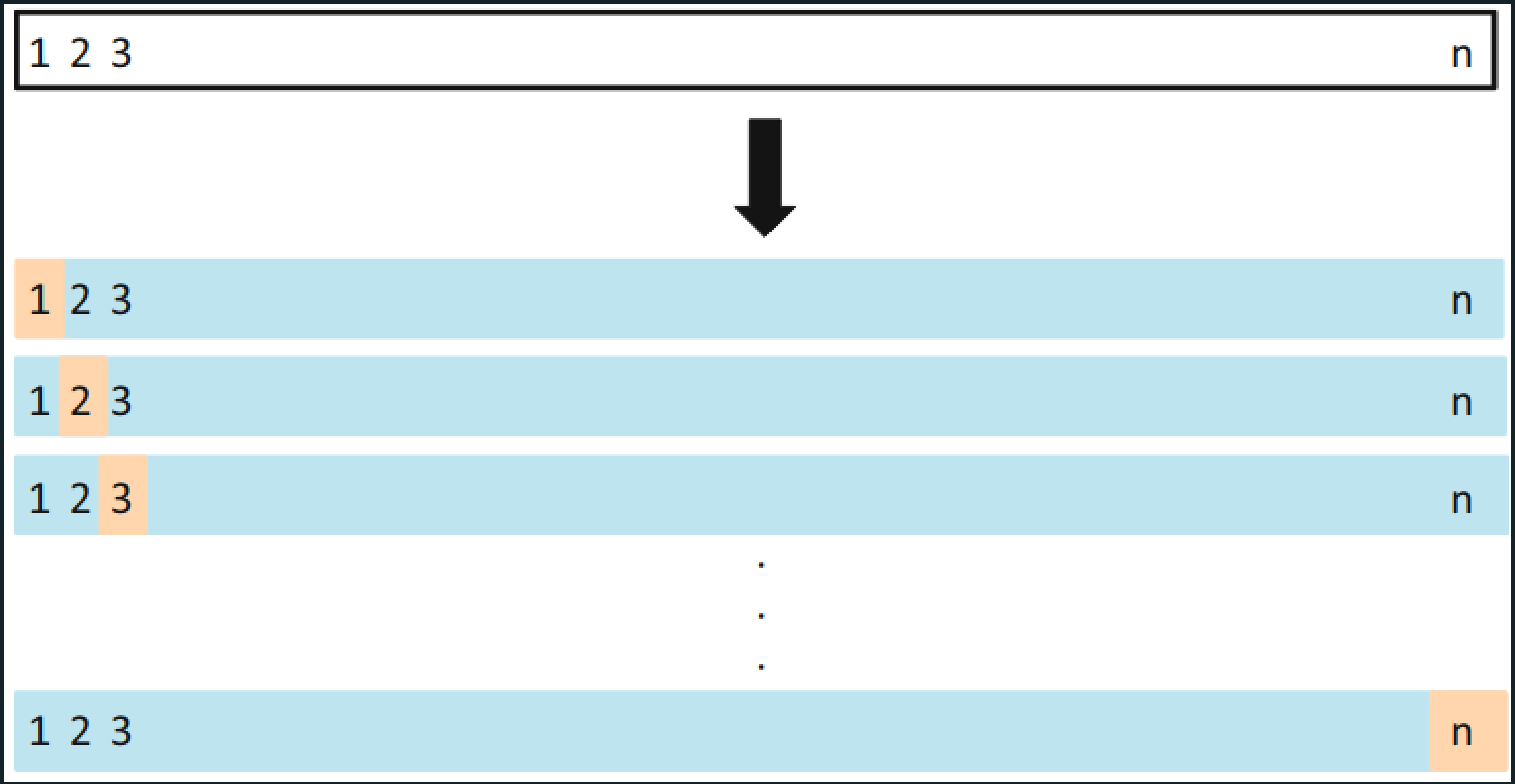
# Leave-One-Out Cross-Validation (LOOC)

- Fit on $n - 1$ training observations, and a prediction the Last
- Iterate $n$ times
- Assess the average model fit across each test set.

Estimate for the test MSE:

$$CV_n = \sum_{i=1}^{n} MSE_i$$

# Leave-One-Out Cross-Validation (LOOC)




- less bias than the validation set approach
- always yield the same results
- potentially too expensive to implement

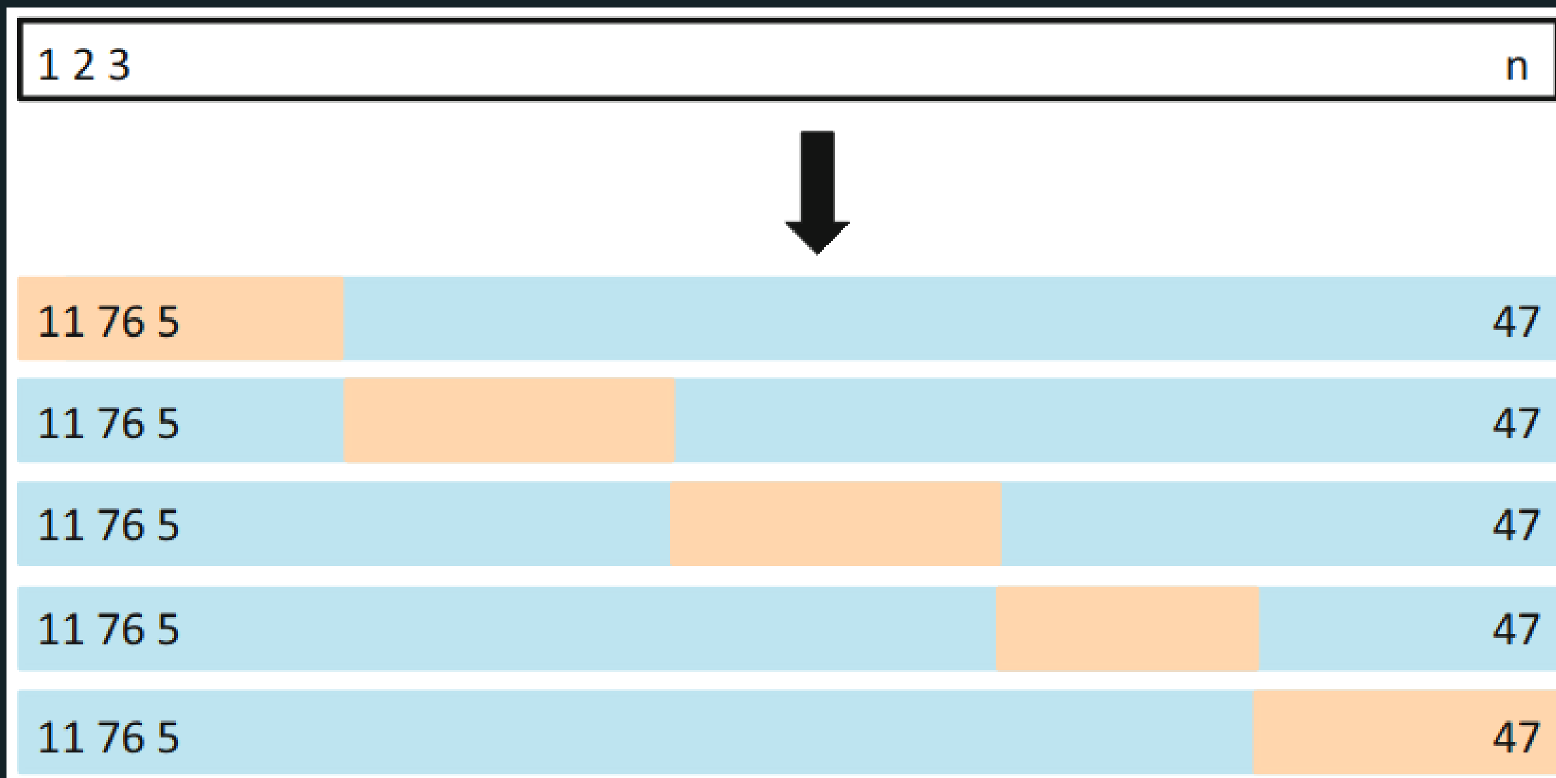
# $k$-fold Cross-validation

- Leave-One-Out Cross-Validation with $k = 1$
- Randomly dividing the data into the set of observations into $k$ groups
- 1st fold is treated as a validation set, and the method is fit on the remaining $k - 1$ folds
- Iterate $k$ times

Estimate for the test MSE:

$$CV_k = \sum_{i=1}^{k} MSE_i$$

# $k$-fold Cross-validation




⇒ Arguably the contribution to econom(etr)ics: Cross-validation (to estimate test MSE)!

# Bias-Variance Trade-Off $f$-Fold Cross-Validation

Bias

- **validation set approach** can lead to overestimates of the test error rate
- **1-fold validation:** almost unbiased estimates of the test error
- **k-fold validation** is in between

Variance

- **1-fold validation:** higher variance
- **k-fold validation:** lower variance

$k = 5$ or $k = 10$ is a good benchmark

# Conclusion:

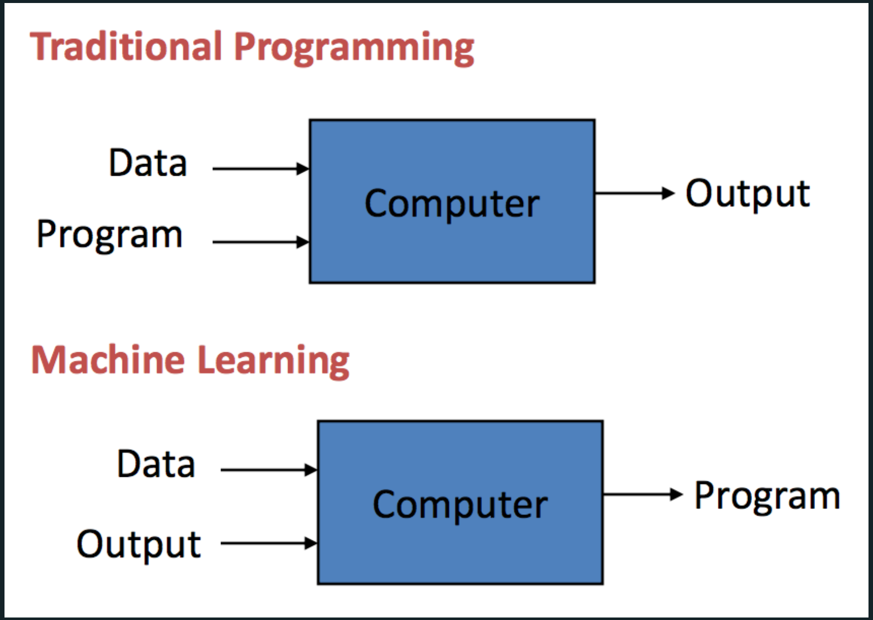
## Econometrics vs. Machine Learning

# Econometrics vs. Machine Learning (1)

- **Common objective**: to build a predictive model, for a variable of interest, using explanatory variables (or features)
- **Different cultures**:
  - *E*: probabilistic models designed to describe economic phenomena
  - *ML*: algorithms capable of learning from their mistakes

Charpentier A., Flachaire, E. & Ly, A. (2018). Econometrics and Machine Learning. *Economics and Statistics*, 505-506, 147–169.

# Econometrics vs. Machine Learning (2)




- **Classical computer programming:** humans input the rules and the data, and the computer provides answers.
- **Machine learning:** humans input the data and the answer, and the computer learns the rules.

# The Machine learning workflow

1. Look at the big picture.
2. Get the data.
3. Discover and visualize the data to gain insights.
4. Prepare the data for Machine Learning algorithms.
5. Select a model and train it.
6. Fine-tune your model.
7. Present your solution.
8. Launch, monitor, and maintain your system

Aurelien Geron, *Hands-on machine learning with Scikit-Learn & TensorFlow*, Chapter 2