Course 2: Statistical Learning for Data Science

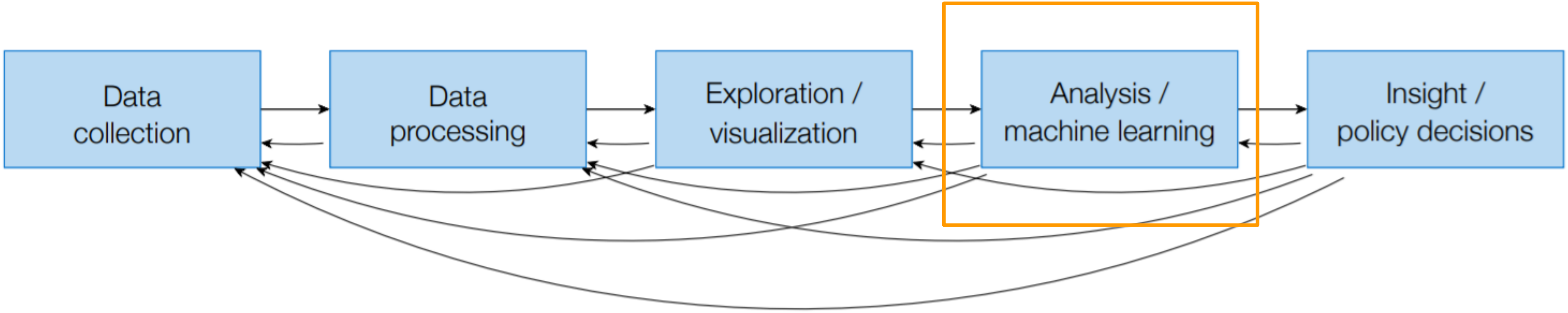
Instructor: Glen Berseth

Université de Montréal

DIRO

Mila

Data science



1. Why do we want machine learning?

Source: https://www.datasciencecourse.org/slides/intro.pdf

Outline

- What is Machine Learning
- Mains types of ML tasks in the context of DS:
    - Classification
    - Regression

    Supervised ML (e.g. requires input-output pairs (x_i,y_i))

    - Clustering
    - Dimensionality reduction

    Unsupervised ML (e.g. only requires inputs (x_i))
- Goal: Given data, understand which ML method to use

Want to read more about it:

- [ISLR book Chapter 4.1, 4.2 and 12.1]
- [Python data Science Book Chapter 5 "What is ML?"]

Disclaimer:
This course approaches statistical learning in a superficial way to focus on the data science part.

What is Machine Learning?

- Thought question.
- Many attempts to define it:

1. “Machine Learning at its most basic is the practice of using algorithms to parse data, learn from it, and then make a determination or prediction about something in the world.” – Nvidia
2. “Machine learning is the science of getting computers to act without being explicitly programmed.” – Stanford
3. “Machine learning is based on algorithms that can learn from data without relying on rules-based programming.”- McKinsey & Co.
4. “Machine learning algorithms can figure out how to perform important tasks by generalizing from examples.” – University of Washington
5. “The field of Machine Learning seeks to answer the question “How can we build computer systems that automatically improve with experience, and what are the fundamental laws that govern all learning processes?” – Carnegie Mellon University

What is Machine Learning?

- Thought question
- In this course: focus on ML in the context of DS.

We can think machine learning as a means of building models of Data.
[Python Data Science Book]

- ML: building models $\boldsymbol{y}_i \leftarrow f(\boldsymbol{x}_i|\theta)$ to help understand the data.
- Fit model parameters $\boldsymbol{\theta}$ on some collected data $\boldsymbol{D}$
- Can be used then on newly arriving data (for prediction or other things …)

Categories of Machine Learning

Supervised Learning

Categories of Machine Learning

Two big main categories:

- Supervised learning

  Our (rough) definition: train a model θ when we only have input-output pairs (x_i,y_i)
- Unsupervised learning

  Our (rough) definition: train a model θ when we only have the inputs (x_i)

- Many more versions (out of scope for this course):
  - RL,
  - semi-supervised learning,
  - self-supervised learning…

Sometime, unclear… Q: what does *supervision* mean?

Supervised Learning: Input - Output

Across applications we often want to model an output variable as a function of many inputs,

- Genetic profile → Chance of developing disease
- Person’s characteristics → Whether they’ll vote
- Marketing plan → Total sales amount
- Image pixel values → What’s in the image

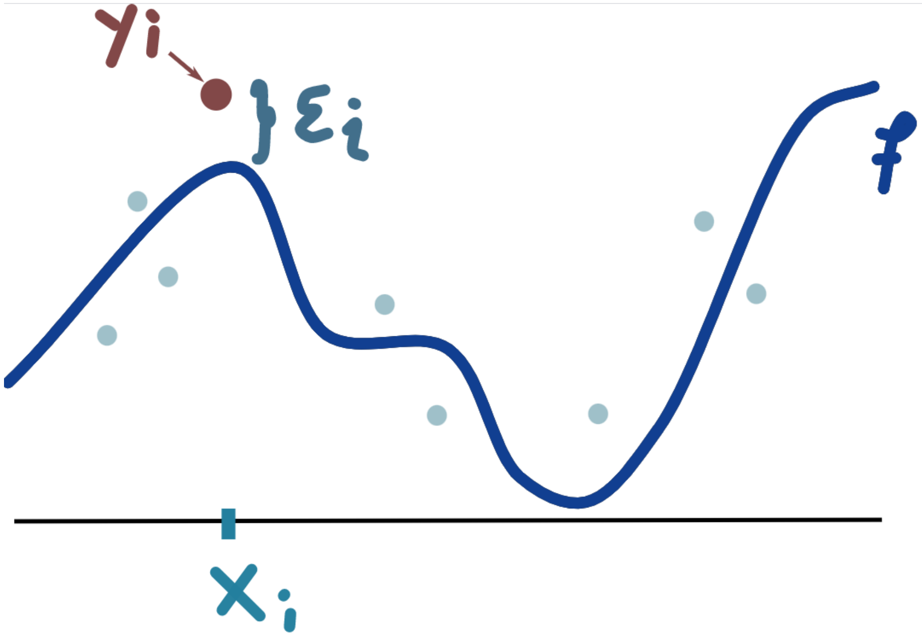
$$x_i = (x_{i1}, \ldots, x_{ip}) \leftarrow \text{all the inputs}$$

$$y_i = f(x_i) \leftarrow \text{inputs to output relationship}$$

Learning Input - Output mapping

- $\boldsymbol{y}_i \leftarrow f(\boldsymbol{x}_i|\boldsymbol{\theta})$ is a "simple" function that describe the relationship between x and **y**
- $\boldsymbol{\varepsilon}_\mathrm{i}$ reflects the variations whose source is unknown to us.

  (because not enough features or too complex function) Example: [coin toss]

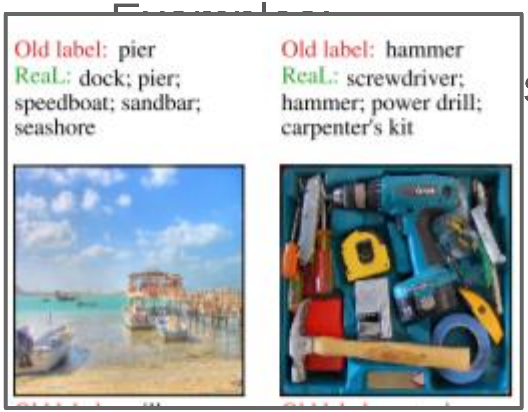


Supervised Learning: Input - Output

Goal: Given an input x, predict and output y

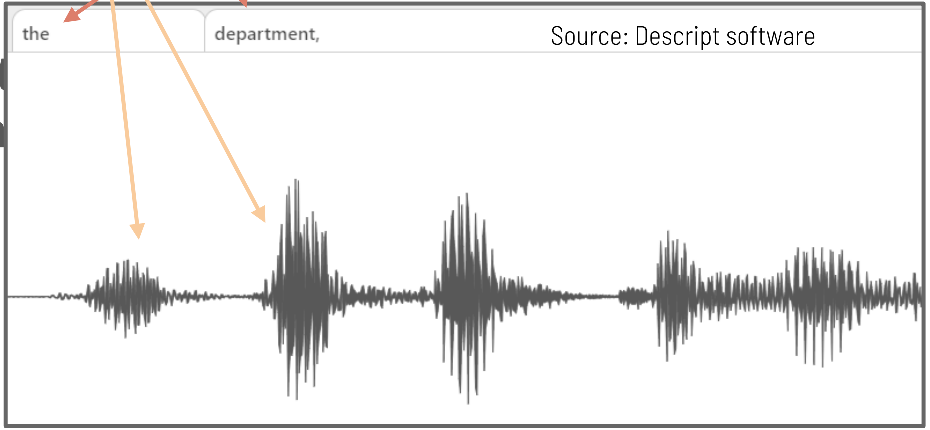
What we have: Observations: $(\boxed{x_i}, \boxed{y_i}),\ i = 1, \ldots, n.$

Examples:



sound, text

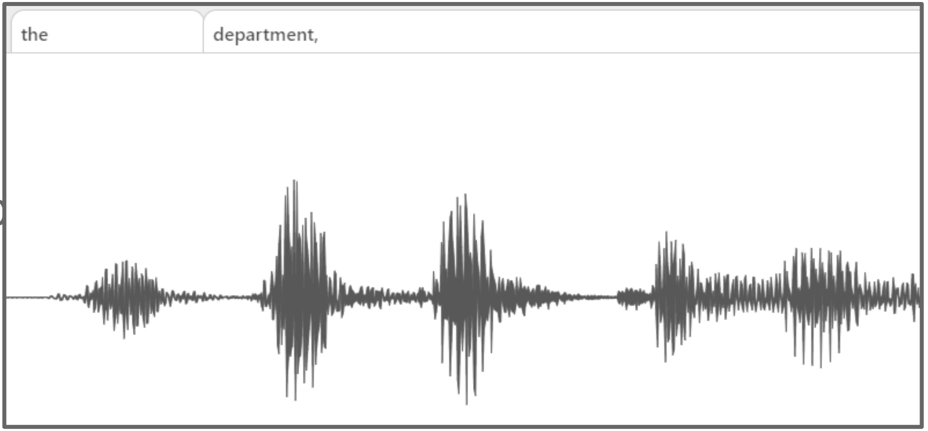
(prediction



Supervised Learning: Input - Output

k, predic

vations:



Why it is "hard":

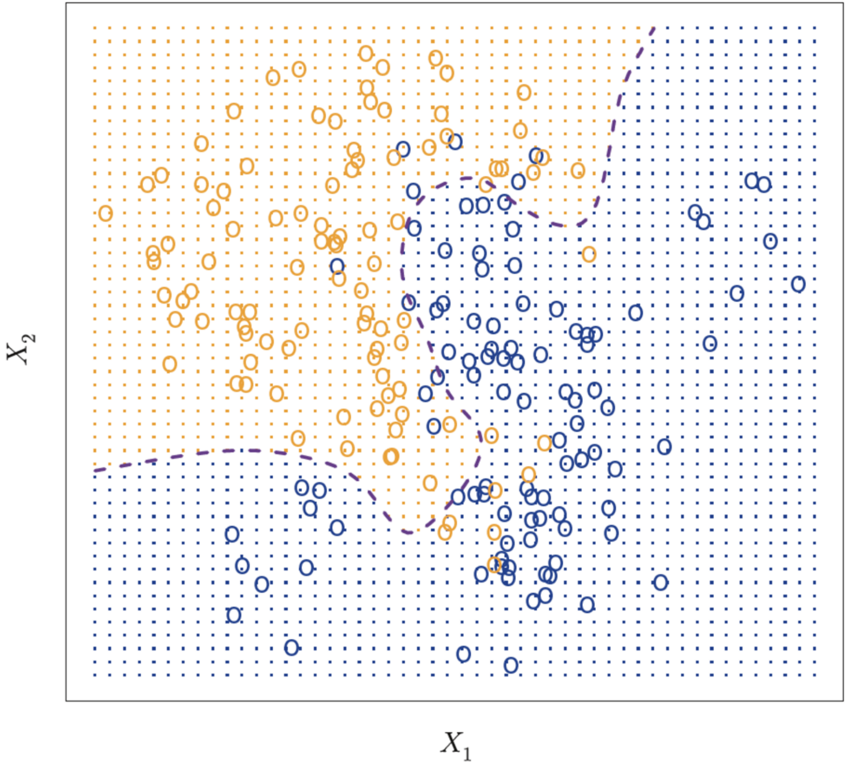
- Labels may be stochastic:
- The prediction may be "complex" (non-linear in high dimension)
- Only few pairs observed. (Bad labels)

Supervised Learning: Classification

- Prediction of **discrete** classes. (e.g. cats vs dogs)
- Example here:
  - Input: 2D position
  - Output (Labels) blue or yellow

- Goal: each region of the space should be blue or yellow.

- Evaluation: 0-1 loss on test set



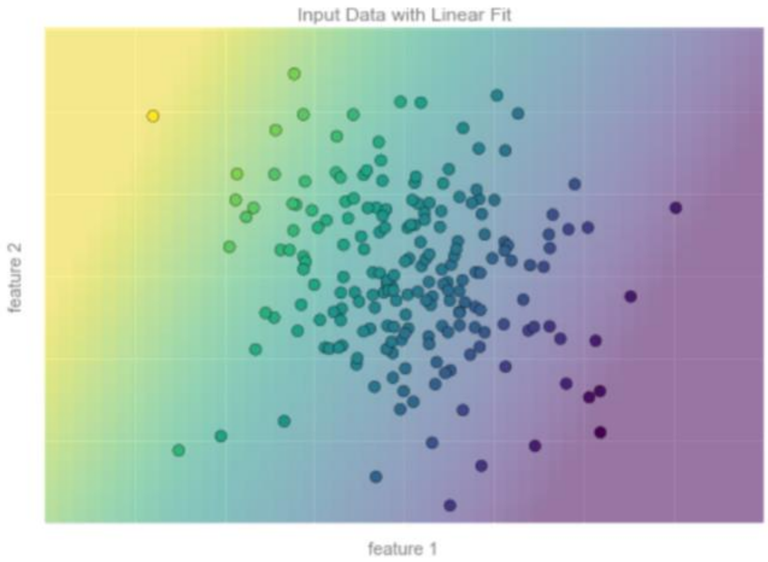
Supervised Learning: Regression

- Prediction of **continuous labels**. (e.g. temperature)
- Example here:
  - Input: 2D position
  - Output: value (represented by a color)

- Goal: each region of the space should have a value

Supervised Learning: Regression

- Prediction of **continuous labels**. (e.g. temperature)
- Example here:
  - Input: 2D position
  - Output: value (represented by a color)



- Goal: each region of the space should have a value
- Evaluation: loss of accuracy on a test set.

Supervised Learning: Regression vs. classification

More connected than it appears.

Let us consider a classification setting with 2 classes.

- Hard to learn from 0-1 loss (actually NP hard) !
- Instead one often will try to learn the function f:

$$f(x) = \mathbb{P}(y = 1|x) \in [0, 1]$$ (prob of being labeled as y given the input x)

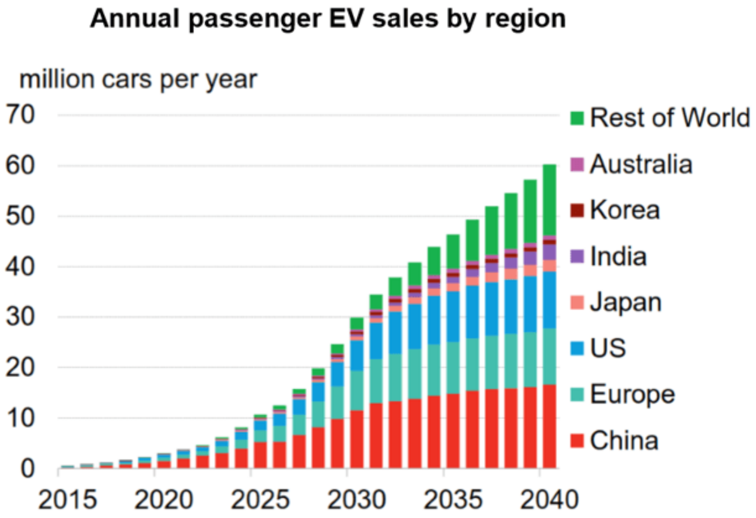
- Now we have a regression problem!!!
- Q: There is still some differences with a regression task: can you see which ones?

Regression: applications

Electric vehicle sales prediction

- **Purpose: to predict sales in each country based on**

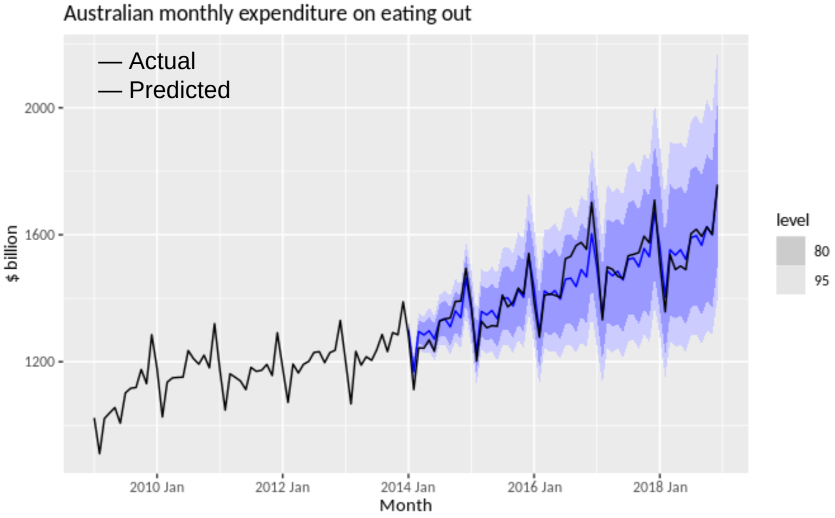
- **Economic factors**
- **Social factors**
- **Geopolitical context**
- **Etc.**



Reference... "The Economic Crash Will Slow Down The Electric Vehicle Revolution... But Not For Long!" - City Vitae

Consumer Spending Prediction

Purpose: To predict monthly restaurant expenses based on revenue from various types of restaurants.



13.4 Forecast combinations | Forecasting: Principles and Practice (3rd ed) (otexts.com)

Home insurance prediction

Goal: Predict the insurance premiums to be paid according to

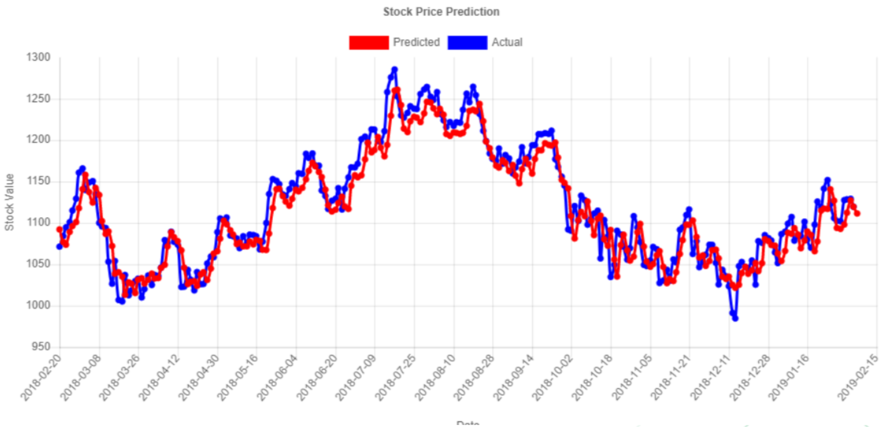
- **Characteristics of a house**
- **Land value**
- **Location**
- **Flood history**
- **etc**

The (Mostly) Definitive Guide to Home Insurance (realtor.com)

Stock market predictions (finance)

Goal: to predict the value of a company's shares based on

- **Micro and macro-economic factors**
- **Geopolitical context**
- **Social phenomena**
- **Competitor Strategies**
- **etc**

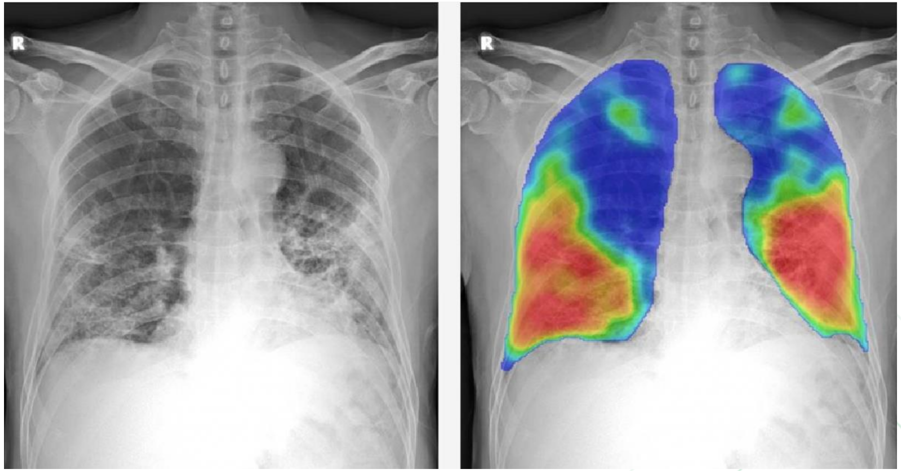


Stock Price Prediction System using 1D CNN with TensorFlow.js-Machine Learning Easy and Fun | by Gavril Ognjanovski | Towards Data Science

Help with medical diagnosis

Aim: To predict the probability of the presence of Covid-19 in the lungs of patients based on the following factors:

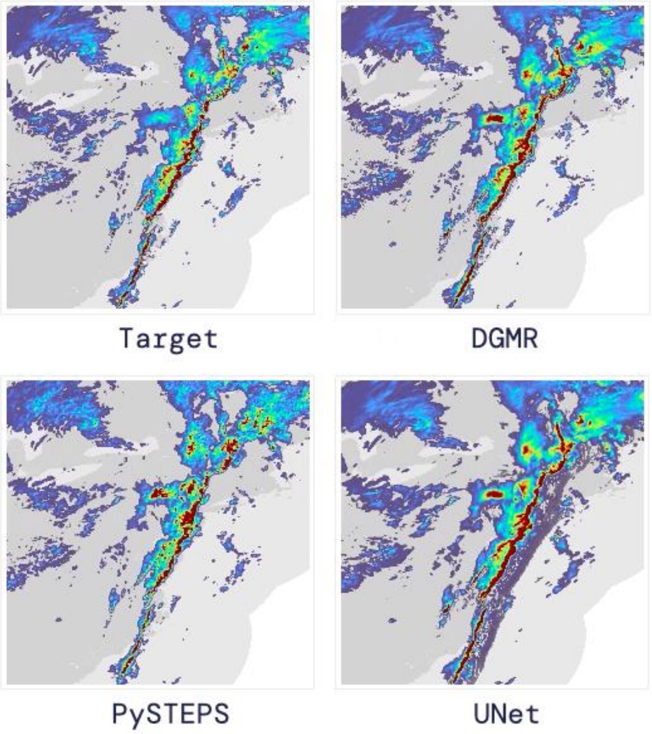
X-rays
Laboratory tests
Medical file
Way of life



COVID-19 catch-up: New AI tool detects affected lung tissue by analyzing X-Ray images | Silicon Channels

Weather prediction

Source: https://www.deepmind.com/blog/nowcasting-the-next-hour-of-rain



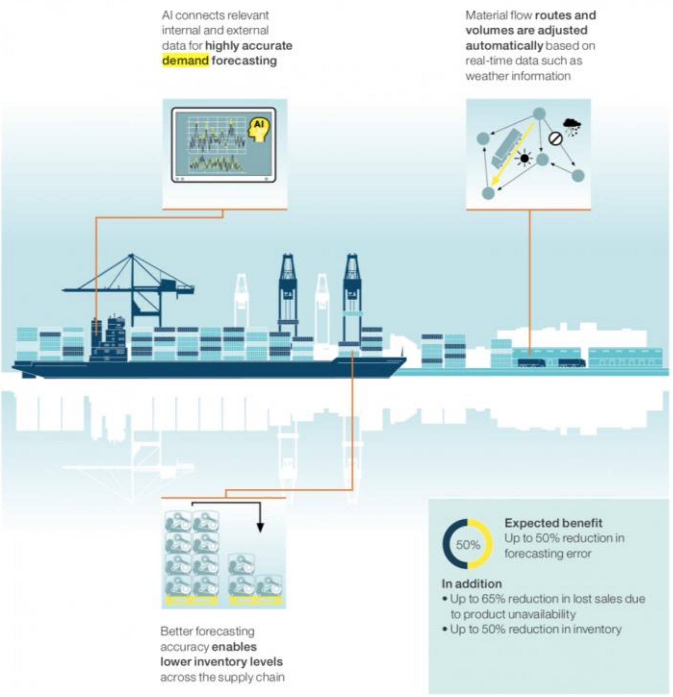
Goals: Predict global rainfall from

- **Satellite images**
- **Buoy surveys**
- **Historical observations**
- **El Niño and La Niña years**
- **etc**

Supply and demand predictions

Goals: To foresee the following aspects for an international trading company

- **Inventory evolution**
- **Evolution of demand**

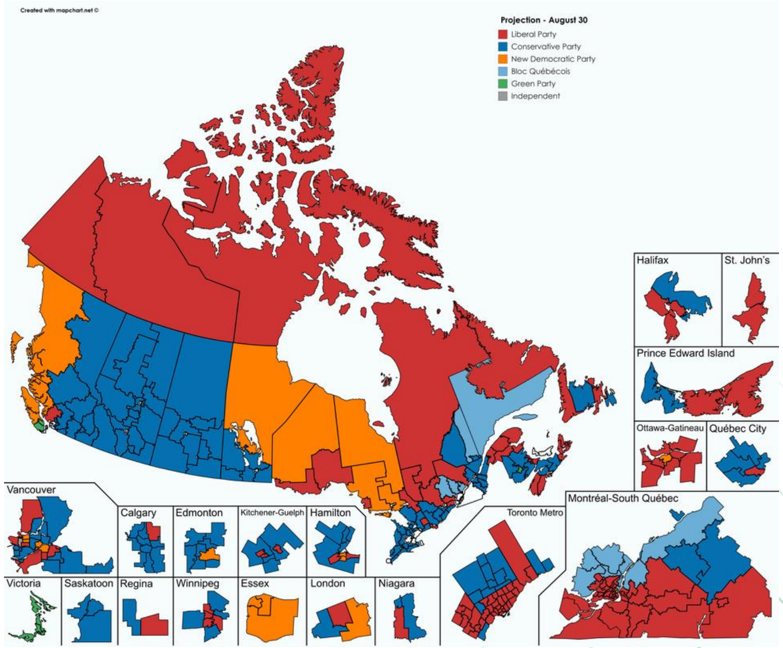


True Stock | Top 5 Benefits of using Machine Learning for Demand Forecasting

Election projections

Goals: Predict the percentage of votes for each party based on the following factors:

- **Historical results**
- **Micro and macro-economic factors**
- **Geopolitical context**
- **Social networks**
- **etc**

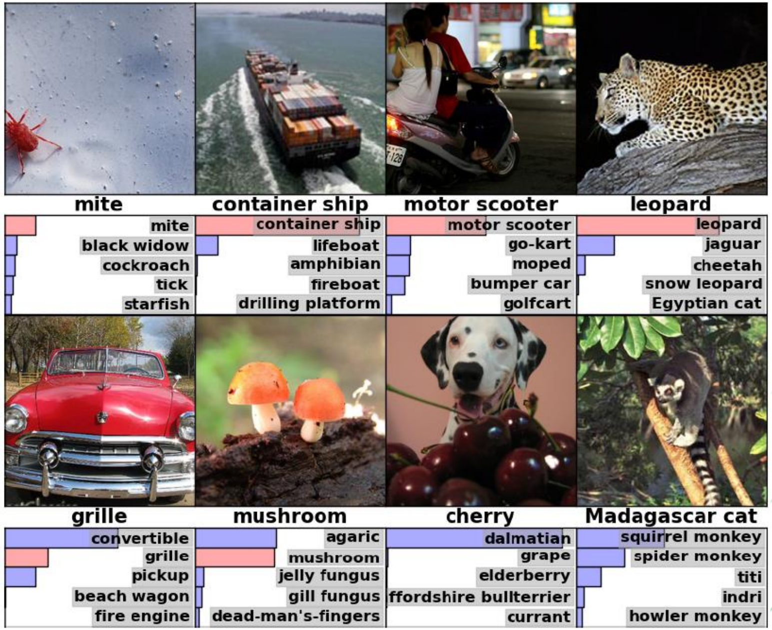


2019 Canadian federal election - screening as of August 30, 2019 : MapPorn (reddit.com)

Classification: Applications

Object recognition

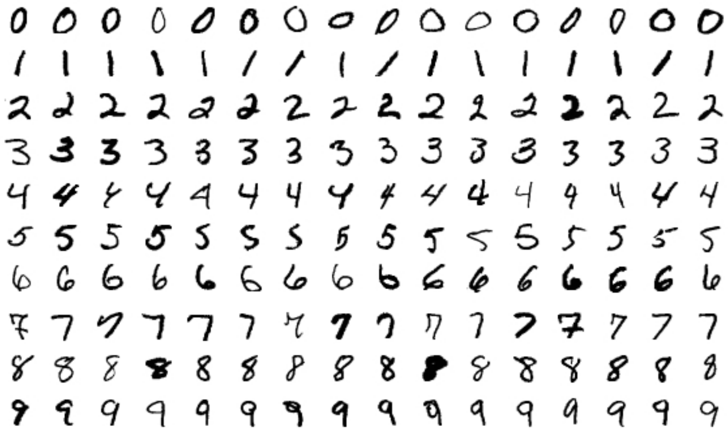
Aim: to recognize the types of objects present in an image despite the variations of objects and poses.



A. Krizhevsky, I. Sutskever and G.E. Hinton. Imagenet Classification with Deep Convolutional Neural Networks. In Advances in Neural Information Processing Systems, 2012.

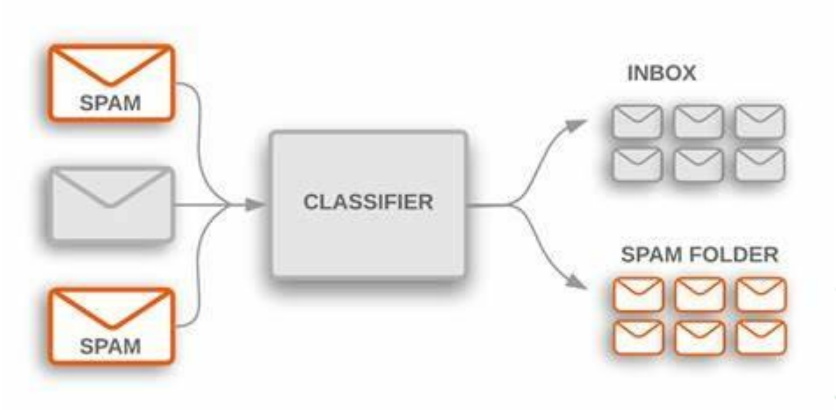
Recognition of handwritten characters

Aim: to recognize each character despite different writing styles.



Natural language processing

Purpose: to recognize combinations and arrangements of words to filter SPAM/SPAM.

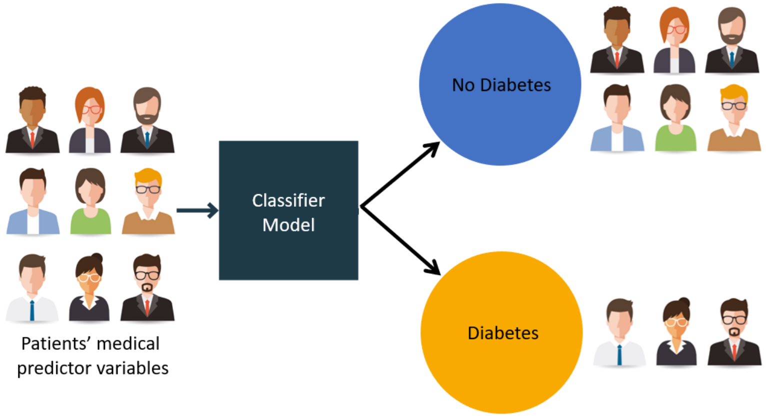


Binary Trees | CSE 143 (washington.edu)

Help with medical diagnosis

Purpose: to identify people with diabetes from

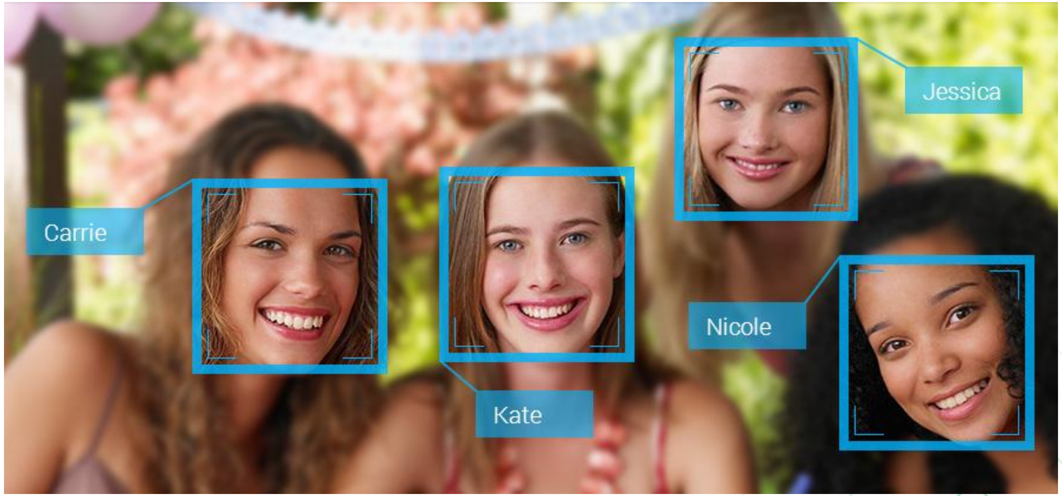
Laboratory tests
Medical file
Way of life



Classification Algorithm and its types in Machine Learning | by Iprathore | Medium

Facial recognition (Biometrics)

Goals: identify
Last name
Gender
Age range
Health
Etc.



Face Recognition for Beginners. Face Recognition is a recognition… | by Divyansh Dwivedi | Towards Data Science

Speech Recognition

Goals: identify a speaker from

Time dependence of information.

Dictionaries of valid words/structures.

Siri richtig nutzen: nützliche Tipps & Befehle auf einen Blick (sparhandy.de)

Autonomous car

Purpose:
to identify the pixels corresponding to the

- **pedestrians**
- **buildings**
- **trees**
- **types of terrain**

despite the movements of the objects and the car!

We are interested in areas of the image containing objects of the same type (semantics).

Beanie-inception: Prediction of Cityscapes Test Sequences 00, 01, and 02. 768x384px - YouTube

Very important to have robust systems

- Robust to different situations.
- Especially the ones that have never been met.
- Humans are very good at that.

Unsupervised learning

Clustering: Inferring labels on unlabeled data

What can we do when we do not have the labels y???

Hypothesis: point x with the same label should me **similar**

Clustering: Model that aims at regrouping "similar" input data points.

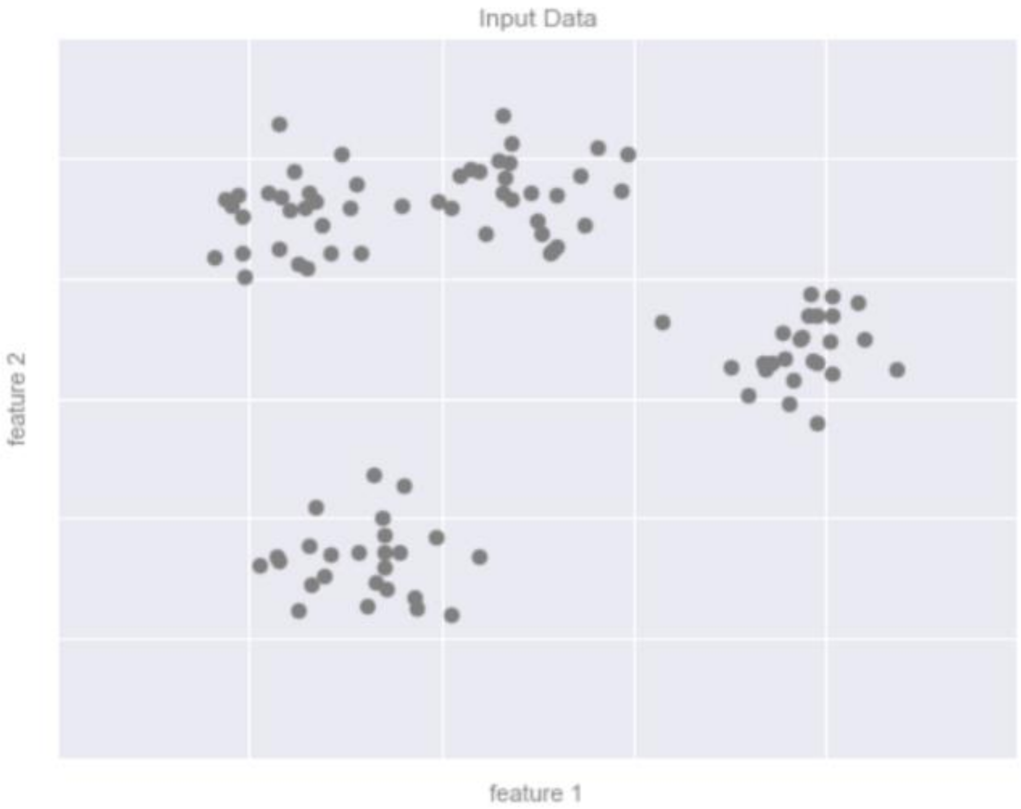
A lot of the complexity boils down to how we will define "similar"

- Example: data points close with respect to a certain distance (which distance???)

Importance of the right features: will better capture similarity.

Clustering

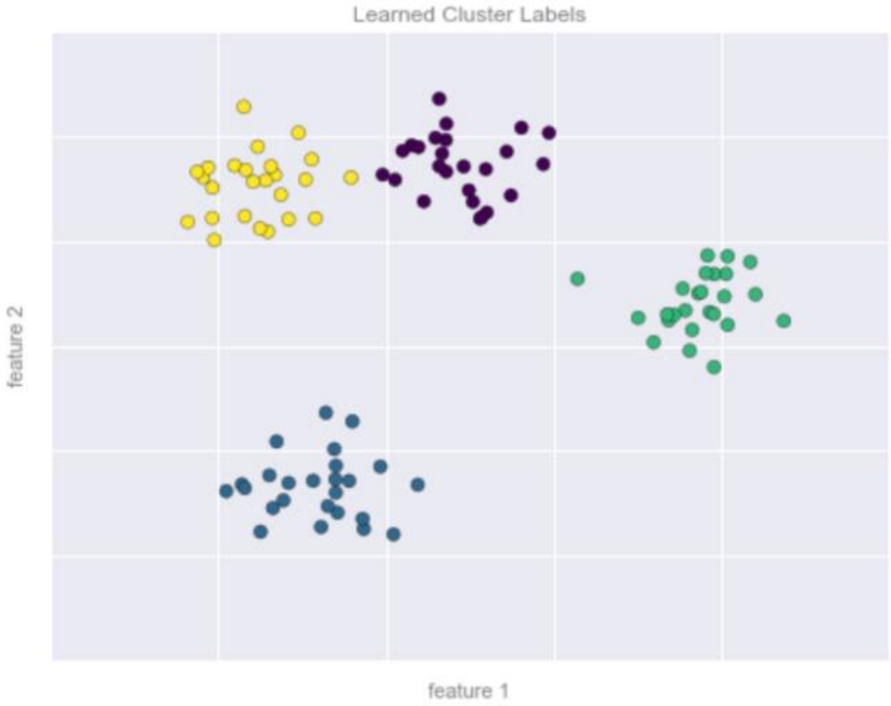
- By eye it is clear that there is 4 clusters!
- Importance of data visualisation!!!!
  (now we know that we need to look for 4 clusters)



Clustering

- By eye it is clear that there is 4 clusters!
- Importance of data visualisation!!!!
  (now we know that we need to look for 4 clusters)
- After running k-means ->
- $c \leftarrow f(\boldsymbol{x}_i|\theta)$
  - C is discrete 0 … k

  (will be covered on week 7)



Dimensionality reduction: Inferring structure of unlabeled data

Problem: data is high dimensional ! (e.g. images, text)

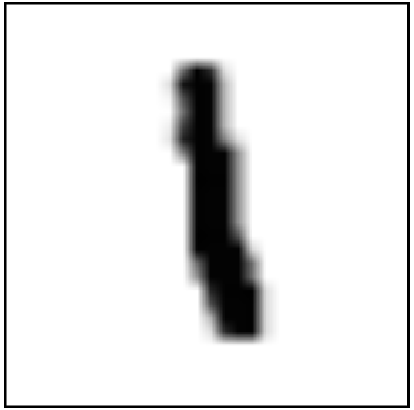
Hypothesis: There exists a low dimensional *latent* $\boldsymbol{z}$ representation of these data

In other words: the data lies on a low dimensional manifold (known as the manifold hypothesis).

- function: $\boldsymbol{z} \leftarrow f(\boldsymbol{x}_i|\theta)$, $\boldsymbol{z}$ is continuous (usually)

Dimensionality reduction

- Why this hypothesis?
- Uniformly Random generated data does not look like data.
- Example: MNIST:



$$\simeq \begin{bmatrix}
0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&.6&.8&0&0&0&0&0&0\\
0&0&0&0&0&0&.7&1&0&0&0&0&0&0\\
0&0&0&0&0&0&.7&1&0&0&0&0&0&0\\
0&0&0&0&0&0&.5&1&.4&0&0&0&0&0\\
0&0&0&0&0&0&0&1&.4&0&0&0&0&0\\
0&0&0&0&0&0&0&1&.4&0&0&0&0&0\\
0&0&0&0&0&0&0&1&.7&0&0&0&0&0\\
0&0&0&0&0&0&0&1&1&0&0&0&0&0\\
0&0&0&0&0&0&0&.9&1&.1&0&0&0&0\\
0&0&0&0&0&0&0&.3&1&.1&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0
\end{bmatrix}$$

Dimensionality reduction

- Why this hypothesis?
- Uniformly Random generated data does not look like data.
- Example: MNIST:

Picking pixel values uniformly btw 0 and 1:

Illustration of the manifold hypothesis

The data approximately lies on a 1D manifold:

Goal: learn the system of coordinate of the manifold

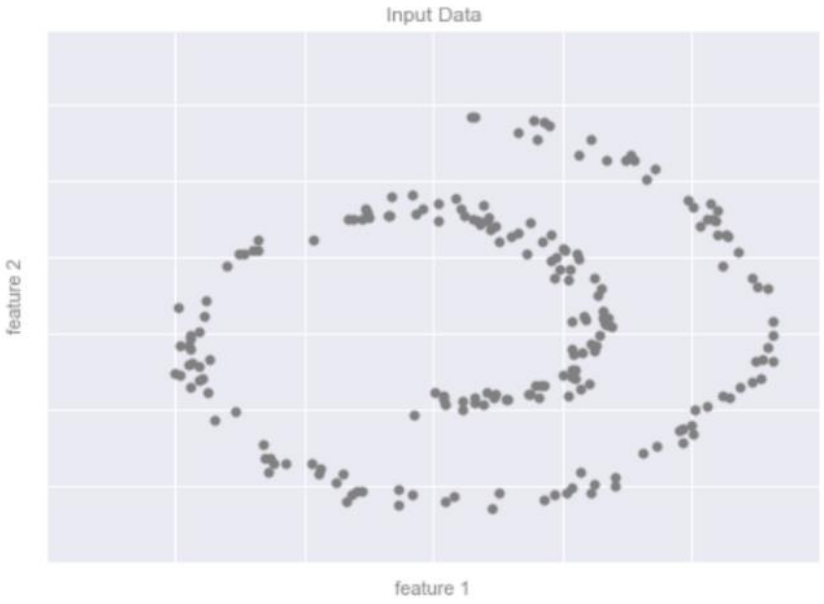
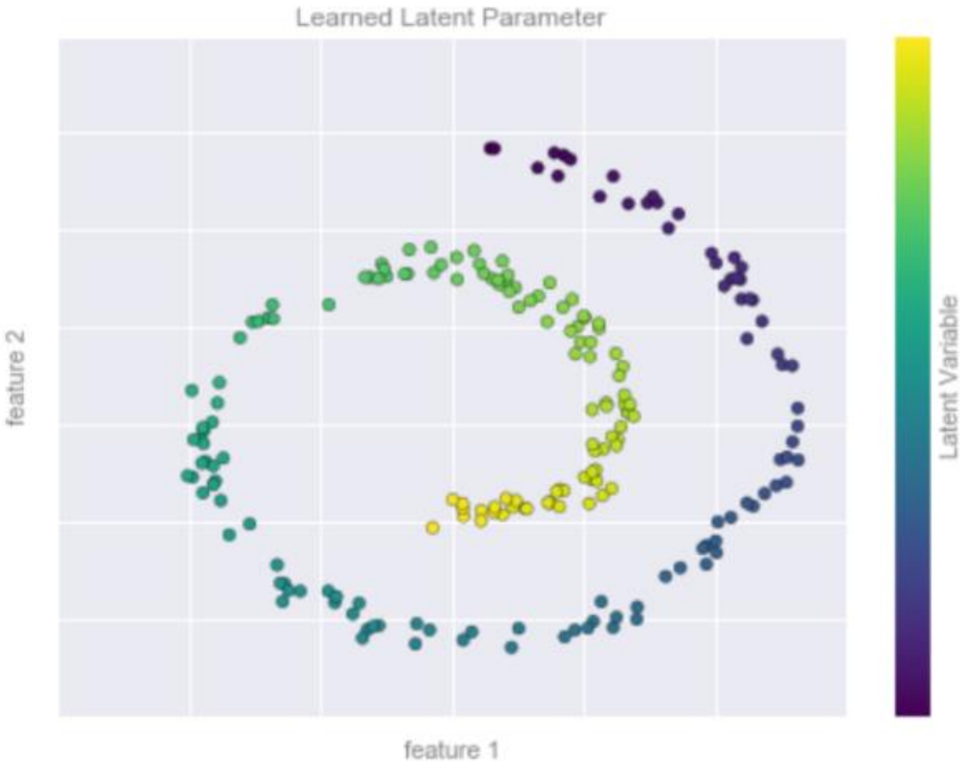


Illustration of the manifold hypothesis

The data approx

<u>Goal:</u> learn the s



Dimensionality reduction: conclusion

- Maybe the less clear setting
- High level idea: find better representations
- Example: PCA (covered in week 7)
- Useful to eventually solve other tasks (e.g. classification with only few labels)
    - ML using few features

Conclusion (part 1)

The ML tasks you might encounter in DS depend on your data (whether or not you have access to labels)

Two main paradigms:

- Supervised learning (with labels)
- Unsupervised learning (without labels)

Part 2: some concrete supervised learning algorithms.

- Watch IFT 6390 if you want to learn more about the theory of supervised learning algorithms.
- How to use these algorithms in practice (with sk-learn) in future labs.

Supervised learning algorithms

Logistics Regression

Logistics Regression

Logistic Regression: Estimating $P(C_1|\mathbf{x})$

$$y = \hat{P}(C_1|\mathbf{x}) = \frac{1}{1 + \exp[-(\mathbf{w}^\top \mathbf{x} + w_0)]}$$

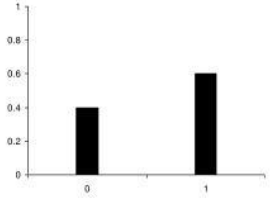
Learn w and w0 from $\mathcal{X} = \{\mathbf{x}^t, r^t\}$, with $r^t \in \{0, 1\}$.

Logistics Regression

Discrete distributions

<u>The Bernoulli distribution</u>

$$p(x) = P[X = x] = \begin{cases} q = 1 - p & x = 0 \\ p & x = 1 \end{cases}$$



r^t for some x^t follows a Bernoulli distribution with probability

$$y^t = P(C_1 | \mathbf{x}^t)$$

Sampling likelihood of $\mathcal{X} = \{\mathbf{x}^t, r^t\}$ according to w and w0:

$$l(\mathbf{w}, w_0 | \mathcal{X}) = \prod_t (y^t)^{(r^t)} (1 - y^t)^{(1 - r^t)}$$

Error to maximize log-likelihood

$$E_{entr}(\mathbf{w}, w_0 | \mathcal{X}) = -\log l(\mathbf{w}, w_0 | \mathcal{X}) = -\sum_t r^t \log y^t + (1 - r^t) \log(1 - y^t)$$

Also called cross-entropy.

Learning algorithm

Randomly initialize the weights,wj ~U(-0.01,\,0.01).

Repeat until convergence:

Prediction:

$$y^t = \frac{1}{1 + \exp[-(\mathbf{w}^\top \mathbf{x}^t + w_0)]}, \qquad t = 1, \ldots, N$$

No gradient:

$$\begin{cases} w_j = w_j + \eta \sum_t (r^t - y^t) x_j^t, \qquad j = 1, \ldots, D \\ w_0 = w_0 + \eta \sum_t (r^t - y^t) \end{cases}$$

Interpretation

- Prediction: $y = \hat{P}(C_1|\mathbf{x}) = \dfrac{1}{1 + \exp[-(\mathbf{w}^\top \mathbf{x} + w_0)]}$

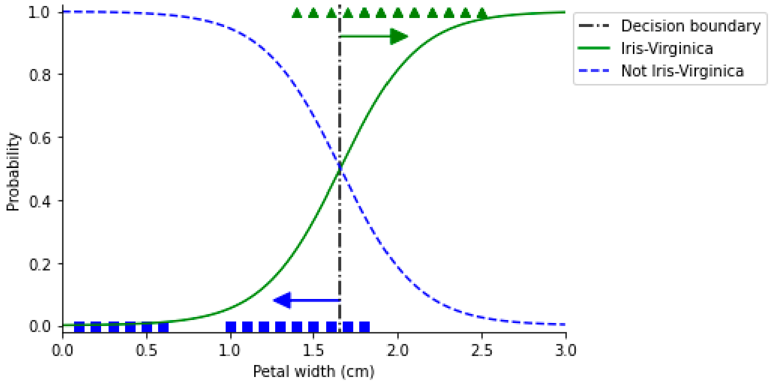
- Interpretation of wi coefficients:
  - the larger w_i, the more important x_i is for the prediction.
  - If w_i is zero x_i is not used.

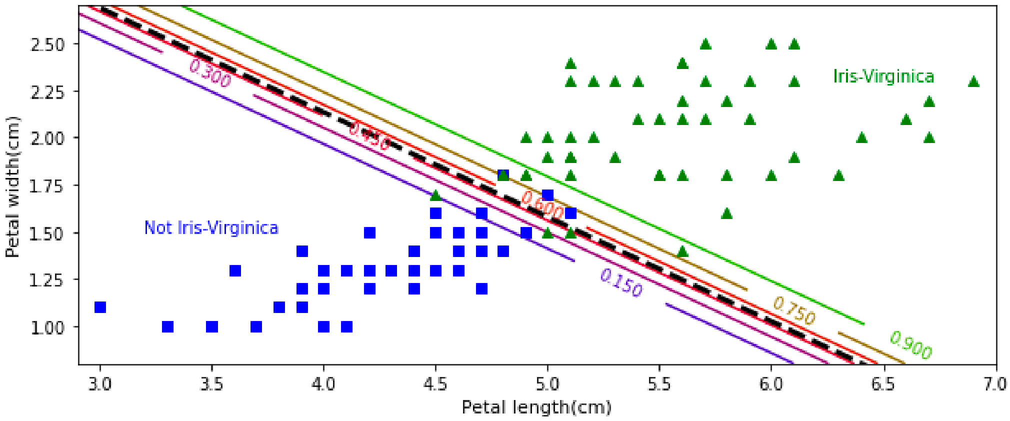
Example on the IRIS dataset

4 features: the length and width (in centimeters) of the sepals and petals of the flowers.

Proposed by Fisher in 1936.

Two of the three species were collected in Gaspésie (Quebec, Canada).

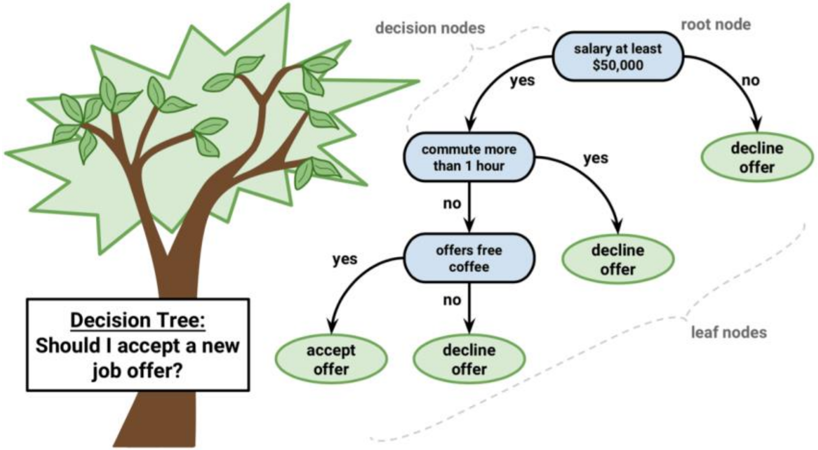




decision trees

Decision tree

- Performs a hierarchical splitting of the input space
- Each node corresponds to a test on a characteristic (input variable)
- The leaves of the tree correspond to the predictions



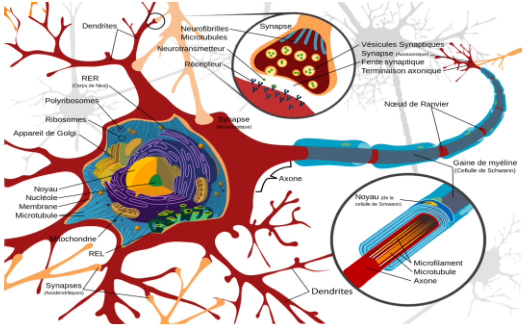
Characteristics of trees

- They are built from the top down
- They tend to over-specialize and over-learn
    - Pruning often required after construction to simplify the structure
- They make it possible to obtain interpretable predictions
- Classifiers with low bias and high variance $\longrightarrow$ Instability
- (concept of bias and variance will be seen in more detail in future courses.)

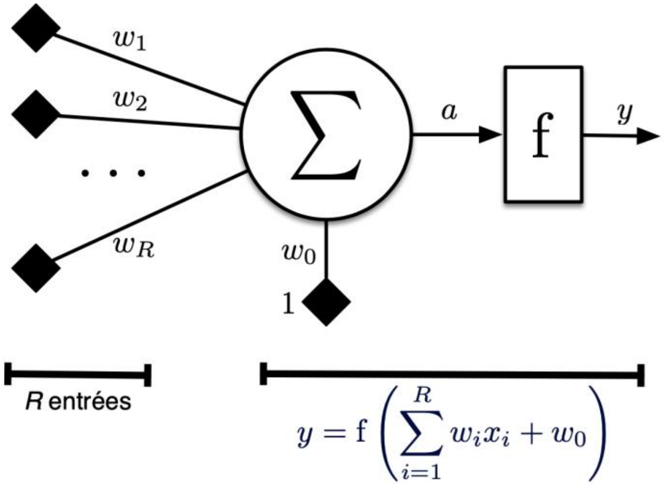
Neural Networks

Artificial neuron

- Inspired by biological neurons



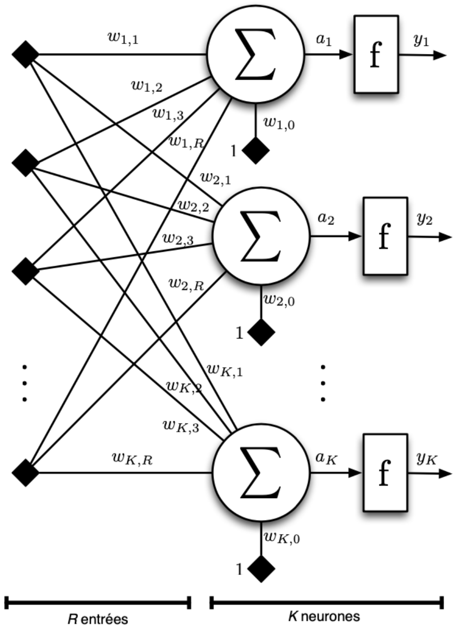
- A neuron is an affine function whose output is subject to an activation function



- Linear discriminant

Layer

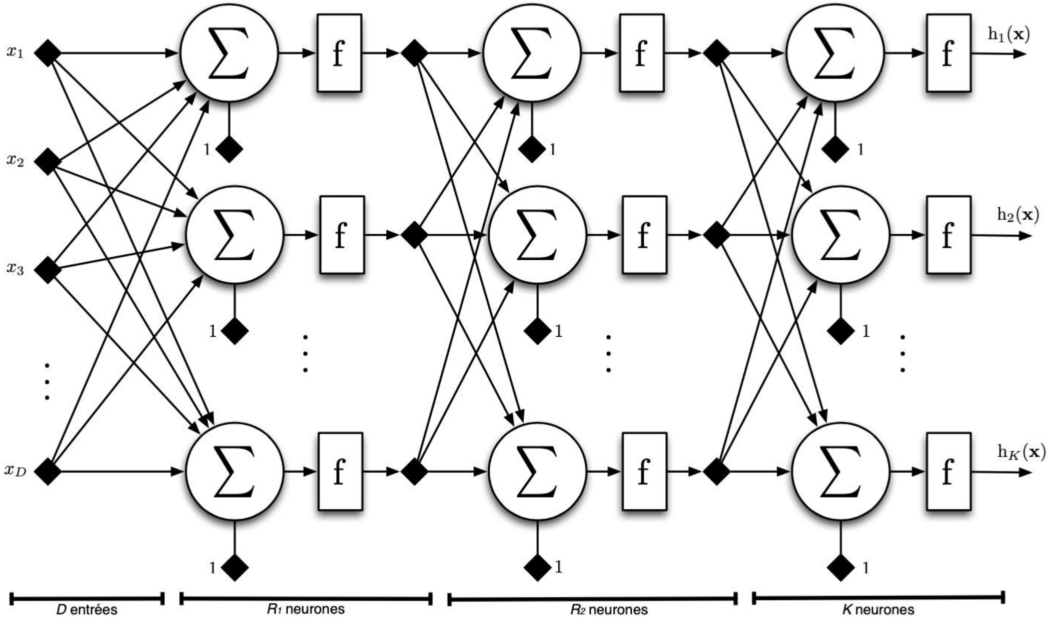
- A layer is a set of vertically stacked neurons
- Each neuron processes input and produces an output
- Linear discriminant



Multilayer Perceptron

- Several connected layers form a deep artificial neural network
- Three types of layers: input, output, hidden layers
- The outputs of one layer are the inputs of the next
- Discriminant linear or not according to the activation function

Multilayer Perceptron



Activation function

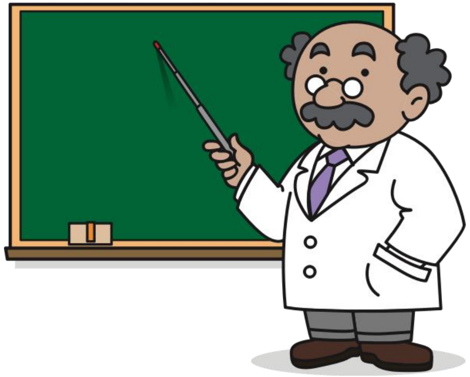
Definition

- Function applied to the output of an artificial neuron

- There are several functions whose usefulness depends on the context:
  - Relu: $f(x) = \max(x, 0)$
  - Sigmoid: $f(x) = \dfrac{1}{1+e^{-x}}, \quad x \in [-\infty, \infty], \quad f(x) \in [0,1]$
  - Hyperbolic Tangent: $f(x) = \dfrac{e^{x} - e^{-x}}{e^{x} + e^{-x}}, \quad x \in [-\infty, \infty], \quad f(x) \in [-1,1]$

Last layer

The activation function of the **last layer** of an artificial neural network allows to **adapt the output** to the task we want to accomplish



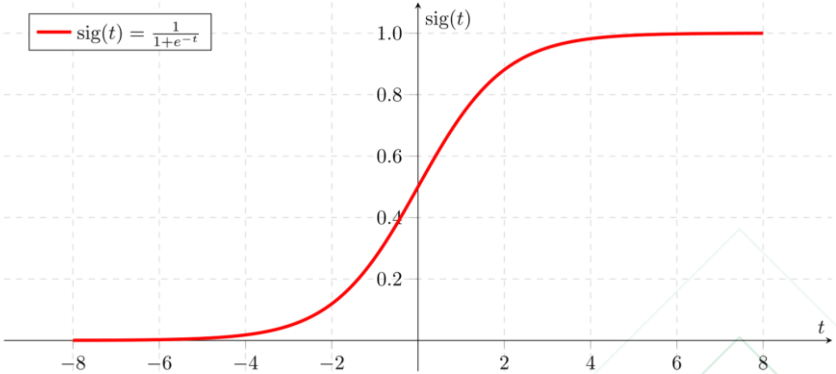
Example: binary classification

- The Sigmoid function transforms an output into a probability between 0 and 1

$$f(x) = \frac{1}{1+e^{-x}}$$

- Useful for binary classification

- Output compared to 0.5 (50%)



Example: multiclass classification

- The Softmax function transforms a vector of outputs into a $\mathbf{z}$ discrete probability distribution

$$f(\mathbf{z})_i = \frac{e^{z_i}}{\sum_{j=1}^{K} e^{z_j}}$$

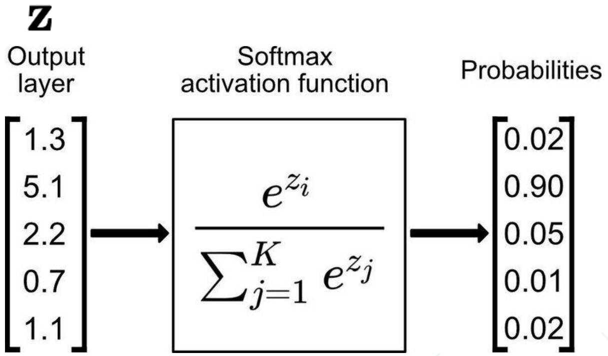
- Indeed, we have

$$f(\mathbf{z})_i \in [0, 1]$$

$$\sum_{i=1}^{K} f(\mathbf{z})_i = 1$$

- Very useful for multiclass classification

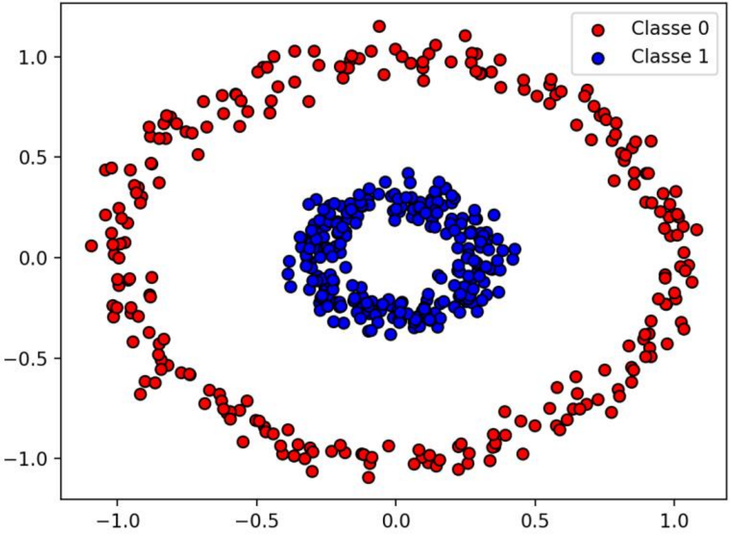
- The class with the highest probability is chosen



Importance of nonlinearity

- The identity function used as an activation for the hidden layers serves as a linear discriminant
- The functions Sigmoid, Tanh, ReLU, etc., allow to process linearly non-separable data
- These last functions bring **non-linearity** to the networks
- Nonlinearity allows networks to learn
- Without nonlinearity, a network is a linear filter

Example: Data classification with a perceptron

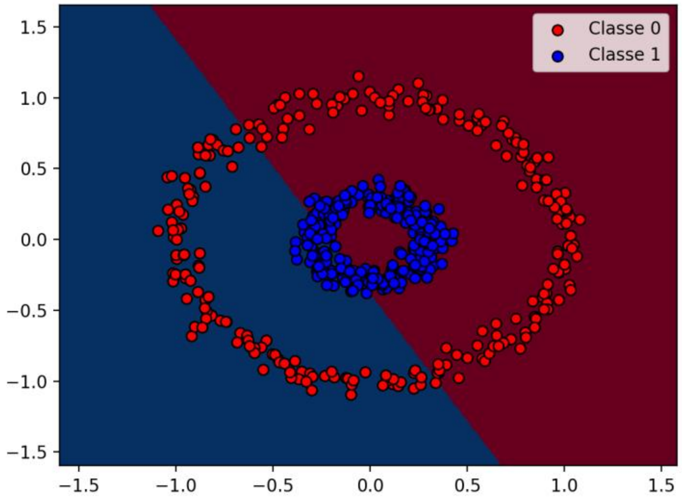


- Distribution of data to be classified
  - 2-D data
  - Two well-separated classes
  - Non-linearly separable boundary

Example: linear network

- A hidden layer: 4 neurons
- Hidden layer activation function: **identity**
- The network poorly separates the data.

$$f(x) = x$$



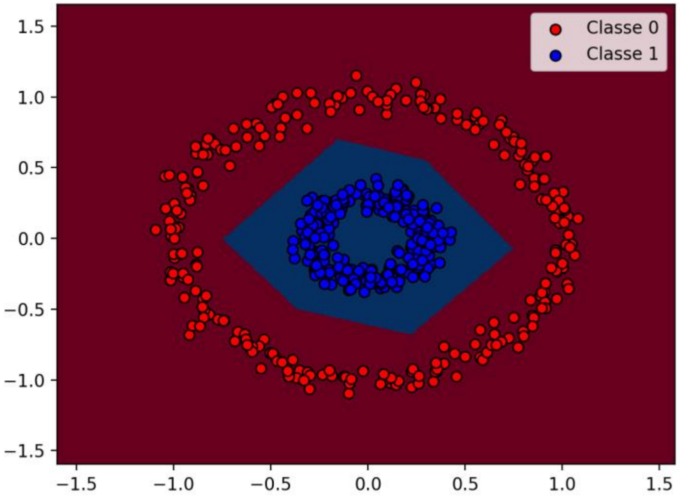
Example: non-linear network

A hidden layer: 4 neurons

Hidden layer activation function: **ReLU**

$$f(x) = max(0, x)$$

The non-linearity of ReLU allows the network to separate the data well.



Refs used to build this course

- Zico Kolter's course on Data Science.
- MOOC IVADO