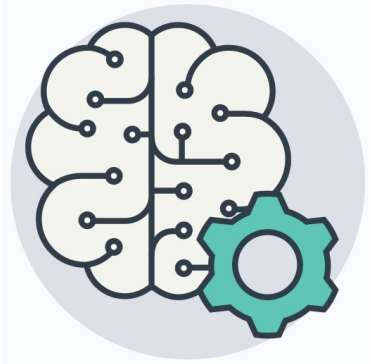
# Aprendizado de Máquina

# Introdução



Prof. Regis Pires Magalhães

regismagalhaes@ufc.br - http://bit.ly/ufcregis

# What is Machine Learning?

- Machine Learning is the science (and art) of programming computers so they can learn from data.

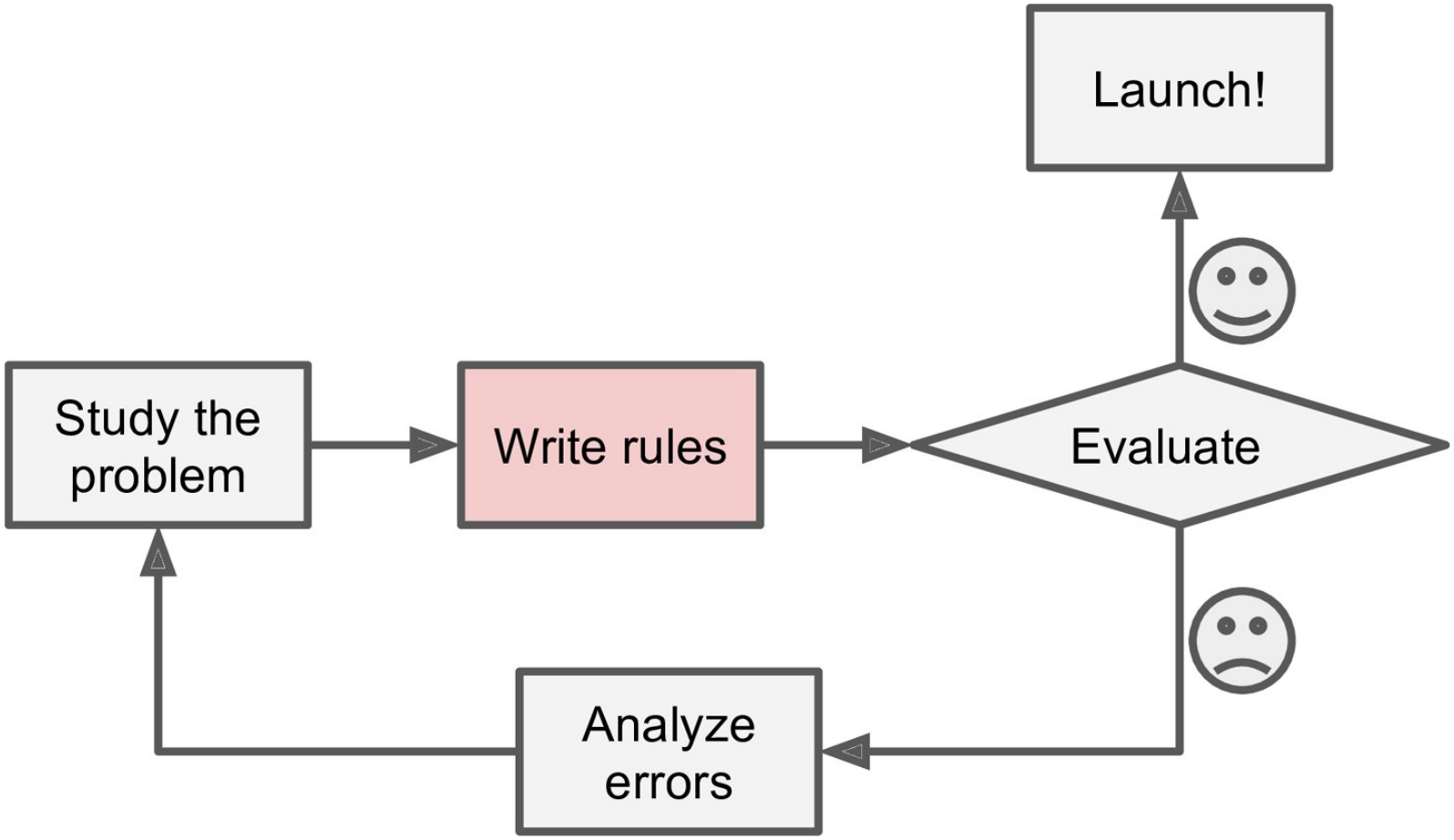
- [Machine Learning is the] field of study that gives computers the ability to learn without being explicitly programmed.
  —Arthur Samuel, 1959

- A computer program is said to learn from experience E with respect to some task T and some performance measure P, if its performance on T, as measured by P, improves with experience E.
  —Tom Mitchell, 1997

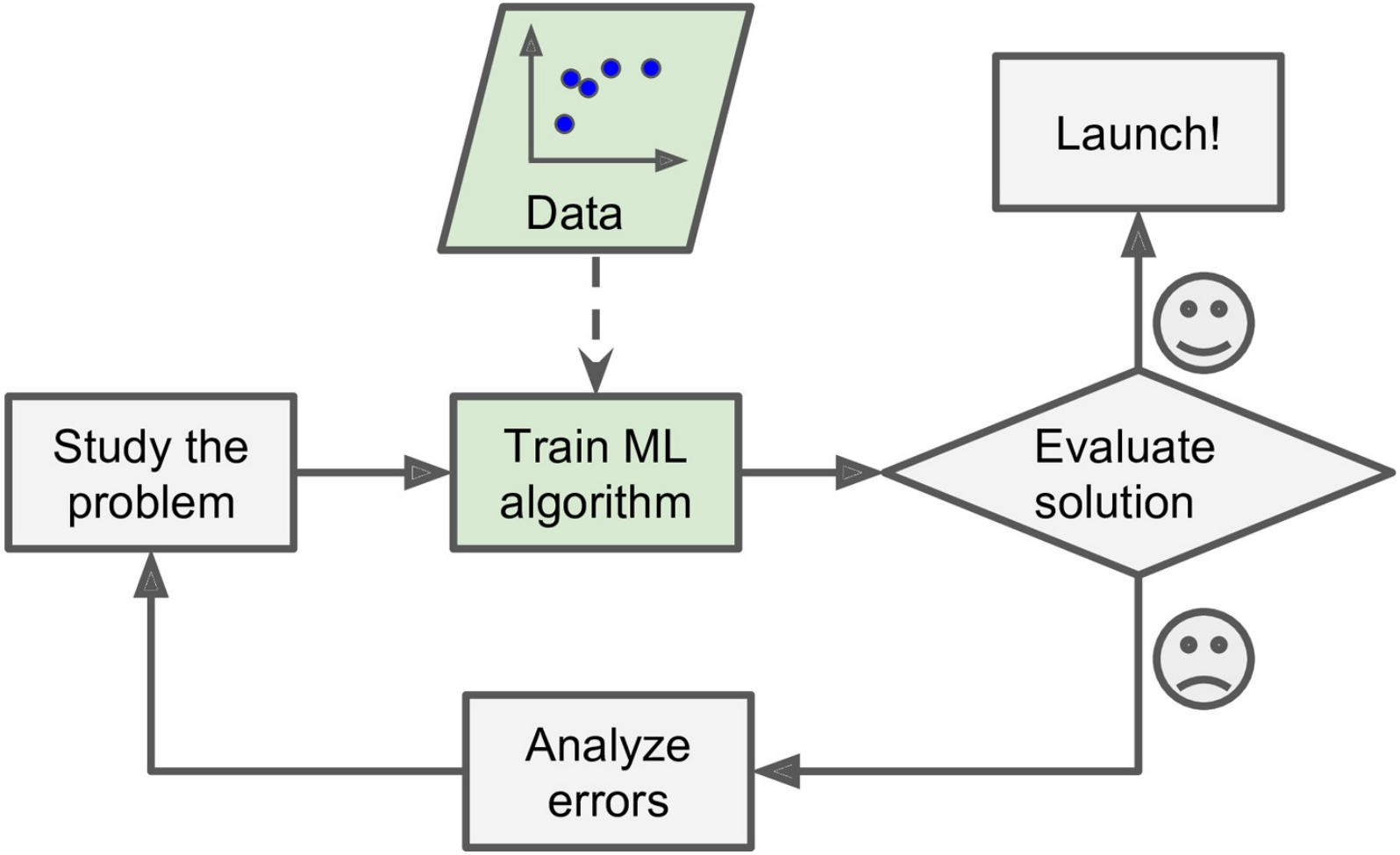
# Giving Computers the Ability to Learn from Data

- Intuição:
  - Exemplo do empréstimo pessoal

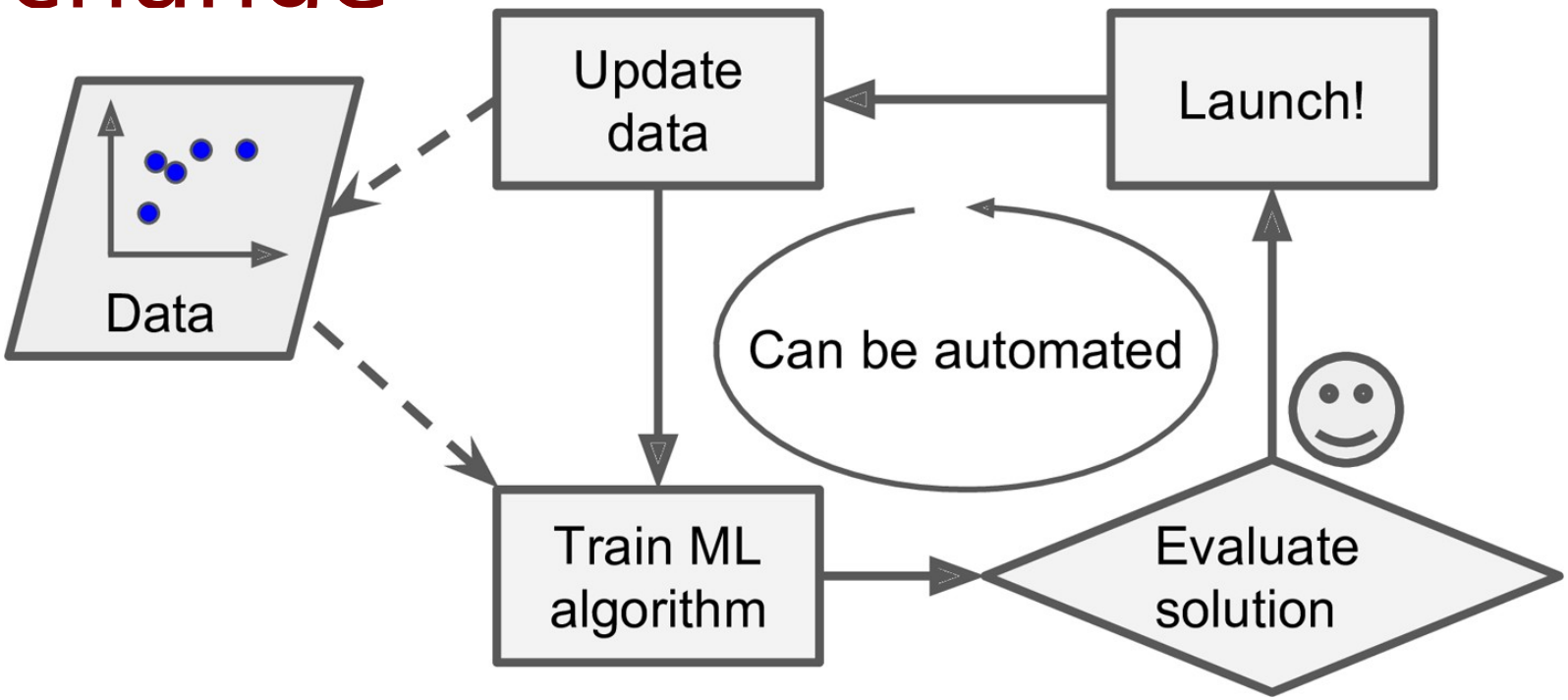
# Traditional approach

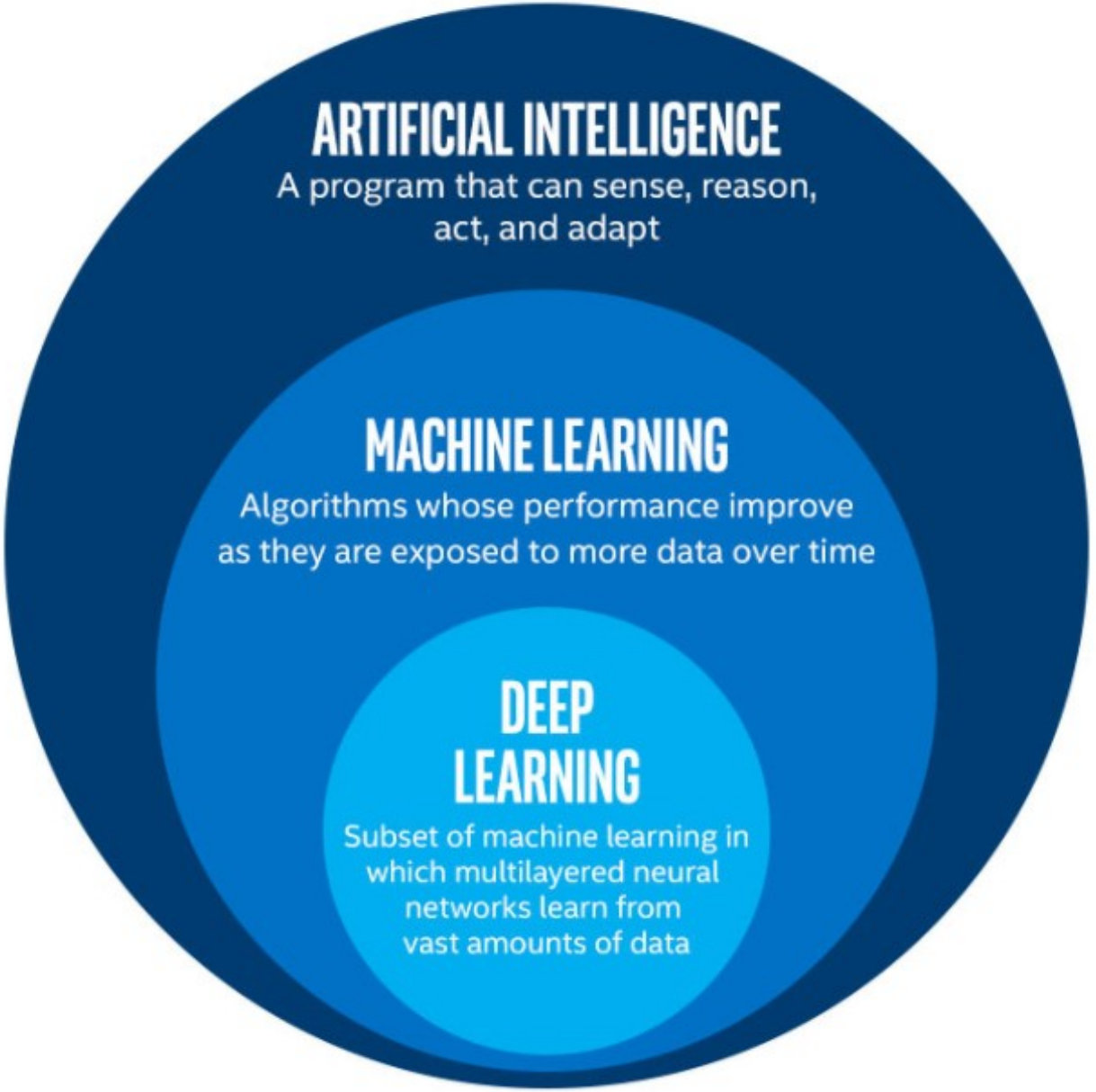


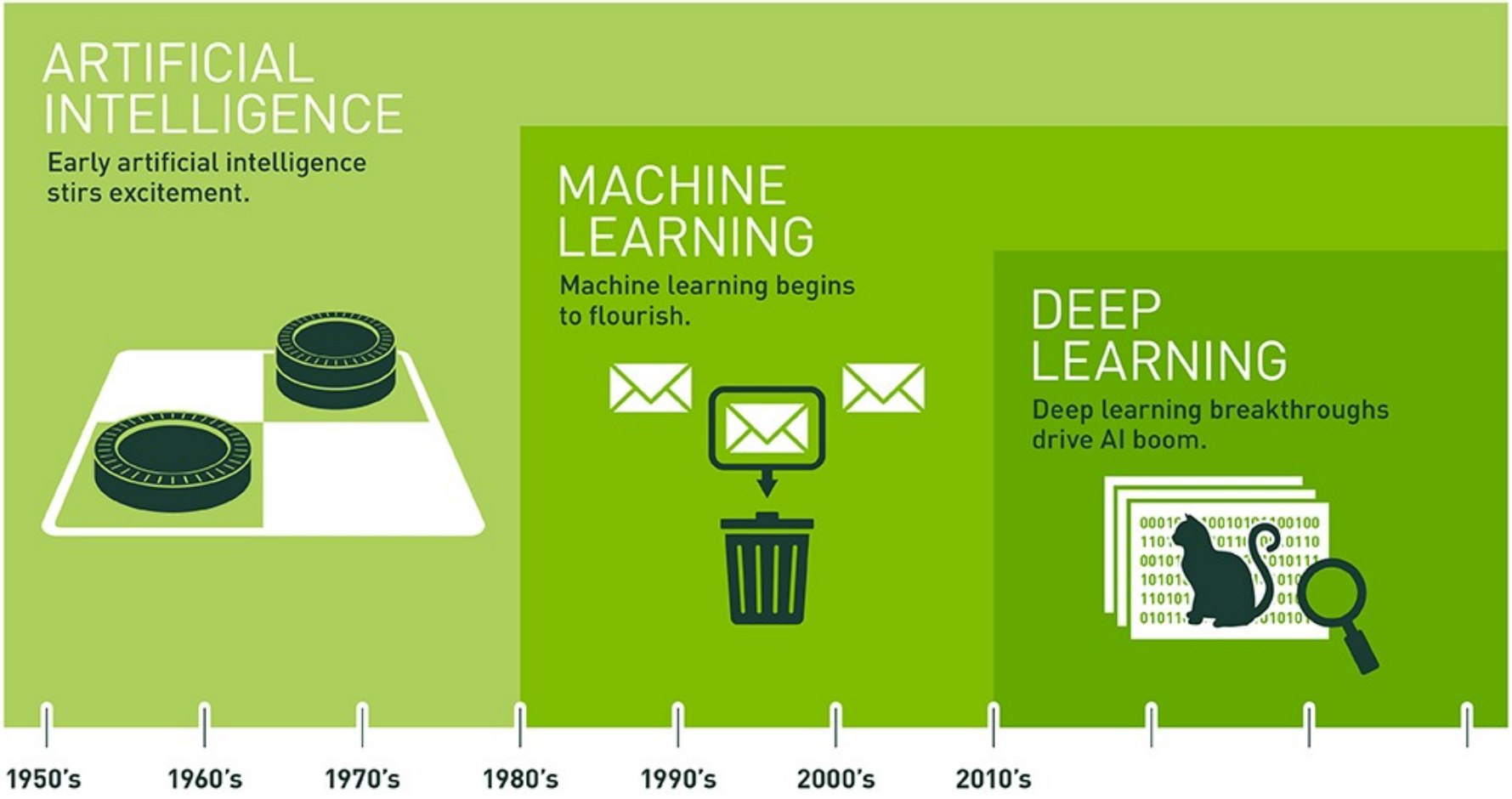
# Machine Learning approach



# Automatically adapting to change




ARTIFICIAL INTELLIGENCE
A program that can sense, reason, act, and adapt
MACHINE LEARNING
Algorithms whose performance improve as they are exposed to more data over time
DEEP LEARNING
Subset of machine learning in which multilayered neural networks learn from vast amounts of data

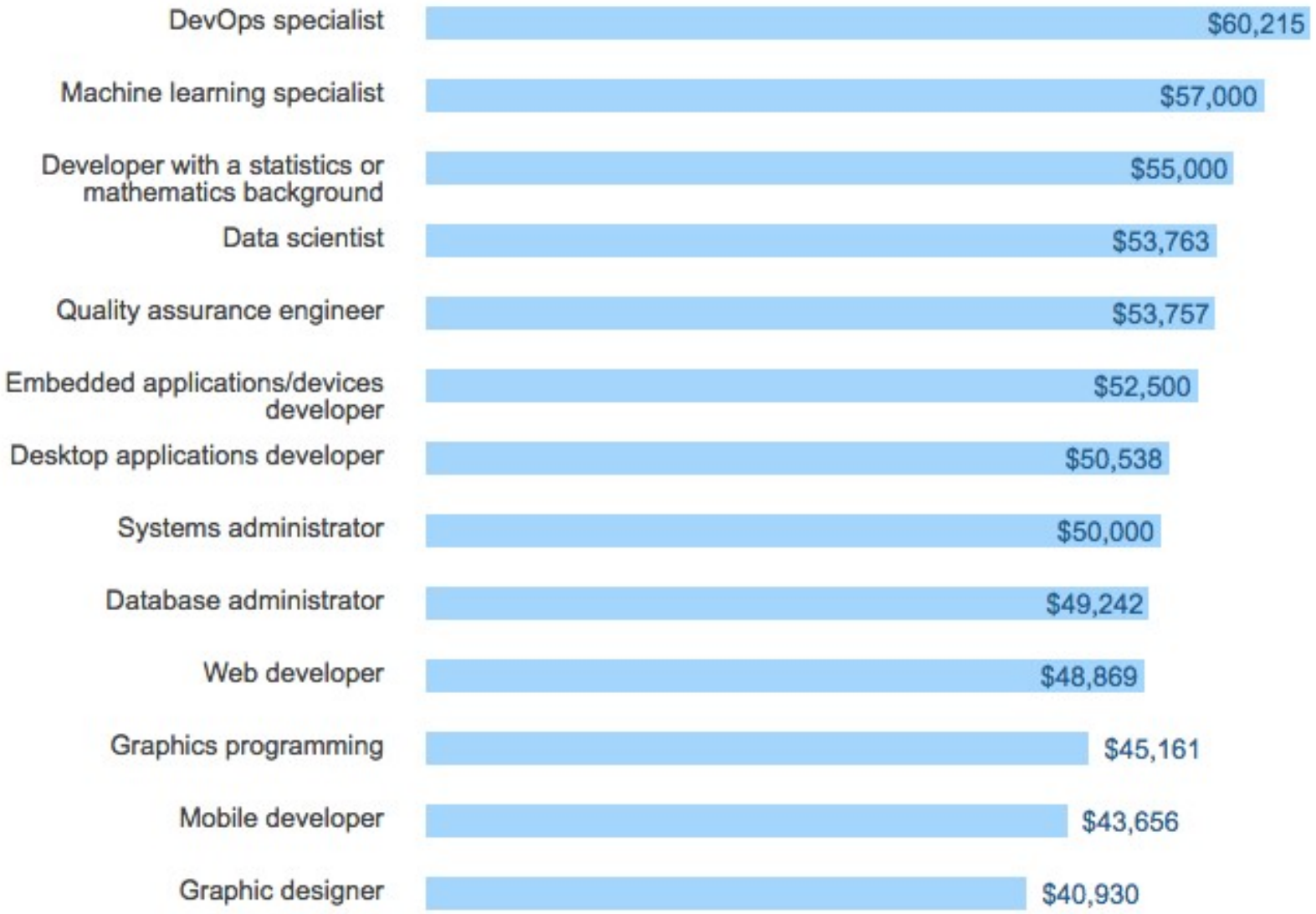



Since an early flush of optimism in the 1950s, smaller subsets of artificial intelligence – first machine learning, then deep learning, a subset of machine learning – have created ever larger disruptions.

# Motivações

- Barateamento e popularização dos dispositivos móveis e dos mais diversos tipos de sensores para coleta de dados.
- Aumento do poder de processamento e armazenamento dos computadores.
- Otimização de recursos em um mercado global cada vez mais competitivo.
- Amadurecimento, melhoria e disseminação de técnicas de aprendizagem de máquina.


## Salary by Developer Type

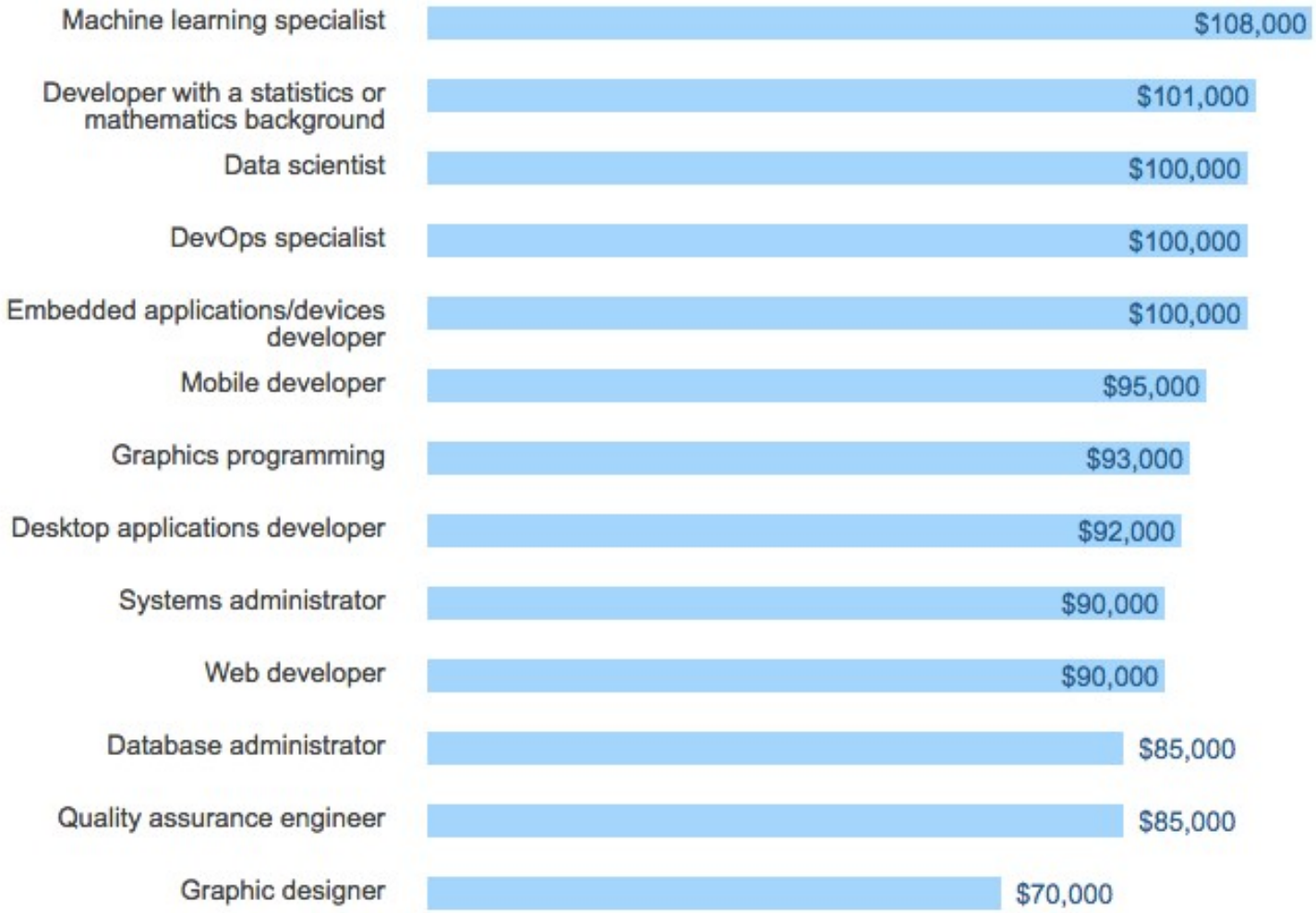



*Median of 12,475 responses; USD*

https://stackoverflow.com/insights/survey/2017/

## Salaries by Geography

United States | Canada | United Kingdom | Germany | France

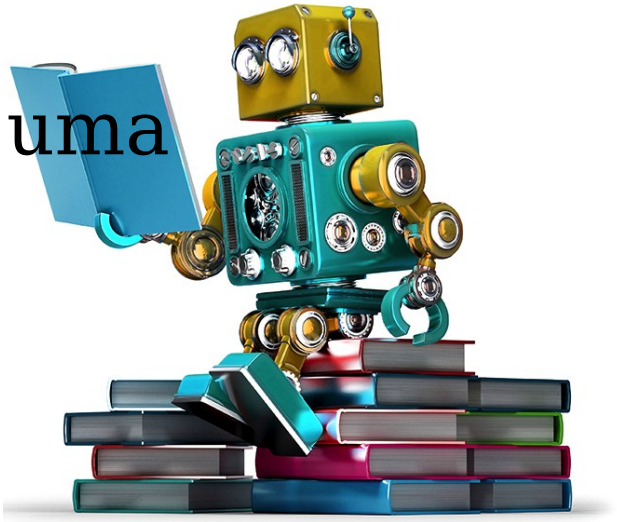



*Median of 3,630 responses; USD*

https://stackoverflow.com/insights/survey/2017/

# Exemplos de uso de técnicas de aprendizado de máquina

- Risco de empréstimo / seguro / plano de saúde
- Preços de imóveis / carros
- Produtos com maior chance de venda
- Sistemas de recomendação
- Evasão de cursos
- Probabilidade de desenvolver uma doença
- Sistema de busca
- Sistemas de segurança
- etc.



# Top 10 Use Cases for Data Science & Machine Learning

**HEALTHCARE:**
Patient Diagnosis

**FINANCE:**
Fraud Detection

**MANUFACTURING:**
Anomaly Detection

**RETAIL:**
Inventory Optimization

**GOVERNMENT:**
Smarter Services

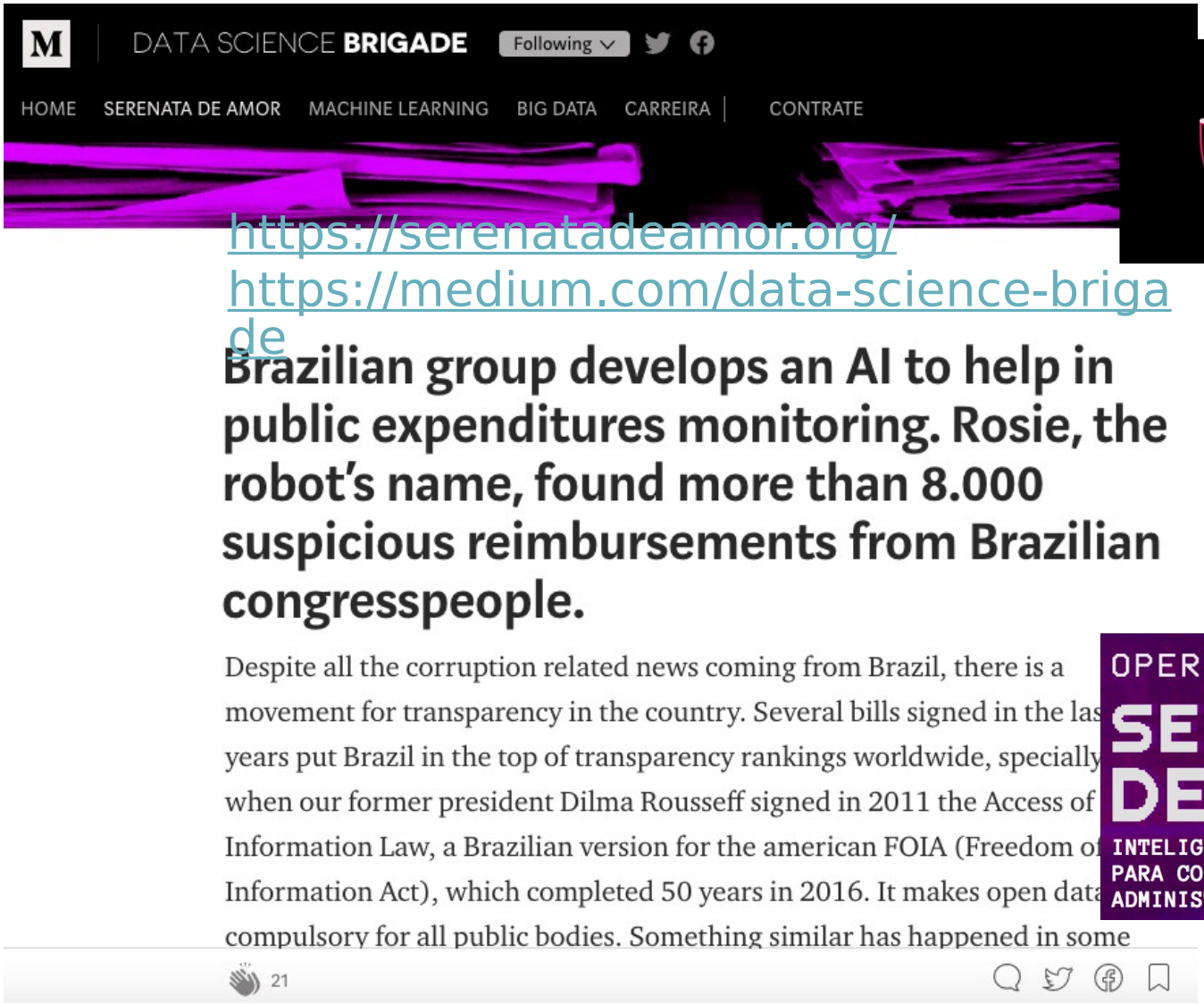
**TRANSPORTATION:**
Demand Forecasting

**NETWORKS:**
Intrusion Detection

**E-COMMERCE:**
Recommender Systems

**MEDIA:**
Interaction & Speed

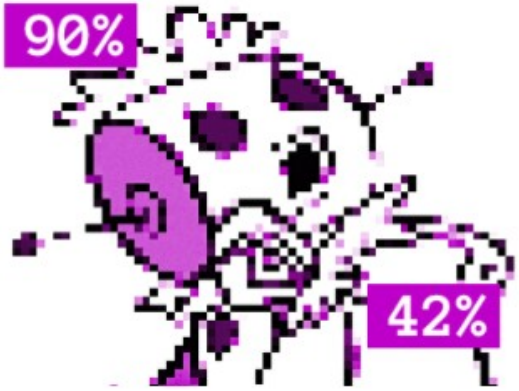
**EDUCATION:**
Research Insight

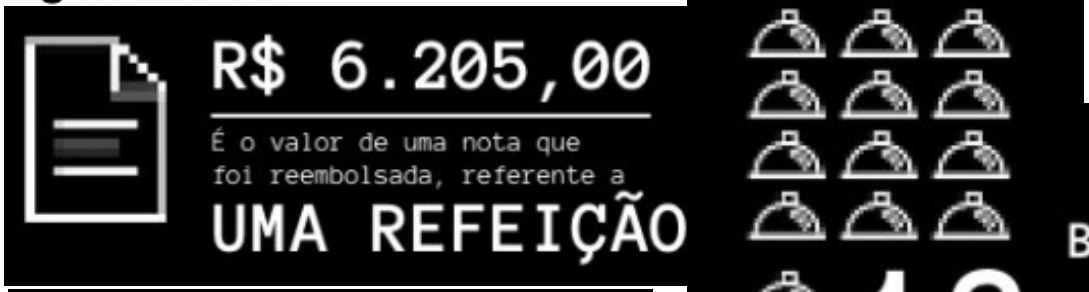

DATA SCIENCE BRIGADE
Following
HOME
SERENATA DE AMOR
MACHINE LEARNING
BIG DATA
CARREIRA
CONTRATE
https://serenatadeamor.org/
https://medium.com/data-science-brigade
Brazilian group develops an AI to help in public expenditures monitoring. Rosie, the robot's name, found more than 8.000 suspicious reimbursements from Brazilian congresspeople.
Despite all the corruption related news coming from Brazil, there is a movement for transparency in the country. Several bills signed in the las years put Brazil in the top of transparency rankings worldwide, specially when our former president Dilma Rousseff signed in 2011 the Access of Information Law, a Brazilian version for the american FOIA (Freedom of Information Act), which completed 50 years in 2016. It makes open data compulsory for all public bodies. Something similar has happened in some
21

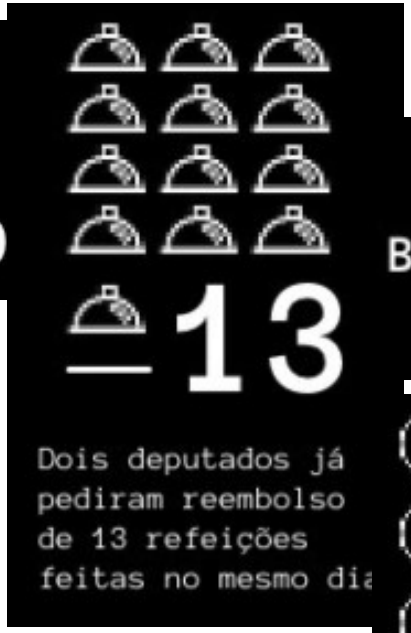

DATA SCIENCE
BRIGADE


OPERAÇÃO
SERENATA
DE AMOR
INTELIGÊNCIA ARTIFICIAL
PARA CONTROLE SOCIAL DA
ADMINISTRAÇÃO PÚBLICA

**A Operação Serenata de Amor criou a Rosie – uma inteligência artificial capaz de analisar cada pedido de reembolso dos deputados e identificar a probabilidade de ilegalidade.**

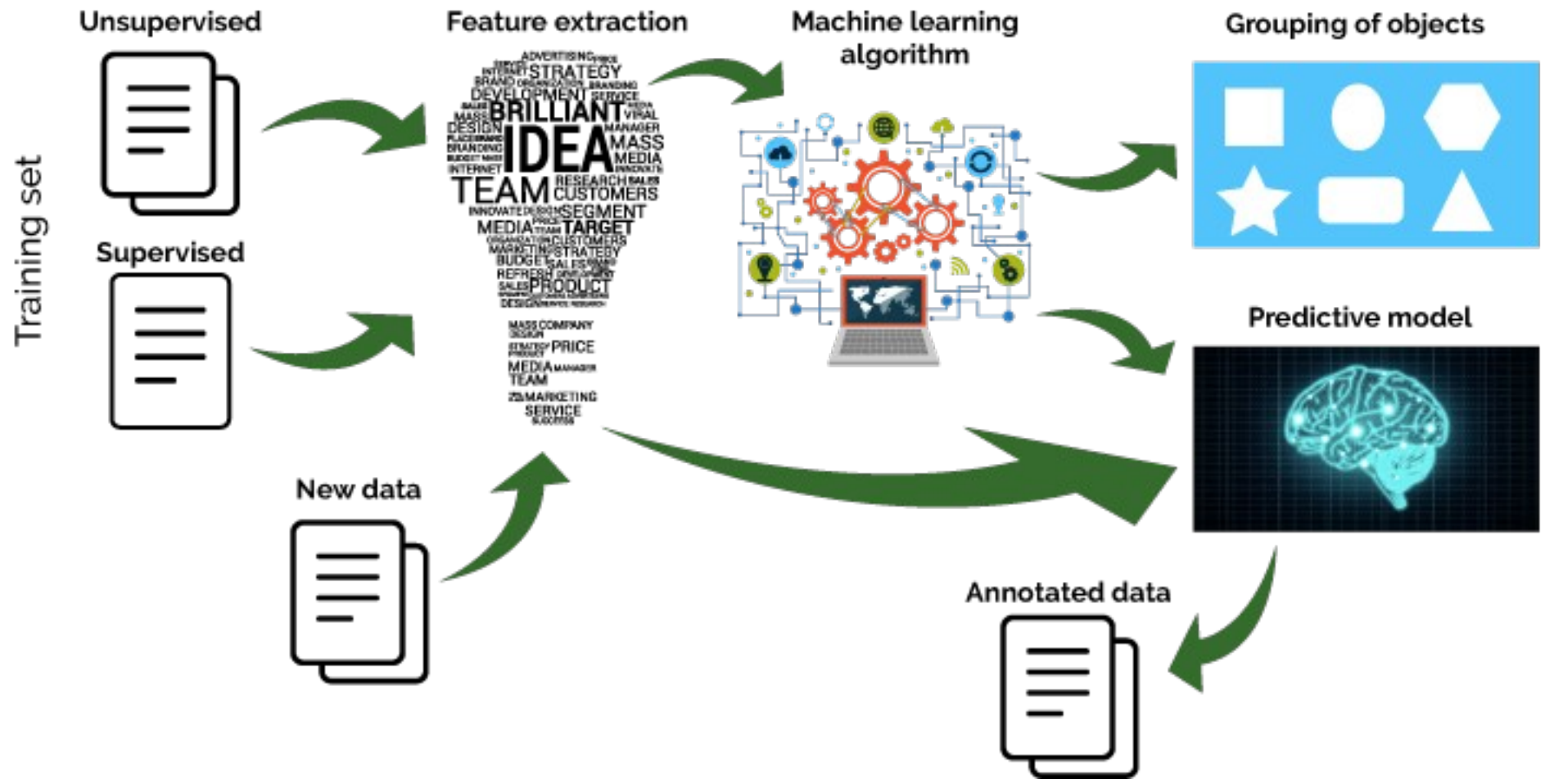


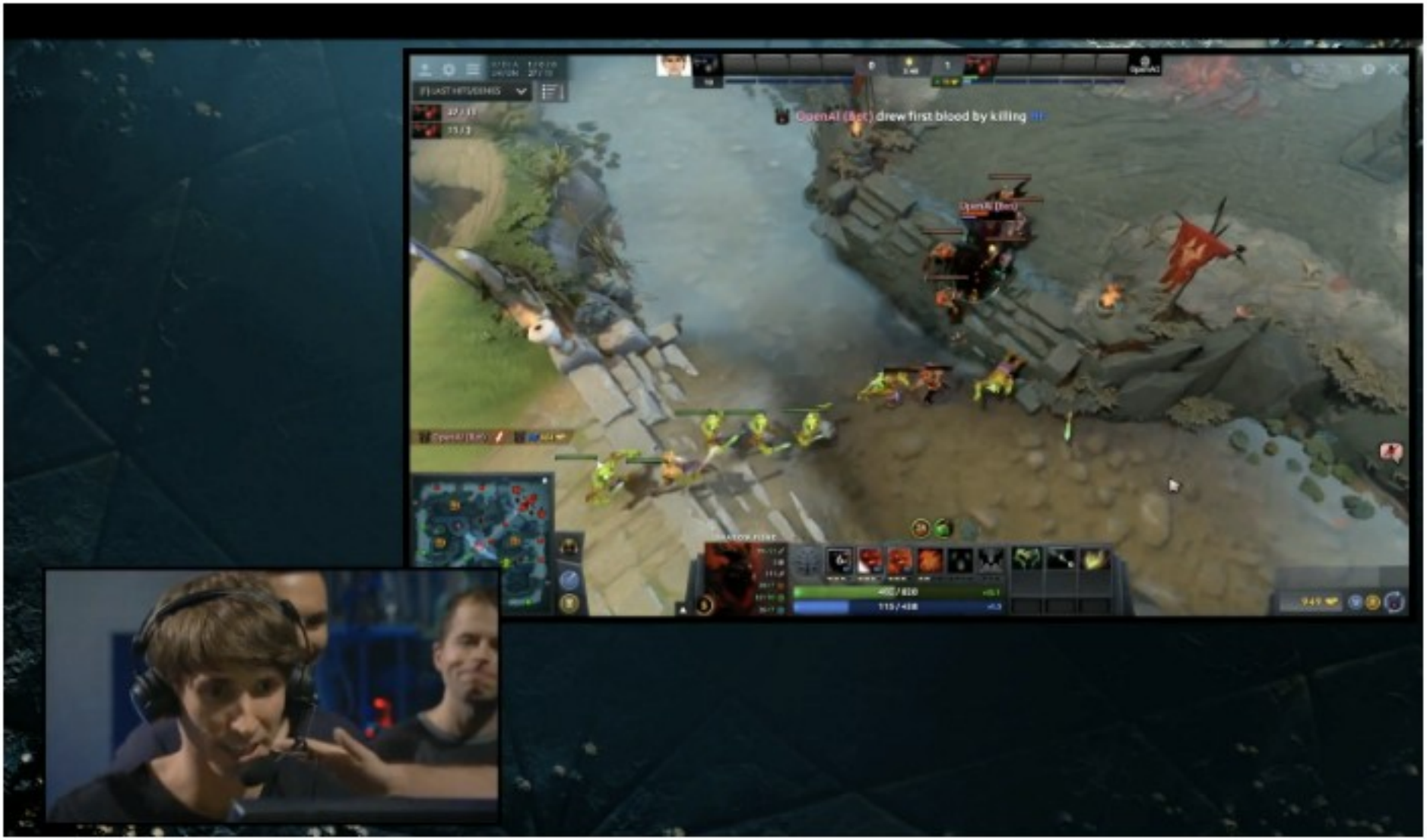





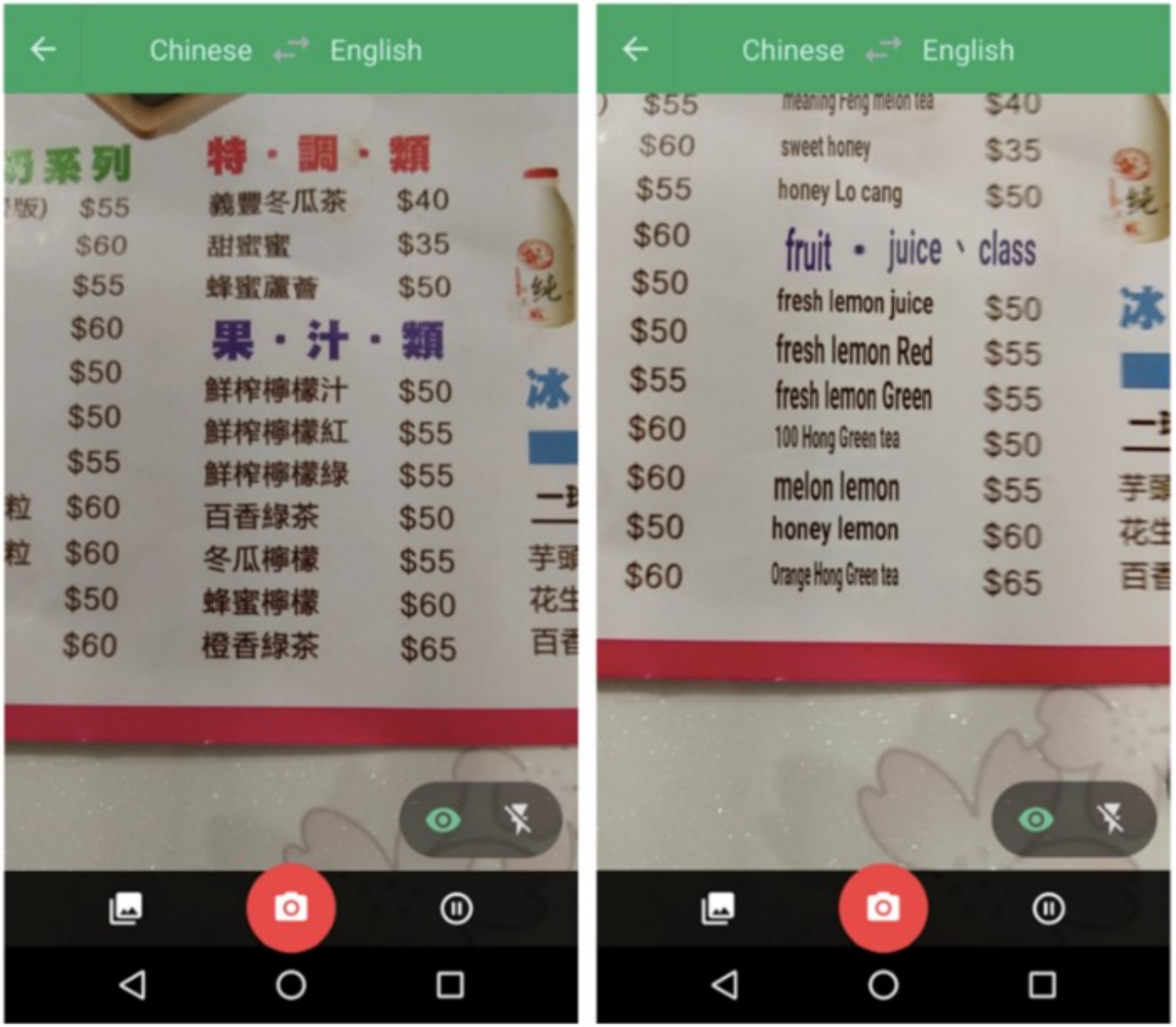

# Machine Learning






See the full match at The International 2017, with Dendi (human) vs. OpenAI (bot), on YouTube.

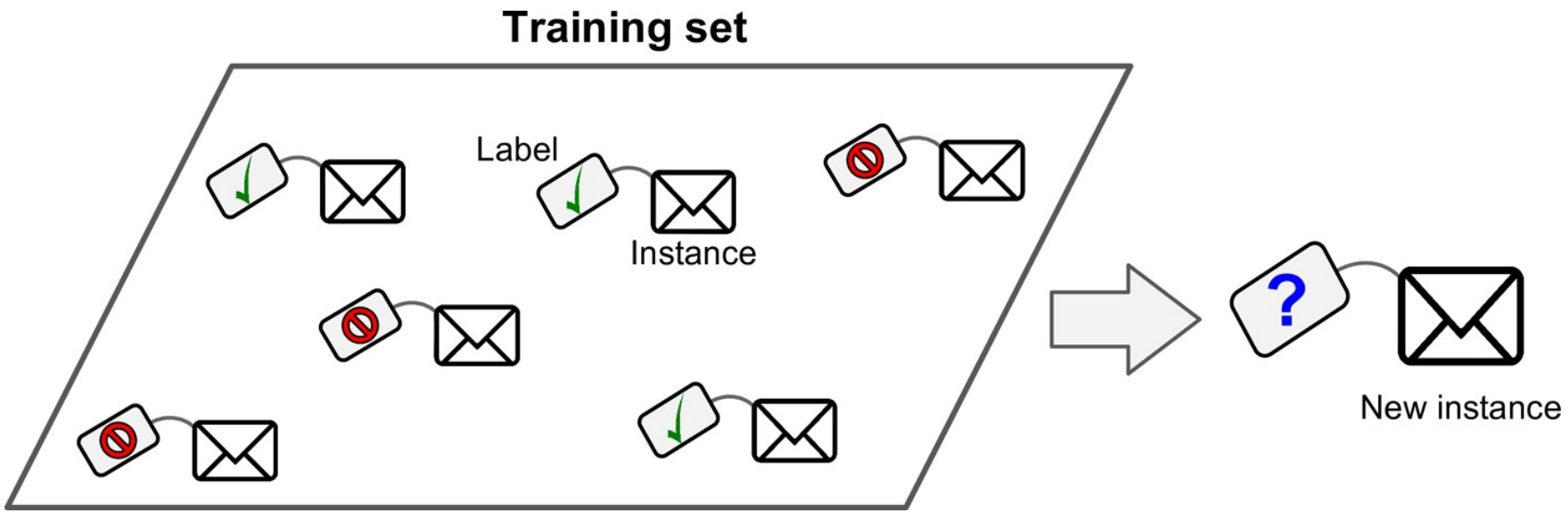
https://www.youtube.com/watch?v=wiOopO9jTZw




Google Translate overlaying English translations on a drink menu in real time using convolutional neural networks.

# Usos de Machine Learning

# Spam classification



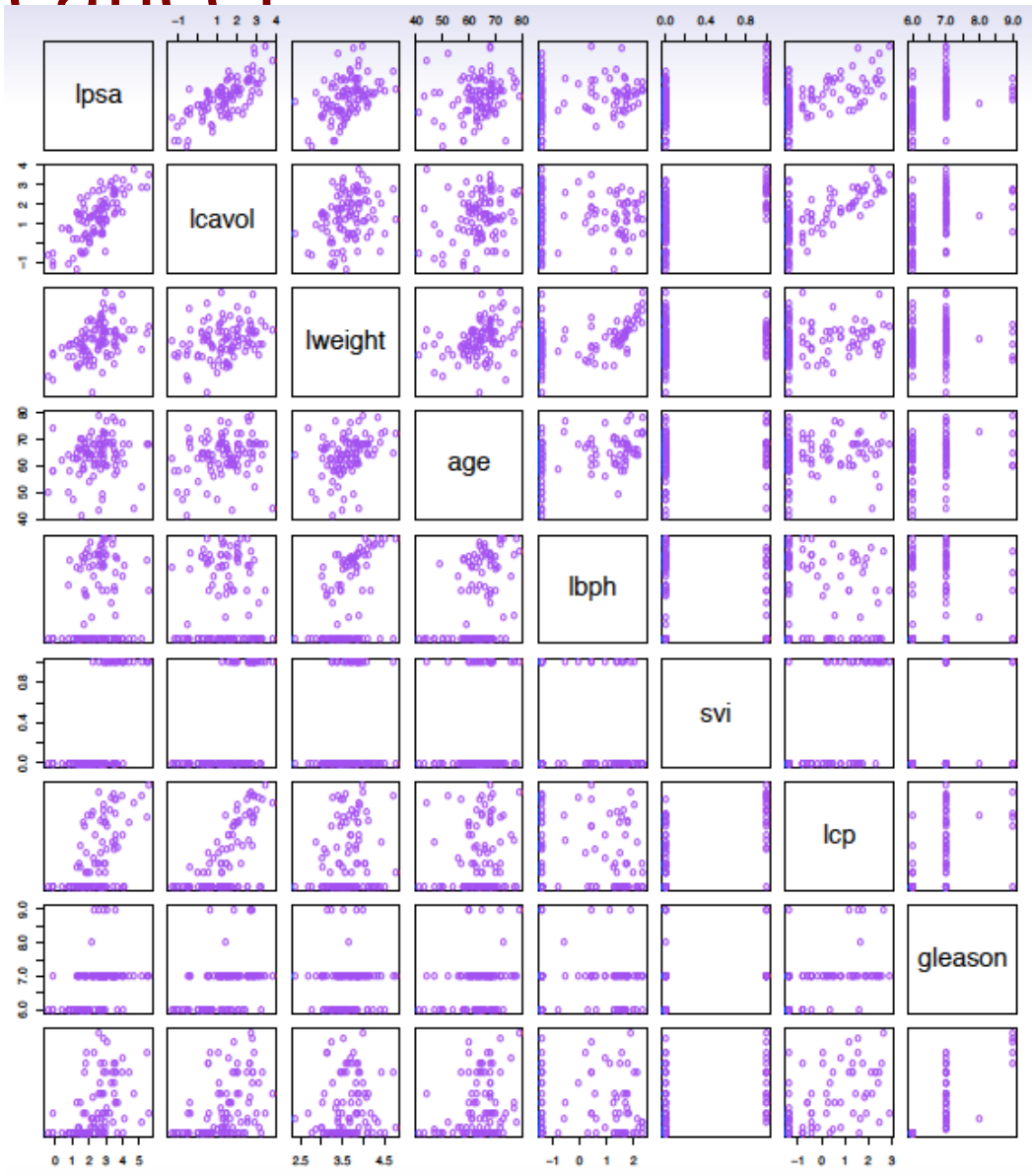
# Spam Detection

- data from 4601 emails sent to an individual (named George, at HP labs, before 2000). Each is labeled as spam or email.
- goal: build a customized spam filter.
- input features: relative frequencies of 57 of the most commonly

| | george | you | hp | free | ! | edu | remove |
|---|---|---|---|---|---|---|---|
| spam | 0.00 | 2.26 | 0.02 | 0.52 | 0.51 | 0.01 | 0.28 |
| email | 1.27 | 1.27 | 0.90 | 0.07 | 0.11 | 0.29 | 0.01 |

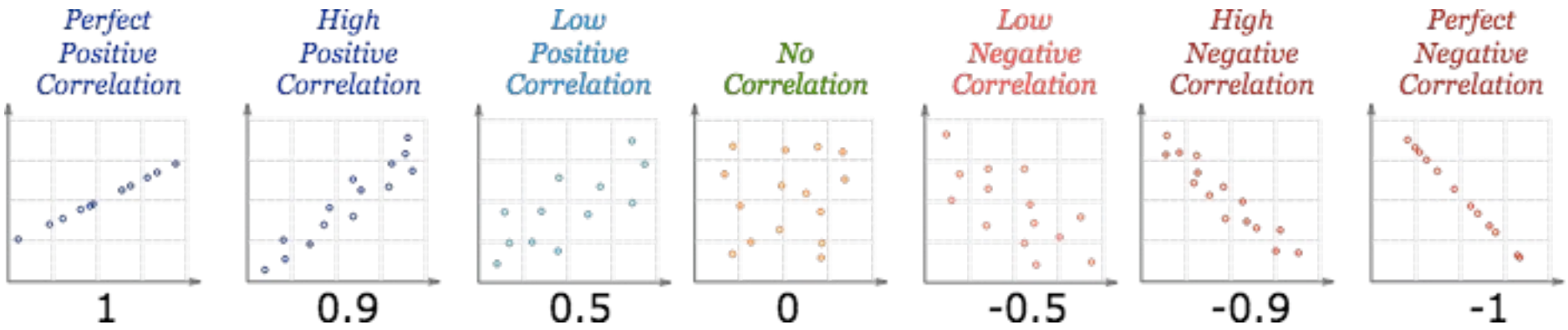
Average percentage of words or characters in an email message equal to the indicated word or character. We have chosen the words and characters showing the

# Identify the risk factors for prostate cancer

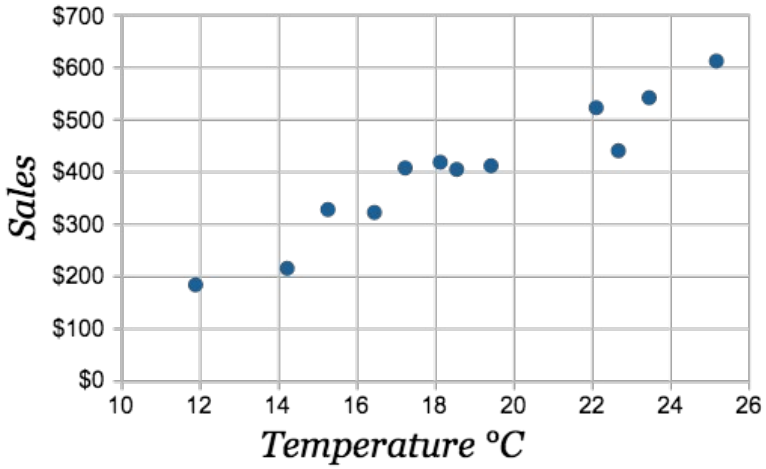



Scatter plot correlation

# Correlação



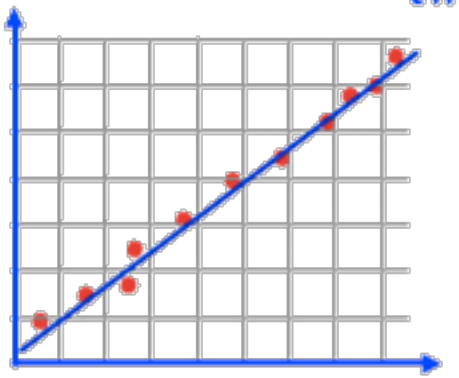

| Ice Cream Sales vs Temperature | |
|---|---|
| **Temperature °C** | **Ice Cream Sales** |
| 14.2° | $215 |
| 16.4° | $325 |
| 11.9° | $185 |
| 15.2° | $332 |
| 18.5° | $406 |
| 22.1° | $522 |
| 19.4° | $412 |
| 25.1° | $614 |
| 23.4° | $544 |
| 18.1° | $421 |
| 22.6° | $445 |
| 17.2° | $408 |



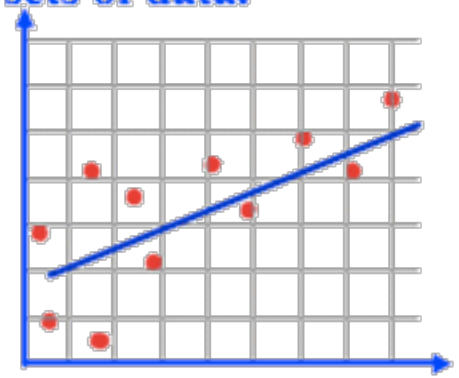

the correlation is 0.9575

# SCATTERPLOTS & CORRELATION

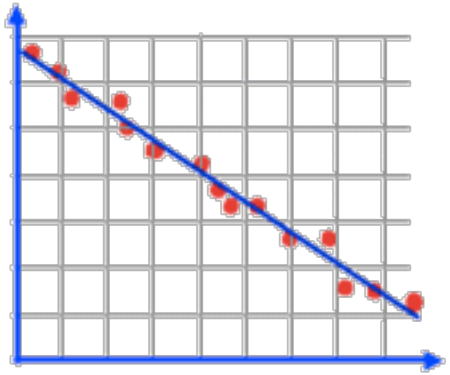
**Correlation - indicates a relationship (connection) between two sets of data.**



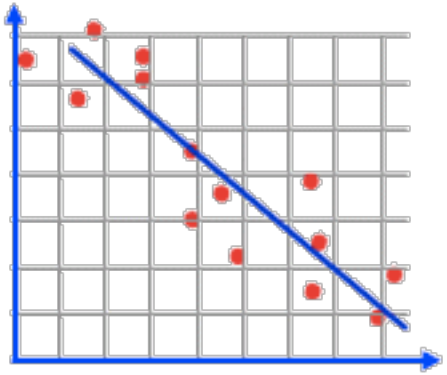
Strong positive correlation



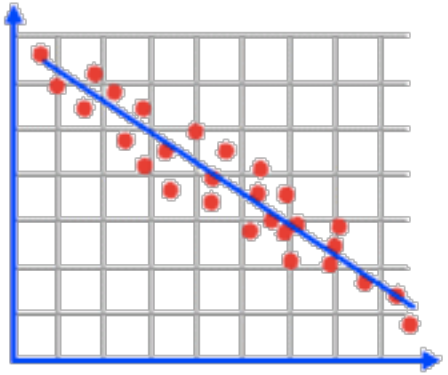
Weak positive correlation



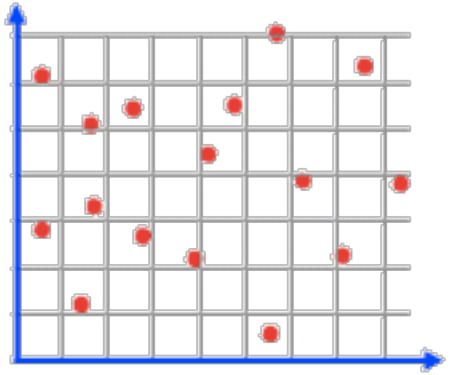
Strong negative correlation



Weak negative correlation

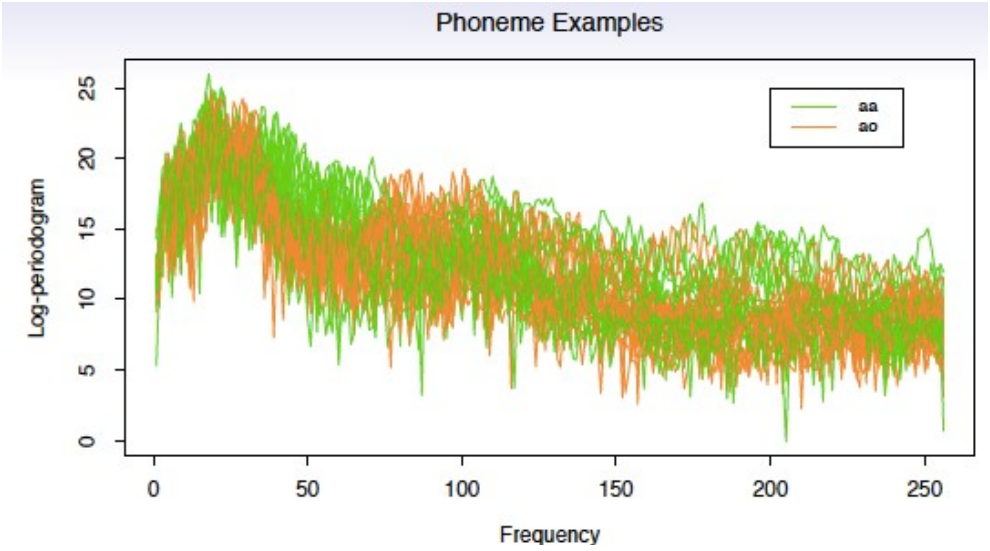


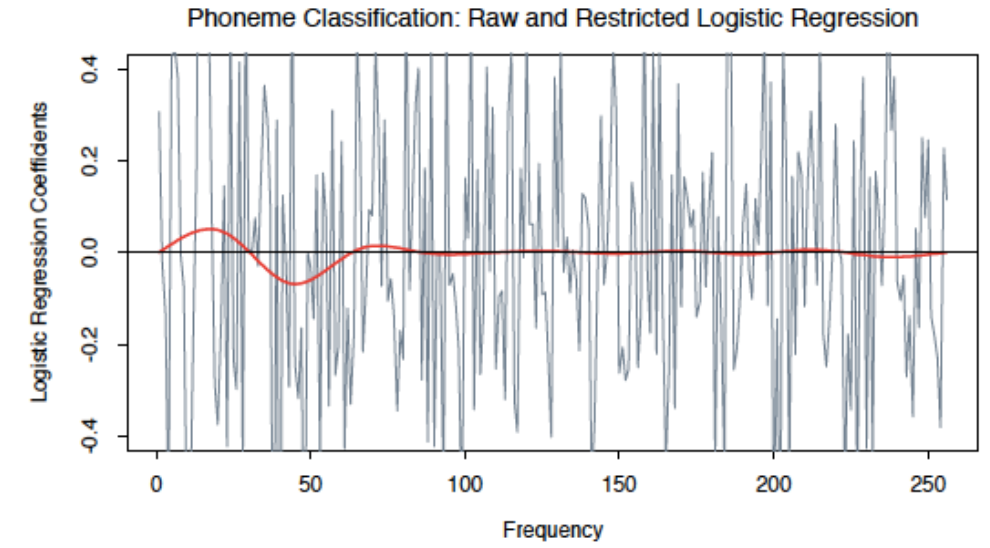
Moderate negative correlation



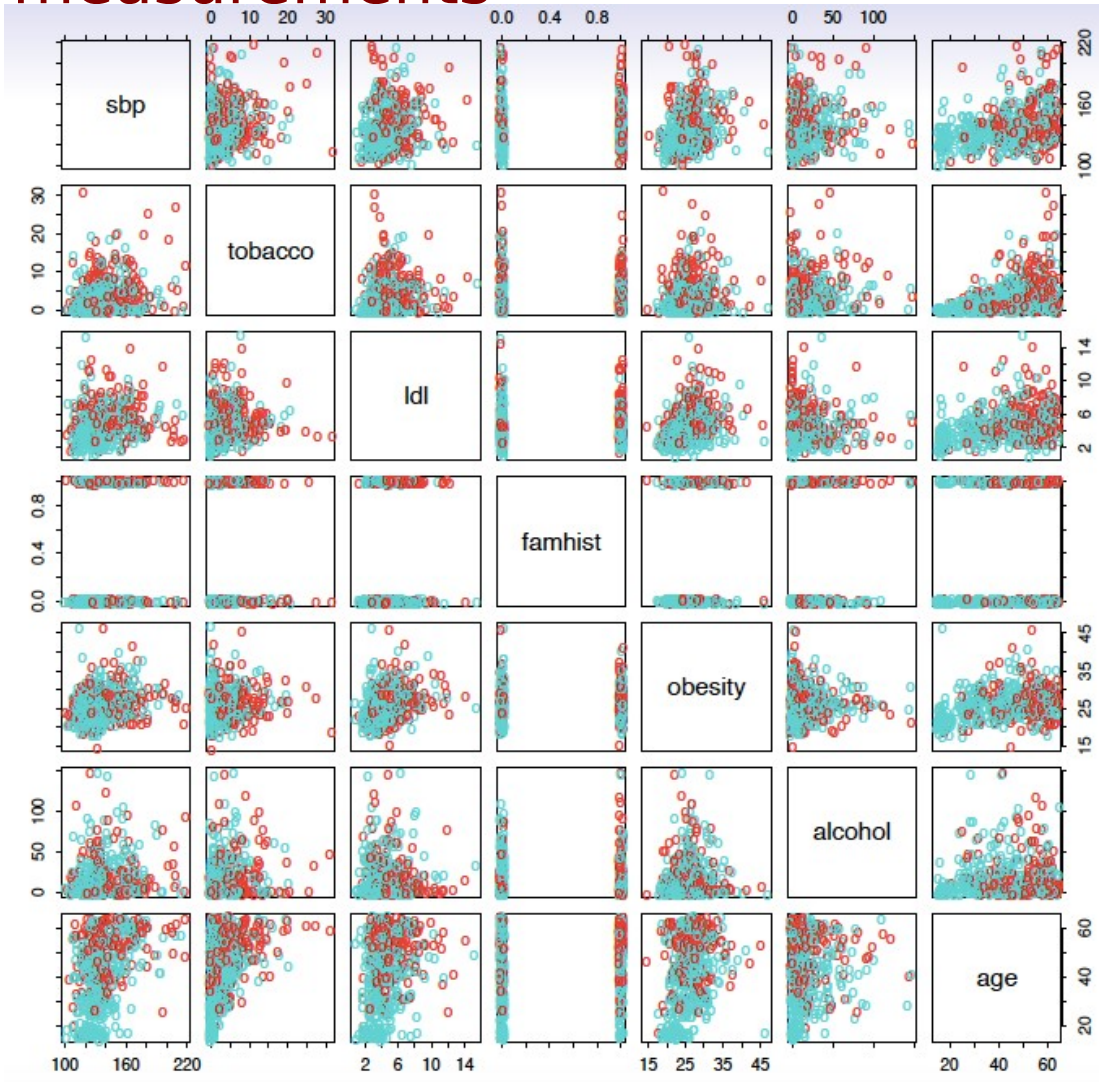
No correlation

# Classify a recorded phoneme based on a log-periodogram

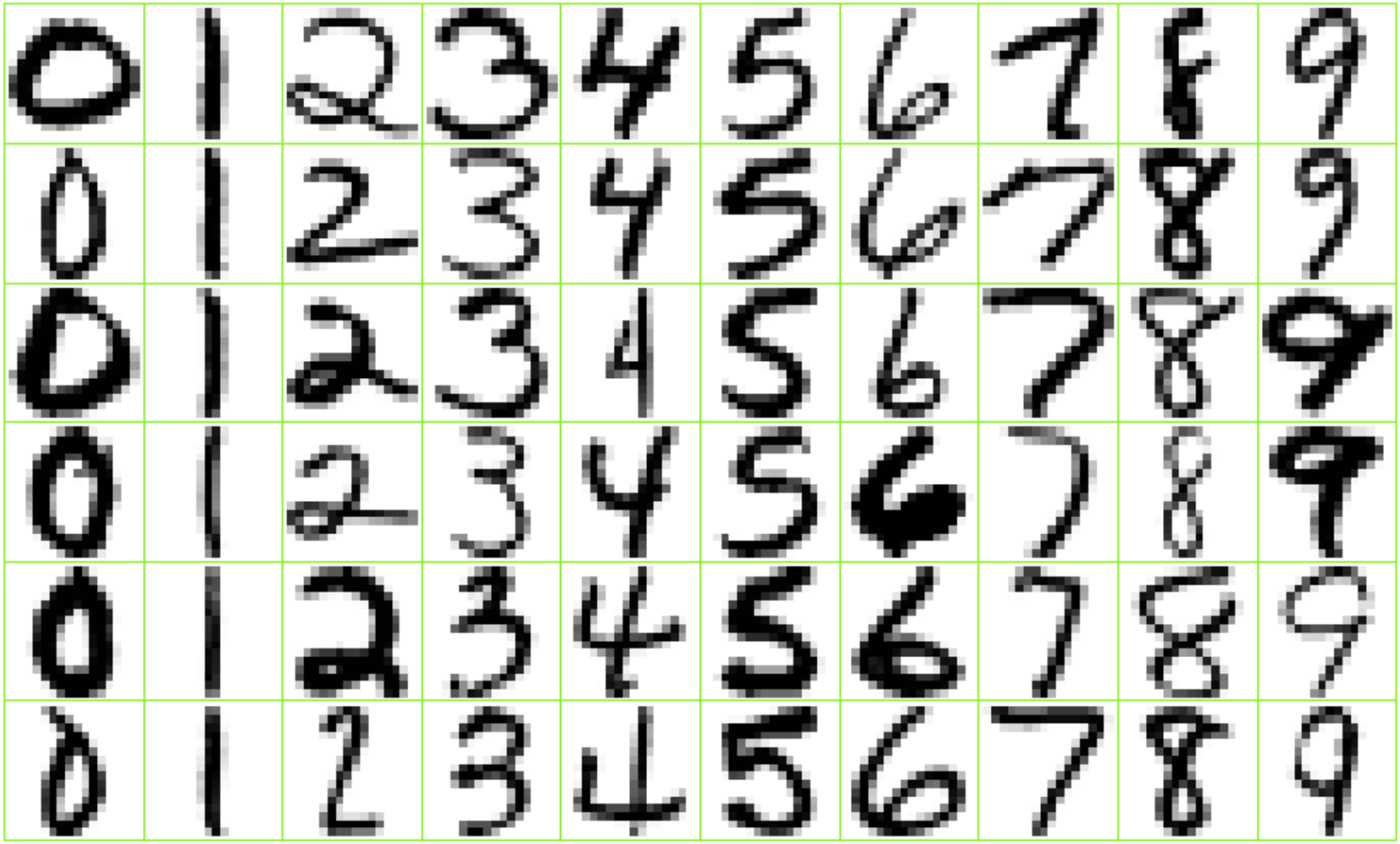


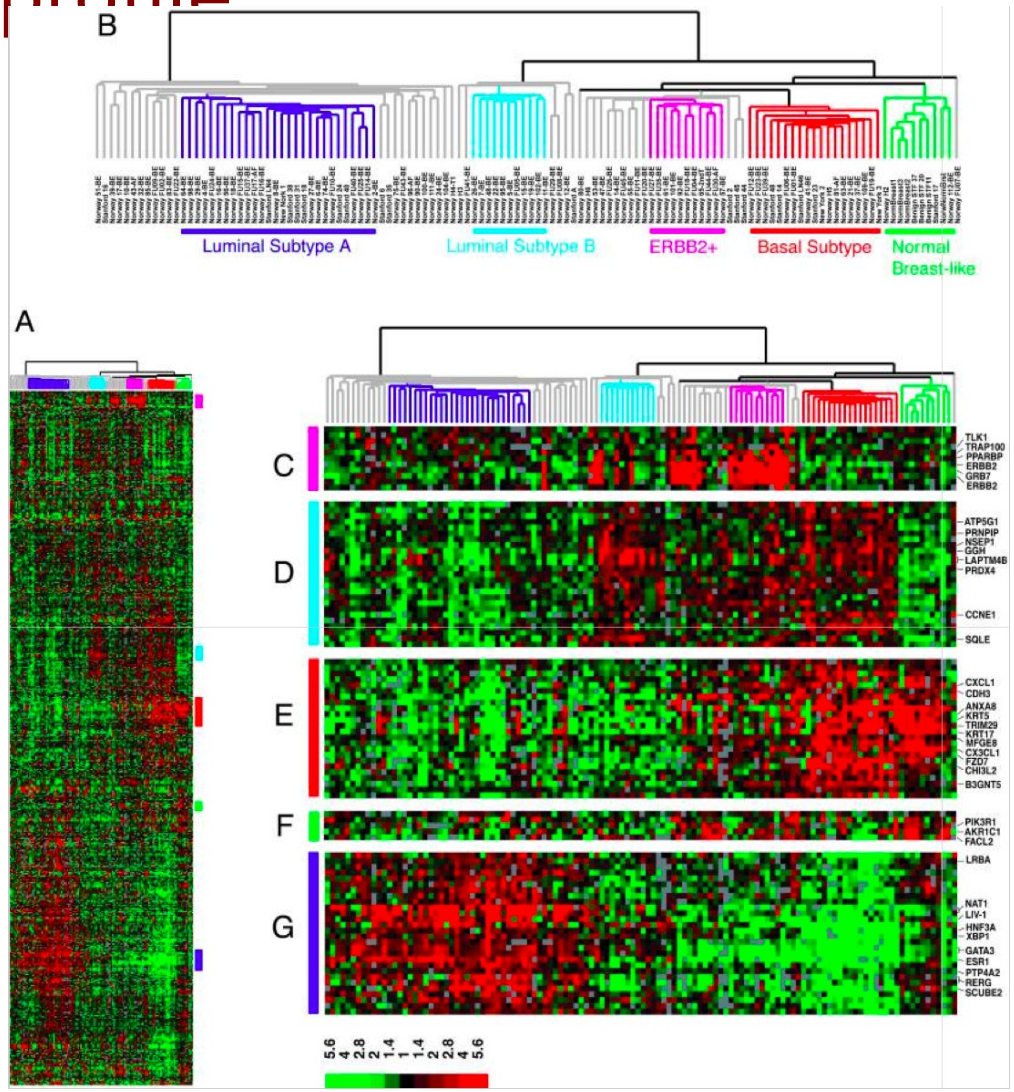
# Predict whether someone will have a heart attack on the basis of demographic, diet and clinical measurements



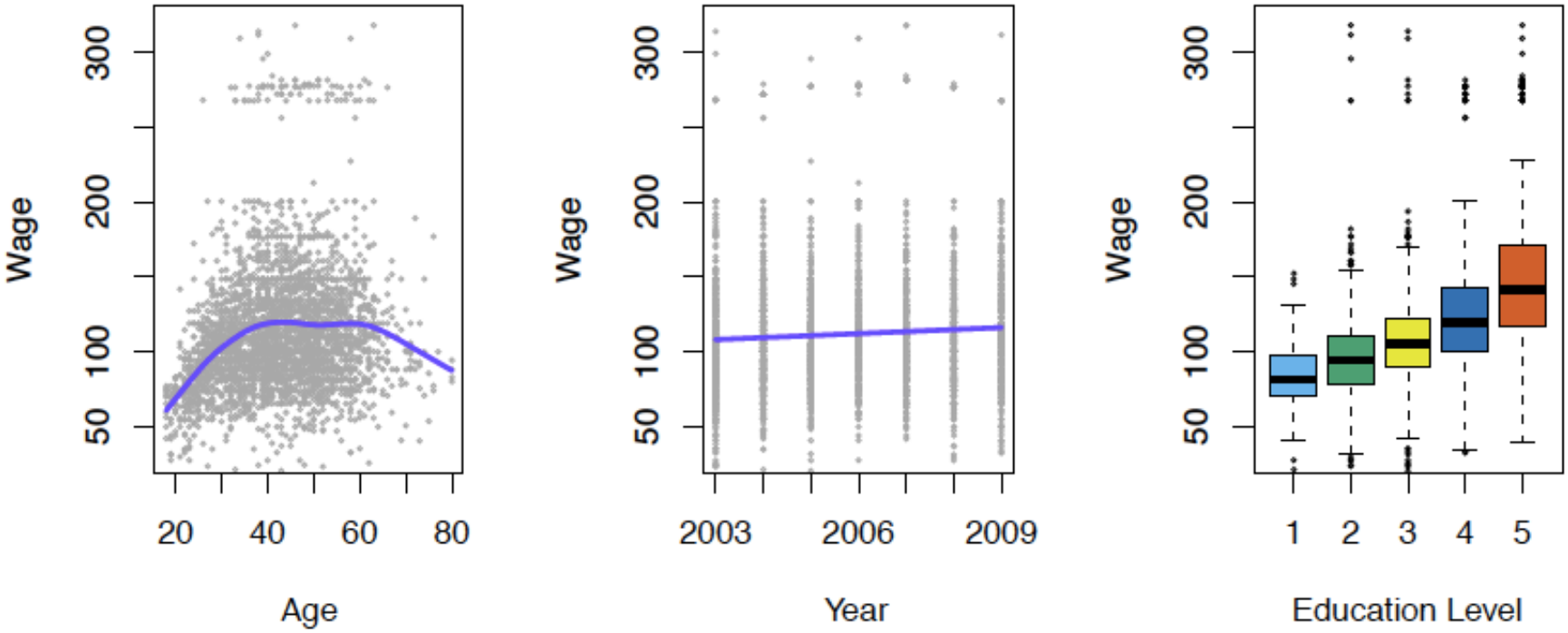
# Identify the numbers in a handwritten zip code  multiclass classification



# Classify a tissue sample into one of several cancer classes, based on a gene expression profile

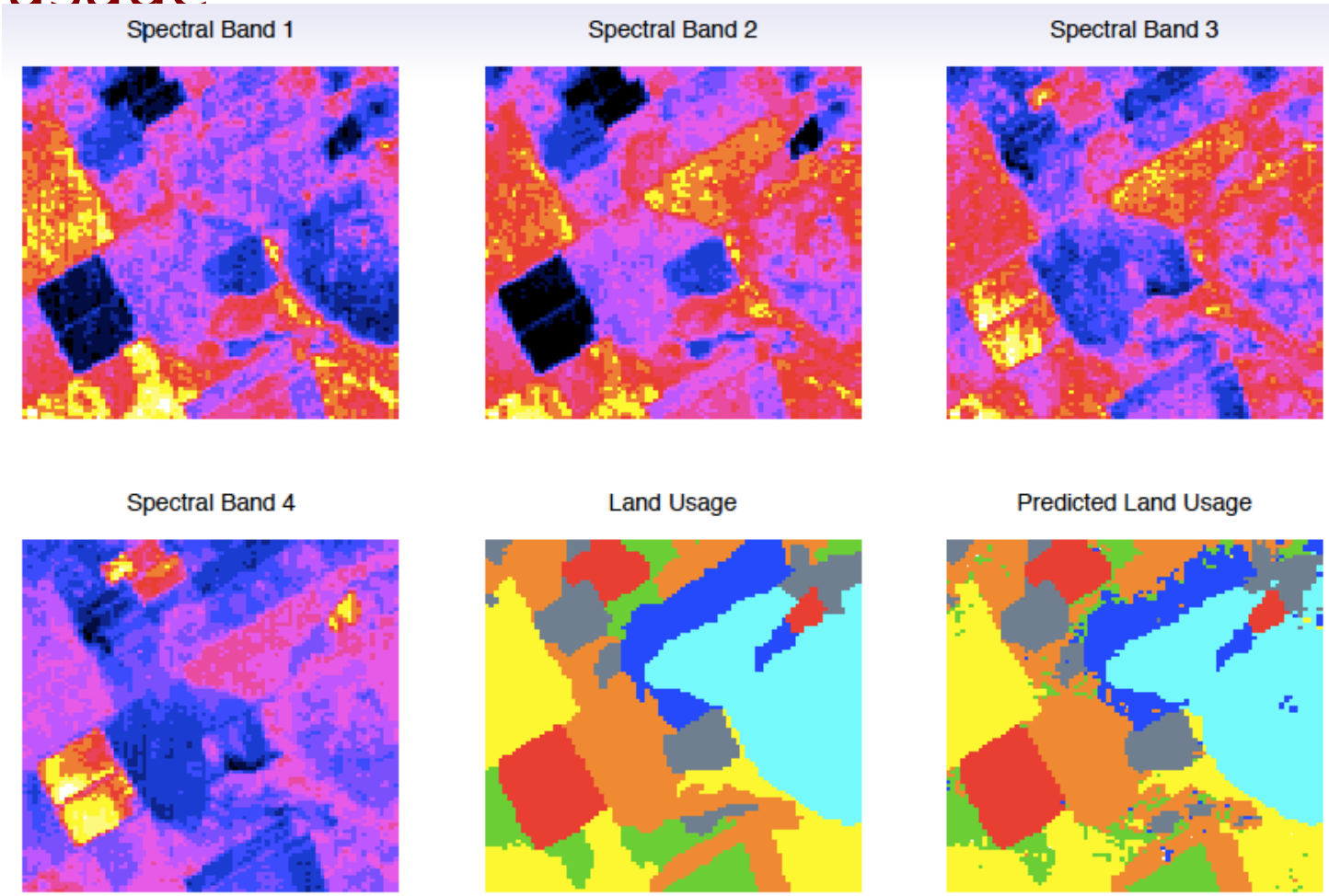


# Establish the relationship between salary and demographic variables in population survey data

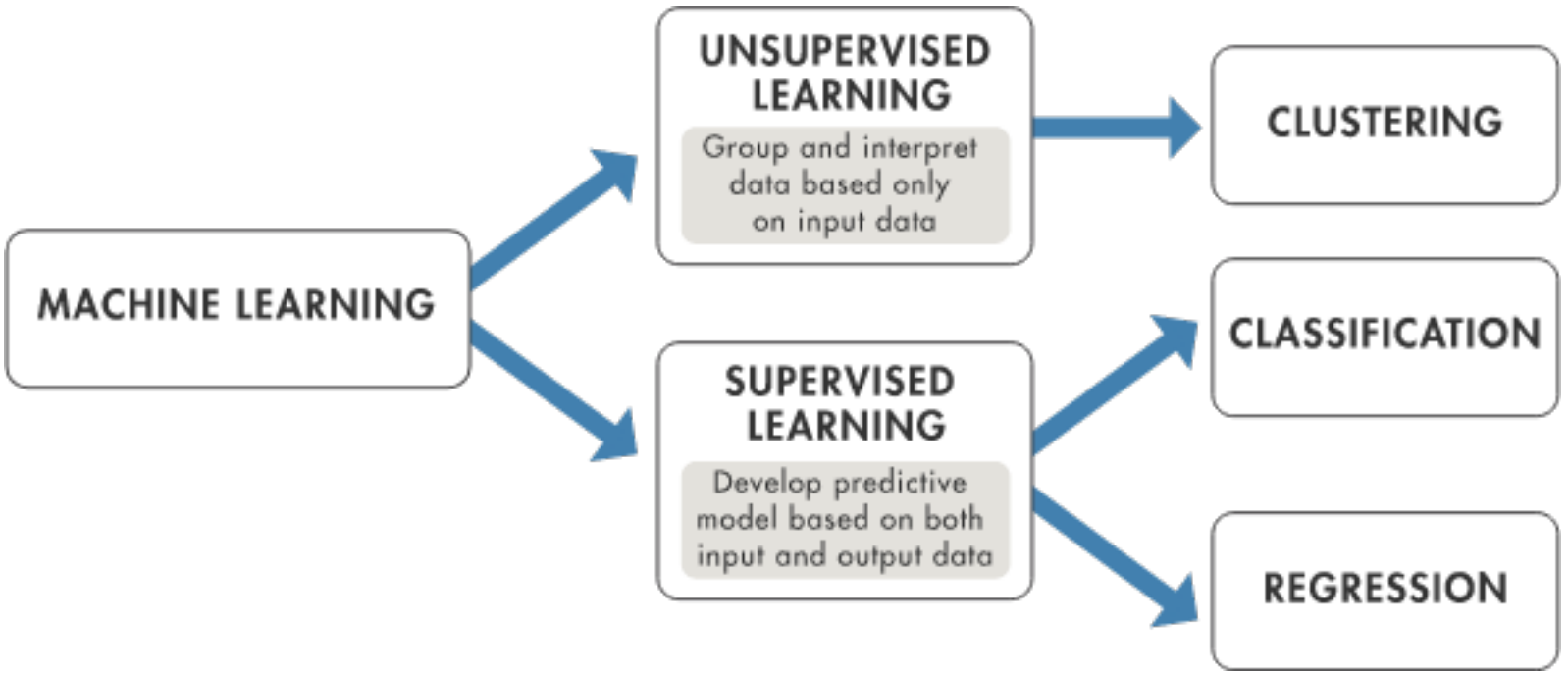



Income survey data for males from the central Atlantic region of the USA in 2009.

# Classify the pixels in a LANDSAT image, by usage




Usage $\in$ {red soil, cotton, vegetation stubble, mixture, gray soil, damp gray soil}


MACHINE LEARNING
UNSUPERVISED LEARNING
Group and interpret data based only on input data
SUPERVISED LEARNING
Develop predictive model based on both input and output data
CLUSTERING
CLASSIFICATION
REGRESSION

# Machine learning ⊆ artificial intelligence

## ARTIFICIAL INTELLIGENCE

Design an intelligent agent that perceives its environment and makes decisions to maximize chances of achieving its goal.
Subfields: vision, robotics, machine learning, natural language processing, planning, …

## MACHINE LEARNING

Gives "computers the ability to learn without being explicitly programmed" (Arthur Samuel, 1959)

### SUPERVISED LEARNING

Classification, regression

### UNSUPERVISED LEARNING

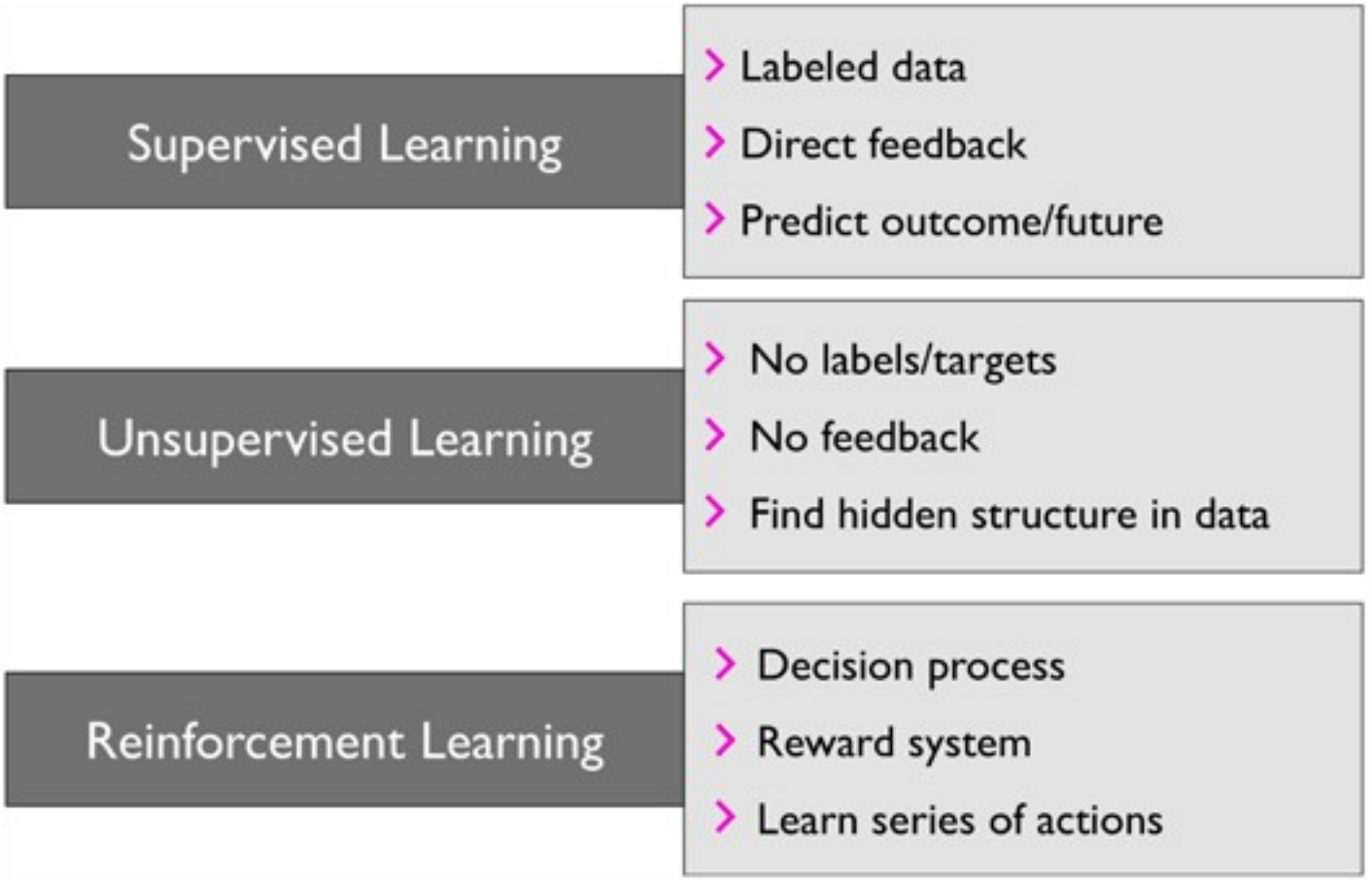
Clustering, dimensionality reduction, recommendation

### REINFORCEMENT LEARNING

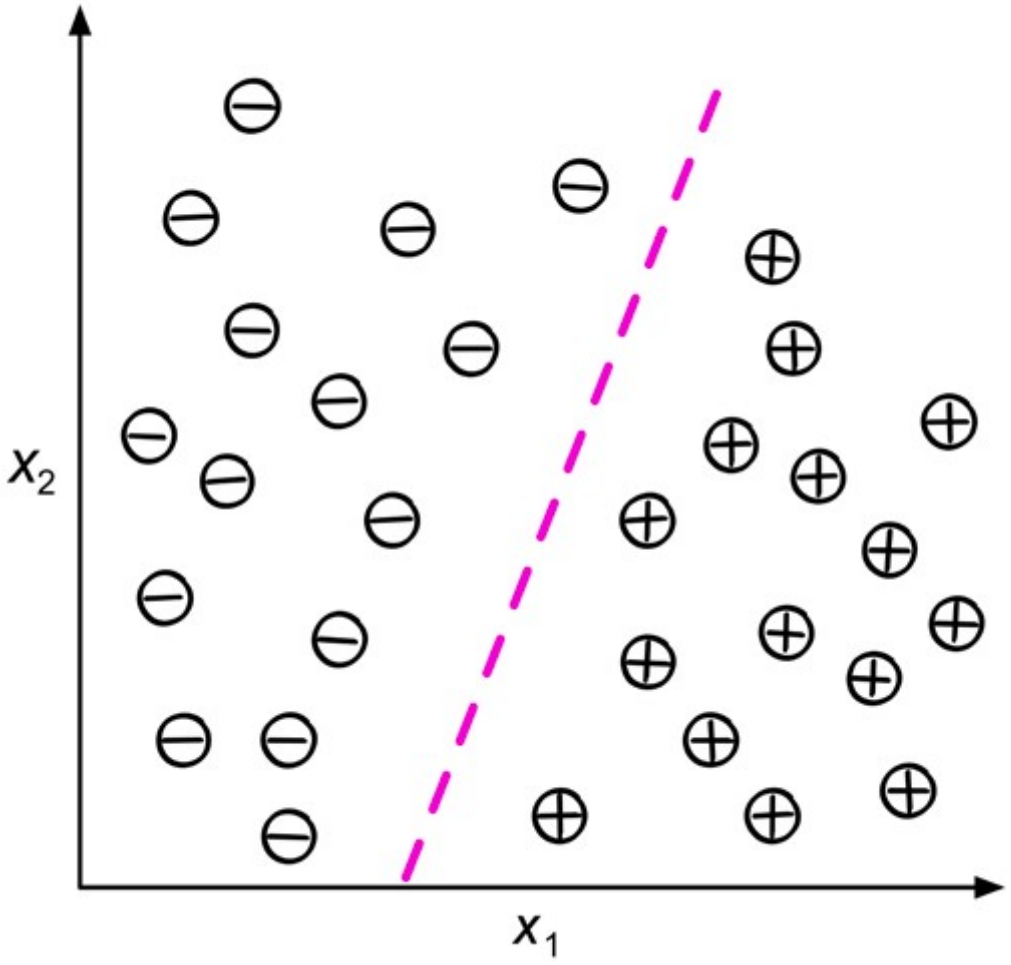
Reward maximization

*Machine Learning for Humans* 🤖👶

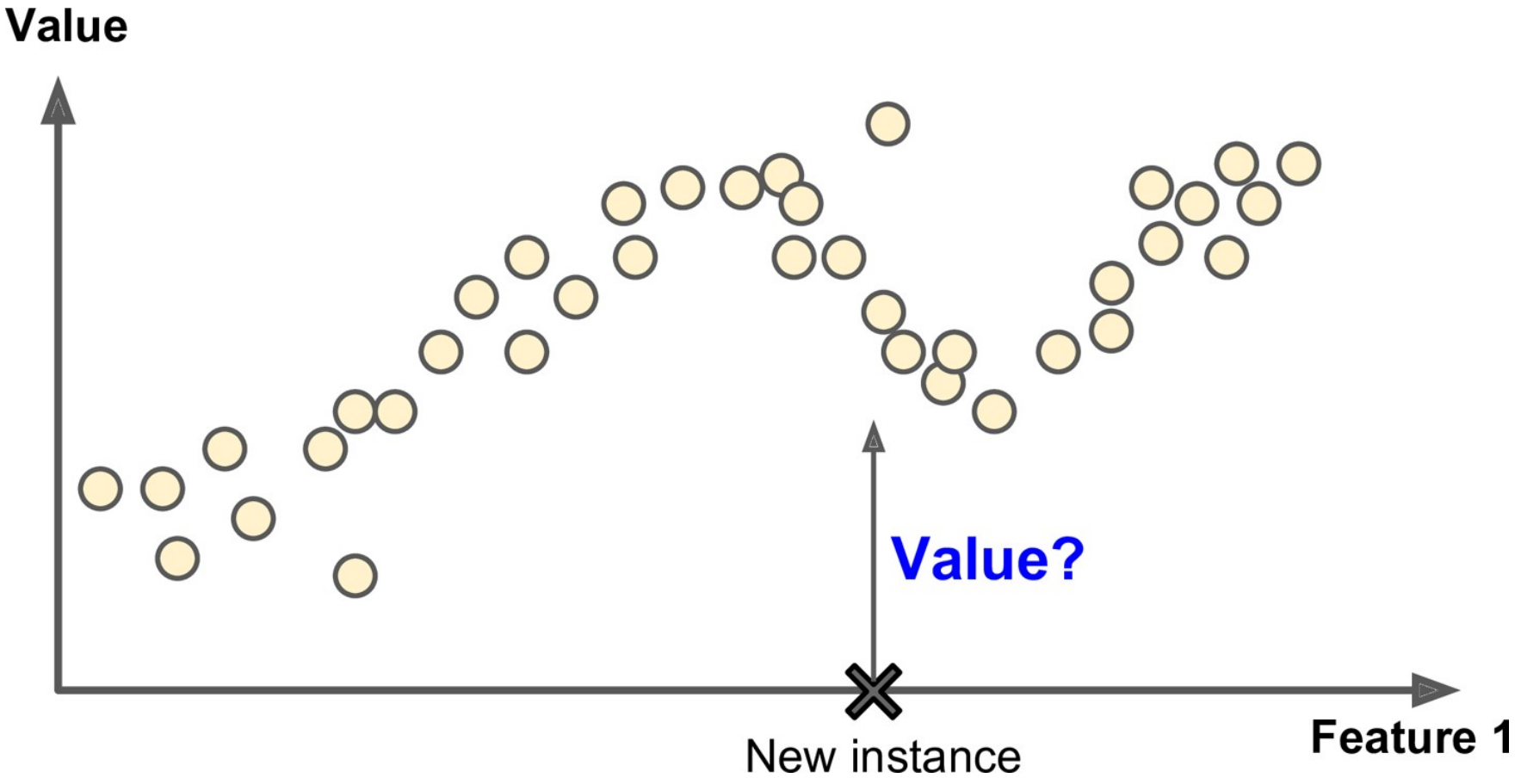
# Types of machine learning



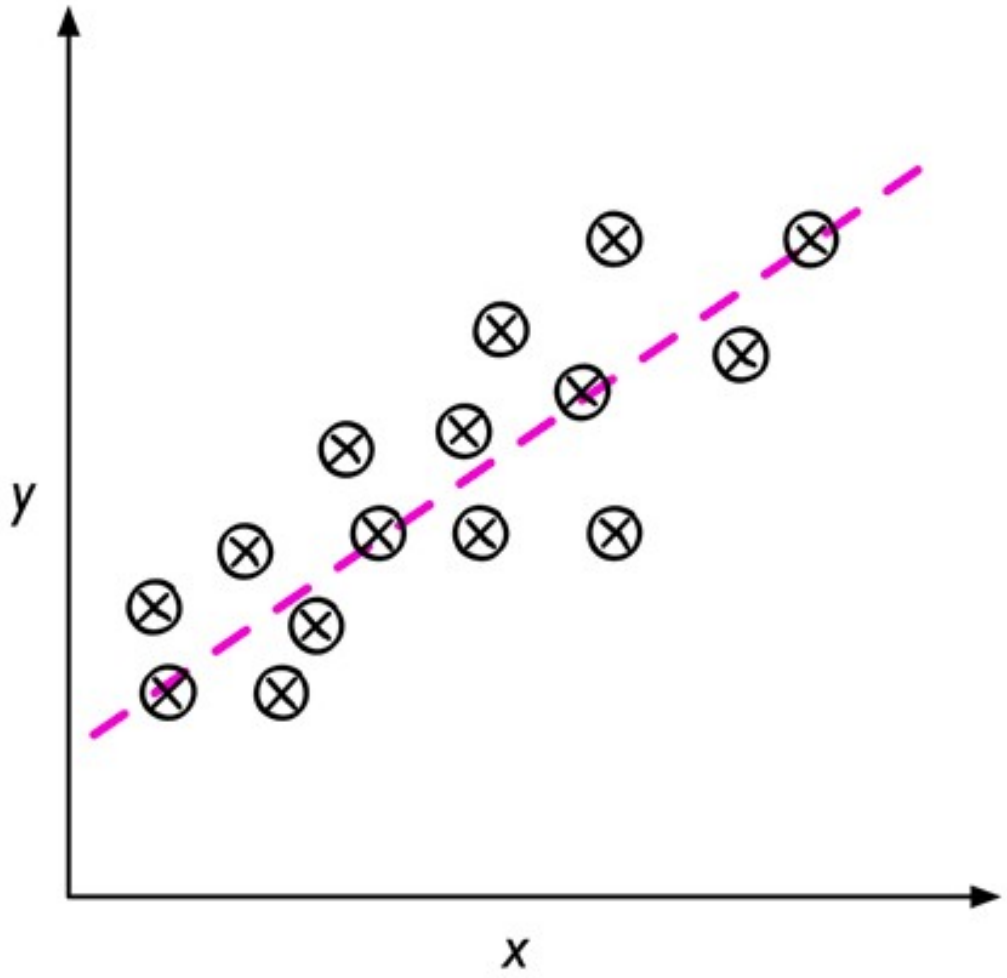
# Binary classification



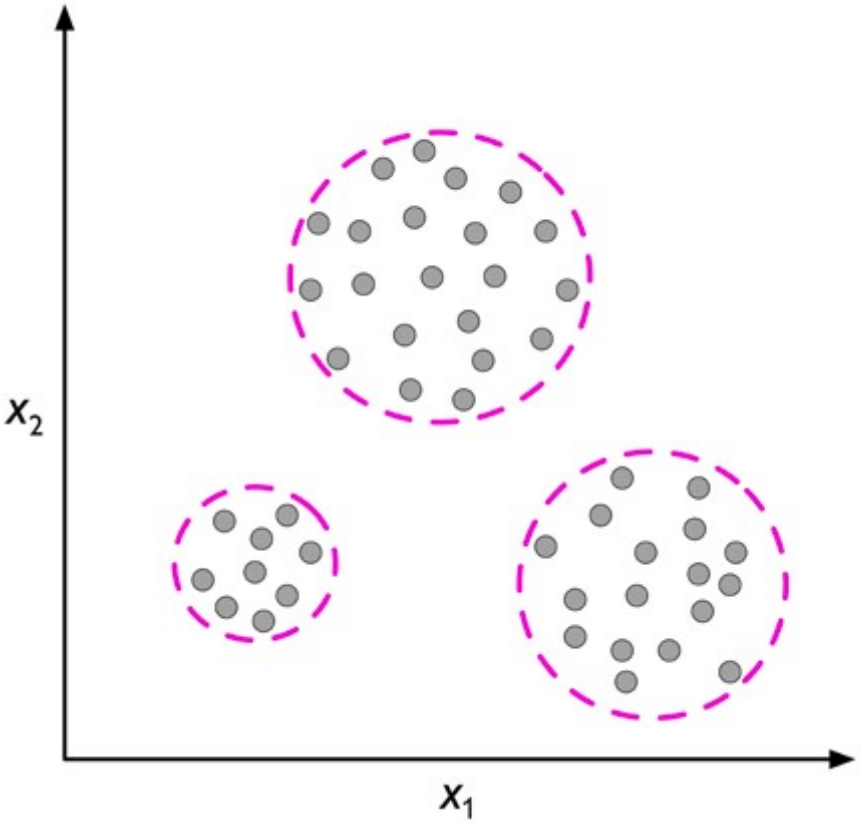
# Regression



# Regression



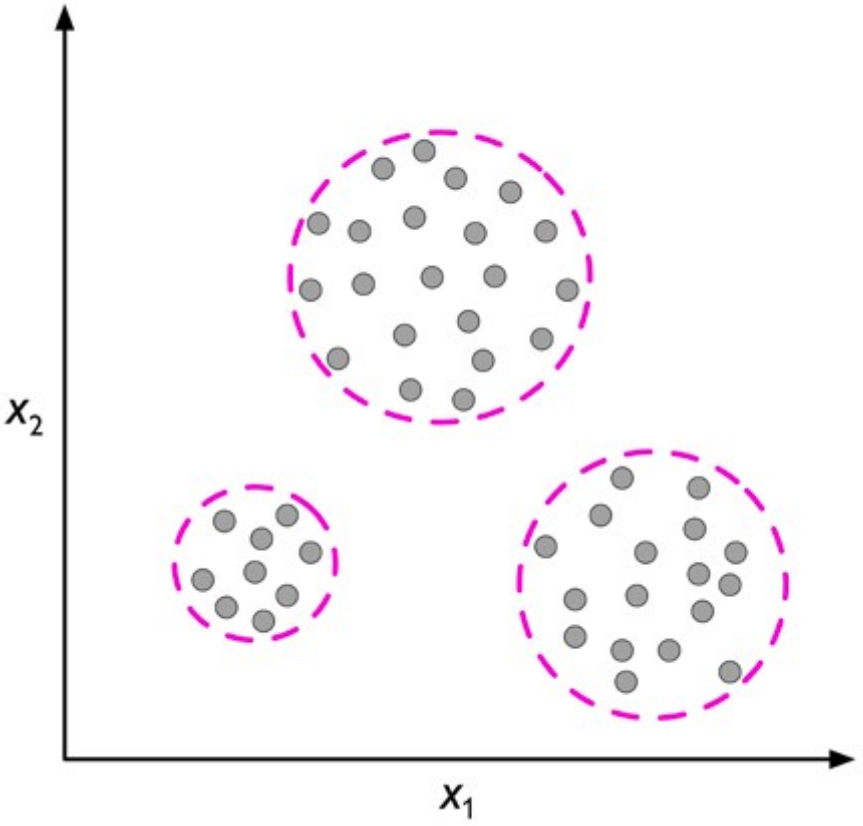
# Clustering




Exploratory data analysis technique that allows to organize a pile of information into meaningful subgroups (clusters) without having any prior knowledge of their group memberships.

Each cluster defines a group of objects that share a certain degree of similarity but are more dissimilar to objects in other clusters.

# Clustering




Clustering can be applied to organizing unlabeled data into three distinct groups based on the similarity of their features $x_1$ and $x_2$.

For example, it allows marketers to discover customer groups based on their interests, in order to develop distinct marketing programs.

# Dimensionality reduction

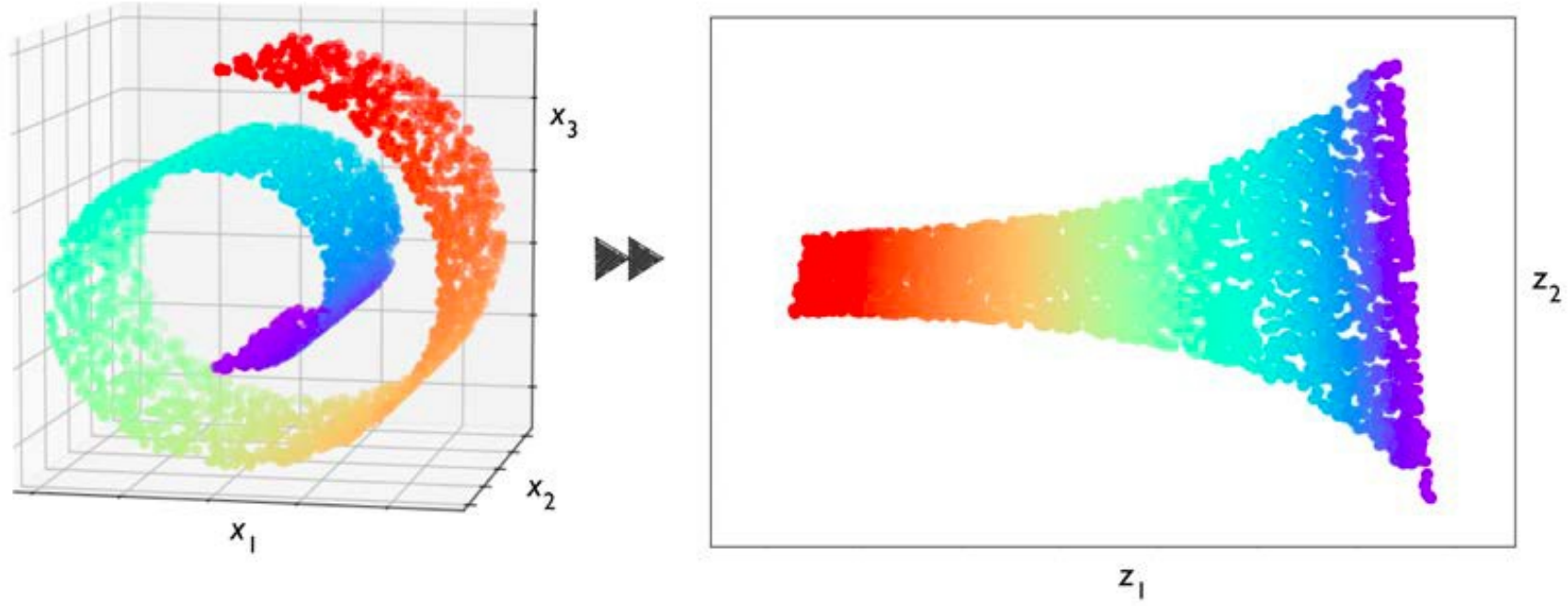
Commonly used approach in feature preprocessing to **remove noise from data**, which can also degrade the predictive performance of certain algorithms, and **compress the data** onto a smaller dimensional subspace while **retaining most of the relevant information**.

It also be useful for visualizing data:

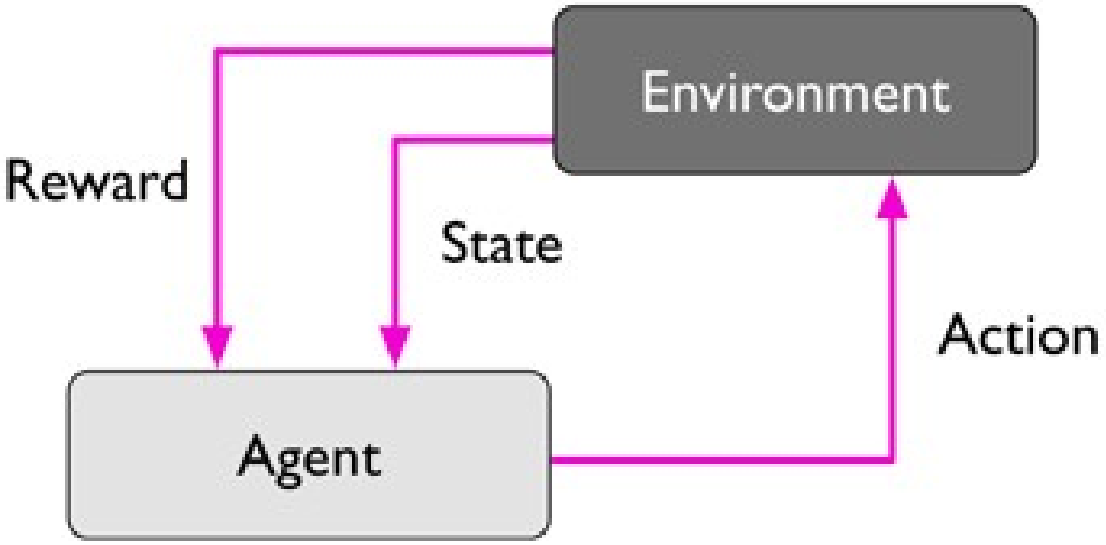
- A high-dimensional feature set can be projected onto one-, two-, or three-dimensional feature spaces in order to visualize it via 3D or 2D scatterplots or histograms.

# Dimensionality reduction

Nonlinear dimensionality reduction applied to compress a 3D Swiss Roll onto a new 2D feature subspace:

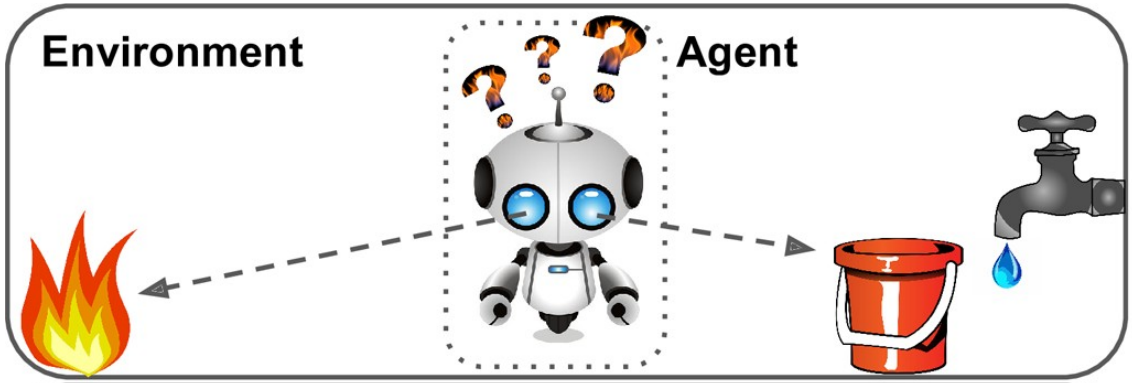


# Reinforcement learning

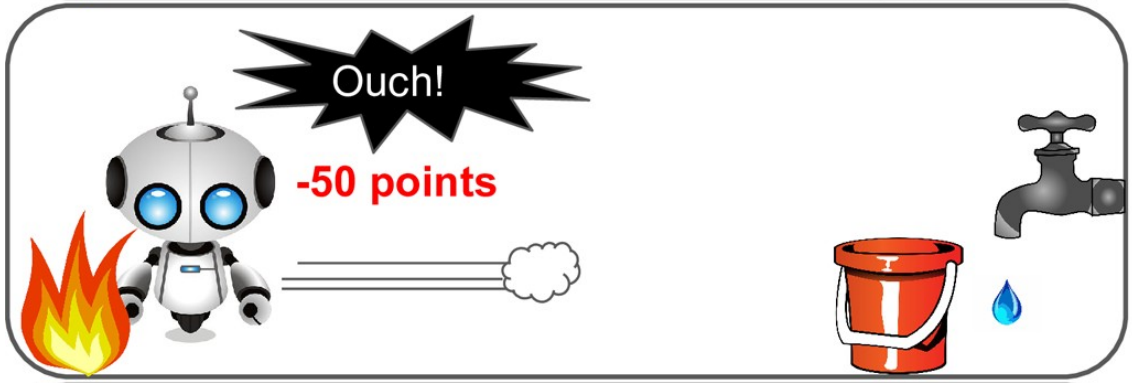



Reward can be defined as win or lose at the end of the game.

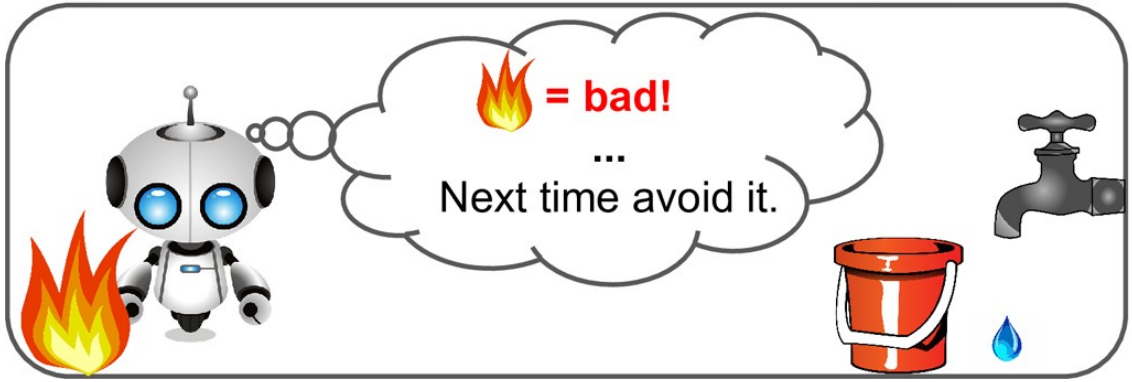
# Reinforcement learning




1. Observe
2. Select action using policy




3. Action!
4. Get reward or penalty




5. Update policy (learning step)
6. Iterate until an optimal policy is found

# Reinforcement learning

# Reinforcement learning

DeepMind's AlphaGo program learned its winning policy by analyzing millions of games, and then playing many games against itself.

Note that learning was turned off during the games against the champion; AlphaGo was just applying the policy it had learned.

# Deep Reinforcement Learning

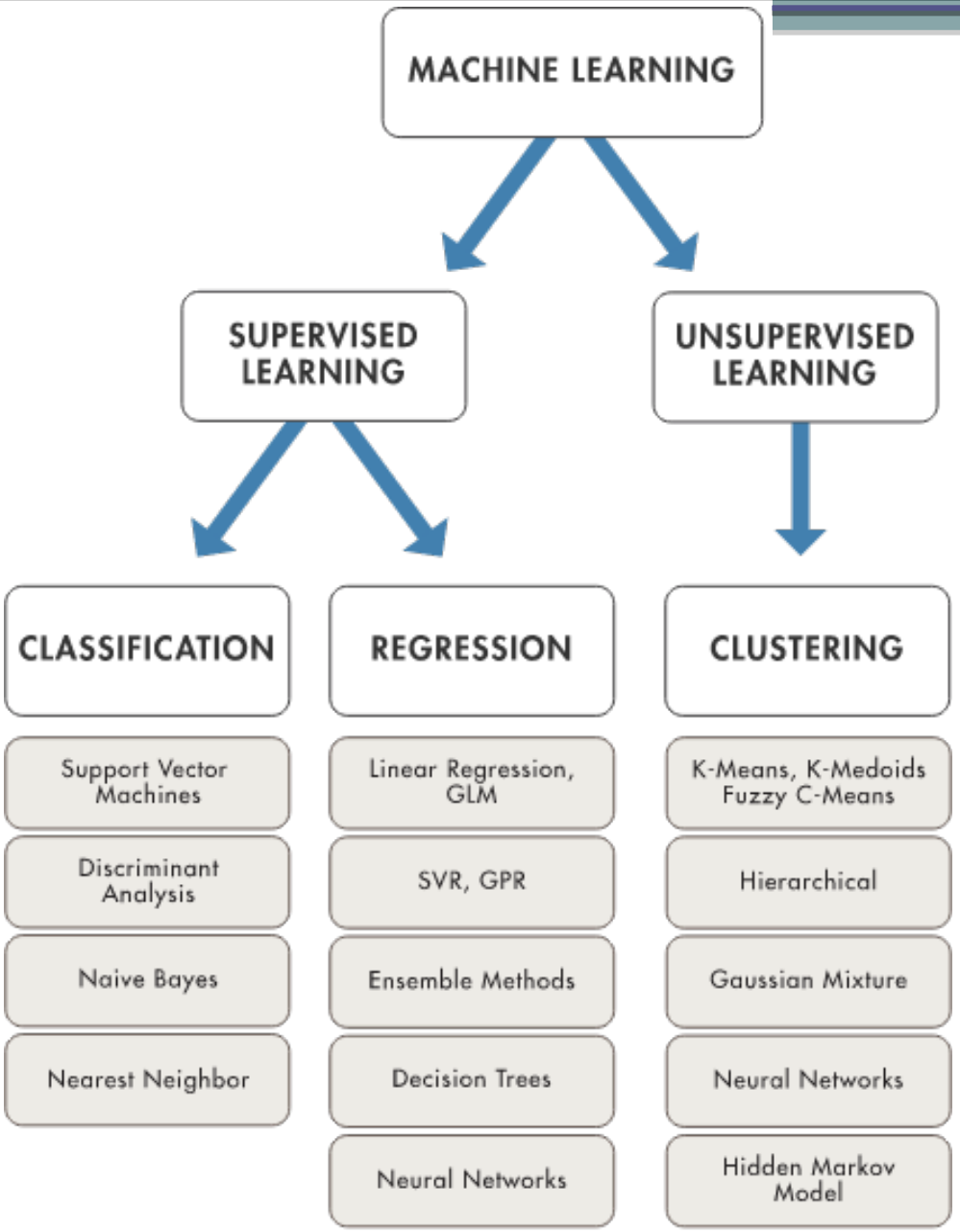
2015 – AlphaGo learned from moves based on the historical tournament data.

2017 – AlphaGo Zero - AlphaGo Zero's strategies were self-taught.

2018 - AlphaZero, a modified version of AlphaGo Zero, gained superhuman abilities at chess and shogi. Like AlphaGo Zero, AlphaZero learned solely through self-play.

2018 – AlphaFold - protein-folding, one of the toughest problems in science.

2019 – AlphaStar - knowledge equivalent to 200 years of playing time.


MACHINE LEARNING
SUPERVISED LEARNING
UNSUPERVISED LEARNING
CLASSIFICATION
REGRESSION
CLUSTERING
Support Vector Machines
Discriminant Analysis
Naive Bayes
Nearest Neighbor
Linear Regression, GLM
SVR, GPR
Ensemble Methods
Decision Trees
Neural Networks
K-Means, K-Medoids Fuzzy C-Means
Hierarchical
Gaussian Mixture
Neural Networks
Hidden Markov Model

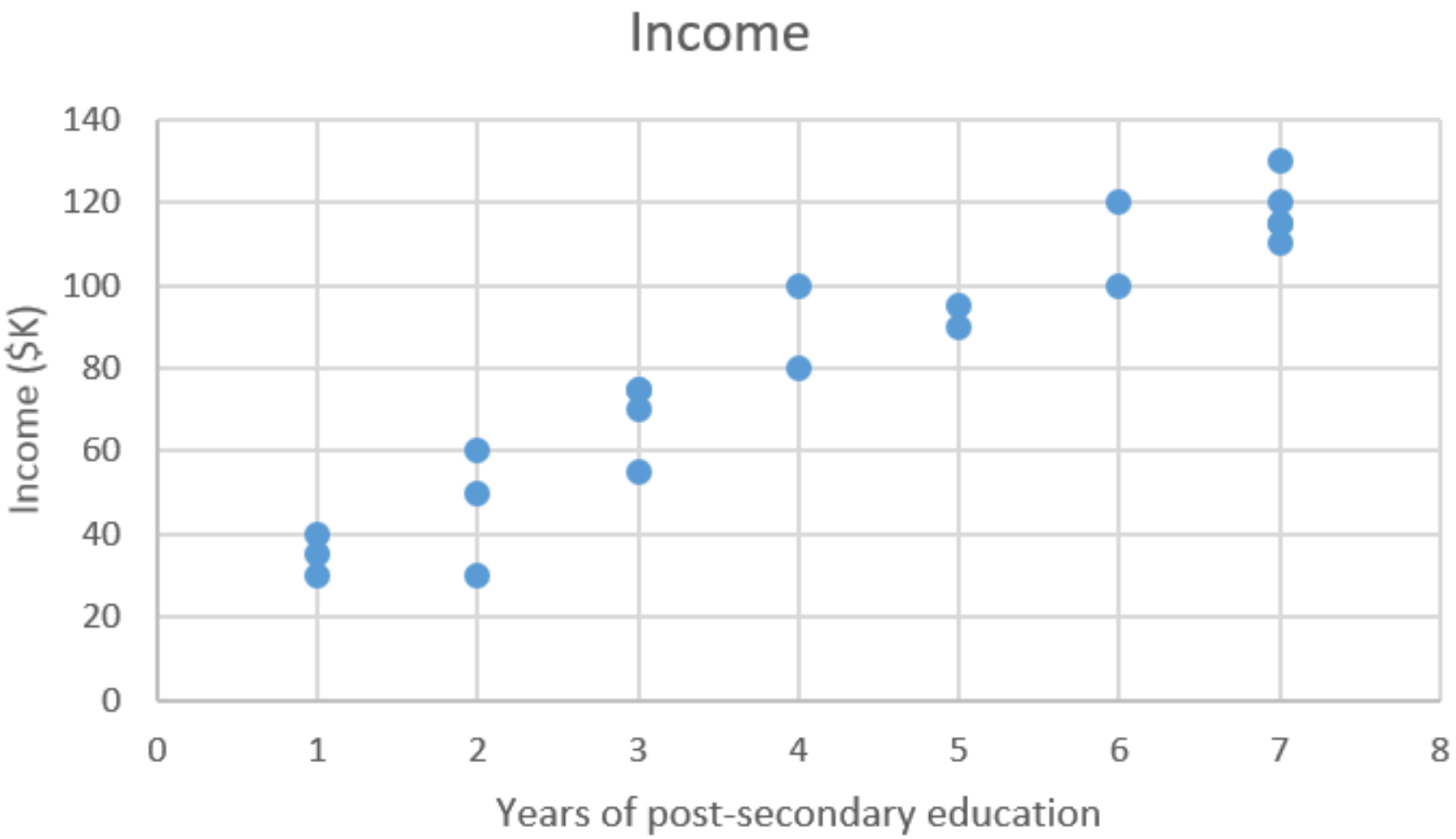
# Supervised Learning: Regression

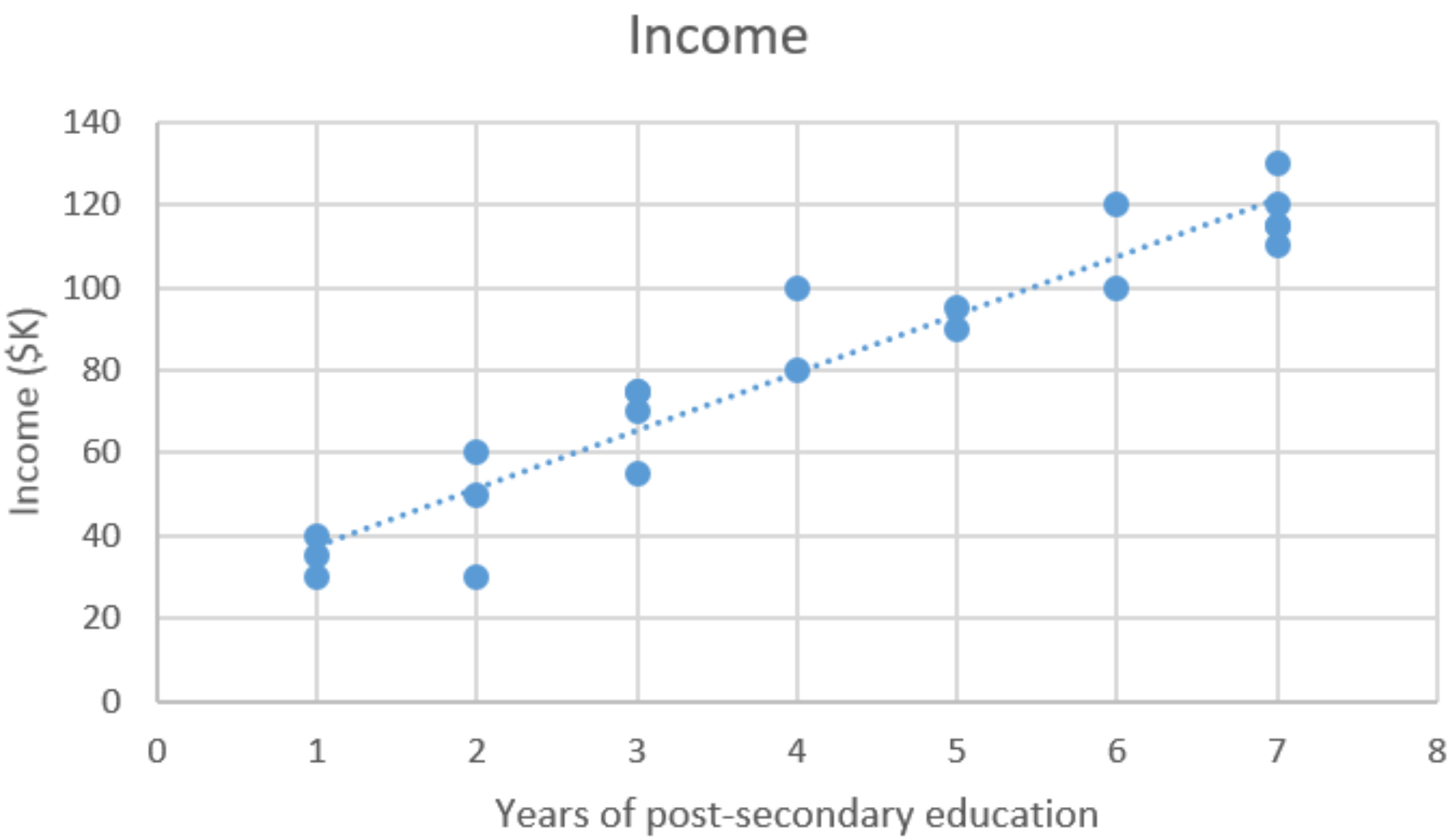
training set

| Observation # | Years of Higher Education (X) | Income (Y) |
|---|---|---|
| 1 | 4 | $80,000 |
| 2 | 5 | $91,500 |
| 3 | 0 | $42,000 |
| 4 | 2 | $55,000 |
| … | … | … |
| N | 6 | $100,000 |

test set

| | | |
|---|---|---|
| 1 | 4 | ??? |
| 2 | 6 | ??? |


Income
Income ($K)
140
120
100
80
60
40
20
0
0
1
2
3
4
5
6
7
8
Years of post-secondary education


Income
Income ($K)
140
120
100
80
60
40
20
0
0
1
2
3
4
5
6
7
8
Years of post-secondary education

# Supervised Learning: Classification

training set

| Observation # | Input image (X) | Label (Y) |
|---|---|---|
| 1 |  | "dog" |
| 2 |  | "cat" |
| 3 |  | "dog" |
| … | … | … |
| N |  | "dog" |

test set

| | | |
|---|---|---|
| 1 |  | ??? |
| 2 |  | ??? |

# Aprendizagem não supervisionada

# Unsupervised learning algorithms

**Clustering**

- k-Means
- Hierarchical Cluster Analysis (HCA)
- Expectation Maximization

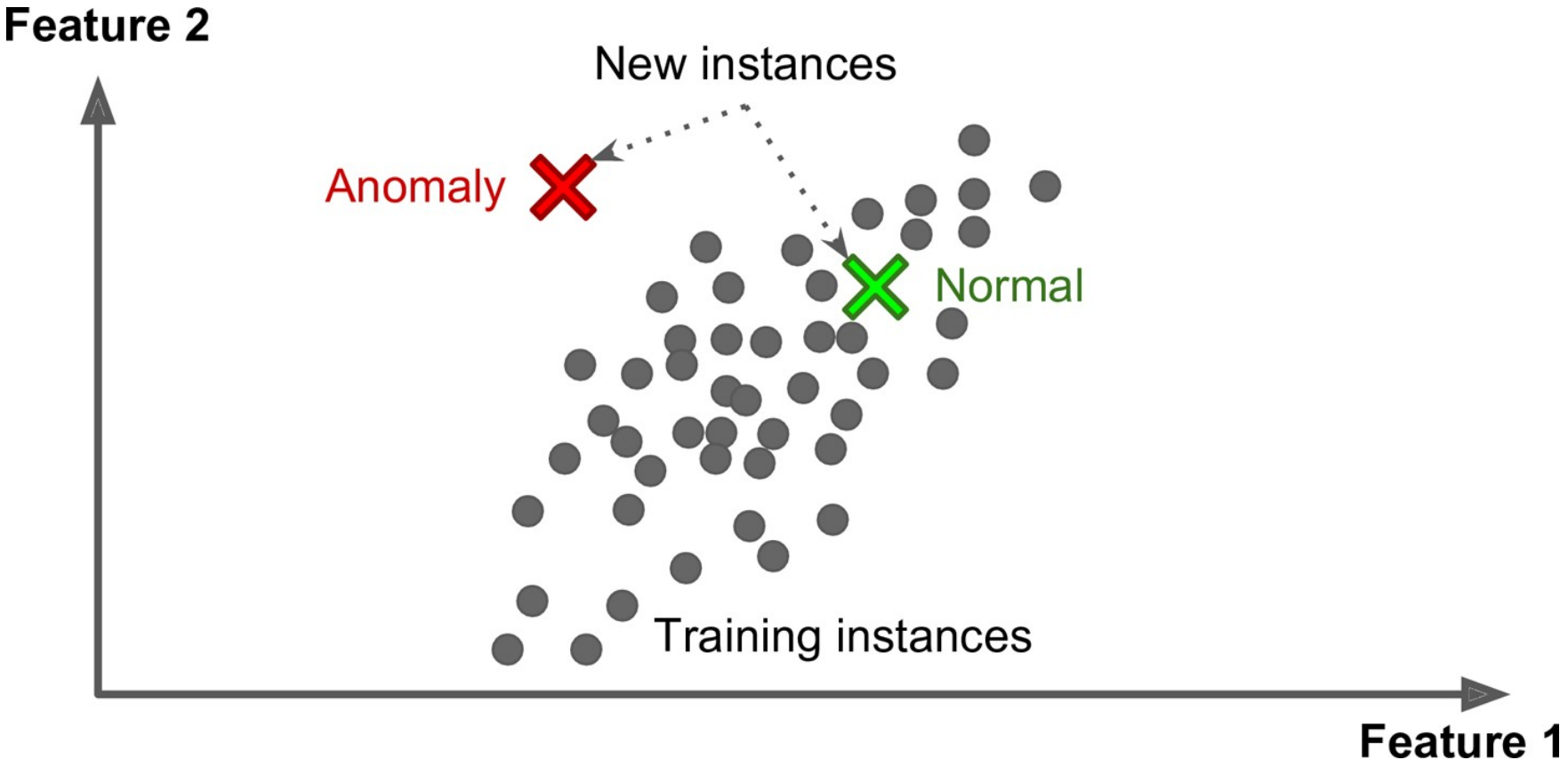
**Visualization and dimensionality reduction**

- Principal Component Analysis (PCA)
- Kernel PCA
- Locally-Linear Embedding (LLE)
- t-distributed Stochastic Neighbor Embedding (t-SNE)

**Association rule learning**

- Apriori
- Eclat

# Unsupervised task - anomaly detection



# Unsupervised task - anomaly detection

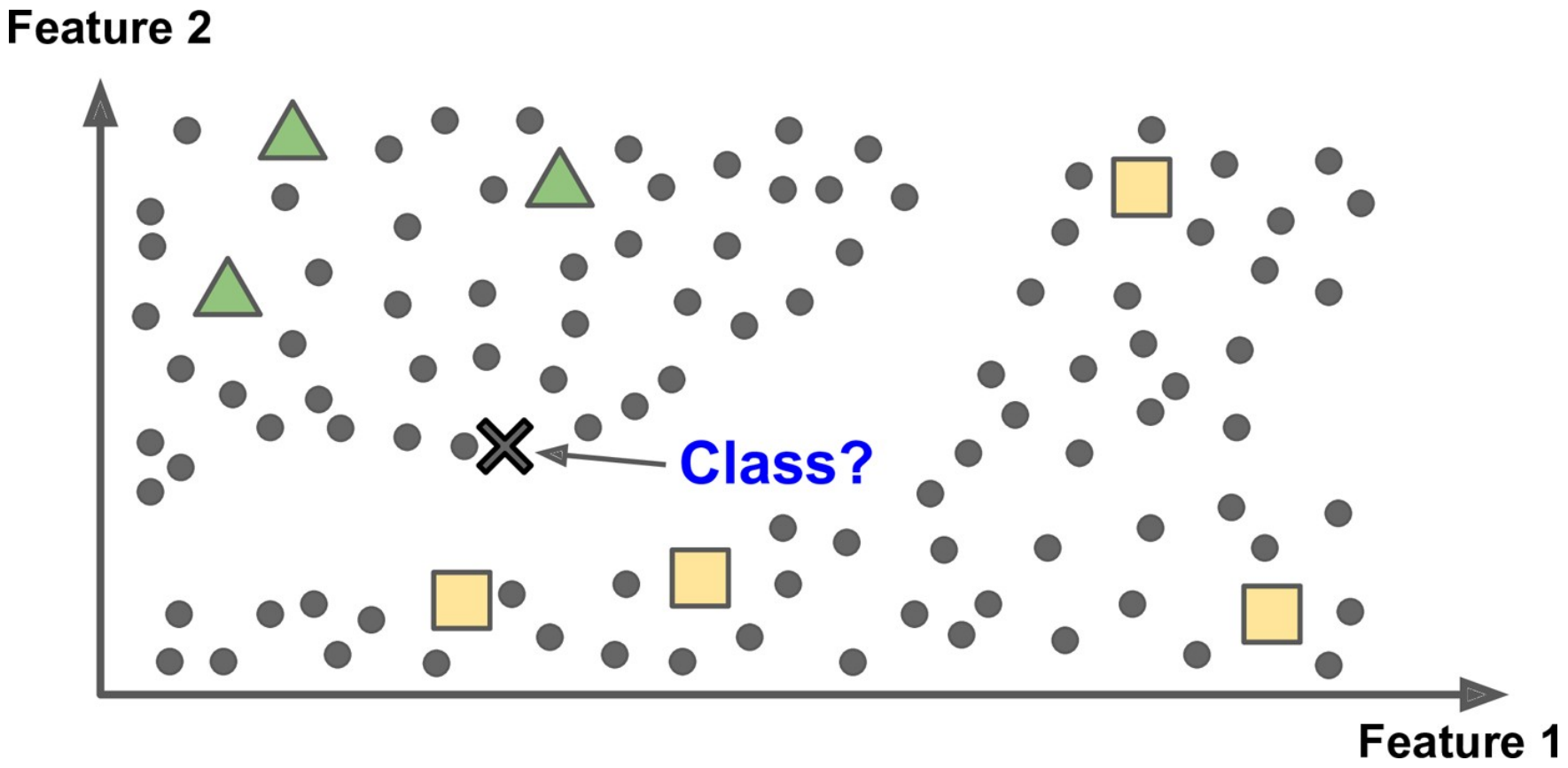
Examples:

- Detecting unusual credit card transactions to prevent fraud;
- Catching manufacturing defects;
- Automatically removing outliers from a dataset before feeding it to another learning algorithm.

The system is trained with normal instances, and when it sees a new
instance it can tell whether it looks like a normal one or whether it is likely an anomaly.

# Semisupervised learning

Some algorithms can deal with partially labeled training data, usually a lot of unlabeled data and a little bit of labeled data.



# Semisupervised learning

Photo-hosting services, such as Google Photos:

- **Unsupervised learning (clustering)**
  - Once you upload all your family photos to the service, it automatically recognizes that the same person A shows up in photos 1, 5, and 11, while another person B shows up in photos 2, 5, and 7.

- **Supervised learning**
  - Tell the system who these people are. Just one label per person, and it is able to name everyone in every photo, which is useful for searching photos.

# O prêmio Netflix

- competition started in October 2006. Training data is ratings for 18, 000 movies by 400, 000 Netflix customers, each rating between 1 and 5.
- training data is very sparse— about 98% missing.
- objective is to predict the rating for a set of 1 million
- customer-movie pairs that are missing in the training data.
- Netflix’s original algorithm achieved a root MSE of 0.953. The first team to achieve a 10% improvement wins one million

# O prêmio Netflix

NETFLIX

## Netflix Prize

COMPLETED

Home | Rules | Leaderboard | Update

## Leaderboard

Showing Test Score. Click here to show quiz score

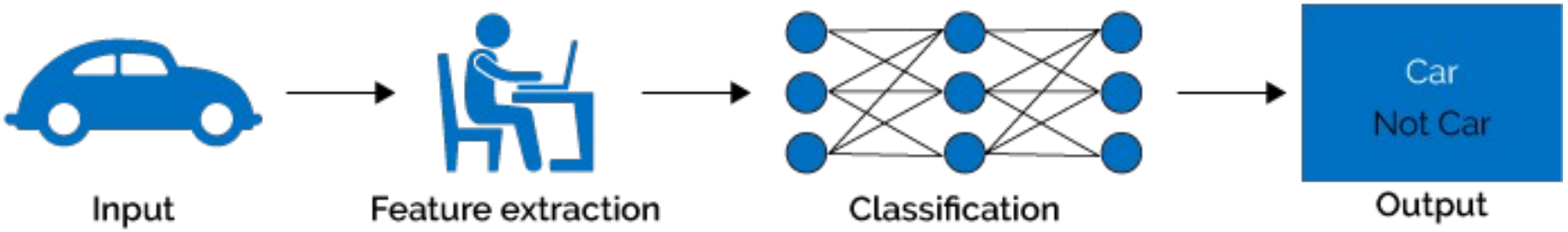
Display top 20 leaders.

| Rank | Team Name | Best Test Score | % Improvement | Best Submit Time |
|---|---|---|---|---|
| Grand Prize - RMSE = 0.8567 - Winning Team: BellKor's Pragmatic Chaos | | | | |
| 1 | BellKor's Pragmatic Chaos | 0.8567 | 10.06 | 2009-07-26 18:18:28 |
| 2 | The Ensemble | 0.8567 | 10.06 | 2009-07-26 18:38:22 |
| 3 | Grand Prize Team | 0.8582 | 9.90 | 2009-07-10 21:24:40 |
| 4 | Opera Solutions and Vandelay United | 0.8588 | 9.84 | 2009-07-10 01:12:31 |
| 5 | Vandelay Industries ! | 0.8591 | 9.81 | 2009-07-10 00:32:20 |
| 6 | PragmaticTheory | 0.8594 | 9.77 | 2009-06-24 12:06:56 |
| 7 | BellKor in BigChaos | 0.8601 | 9.70 | 2009-05-13 08:14:09 |
| 8 | Dace | 0.8612 | 9.59 | 2009-07-24 17:18:43 |
| 9 | Feeds2 | 0.8622 | 9.48 | 2009-07-12 13:11:51 |
| 10 | BigChaos | 0.8623 | 9.47 | 2009-04-07 12:33:59 |
| 11 | Opera Solutions | 0.8623 | 9.47 | 2009-07-24 00:34:07 |
| 12 | BellKor | 0.8624 | 9.46 | 2009-07-26 17:19:11 |

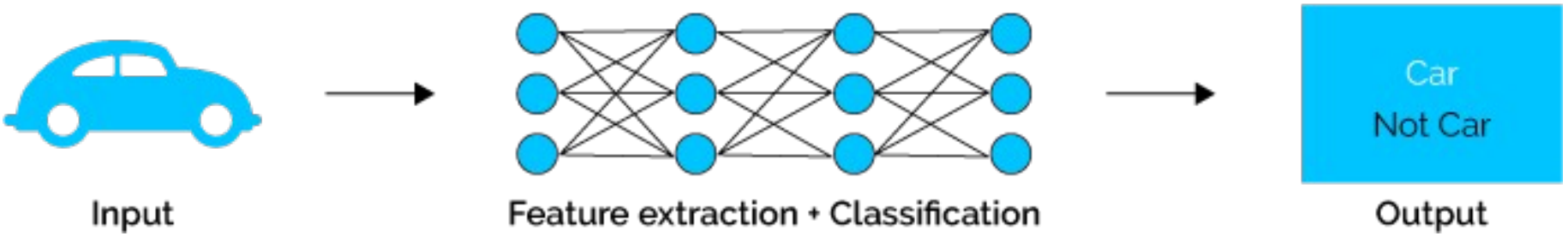
# Statistical Learning versus Machine Learning

- Machine learning arose as a subfield of Artificial Intelligence.
- Statistical learning arose as a subfield of Statistics.
- There is much overlap — both fields focus on supervised and unsupervised problems:
  - Machine learning has a greater emphasis on large scale applications and prediction accuracy.
  - Statistical learning emphasizes models and their interpretability, and precision and uncertainty.

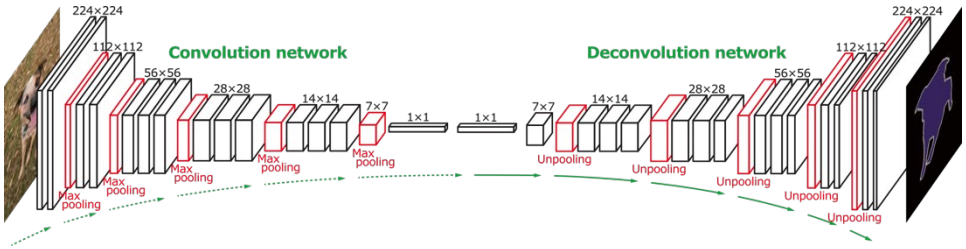
# Machine Learning

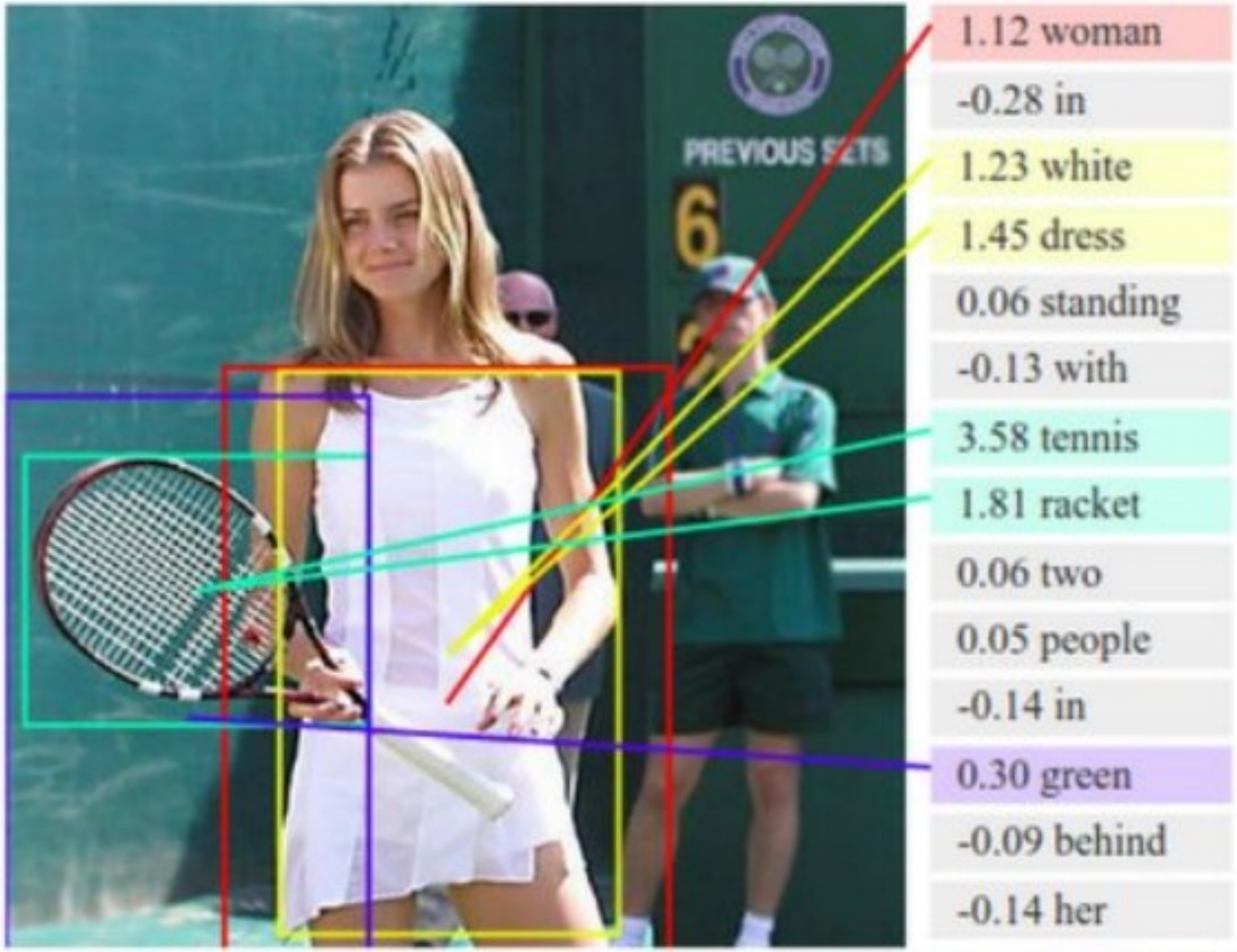



# Deep Learning



# Deep Learning




Image, automatically annotated by Deep Learning.

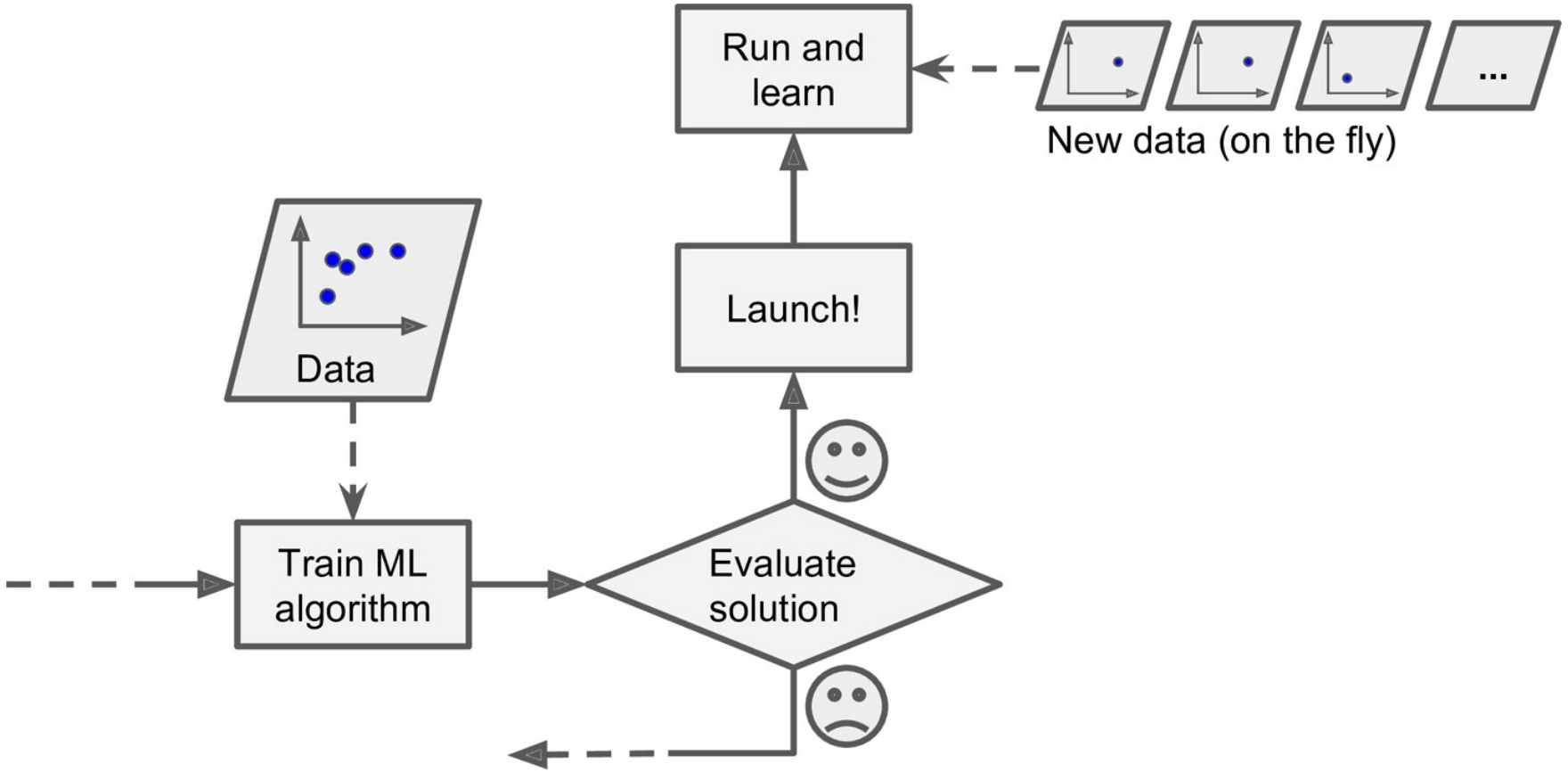
# Batch learning

- In batch learning, the system is incapable of learning incrementally: it must be trained using all the available data.
- This will generally take a lot of time and computing resources, so it is typically done offline (offline learning).
- The whole process of training, evaluating, and launching a Machine Learning system can be automated.
  - Batch learning system can adapt to change.

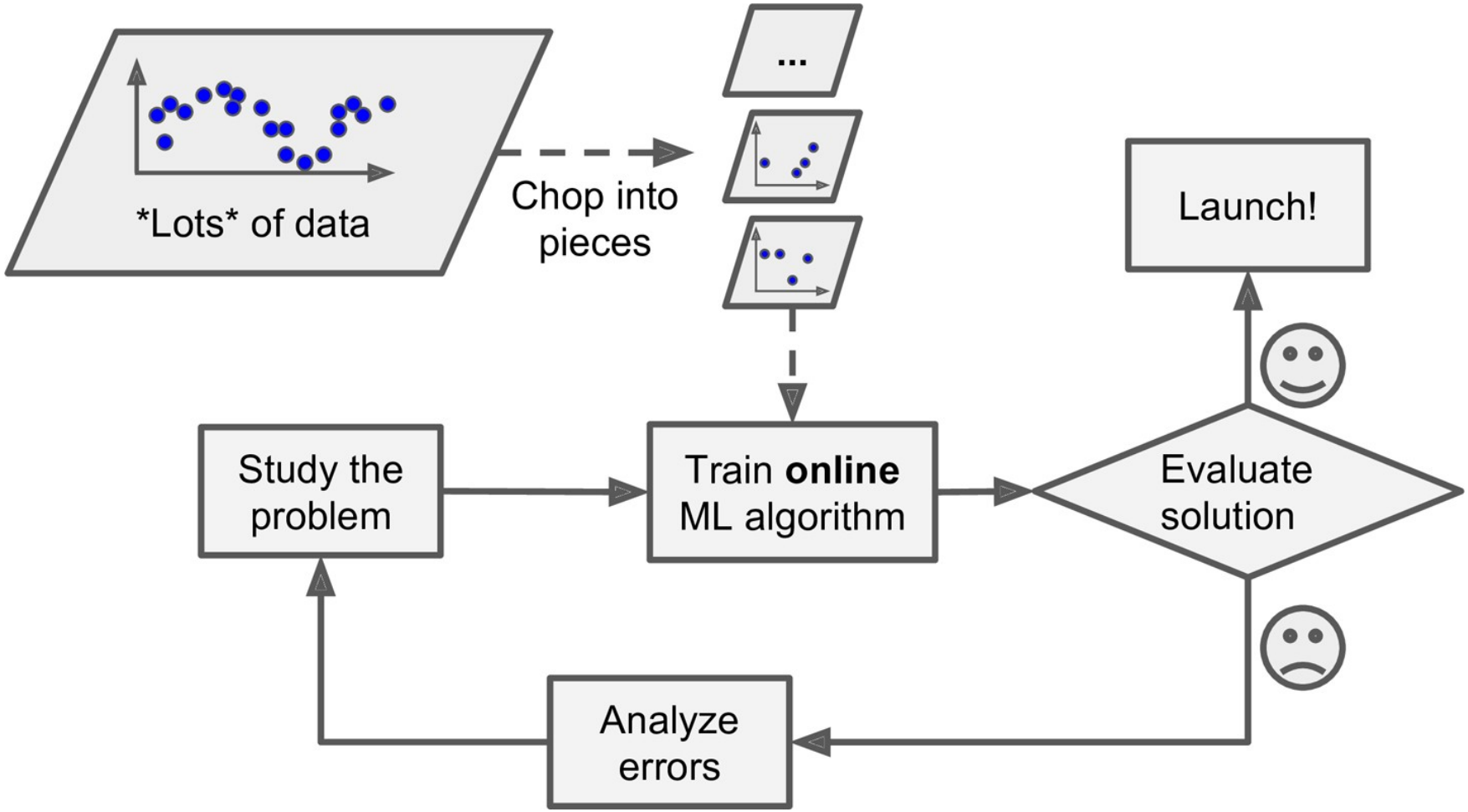
# Online learning

- In online learning, you train the system incrementally by feeding it data instances sequentially, either individually or by small groups called mini-batches.
- Each learning step is fast and cheap, so the system can learn about new data on the fly, as it arrives

# Online learning

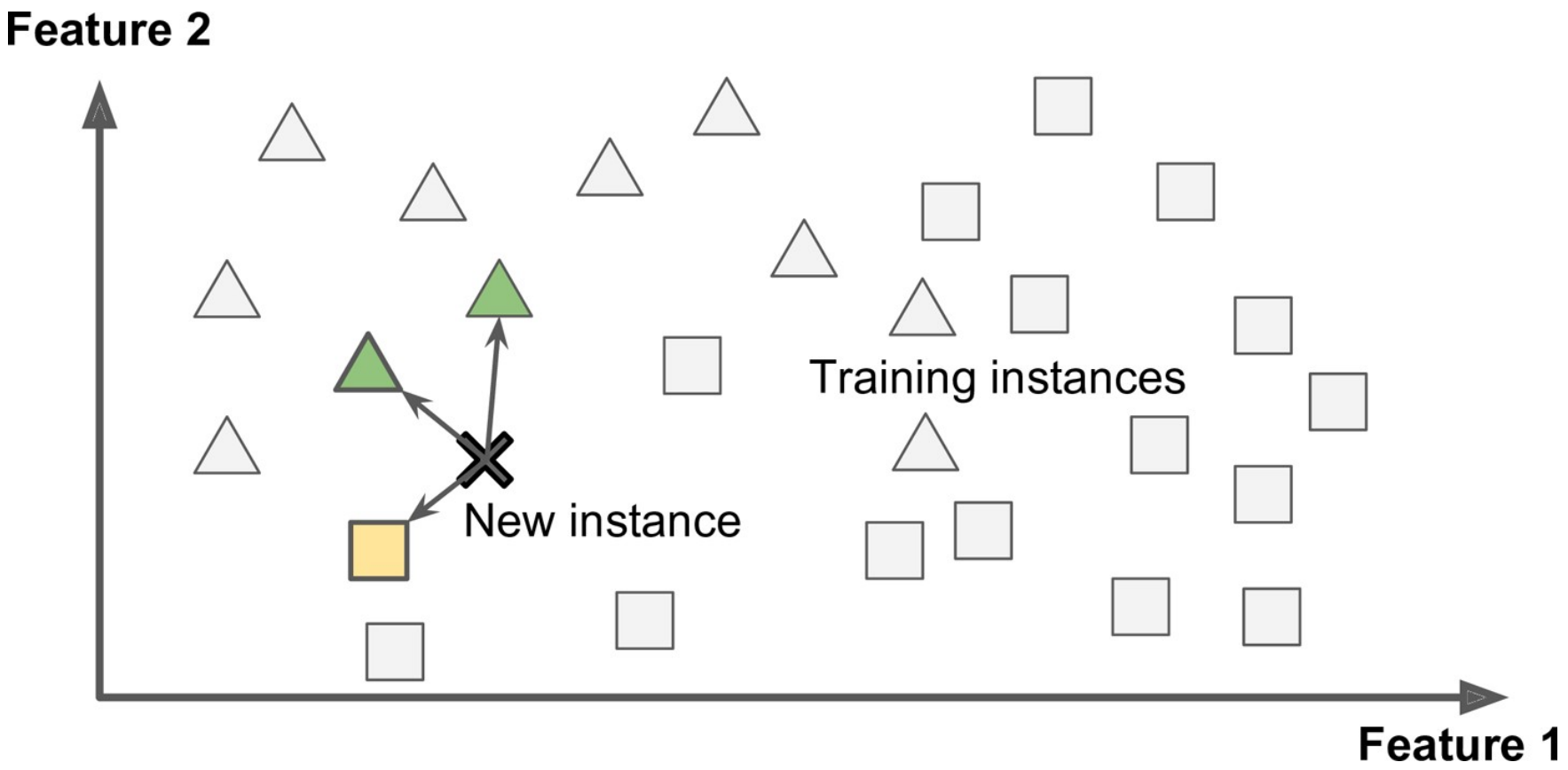


# Online learning to handle huge datasets



# Instance-based learning

The system learns the examples by heart, then generalizes to new cases using a similarity measure.



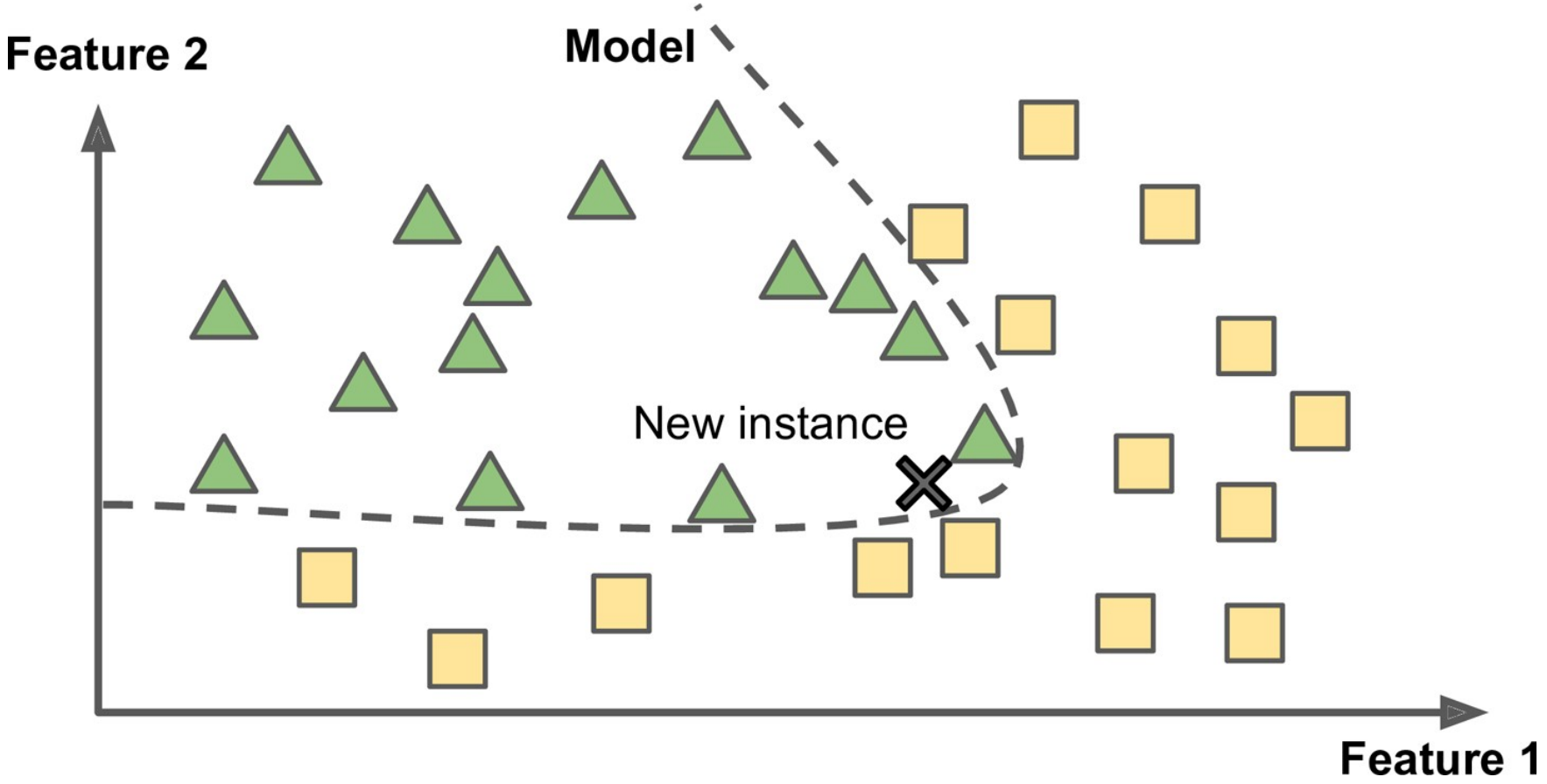
# Instance-based learning

A (very basic) similarity measure between two emails could be to count the number of words they have in common.

The system would flag an email as spam if it has many words in common with a known spam email.

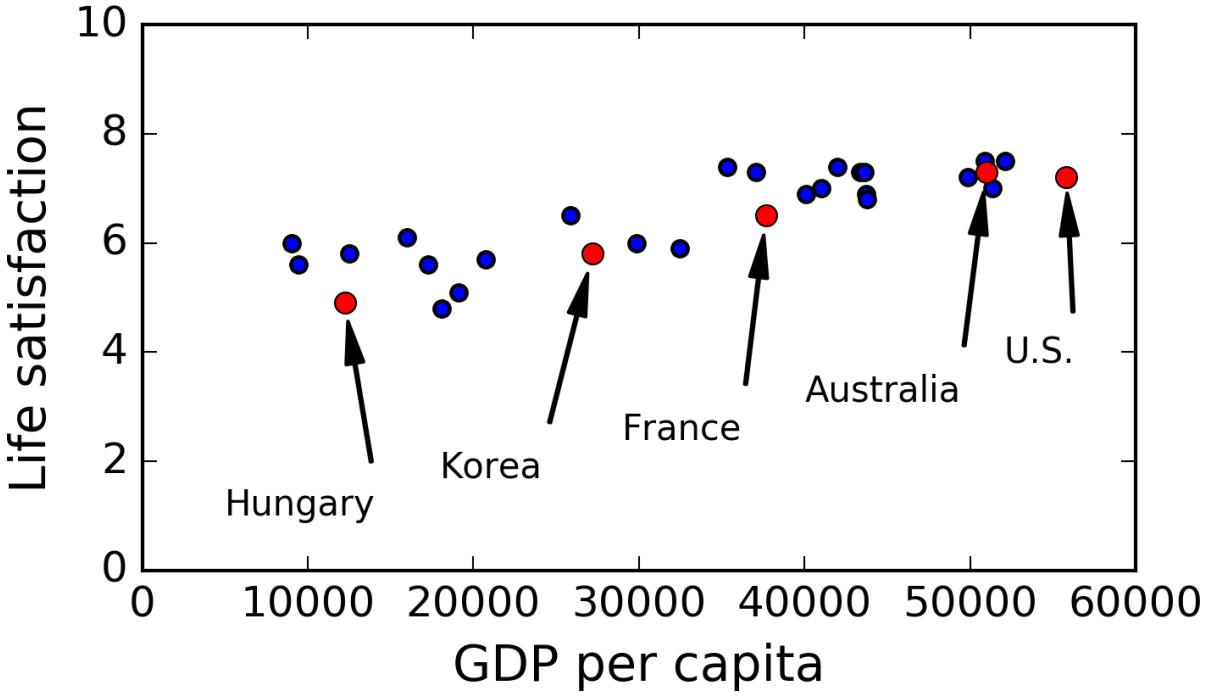
# Model-based learning

Build a model of these examples, then use that model to make predictions.

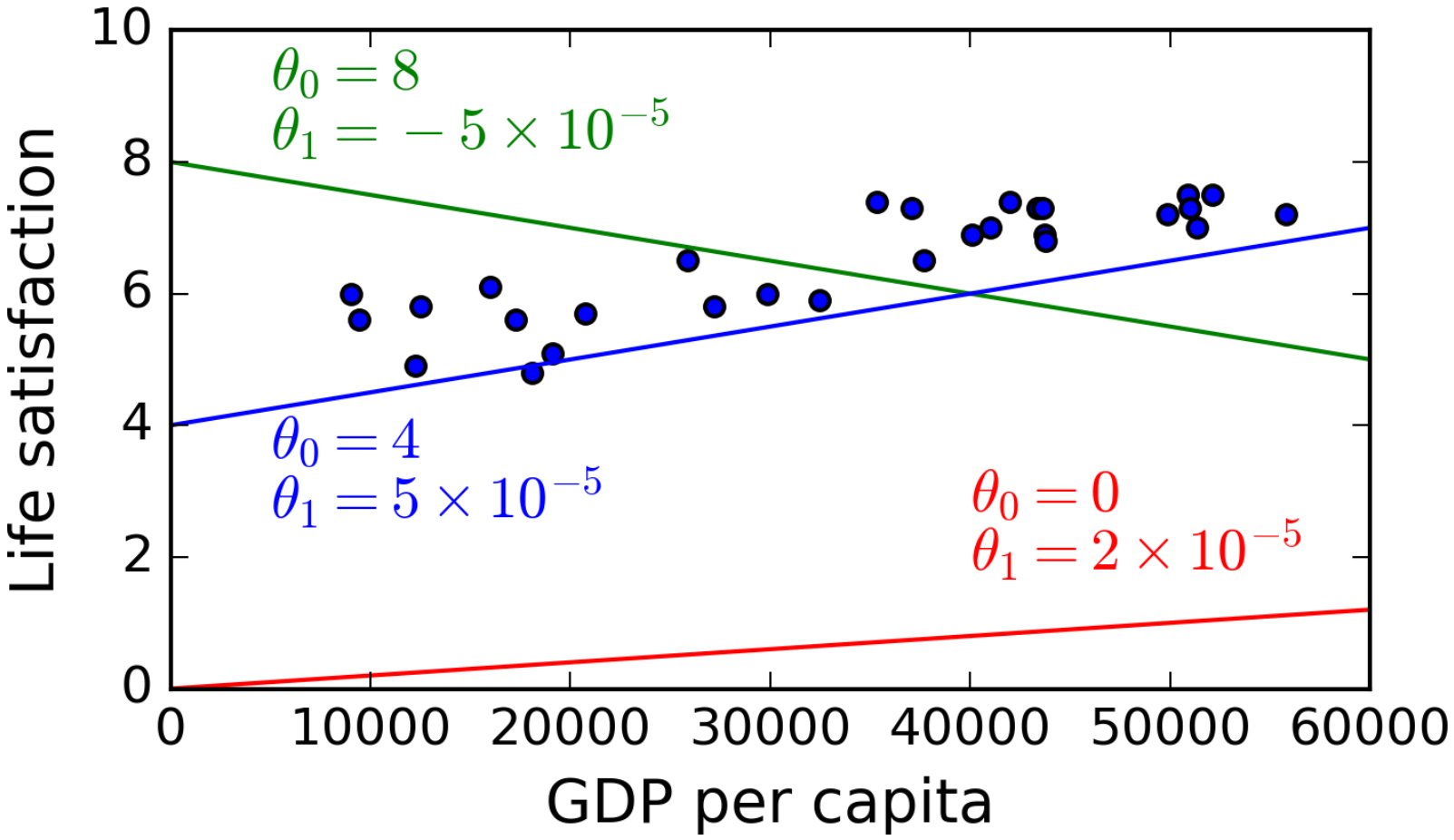


# Model-based learning

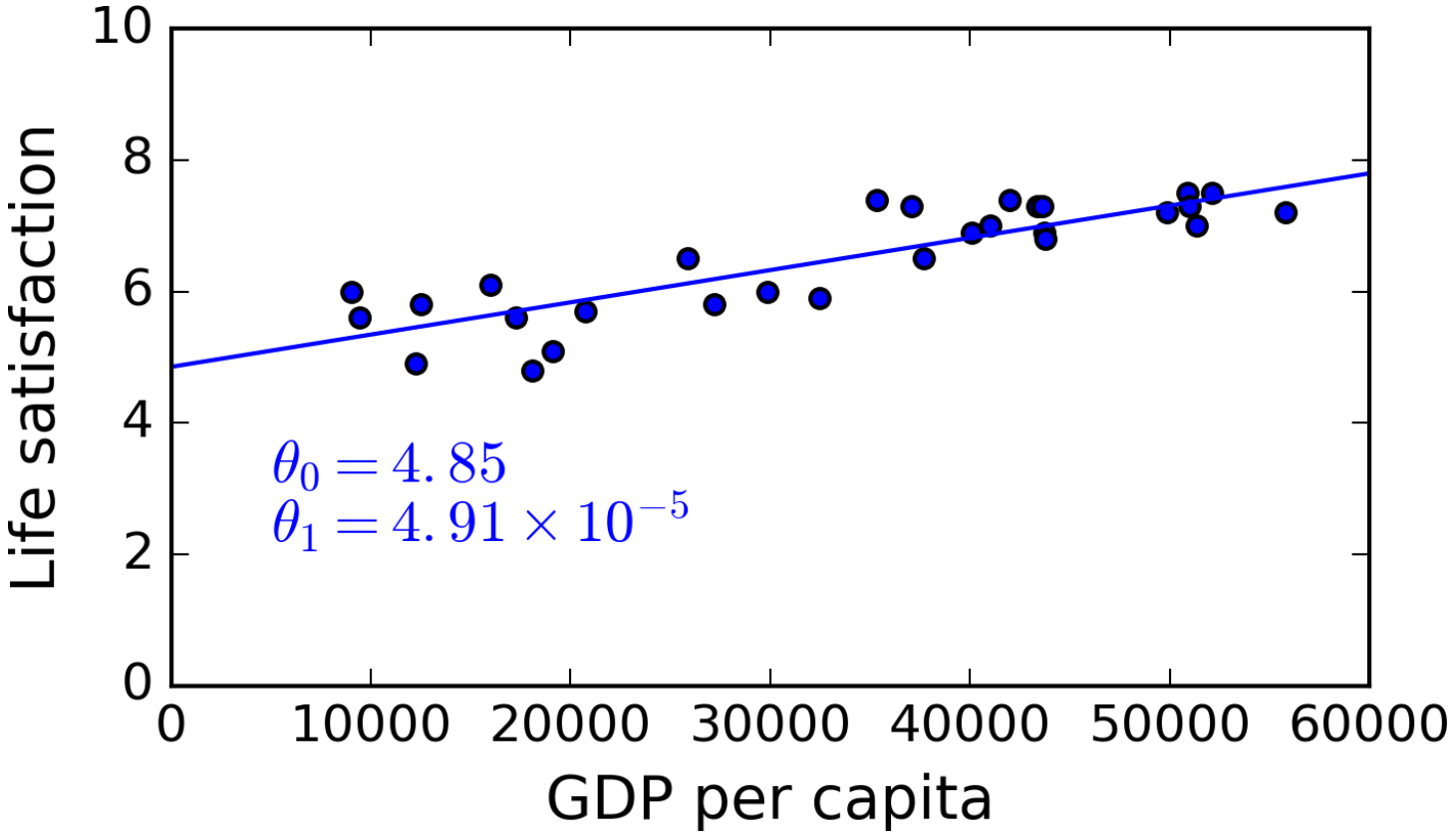
| Country | GDP per capita (USD) | Life satisfaction |
|---|---|---|
| Hungary | 12,240 | 4.9 |
| Korea | 27,195 | 5.8 |
| France | 37,675 | 6.5 |
| Australia | 50,962 | 7.3 |
| United States | 55,805 | 7.2 |



# Model-based learning



# Model-based learning




Cyprus's GDP per capita, find $22,587, and then apply your model and find that life satisfaction is likely to be somewhere around 5.96.

# Instance based learning

Slovenia has the closest GDP per capita to that of Cyprus ($20,732), and since Slovenians' life satisfaction is 5.7, you would have predicted a life satisfaction of 5.7 for Cyprus.

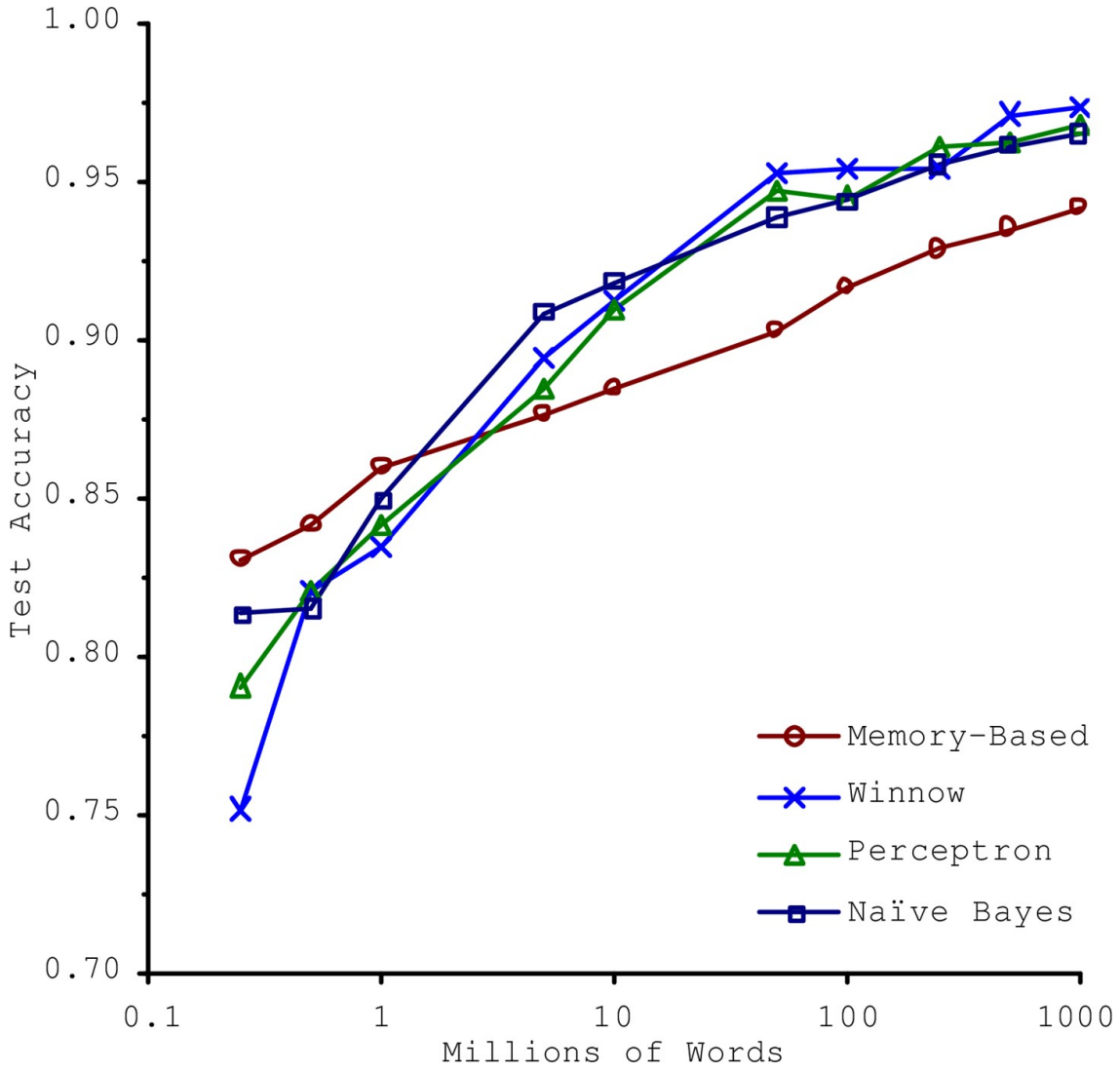
If you zoom out a bit and look at the two next closest countries, you will find Portugal and Spain with life satisfactions of 5.1 and 6.5, respectively.

Averaging these three values, you get 5.77, which is pretty close to your model-based prediction.

This simple algorithm is called k-Nearest Neighbors regression (in this example, k = 3).

# Desafios

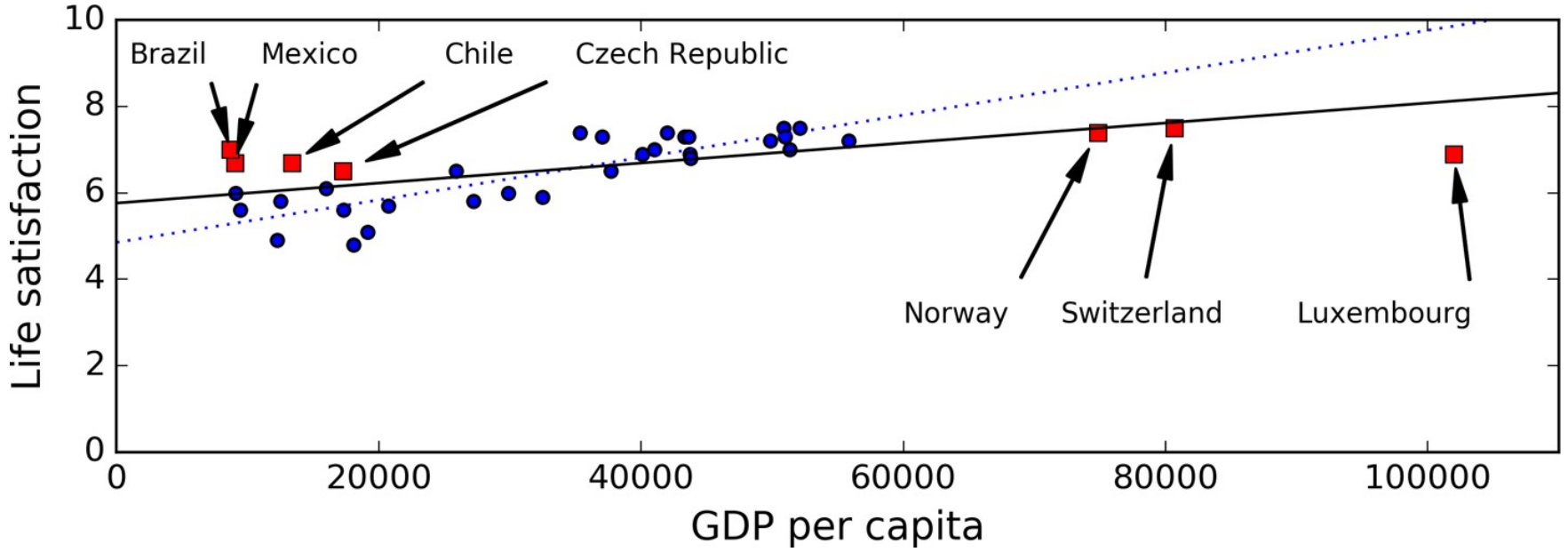
- Insufficient Quantity of Training Data



# Desafios

- Nonrepresentative Training Data

A more representative training sample

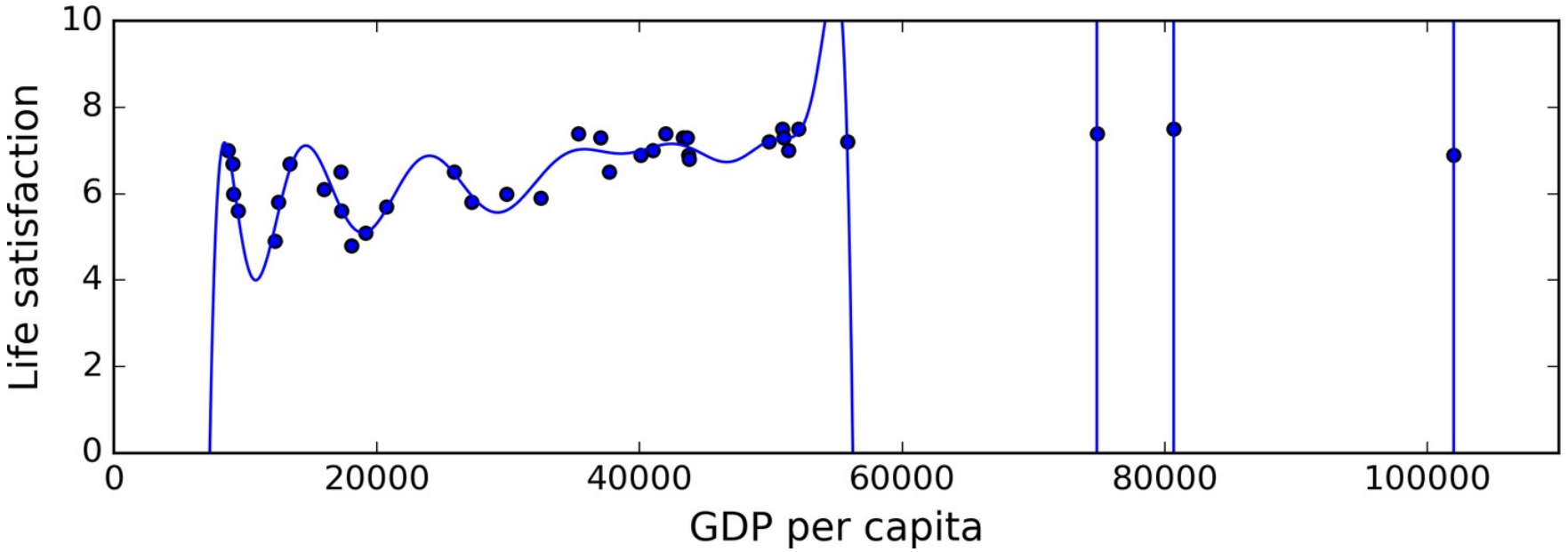


# Desafios

- Poor-Quality Data
- Irrelevant Features
  - Saying: garbage in, garbage out
  - Feature engineering:
    - Feature selection: selecting the most useful features to train on among existing features.
    - Feature extraction: combining existing features to produce a more useful one (dimensionality reduction algorithms can help).
    - Creating new features by gathering new data.

# Desafios

- Overfitting
  - the model performs well on the training data, but it does not generalize well.
  - Overfitting happens when the model is too complex relative to the amount and noisiness of the training data.

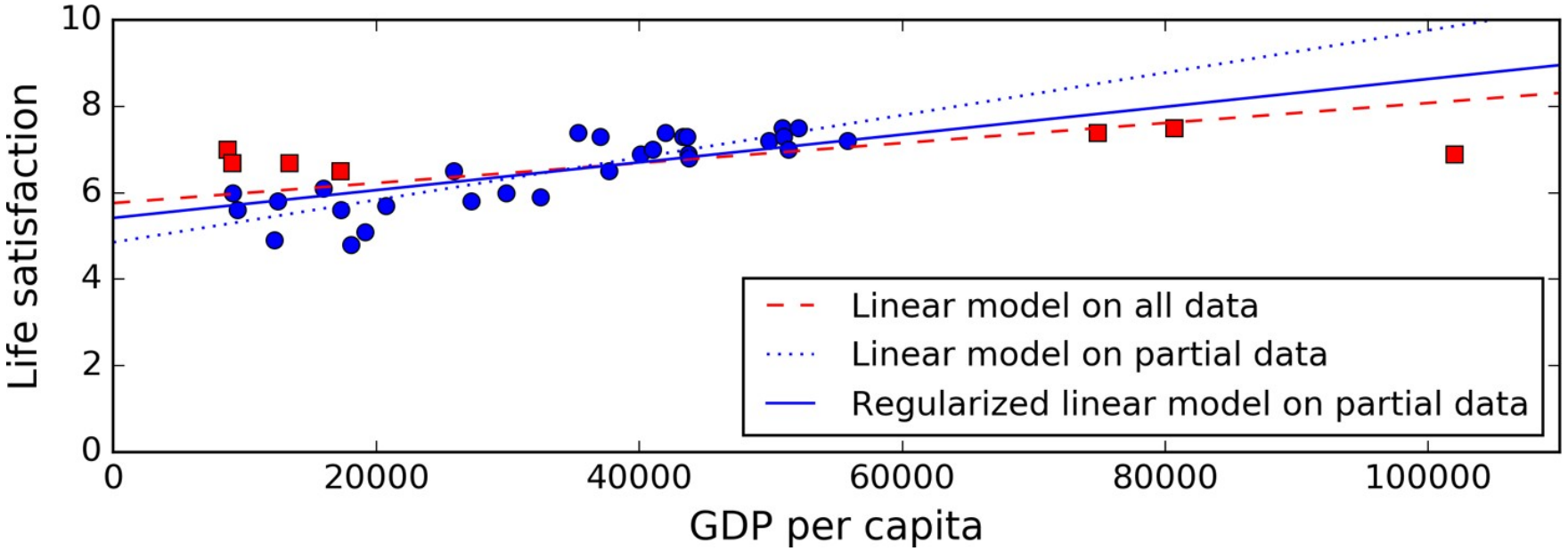


# Desafios

- Overfitting – Soluções
  - To simplify the model by selecting one with fewer parameters, by reducing the number of attributes in the training data or by constraining the model (e.g. regularization constraint);
  - To gather more training data;
  - To reduce the noise in the training data (e.g., fix data errors and remove outliers).

# Desafios

- Overfitting – Soluções
  - Using a regularization constraint hyperparameter



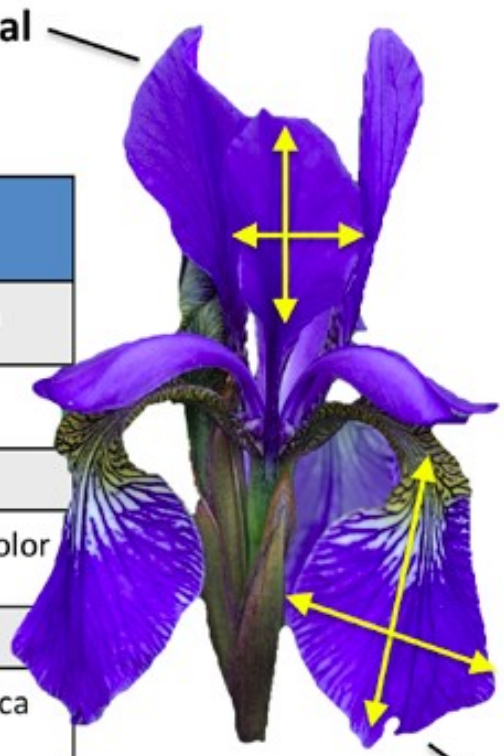
# Desafios

- Underfitting
  - it occurs when your model is too simple to learn the underlying structure of the data.
  - reality is just more complex than the model, so its predictions are bound to be inaccurate, even on the training examples.
  - Soluções:
    - Selecting a more powerful model, with more parameters;
    - Feeding better features to the learning algorithm (feature engineering);
    - Reducing the constraints on the model (e.g., reducing the regularization hyperparameter).

# Terminology and notations

**Samples**
(instances, observations)

| | Sepal length | Sepal width | Petal length | Petal width | Class label |
|---|---|---|---|---|---|
| 1 | 5.1 | 3.5 | 1.4 | 0.2 | Setosa |
| 2 | 4.9 | 3.0 | 1.4 | 0.2 | Setosa |
| | | | ... | | |
| 50 | 6.4 | 3.5 | 4.5 | 1.2 | Versicolor |
| | | | ... | | |
| 150 | 5.9 | 3.0 | 5.0 | 1.8 | Virginica |




**Features**
(attributes, measurements, dimensions)

**Class labels**
(targets)

# Iris dataset

150 samples and four features:

$$X = \begin{bmatrix} x_1^{(1)} & x_2^{(1)} & x_3^{(1)} & x_4^{(1)} \\ x_1^{(2)} & x_2^{(2)} & x_3^{(2)} & x_4^{(2)} \\ \vdots & \vdots & \vdots & \vdots \\ x_1^{(150)} & x_2^{(150)} & x_3^{(150)} & x_4^{(150)} \end{bmatrix}$$

Row vector:

$$\boldsymbol{x}^{(i)} = \begin{bmatrix} x_1^{(i)} & x_2^{(i)} & x_3^{(i)} & x_4^{(i)} \end{bmatrix}$$

Column vector:

$$\boldsymbol{x}_j = \begin{bmatrix} x_j^{(1)} \\ x_j^{(2)} \\ \vdots \\ x_j^{(150)} \end{bmatrix}$$

# Iris dataset

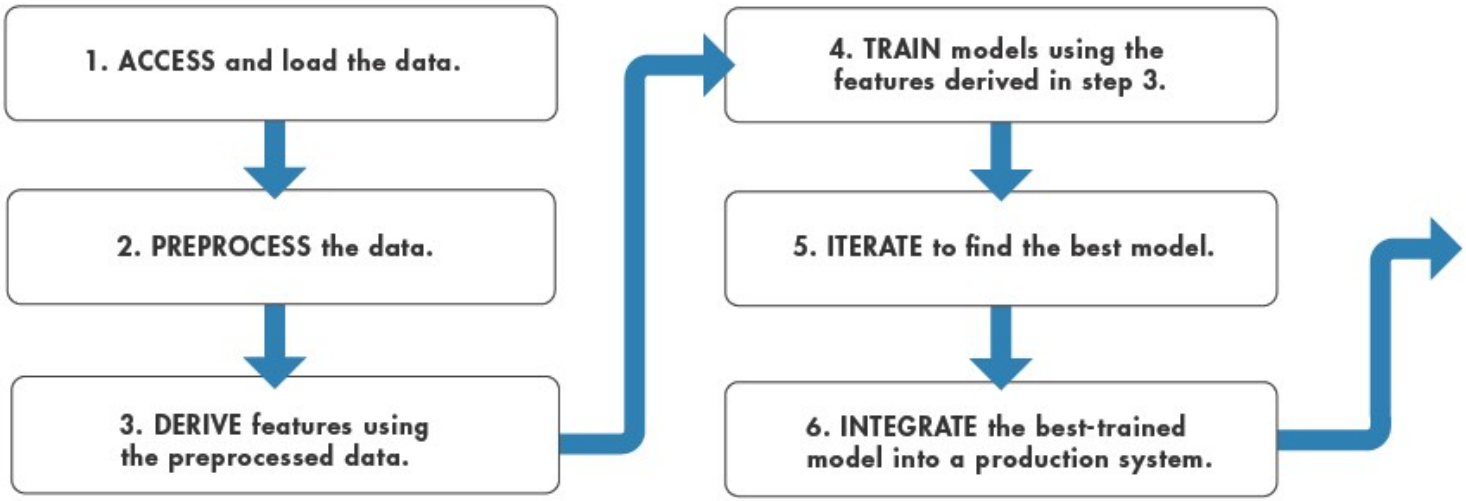
Target variables (class labels) as a 150-dimensional column vector:

$$\boldsymbol{y} = \begin{bmatrix} y^{(1)} \\ \dots \\ y^{(150)} \end{bmatrix} \left( y \in \{\text{Setosa}, \text{ Versicolor}, \text{ Virginica}\} \right)$$

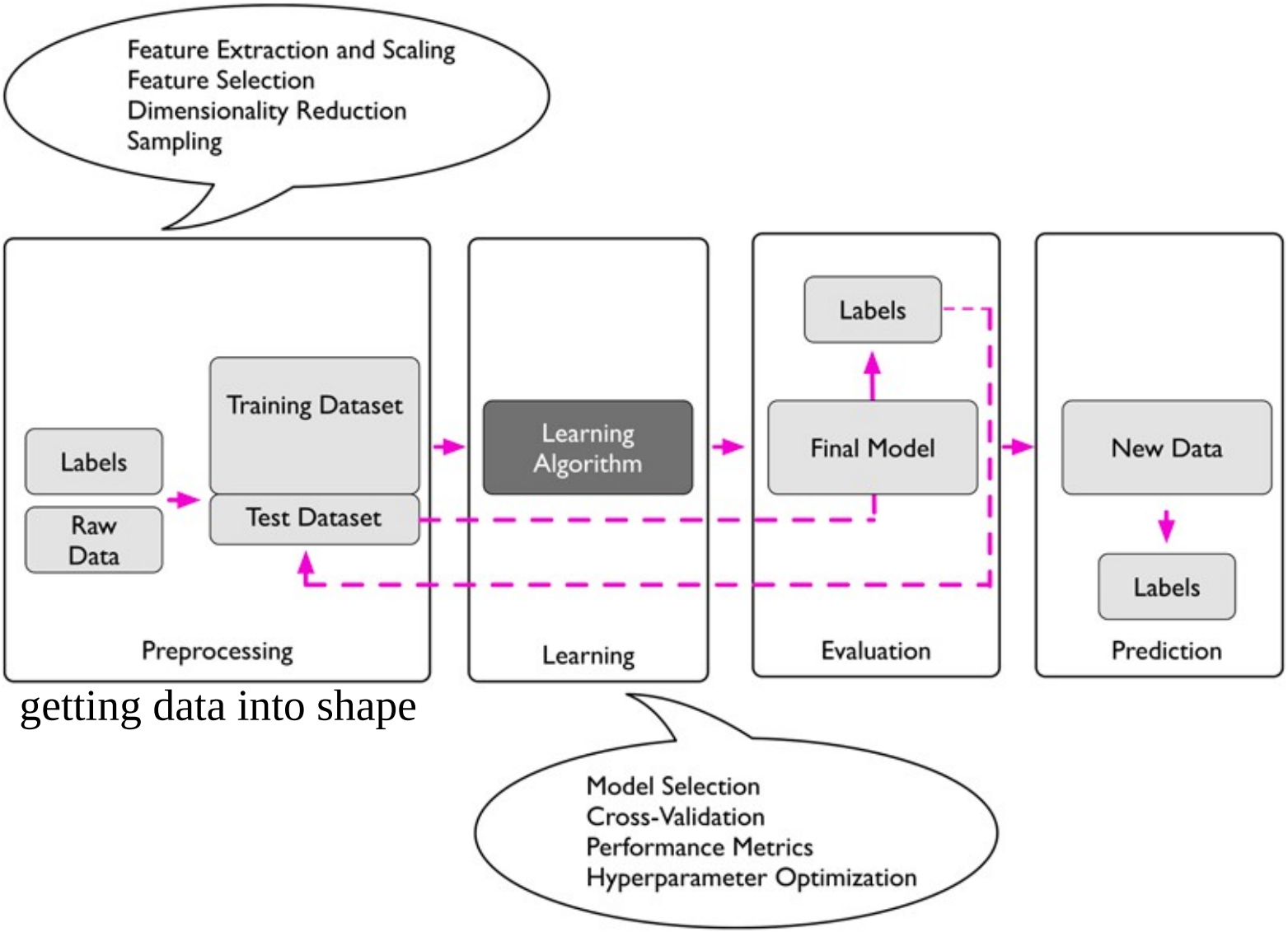
# Workflow 1

1. Definição do problema
2. Análise Exploratória dos Dados
3. Preparar os dados
4. Avaliar Algoritmos
5. Melhorar os resultados
6. Apresentar os resultados

# Workflow 2

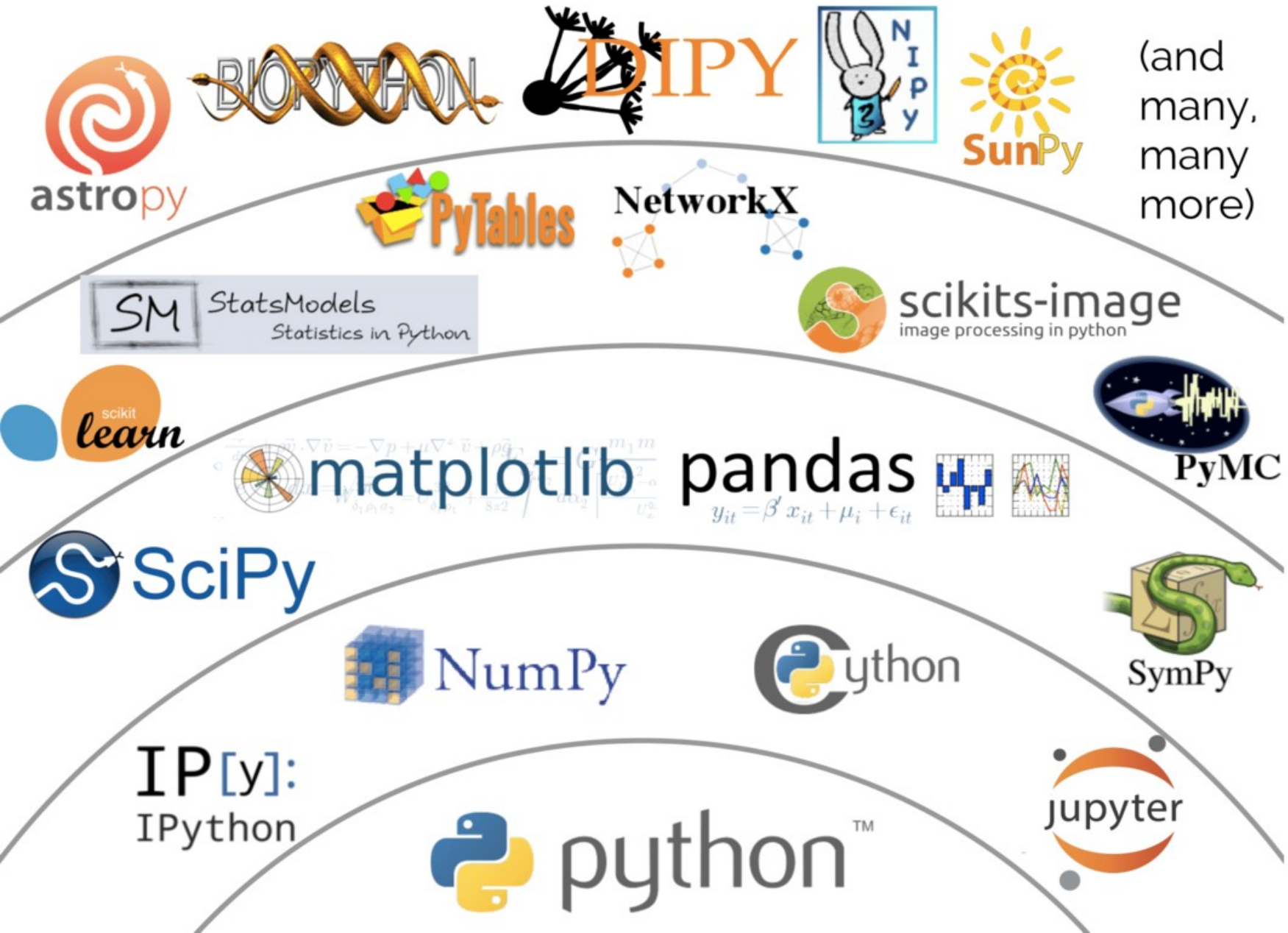

1. ACCESS and load the data.
2. PREPROCESS the data.
3. DERIVE features using the preprocessed data.
4. TRAIN models using the features derived in step 3.
5. ITERATE to find the best model.
6. INTEGRATE the best-trained model into a production system.

# Roadmap for building machine learning systems



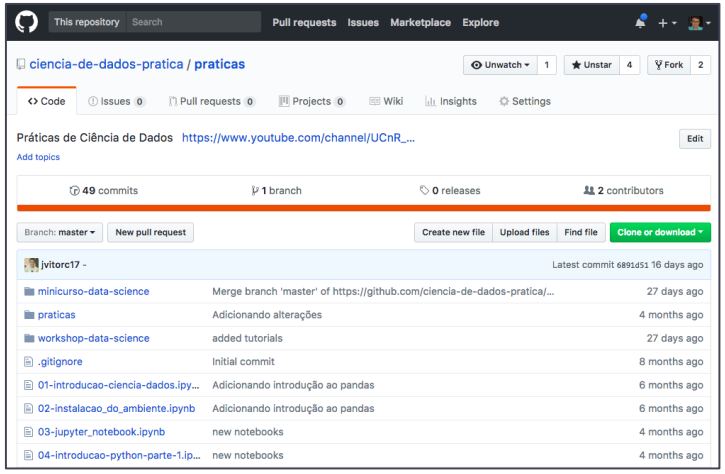

getting data into shape

# Ecossistema Python



# Projeto Ciência de Dados na Prática (cdp)
# Material didático gratuito

- GitHub
  - http://bit.ly/cdpgithub
- YouTube
  - http://bit.ly/cdpvideos
- Facebook
  - http://bit.ly/cdpface




# Obrigado!

Dúvidas, comentários, sugestões?

Regis Pires Magalhães
regismagalhaes@ufc.br
UFC