S&DS 265 / 565

**Introductory Machine Learning**

# Classification and Regression Concepts

September 10

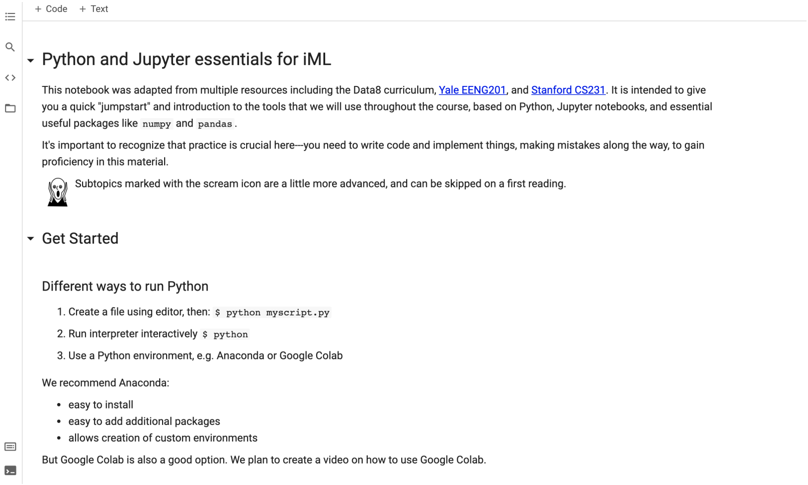
# Logistics

- Assignment 1 has been posted
- Quiz 1 grades and solutions available on Canvas
- Lowest quiz score will be dropped
- Check Canvas / EdD for office hours, Zoom link

# Recall: Last week

- Python elements
- Pandas and linear regression example

# Python elements



## Python and Jupyter essentials for iML

This notebook was adapted from multiple resources including the Data8 curriculum, Yale EENG201, and Stanford CS231. It is intended to give you a quick "jumpstart" and introduction to the tools that we will use throughout the course, based on Python, Jupyter notebooks, and essential useful packages like `numpy` and `pandas`.

It's important to recognize that practice is crucial here---you need to write code and implement things, making mistakes along the way, to gain proficiency in this material.

Subtopics marked with the scream icon are a little more advanced, and can be skipped on a first reading.

## Get Started

### Different ways to run Python

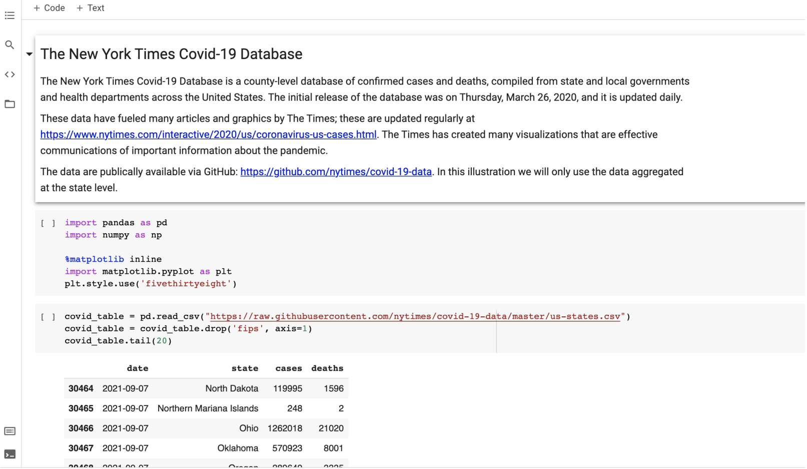
1. Create a file using editor, then: `$ python myscript.py`
2. Run interpreter interactively `$ python`
3. Use a Python environment, e.g. Anaconda or Google Colab

We recommend Anaconda:

- easy to install
- easy to add additional packages
- allows creation of custom environments

But Google Colab is also a good option. We plan to create a video on how to use Google Colab.

# Pandas example



## The New York Times Covid-19 Database

The New York Times Covid-19 Database is a county-level database of confirmed cases and deaths, compiled from state and local governments and health departments across the United States. The initial release of the database was on Thursday, March 26, 2020, and it is updated daily.

These data have fueled many articles and graphics by The Times; these are updated regularly at https://www.nytimes.com/interactive/2020/us/coronavirus-us-cases.html. The Times has created many visualizations that are effective communications of important information about the pandemic.

The data are publically available via GitHub: https://github.com/nytimes/covid-19-data. In this illustration we will only use the data aggregated at the state level.

```
[ ] import pandas as pd
    import numpy as np

    %matplotlib inline
    import matplotlib.pyplot as plt
    plt.style.use('fivethirtyeight')
```

```
[ ] covid_table = pd.read_csv("https://raw.githubusercontent.com/nytimes/covid-19-data/master/us-states.csv")
    covid_table = covid_table.drop('fips', axis=1)
    covid_table.tail(20)
```

| | date | state | cases | deaths |
|---|---|---|---|---|
| **30464** | 2021-09-07 | North Dakota | 119995 | 1596 |
| **30465** | 2021-09-07 | Northern Mariana Islands | 248 | 2 |
| **30466** | 2021-09-07 | Ohio | 1262018 | 21020 |
| **30467** | 2021-09-07 | Oklahoma | 570923 | 8001 |
| **30468** | 2021-09-07 | Oregon | 289640 | 3335 |

# This week

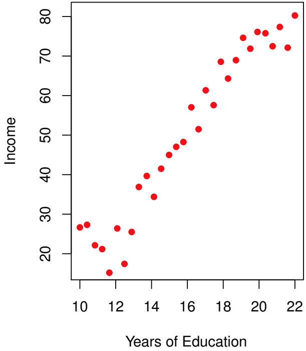
- Overfitting
- Comparing linear and *k*-NN regression
- Classification concepts
- Further examples

# Some Terminology

- supervised vs. unsupervised
- classification vs. regression
- prediction vs. inference

# Regression Example

The `Income` dataset:



Quantitative response $Y$

Predictors $X = (X_1, \ldots, X_p)$

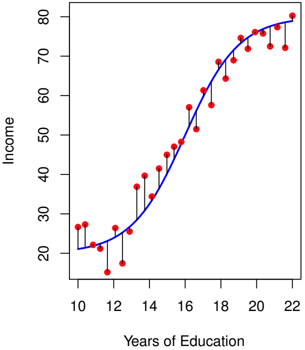
Assume the relationship can be expressed by:

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

where $f$ is a fixed, unknown function and $\epsilon$ is error term.

# Regression Example

The `Income` dataset:



Quantitative response $Y$

Predictors $X = (X_1, \ldots, X_p)$

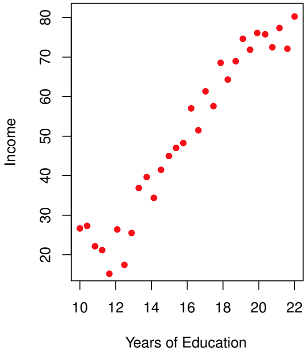
Assume the relationship can be expressed by:

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

where $f$ is a fixed, unknown function and $\epsilon$ is error term.

# Regression Example

Back to regression with $p = 1$:



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

Modeling:

Use a procedure to get $\widehat{f}$. Derive estimates $\widehat{Y} = \widehat{f}(X)$.

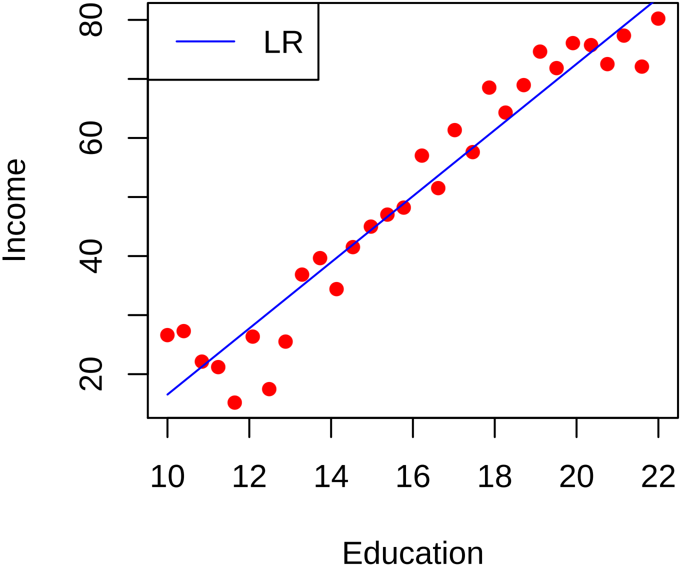
# Possible Regression Approaches

- linear regression
  - Fitting a straight line through the data.
- *k*-nearest neighbors regression
  - Average together the $y_i$ for $x_i$ close to $x$

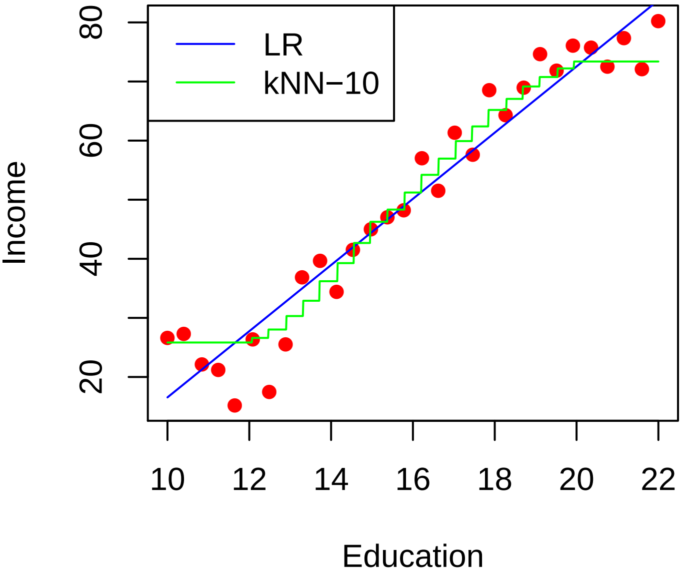
# Americans Are Losing Faith in the Value of College. Whose Fault Is That?

For most people, the new economics of higher ed make going to college a risky bet.

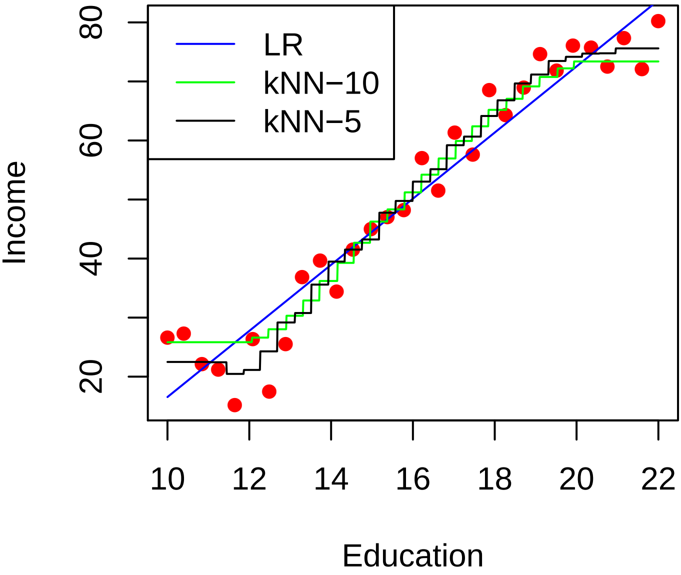
# Possible Regression Approaches



# Possible Regression Approaches

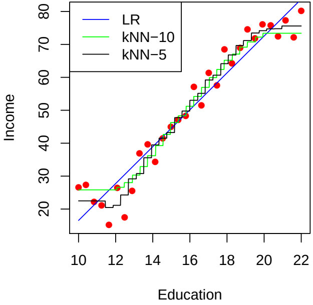


# Possible Regression Approaches



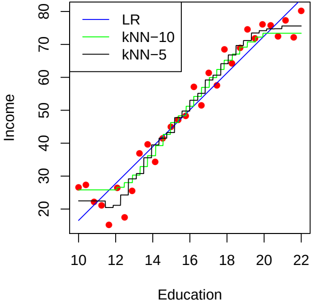
# Possible Regression Approaches

Measuring performance via **Mean Squared Error**



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

# Possible Regression Approaches



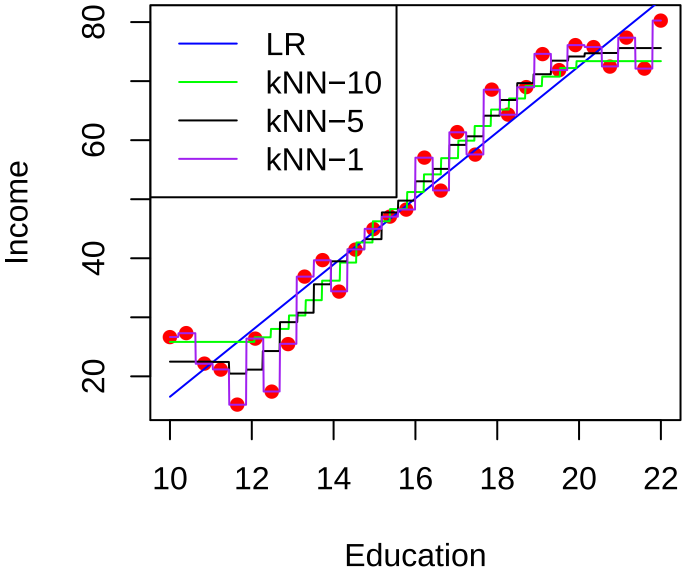
Measuring performance via **Mean Squared Error**

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

MSEs for three methods:

| | |
|---|---|
| Linear Regression | 29.829 |
| k-Nearest Neighbors (k=10) | 23.519 |
| k-Nearest Neighbors (k=5) | 16.21 |

A $k$-nearest neighbors model with $k = 5$ achieves lowest error. Is it the best?


LR
kNN−10
kNN−5
kNN−1
Income
20
40
60
80
Education
10
12
14
16
18
20
22

# Training MSE vs. Test MSE

MSE in the previous table, **training MSE**, was computed based on data used in fitting the model.

We are more interested in **test MSE** computed on *unseen data*.

# Training MSE vs. Test MSE

MSE in the previous table, **training MSE**, was computed based on data used in fitting the model.

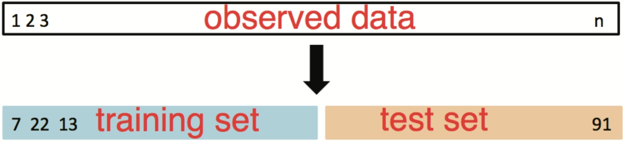
We are more interested in **test MSE** computed on *unseen data*.
What if we don’t have other data?

# Training MSE vs. Test MSE

MSE in the previous table, **training MSE**, was computed based on data used in fitting the model.

We are more interested in **test MSE** computed on *unseen data*. What if we don't have other data?

We can randomly split our data into a test set and a training set.



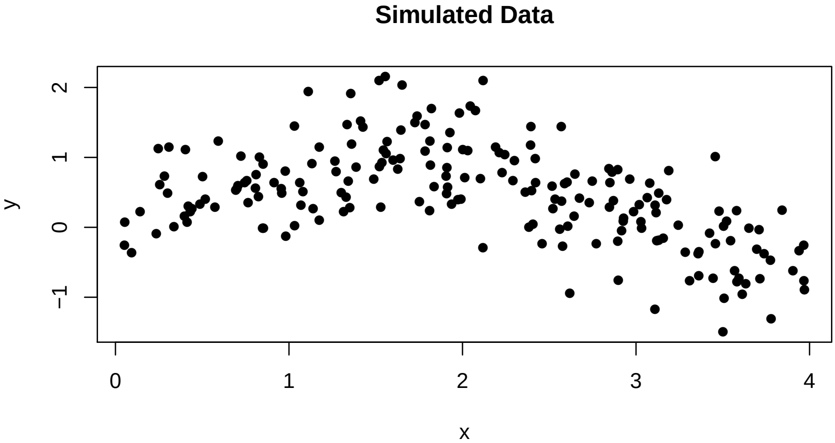
# Overfitting

A method is **overfitting** the data when it has a small training MSE but a large test MSE.

# Overfitting

A method is **overfitting** the data when it has a small training MSE but a large test MSE.

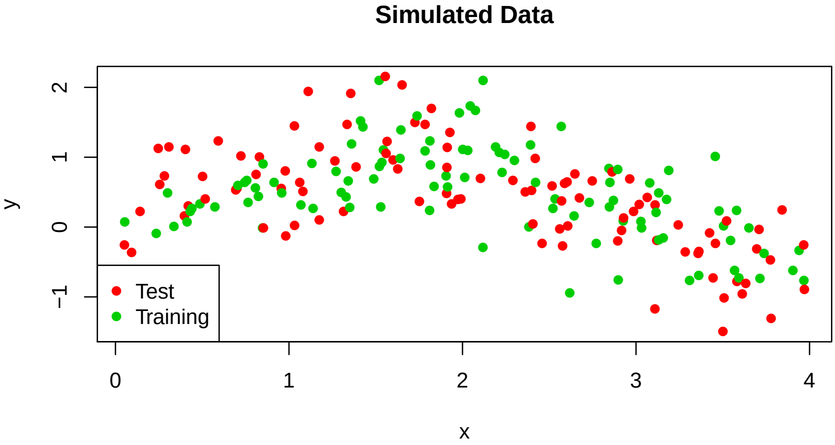
Let’s examine this phenomenon using a bigger dataset:



# Overfitting

A method is **overfitting** the data when it has a small training MSE but a large test MSE.

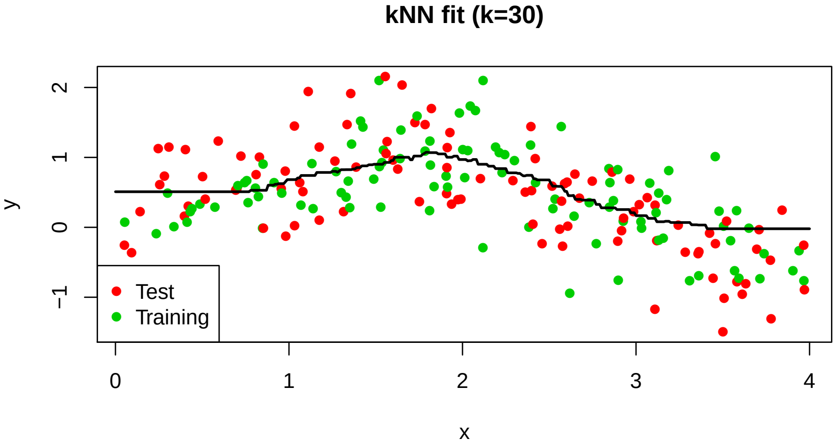
Let's examine this phenomenon using a bigger dataset:



# Overfitting

A method is **overfitting** the data when it has a small training MSE but a large test MSE.

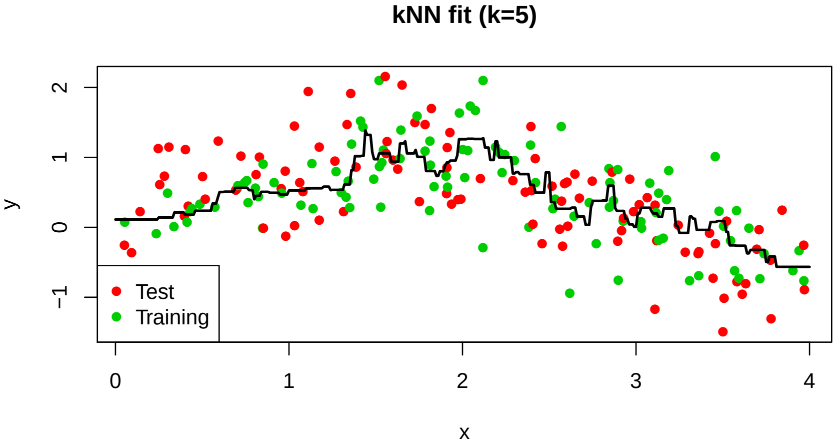
Let's examine this phenomenon using a bigger dataset:



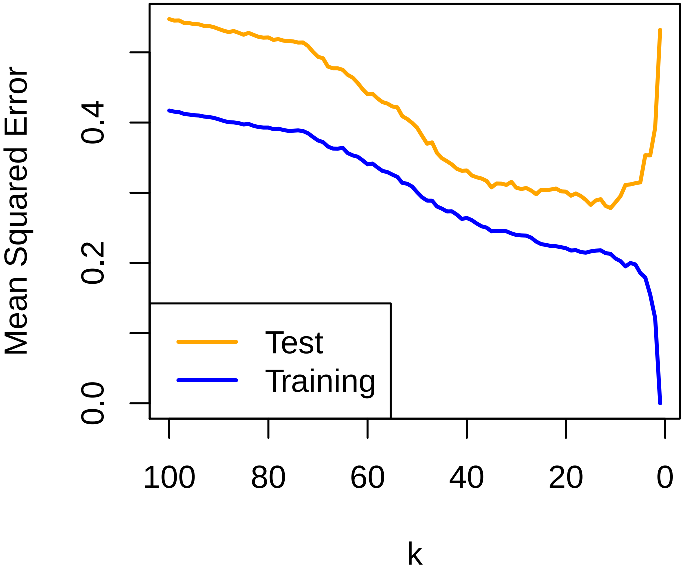
# Overfitting

A method is **overfitting** the data when it has a small training MSE but a large test MSE.

Let's examine this phenomenon using a bigger dataset:



# Overfitting via k-Nearest Neighbors



# *k*-NN vs Linear regression

- *k*-NN is called a “nonparametric” method
- You’ll get practice on this for classification on Assn 1
- Linear regression is a “parametric” method
- Let’s talk about it in more detail

# Linear regression: Why start here?

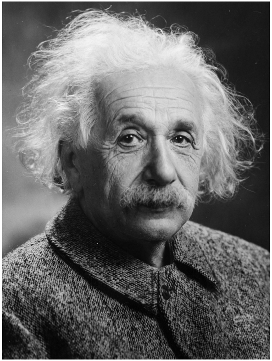
- Linear regression is foundation for more sophisticated topics:
    - Regularization
    - Support vector machines
    - Neural networks

# Linear regression: Why start here?

- Linear regression is foundation for more sophisticated topics:
  - Regularization
  - Support vector machines
  - Neural networks
- Many advanced machine learning methods are generalizations or extensions of linear regression

# Linear regression: Why start here?

- Linear regression is foundation for more sophisticated topics:
  - Regularization
  - Support vector machines
  - Neural networks
- Many advanced machine learning methods are generalizations or extensions of linear regression
- A good place to start — Bay Area traffic story



*Everything should be made as simple as possible, but no simpler.*

https://en.wikiquote.org/wiki/Albert_Einstein

# Estimating the coefficients

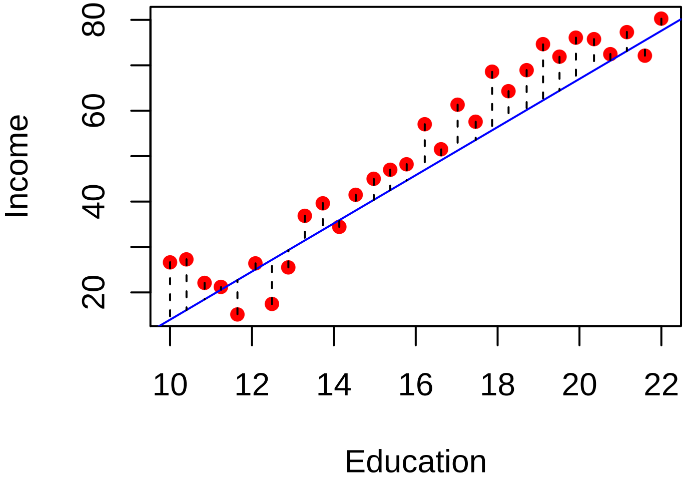
For any $\widehat{\beta}_0, \widehat{\beta}_1$, we predict $\widehat{y}_i = \widehat{\beta}_0 + \widehat{\beta}_1 x_i$. We call these **fitted values.**

# Estimating the coefficients

For any $\widehat{\beta}_0, \widehat{\beta}_1$, we predict $\widehat{y}_i = \widehat{\beta}_0 + \widehat{\beta}_1 x_i$. We call these **fitted values.**

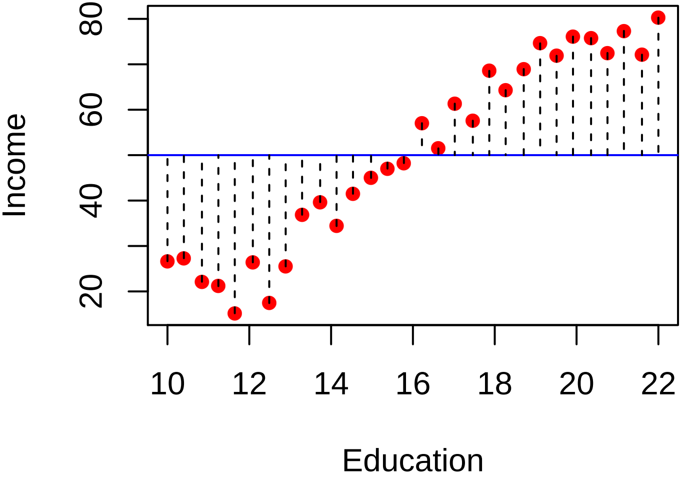
The **residual** $e_i = y_i - \widehat{y}_i$ is difference between the *i*-th observed value and its fitted value.

# Some candidate lines (and residuals)



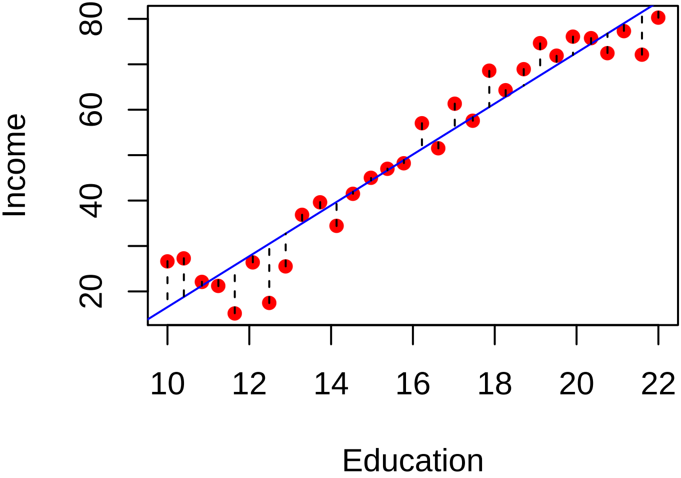
$$\widehat{\beta}_0 = -39, \widehat{\beta}_1 = 5.3$$

# Some candidate lines (and residuals)



$$\widehat{\beta}_0 = 50, \widehat{\beta}_1 = 0$$

# Some candidate lines (and residuals)



$$\widehat{\beta}_0 = -39.4, \widehat{\beta}_1 = 5.6$$

# Estimating the coefficients

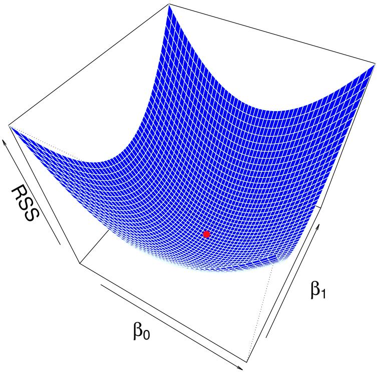
The **least squares** approach selects coefficients $\widehat{\beta}_0$ and $\widehat{\beta}_1$ that minimize the **residual sum of squares** (RSS):

$$RSS = \sum_{i=1}^{n} e_i^2 = \sum_{i=1}^{n} (y_i - \widehat{y}_i)^2.$$

# Estimating the coefficients

The **least squares** approach selects coefficients $\widehat{\beta}_0$ and $\widehat{\beta}_1$ that minimize the **residual sum of squares** (RSS):

$$RSS(\beta_0, \beta_1) = \sum_{i=1}^{n} e_i^2 = (y_1 - \beta_0 - \beta_1 x_1)^2 + \cdots + (y_n - \beta_0 - \beta_1 x_n)^2.$$



# Estimating the coefficients

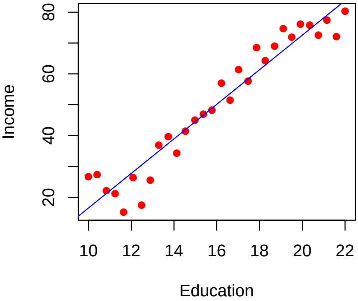
The **least squares** approach selects coefficients $\widehat{\beta}_0$ and $\widehat{\beta}_1$ that minimize the **residual sum of squares** (RSS):

$$RSS(\beta_0, \beta_1) = \sum_{i=1}^{n} e_i^2 = (y_1 - \beta_0 - \beta_1 x_1)^2 + \cdots + (y_n - \beta_0 - \beta_1 x_n)^2.$$

How do we find the minimum?

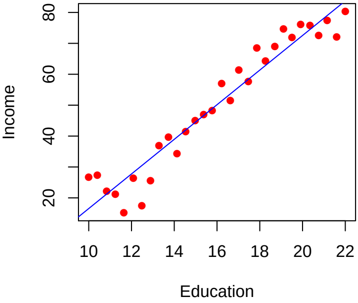
- A little calculus and algebra...
- Or optimization

# Simulated income dataset



$$\widehat{\beta}_0 = -39.45 \qquad \widehat{\beta}_1 = 5.60$$

# Simulated income dataset



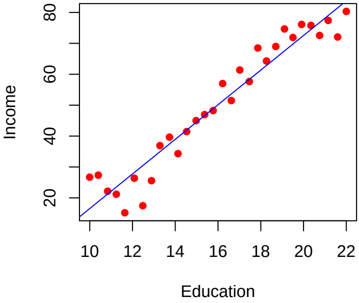
$$\widehat{\beta}_0 = -39.45 \qquad \widehat{\beta}_1 = 5.60$$

$$\widehat{y} = -39.45 + 5.60x$$

Interpretation:

- A one-year increase in education is associated with an increase in average income of 5.6 units.

# Simulated income dataset



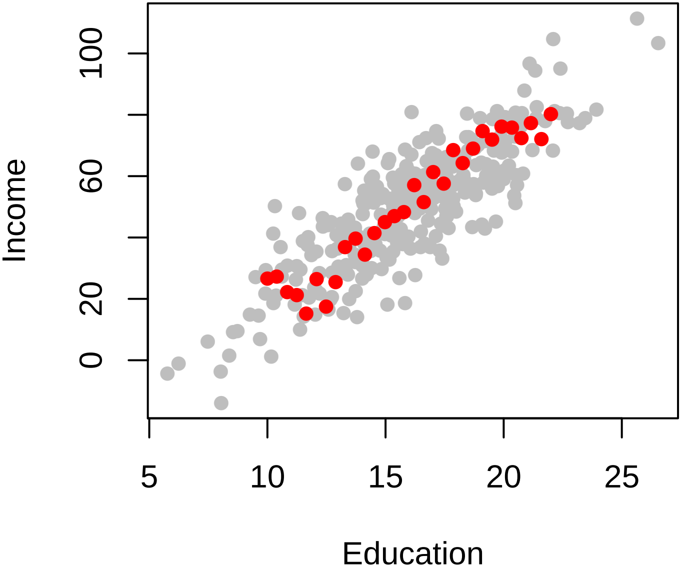
$$\widehat{\beta}_0 = -39.45 \quad \widehat{\beta}_1 = 5.60$$

$$\widehat{Income} = -39.45 + 5.60 \cdot Education$$

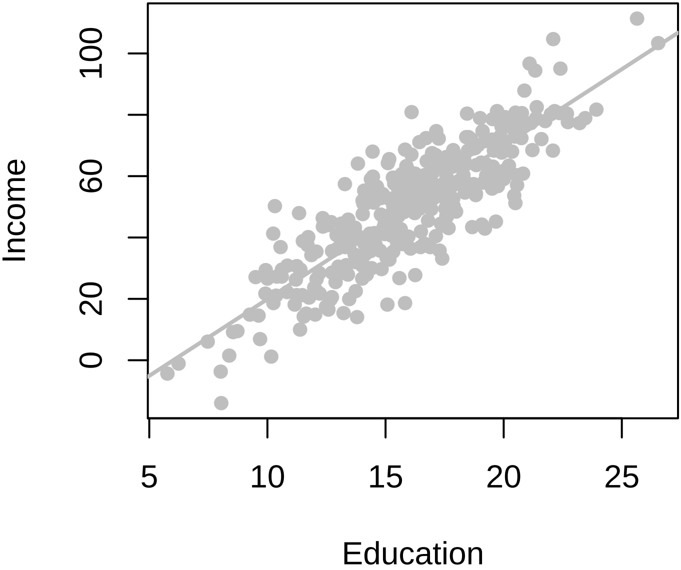
Interpretation:

- A one-year increase in education is associated with an increase in average income of 5.6 units.

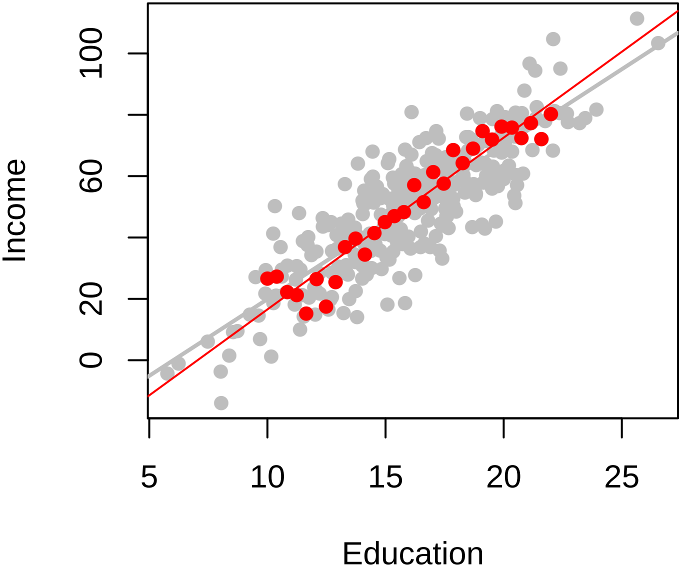
# Population vs. sample



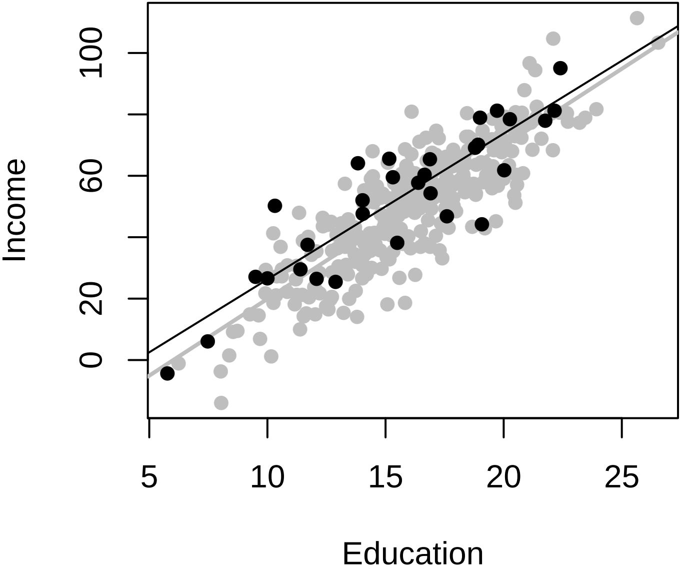
# Population vs. sample



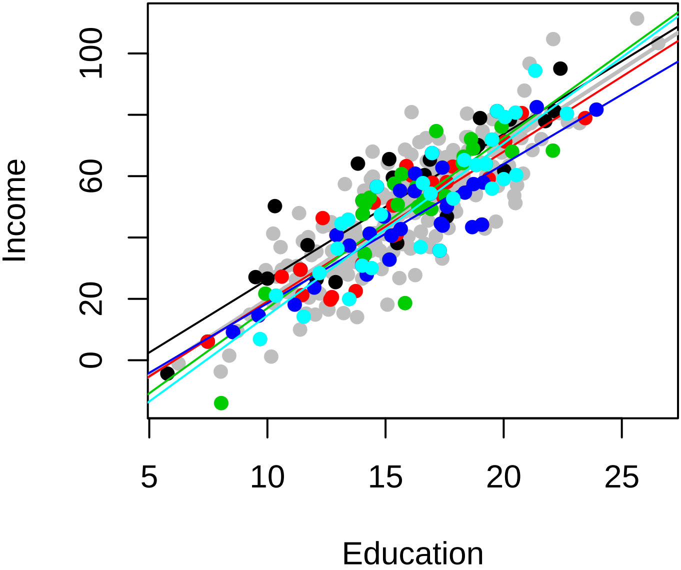
# Population vs. sample



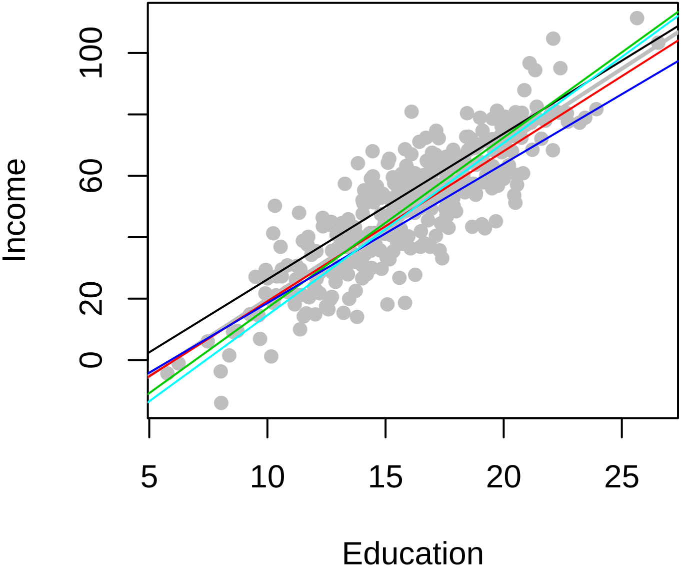
# Different samples



# Different samples



# Different samples



# Sums of squares and $R^2$

Partitioning the sums of squares:

$$\underbrace{\sum(y_i - \bar{y})^2}_{\text{total sum of squares}(TSS)} = \underbrace{\sum(\widehat{y}_i - \bar{y})^2}_{\text{explained sum of squares}(ESS)} + \underbrace{\sum(y_i - \widehat{y}_i)^2}_{\text{residual sum of squares}(RSS)}$$

for least squares linear regression (as some algebra shows):

# Sums of squares and $R^2$

Partitioning the sums of squares:

$$\underbrace{\sum (y_i - \bar{y})^2}_{\text{total sum of squares}(TSS)} = \underbrace{\sum (\widehat{y}_i - \bar{y})^2}_{\text{explained sum of squares}(ESS)} + \underbrace{\sum (y_i - \widehat{y}_i)^2}_{\text{residual sum of squares}(RSS)}$$

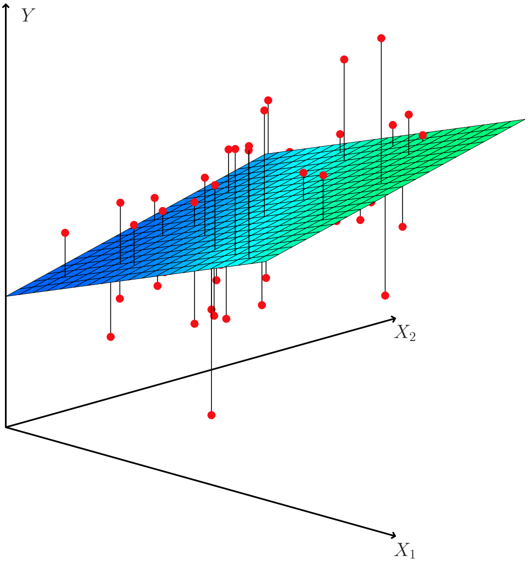
for least squares linear regression (as some algebra shows):

$$R^2 = \frac{ESS}{TSS} = 1 - \frac{RSS}{TSS}$$

We can interpret $R^2$ (**multiple R-squared**) as the proportion of variability in $y$ explained by the model.

- Between 0 and 1
- Doesn't depend on the scale of $Y$.

# Multiple linear regression



# General form

With $p$ predictors $x_1, \ldots, x_p$,

$$y = \beta_0 + \beta_1 x_1 + \cdots + \beta_p x_p + \epsilon,$$

where $\epsilon \sim N(0, \sigma^2)$.

In matrix notation,

$$\begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{pmatrix} = \begin{pmatrix} 1 & x_{1,1} & x_{1,2} & \cdots & x_{1,p} \\ 1 & x_{2,1} & \ddots & & x_{2,p} \\ \vdots & & \ddots & \vdots & \\ 1 & x_{n,1} & x_{n,2} & \cdots & x_{n,p} \end{pmatrix} \begin{pmatrix} \beta_0 \\ \beta_1 \\ \beta_2 \\ \vdots \\ \beta_p \end{pmatrix} + \begin{pmatrix} \epsilon_1 \\ \epsilon_2 \\ \vdots \\ \epsilon_n \end{pmatrix}$$

# General form

With $p$ predictors $x_1, \ldots, x_p$,

$$y = \beta_0 + \beta_1 x_1 + \cdots + \beta_p x_p + \epsilon,$$

where $\epsilon \sim N(0, \sigma^2)$.

In matrix notation,

$$y = X\beta + \epsilon$$

(where the intercept $\beta_0$ corresponds to a column of all 1s)

# Estimating $\beta$

Recall that

$$\widehat{\beta} = \arg\min_{\beta} \mathit{RSS}(\beta).$$

Compute derivatives of $\mathit{RSS}(\beta)$ with respect to $\beta_i$ and set equal to 0.

# Estimating $\beta$

Recall that

$$\widehat{\beta} = \arg\min_{\beta} RSS(\beta).$$

Compute derivatives of $RSS(\beta)$ with respect to $\beta_i$ and set equal to 0.

The $\beta$ that minimizes $RSS(\beta)$ satisfies the **normal equations**:

$$X^T X\beta = X^T y.$$

# Estimating $\beta$

Recall that

$$\widehat{\beta} = \arg\min_{\beta} RSS(\beta).$$

Compute derivatives of $RSS(\beta)$ with respect to $\beta_i$ and set equal to 0.

The $\beta$ that minimizes $RSS(\beta)$ satisfies the **normal equations**:

$$X^T X\beta = X^T y.$$

If the matrix $X^T X$ is invertible, solve to get

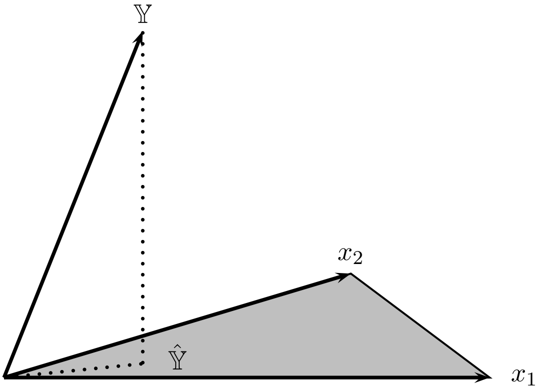
$$\widehat{\beta} = (X^T X)^{-1} X^T y.$$

# Interpretation

The coefficients are just the correlations between the variables $X_j$ and the data $Y$—*after* the variables are “whitened” to become uncorrelated.

# For the geometrically inclined

The **predicted values** (aka **fitted values**) $\widehat{Y} = X\widehat{\beta}$ are the projection of the data $Y \in \mathbb{R}^n$ onto the span of columns $X_1, X_2, \ldots, X_p \in \mathbb{R}^n$



# Discussion

Questions?

# Working with Covid-19 Data

Let’s revisit the Covid-19 example with the new notebook
`covid-trends-revisited.ipynb`

# Summary

- Least squares coefficients correspond to minimum of a quadratic surface
- $R^2$ is a scale-invariant accuracy measure — proportion of variance in $Y$ explained by the model
- Multiple linear regression (many predictors) estimated by solving a linear system, or by optimization