INTRODUCTION TO MACHINE LEARNING

Introduction

Pablo Cañas
pablocanas97@gmail.com

NORTHIN SUMMER SCHOOL
Barcelona, July 2024

Course structure

1. **Introduction:** basic concepts, data preparation, linear regression
2. **Fundamental Machine Learning concepts**: Loss functions, optimization, cross-validation, overfitting
3. **Supervised learning**: naive bayes, k-NN, random forests, …
4. **Unsupervised learning**: clustering, dimensionality reduction.
5. **Introduction to Deep Learning**

What is Machine learning?

- Machine Learning is the science of getting computers to learn and act like humans do, and improve their learning over time in autonomous fashion, by feeding them data and information in the form of observations and real-world interactions

- Machine learning algorithms seek to provide knowledge to computers through **data**, observations, and interaction with the world. It is then used to make **accurate predictions** given new observations.

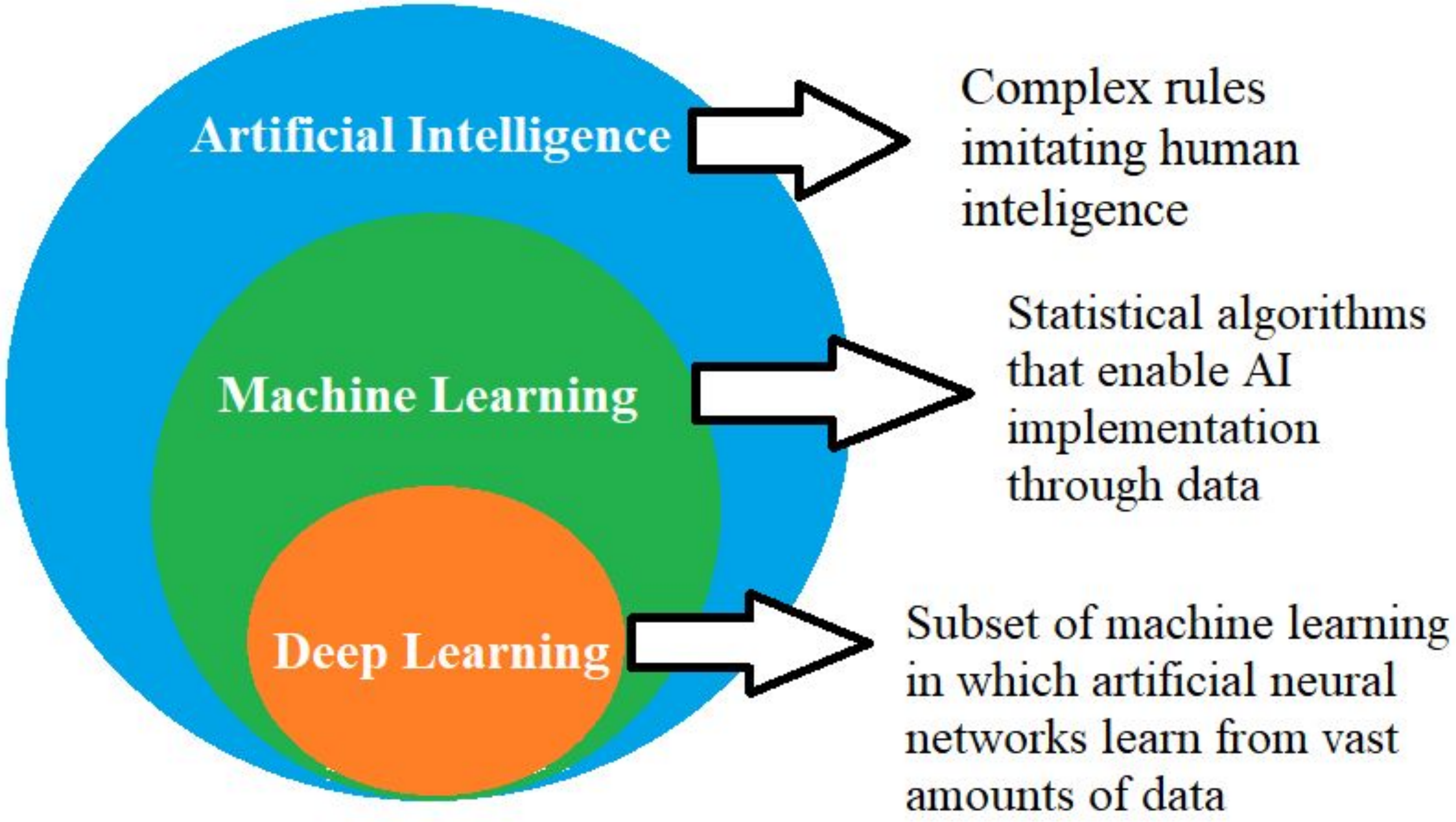
- Machine learning is applied statistics!

Machine Learning/Deep Learning/ AI

Buzzwords, a lot of confusion. What are the differences?

Machine Learning/Deep Learning/ AI

Buzzwords, a lot of confusion. What are the differences?

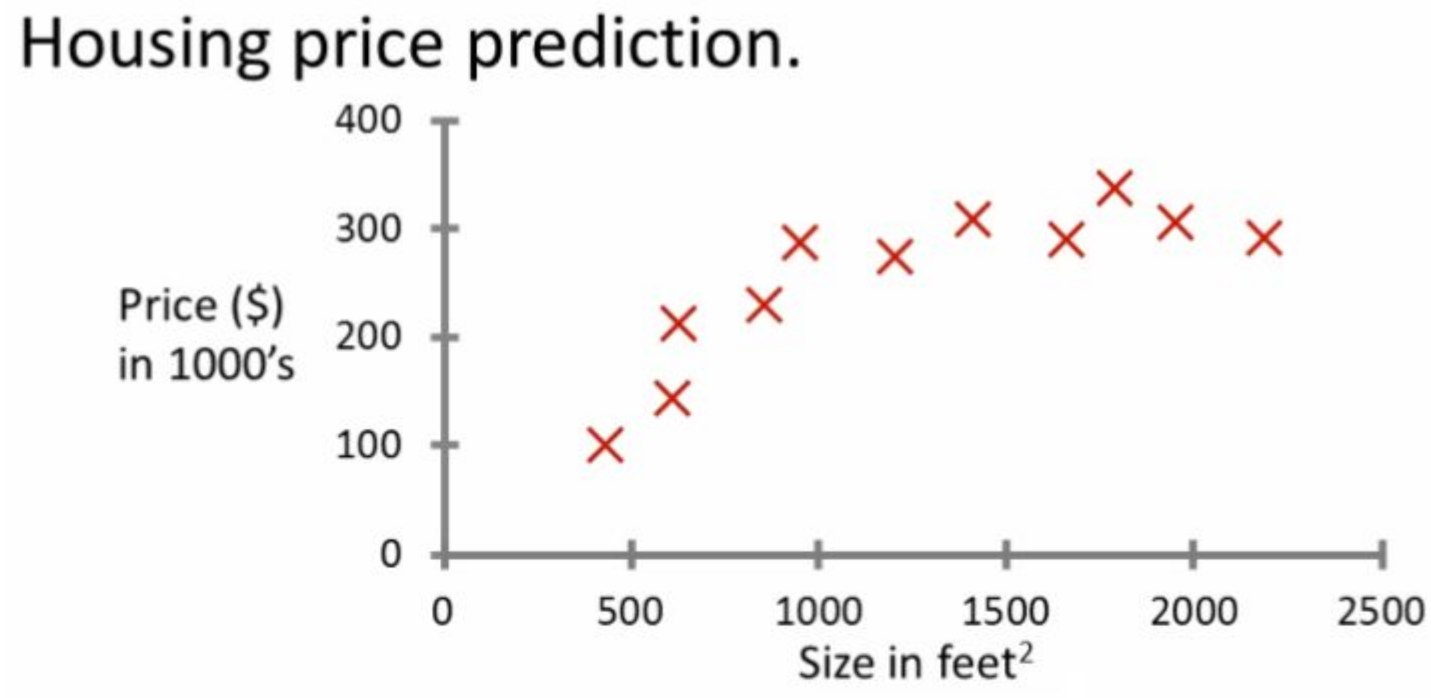


Machine learning algorithms

- **Supervised**: Models are **trained** with **input/output** pairs (X, y) which we relate via a function $y = f(X)$. Model **learns** f to make predictions on new data inputs X.
  - **Classification:** predictions/outputs y are discrete (class labels)
  - **Regression:** y is continuous
- **Unsupervised**: Only **inputs** X are given. We compute f such that $y = f(X)$ is a "simpler" representation
  - **Clustering:** discrete y (groups)
  - **Dimensionality reduction**: continuous y

Supervised Learning: Regression

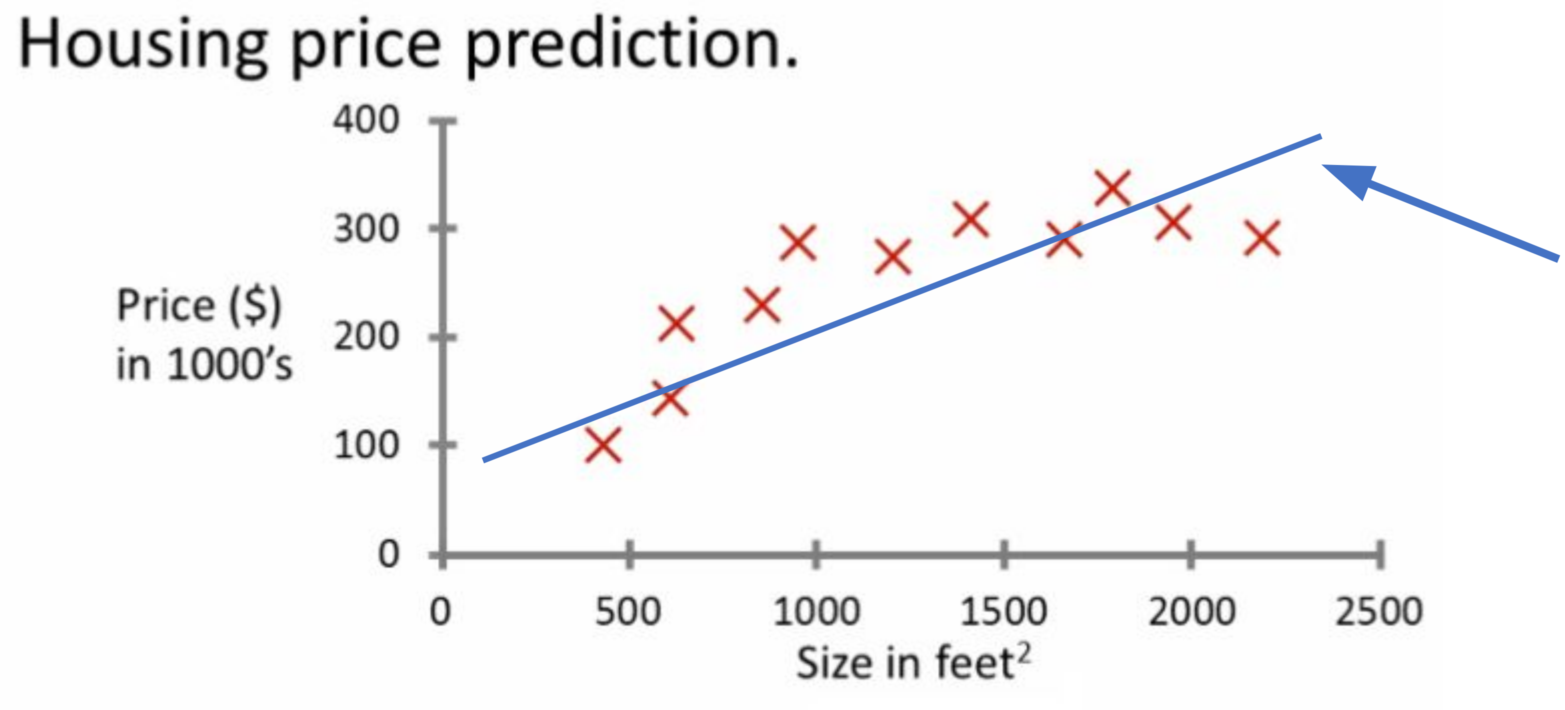
- Predict **continuous** valued output. Ex: Housing price prediction.



Source: Coursera

Supervised Learning: Regression

- Predict **continuous** valued output. Ex: Housing price prediction.

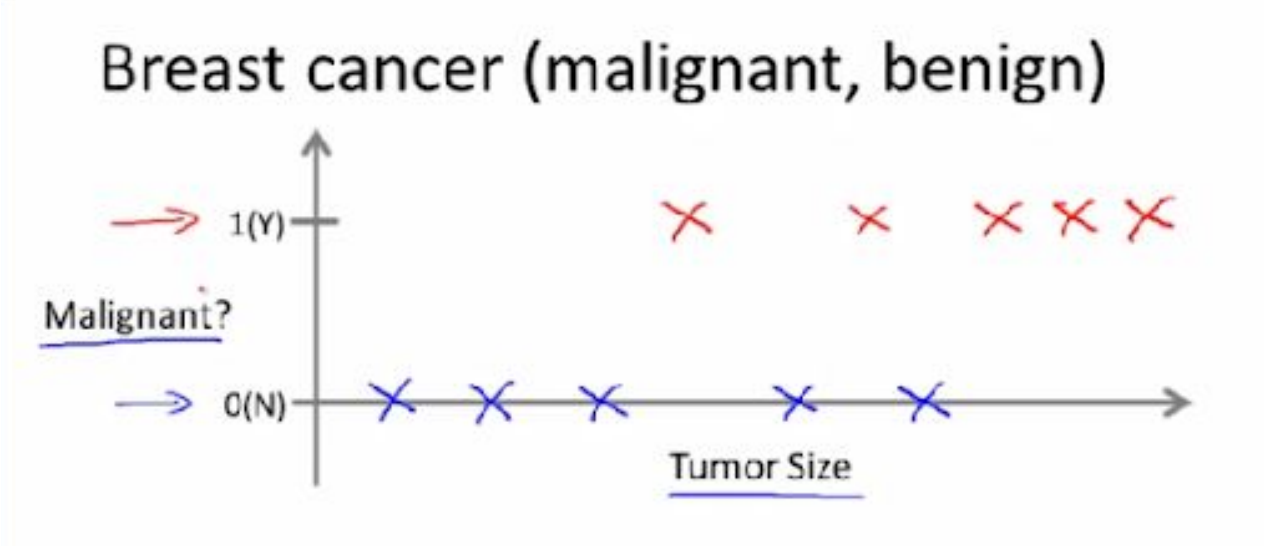


Linear regression model, predict price

Source: Coursera

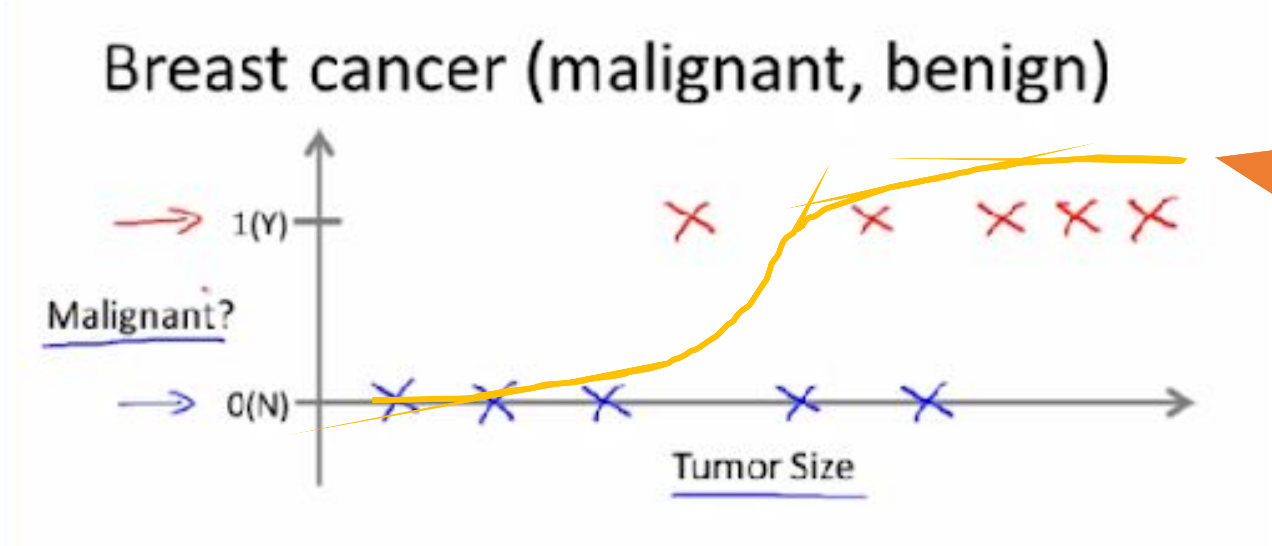
Supervised Learning: Classification

- Predict **discrete** valued output. Ex: Breast cancer diagnose.



Supervised Learning: Classification

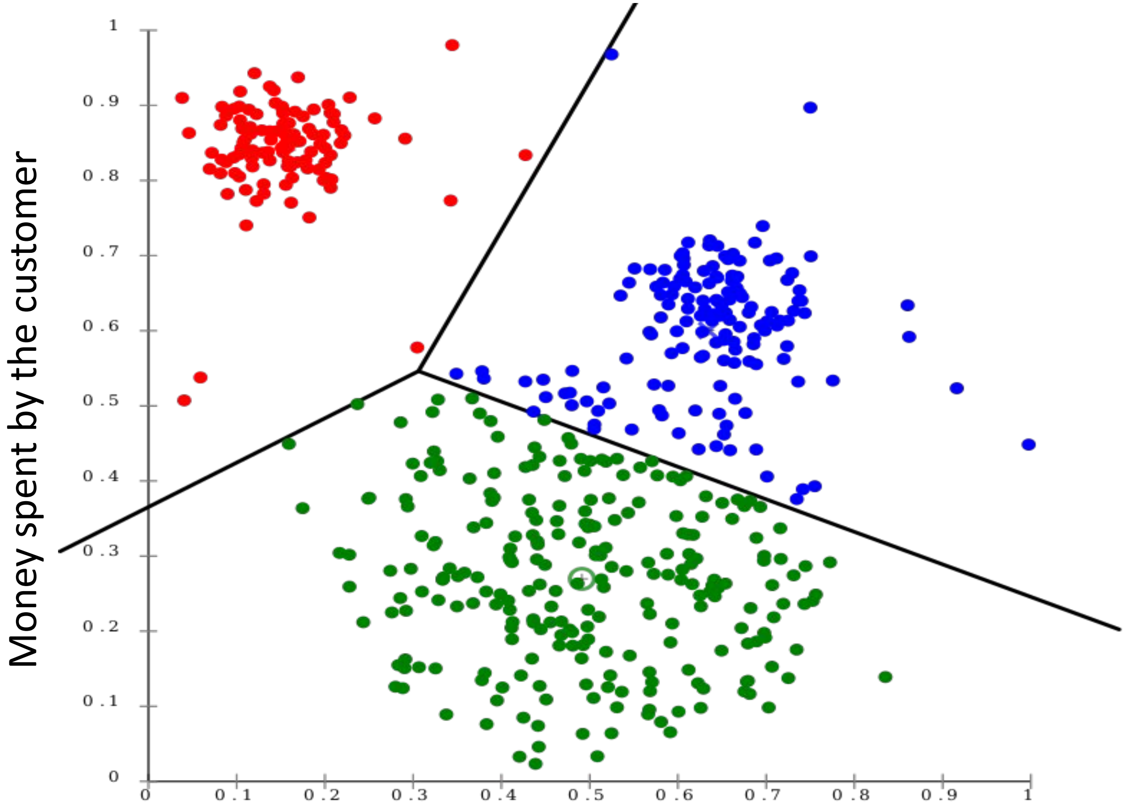
- Predict **discrete** valued output. Ex: Breast cancer diagnose $y \in \{0,1\}$.



Logistic regression model, predict class probabilities

Unsupervised Learning: Clustering

Customer segments

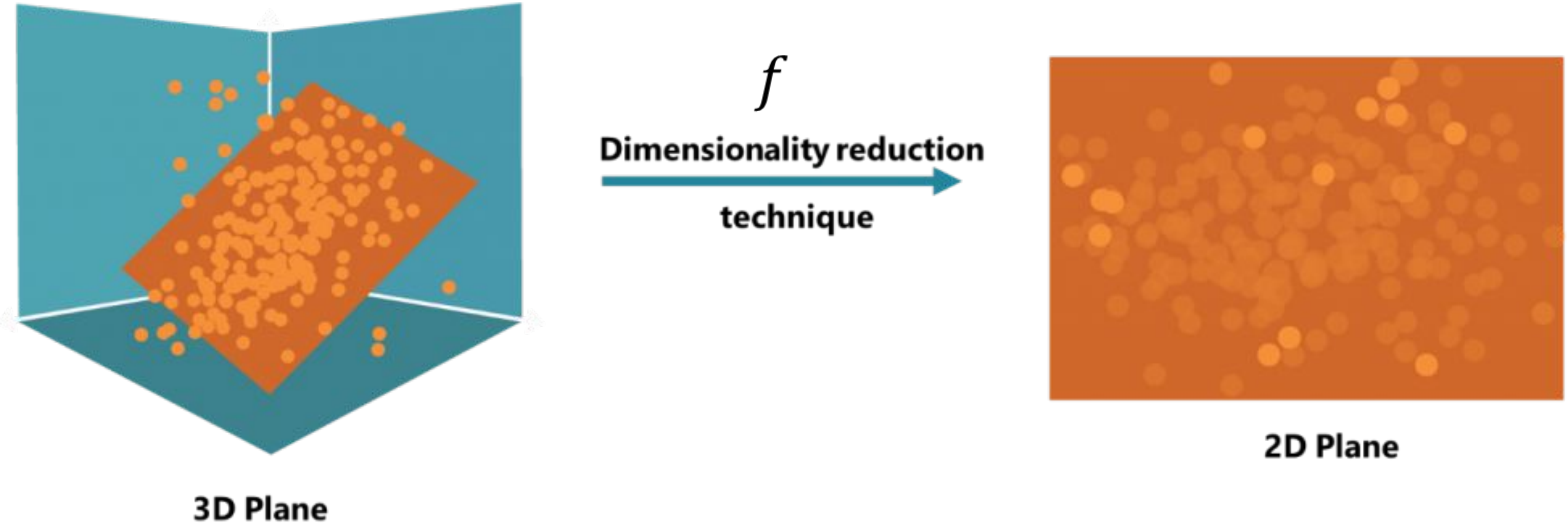


Find natural **clusters** or groups that share **similar** features. Ex: customer segmentation $y \in \{0,1,2 \ldots\}$.

- Red: Poor-buyers
- Green: Middle class-non-buyers
- Blue: Rich-buyers

Unsupervised Learning: Dimension reduction

- **Reduce** number of **features**, while keeping the maximum information
- **Reduce** the **complexity** of the problem!



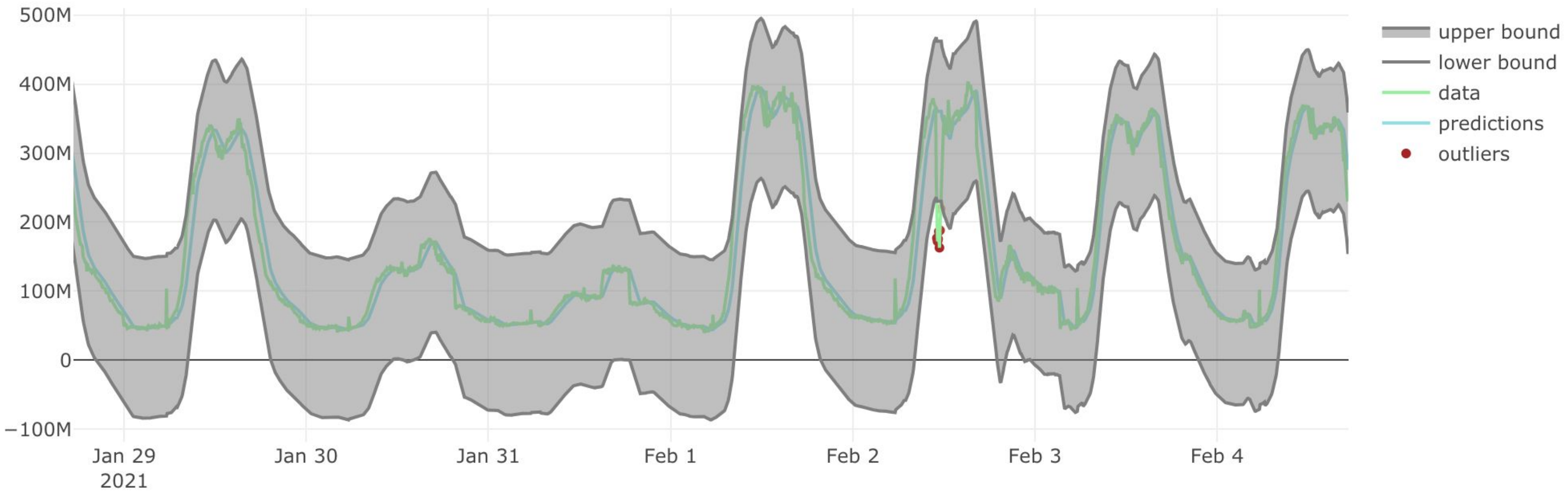
Quiz: Supervised/Unsupervised

- Imagine a Telecomm company wants to **automatically** detect outages or **failures** in the network by identifying when there is a **drop** or **spike** in the network **traffic**.

Should we use a Supervised or Unsupervised algorithm?

Quiz: Supervised/Unsupervised

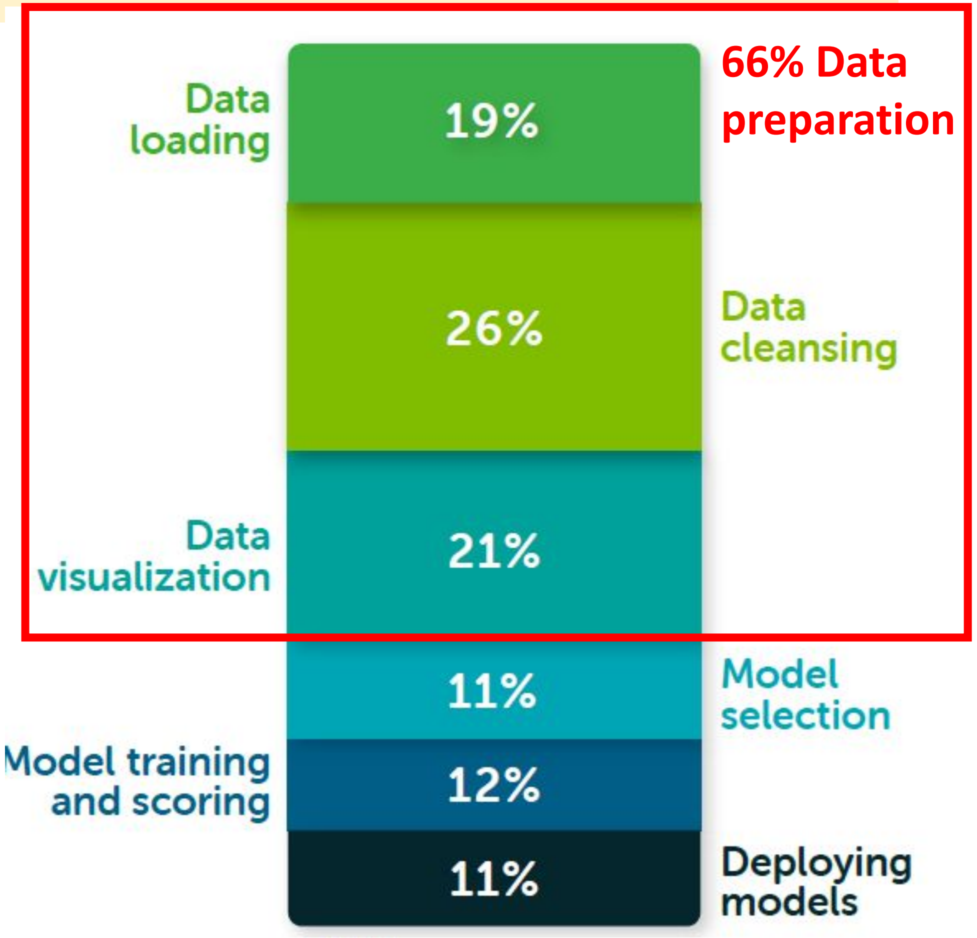
- Imagine a Telecomm company wants to **automatically** detect outages or **failures** in the network by identifying when there is a **drop** or **spike** in the network **traffic**.



Data preparation

- Data is the oil of machine learning
- Prepare and analyse the data before applying machine learning!

Source: Memegenerator



Source: Anaconda

Data preparation with Python: Pandas

- Pandas is a fast, powerful, flexible and easy to use data analysis and manipulation tool, built on top of Python

Source: Reuters

Python working environment

- Option 1: Local Python installation (Anaconda + JupyterLab)
  - Or VS Code Jupyter extension
- Option 2: JupyterLab in the cloud

- Libraries for Machine Learning/Data Science: Pandas, NumPy, Scipy, Matplotlib

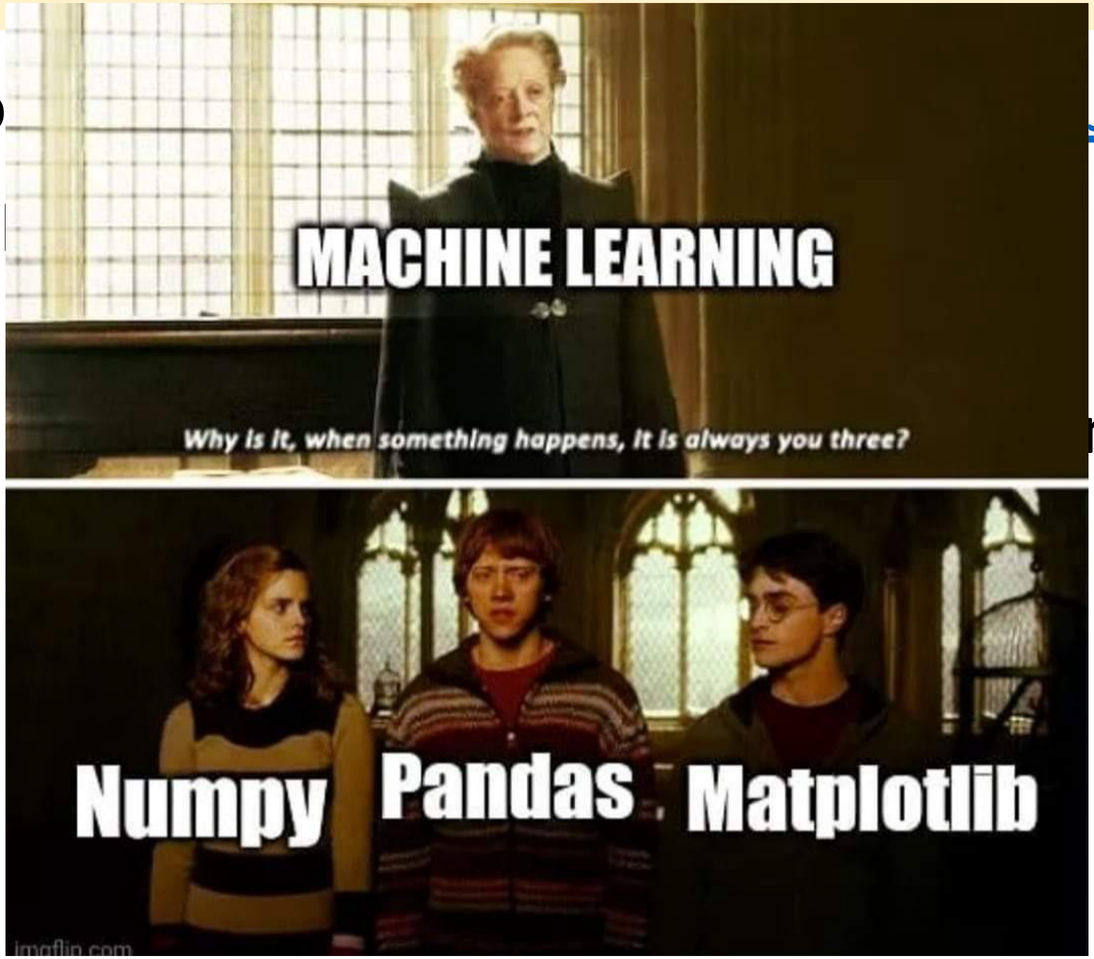
Python working environment

- Option 1: Lo
- Option 2: Ju

JupyterLab)

- Libraries for

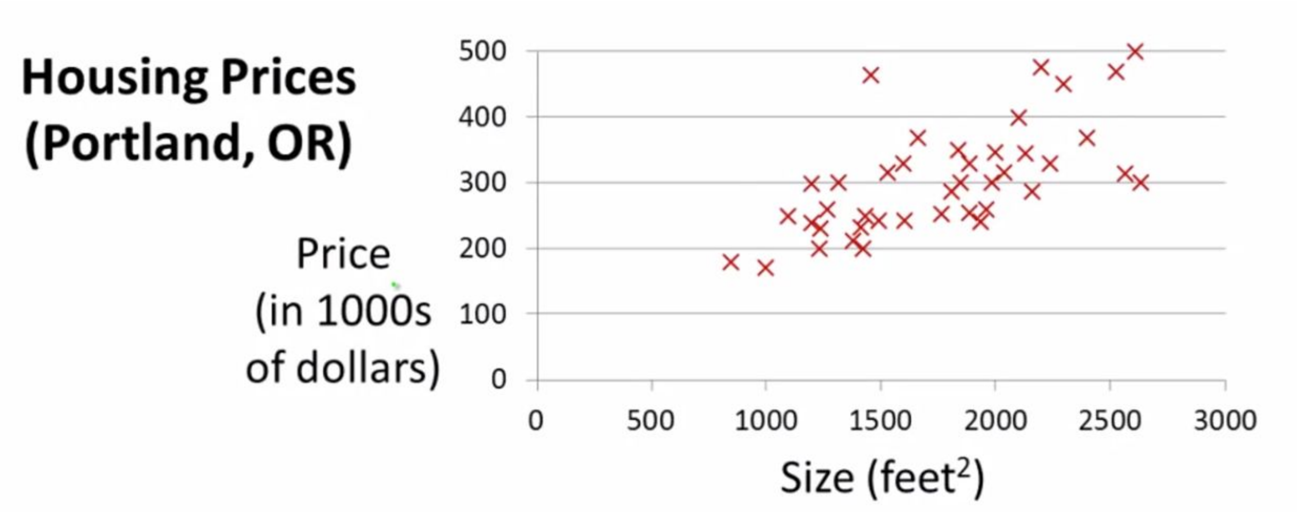
ndas, NumPy, Scipy, Matplotlib



Source: Imgflip

Linear regression: overview

- Predict **continuous**-valued **output**
- Example: given the **size** of a **house**, **predict** its **price**



Source: Coursera

Training set of housing prices (Portland, OR)

Size in feet^2 (x)	Price ($) in 1000's (y)
2104	460
1416	232
1534	315
852	178
…	…

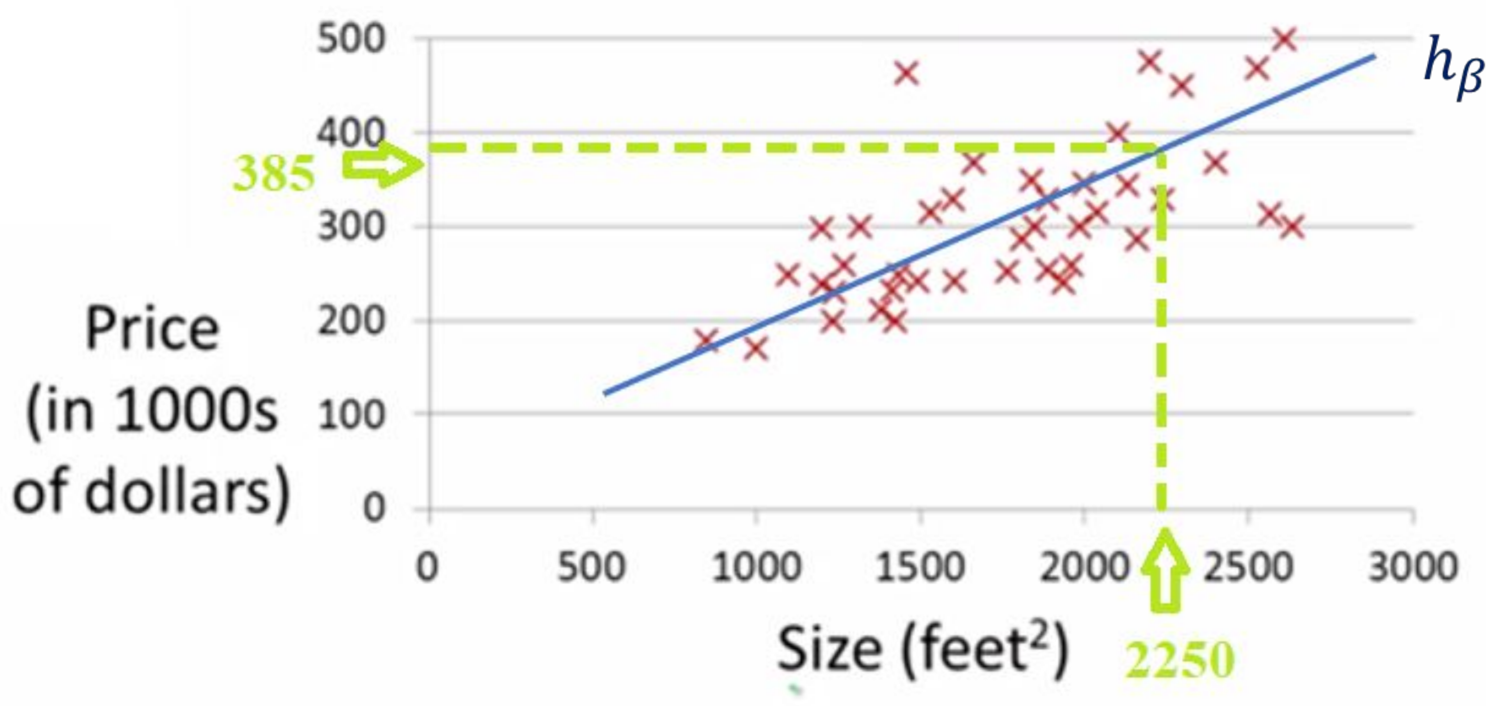
Notation:

x's = "input" variable / features
y's = "output" variable / "target" variable

Source: Coursera

Linear regression: overview

- Predict **continuous**-valued **output**
- Example: given the **size** of a **house**, **predict** its **price**

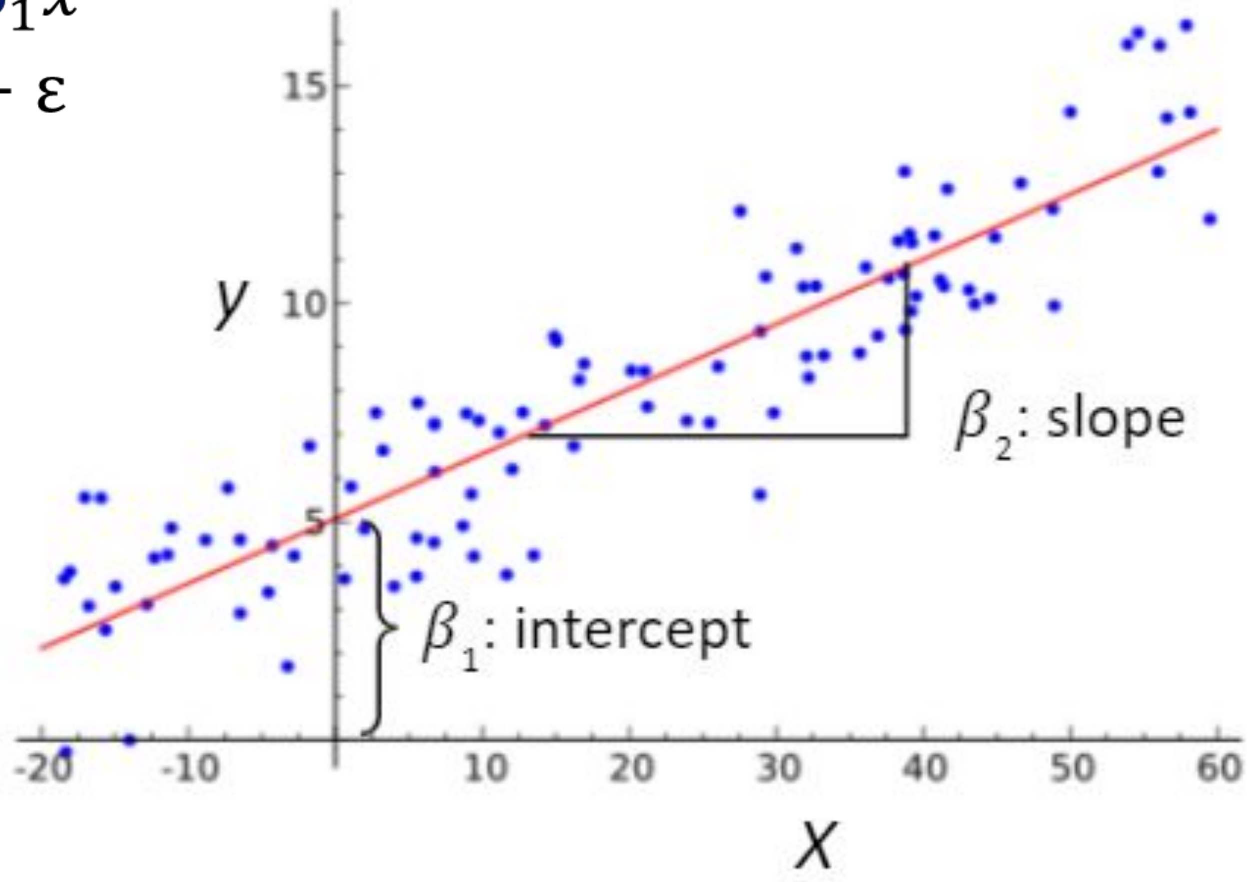


$h_\beta(x) = \beta_0 + \beta_1 x$

Source: Coursera

Linear regression: parameter interpretation

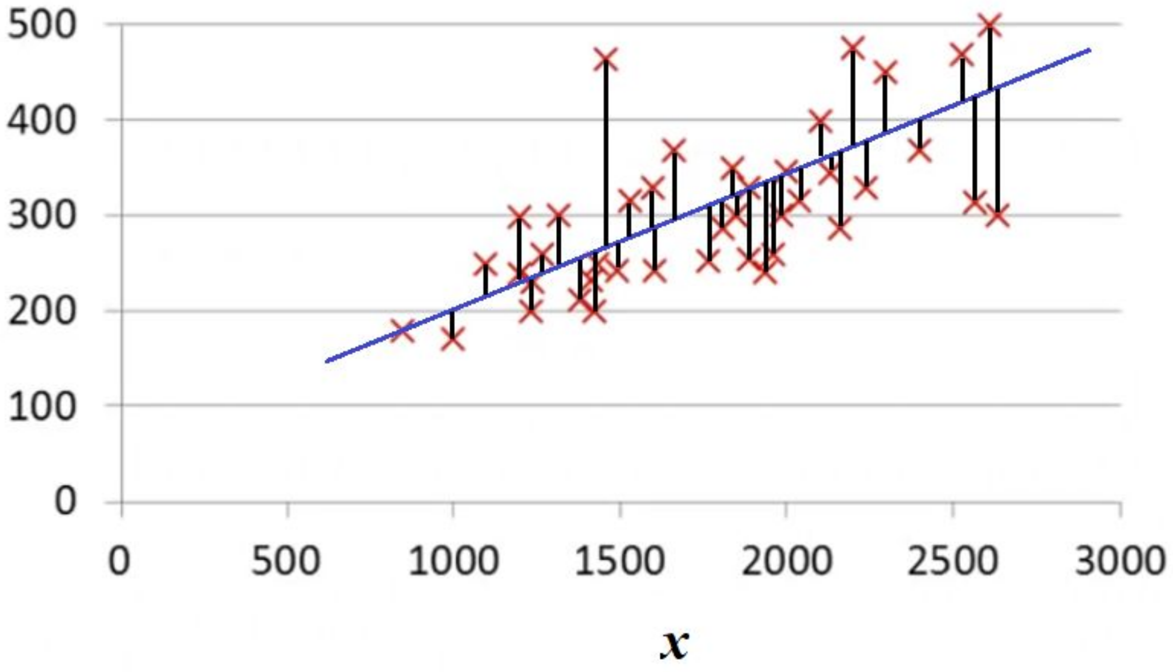
- Hypothesis: $h_\beta(x) = \beta_0 + \beta_1 x$
  $y = \beta_0 + \beta_1 x + \varepsilon$
- Slope β_0
- Intercept β_1



Linear regression: cost function

- Find the **optimal** parameters or weights β_0, β_1 so that $h_\beta(x)$ is close to y for our training simples (x, y).
- **Residuals** or errors: $h_\beta\left(x^{(i)}\right) - y^{(i)}$
- **Minimize** cost function

$$J(\beta_0, \beta_1) = \frac{1}{n}\sum_{i=1}^{n}\left(h_\beta\left(x^{(i)}\right) - y^{(i)}\right)^2$$



Mean squared error (MSE)