

ÜK 259

ICT-Lösungen mit Machine Learning entwickeln





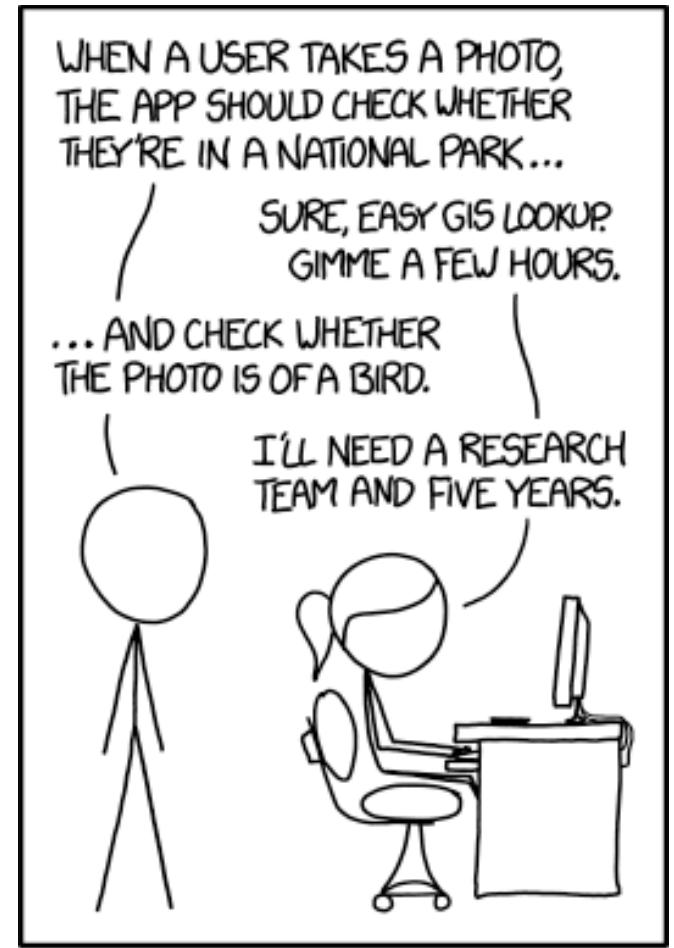






What is Machine Learning?

- What is Intelligence?
- What is Artificial Intelligence?
- What's the difference between Machine learning and conventional computational analysis?



IN CS, IT CAN BE HARD TO EXPLAIN
THE DIFFERENCE BETWEEN THE EASY
AND THE VIRTUALLY IMPOSSIBLE.

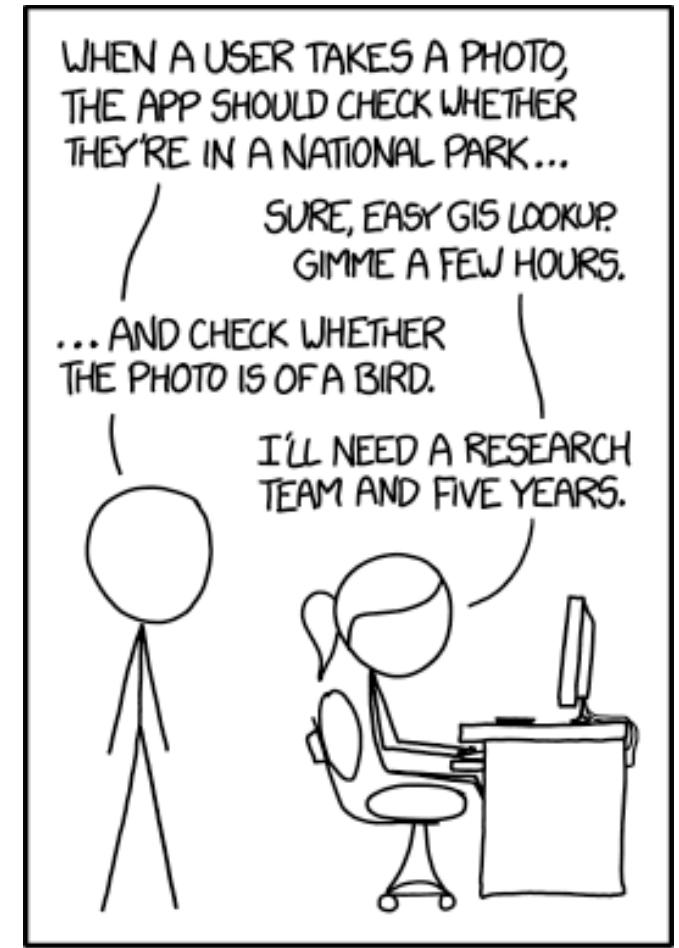
What is Machine Learning?

- **What is Intelligence?**

“Ability to process information and use it to solve a problem”

- **What is Artificial Intelligence?**

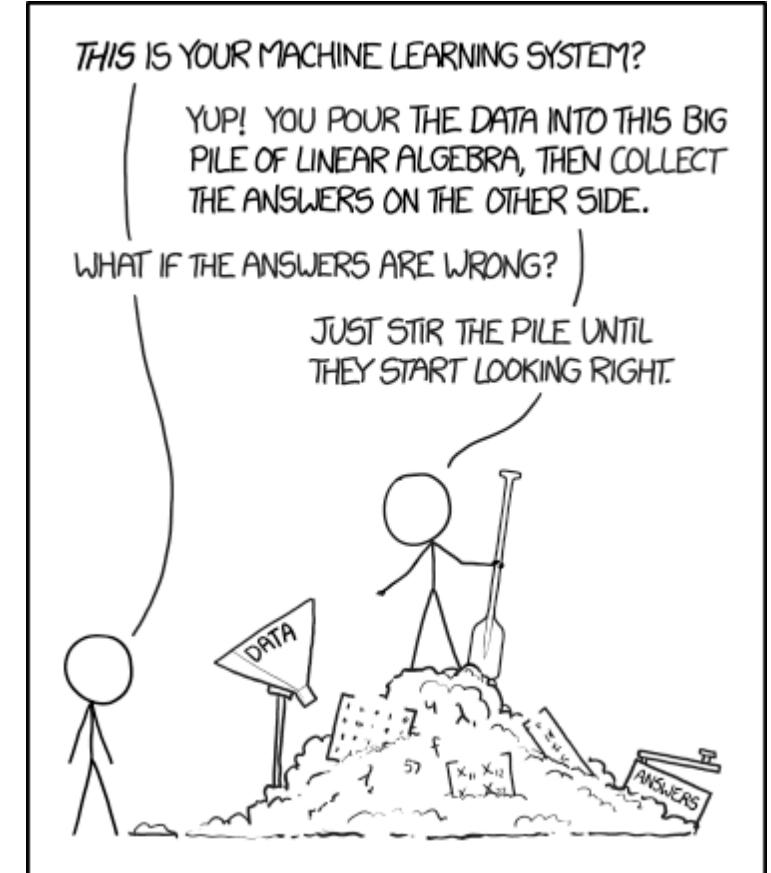
“Algorithms that imitate decision structures of humans to (somewhat) independently solve problems”



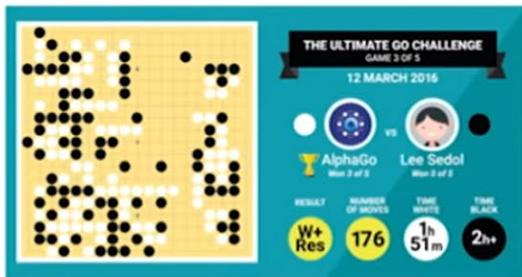
IN CS, IT CAN BE HARD TO EXPLAIN
THE DIFFERENCE BETWEEN THE EASY
AND THE VIRTUALLY IMPOSSIBLE.

Machine Learning is...

- “A lot of Maths”
- A set of algorithms that improve automatically through experience and by the use of data
- A method to solve a problem that we don’t know the solution for (yet)



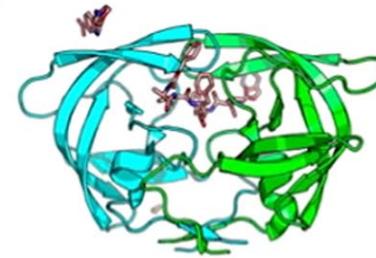
Use Cases



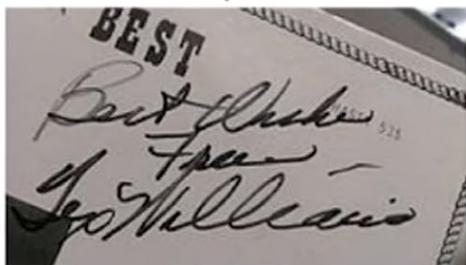
AlphaGo



Recommendation systems



Drug discovery



Character recognition



Hedge fund stock predictions



Voice assistants



Assisted driving



Face detection/recognition

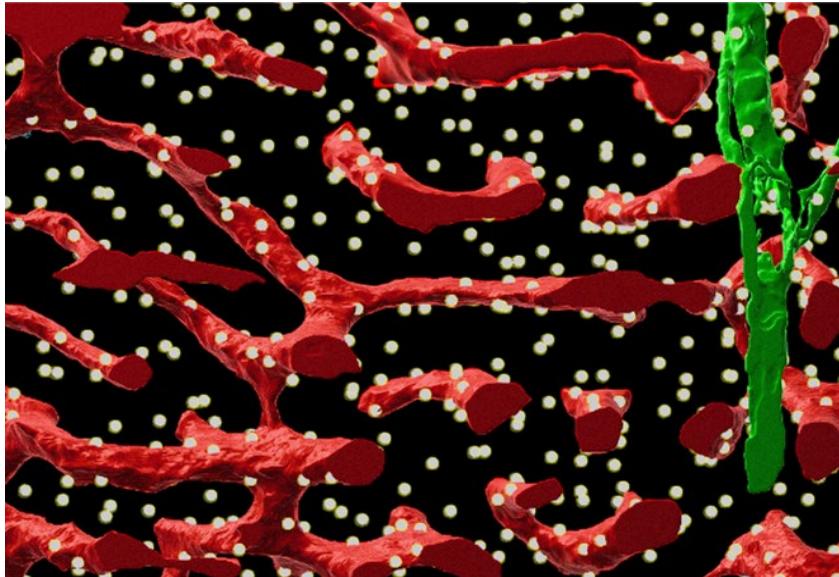
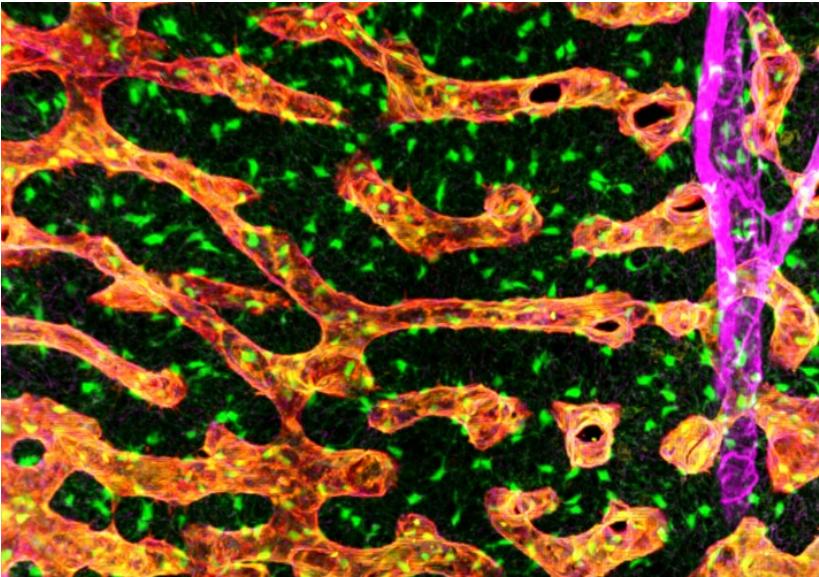


Cancer diagnosis

What can ML do?

Some examples

- AVIA: AI Composer
- This Person does not exist: thispersondoesnotexist.com
- Github CoPilot: <https://copilot.github.com/>
- MINTIF: 3D microscopy segmentation



ÜK 259

Goals

- Understand how machines “learn”
- Get an overview of current ML methods and their applications
- Get a feeling for common problems and workflows in data analysis
- Gather first experiences in applying those methods on real world problems

ÜK 259

HANOKS

- 1.1 Kennt die verschiedenen Technologien im Machine Learning Umfeld und deren Anwendungsbereiche.
- 1.2 Kennt Lösungsvarianten und den Mehrwert für ICT-Lösungen im Vergleich zu bestehenden Lösungen
- 1.3 Kennt die Kategorien des Machine Learning und wählt aus diesen die geeignete Technologie für eine ICT-Lösung aus.
- 1.4 Kennt Modelle und Verfahren im Machine Learning Umfeld
- 2.1 Kennt die gesetzlichen Kriterien zur Bestimmung schützenswerter Daten.
- 2.2 Kennt Massnahmen zur Gewährleistung des Datenschutzes bei der Nutzung und Verarbeitung von schützenswerten Daten.
- 3.1 Kennt die Eigenschaften von Daten und die Vorgehensweise, zur Extraktion von Features für eine Datenanalyse.
- 3.2 Kennt die drei Datentypen und deren Eigenschaften.
- 3.3 Kennt die Möglichkeiten, Daten für eine weitere Verarbeitung aufzubereiten

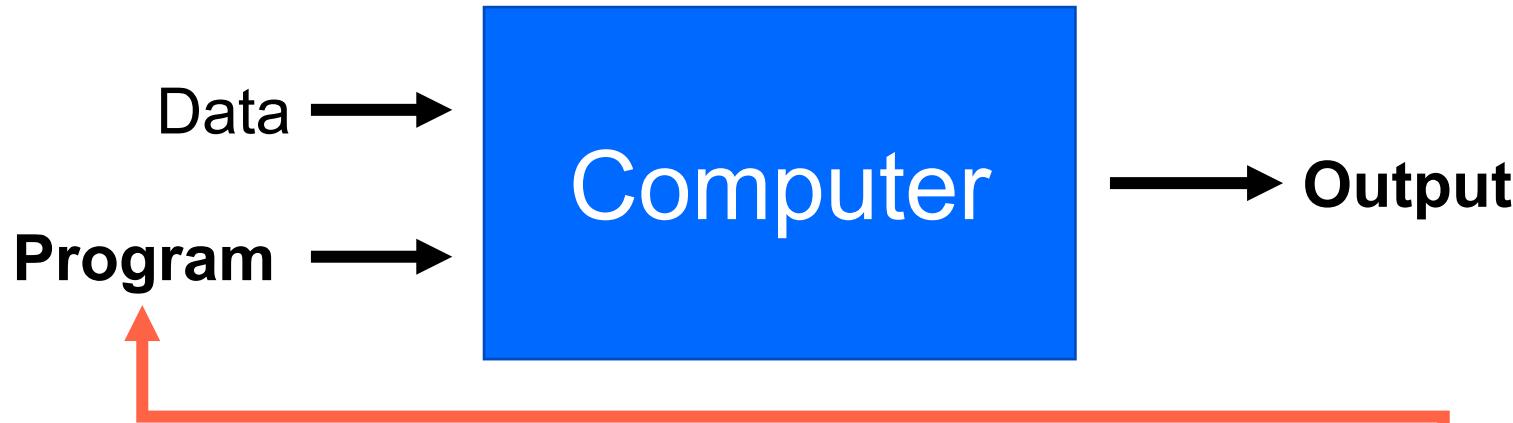
ÜK 259

HANOKS

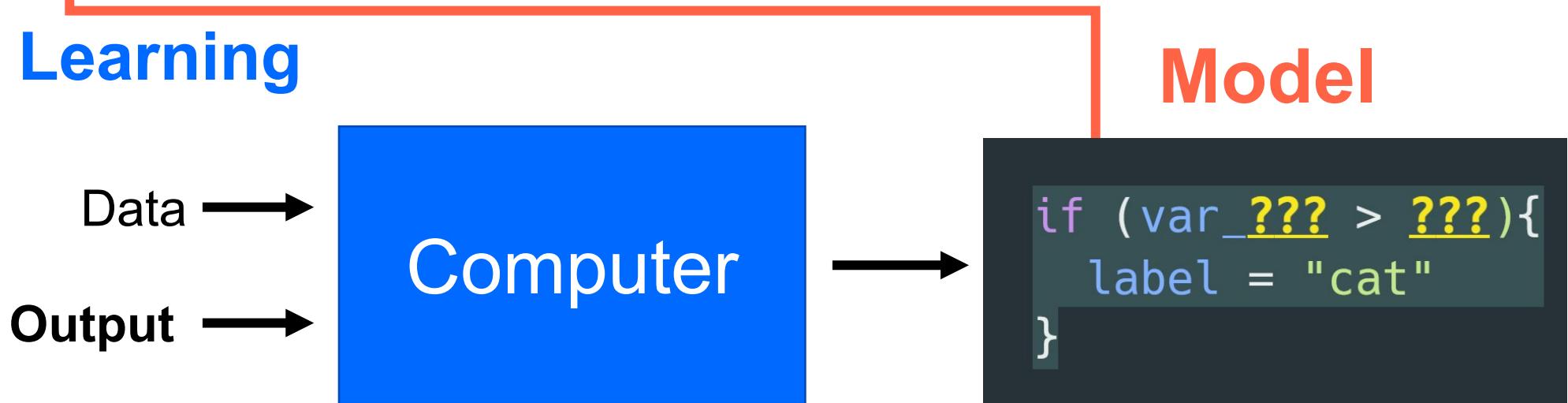
- 4.1 Kennt den Ablauf der Entwicklung einer ICT-Lösung mit Machine Learning gemäss den folgenden Schritten: Zieldefinition, Datenbeschaffung und Aufbereitung der Daten, Lernphase, Interpretation der Ergebnisse und produktivem Einsatz.
- 4.2 Kennt das Vorgehen zum Trainieren und Testen eines Modells
- 5.1 Kennt die Wahrheitsmatrix und deren Funktion.
- 5.2 Kennt die statistischen Gütekriterien zur Beurteilung der Wahrheitsmatrix.
- 6.1 Kennt die erforderlichen Zugriffsmechanismen und die benötigten Schnittstellen eines Machine Learning Dienstes.
- 6.2 Kennt die Komponenten und Dienste sowie das Vorgehen zur Konfiguration nach Vorgaben.
- 7.1 Kennt eine Programmierumgebung und deren Framework zur Entwicklung einer ICT-Lösung mit Machine Learning.
- 7.2 Kennt verschiedene Pipelines zur Entwicklung eines Machine Learning Modells.

Traditional Programming vs. ML

Traditional Programming



Machine Learning



What is a Model?

- A model is a set of instructions (Algorithm) used to come to a decision based on input data
 - It represents the general set of solutions / strategies used to solve the problem
 - A “trained” model is such an algorithm with tuned variables to solve a specific problem.



What is a Model?

- A model is a set of instructions (Algorithm) used to come to a decision based on input data
 - It represents the general set of solutions / strategies used to solve the problem
 - A “trained” model is such an algorithm with tuned variables to solve a specific problem.
E.g. Predicting the weather for tomorrow based on todays weather vs. predicting tomorrows stock price

Problems to solve with machine learning



Classification



Clustering



Regression



Dimensionality
Reduction

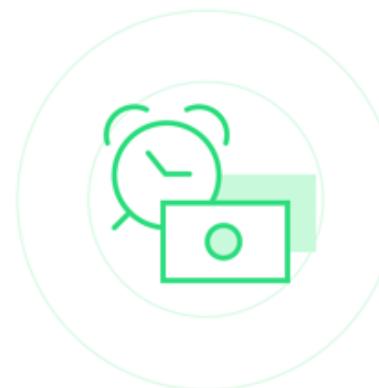
When is traditional computing better than machine learning?



Not enough data



Noisy Data

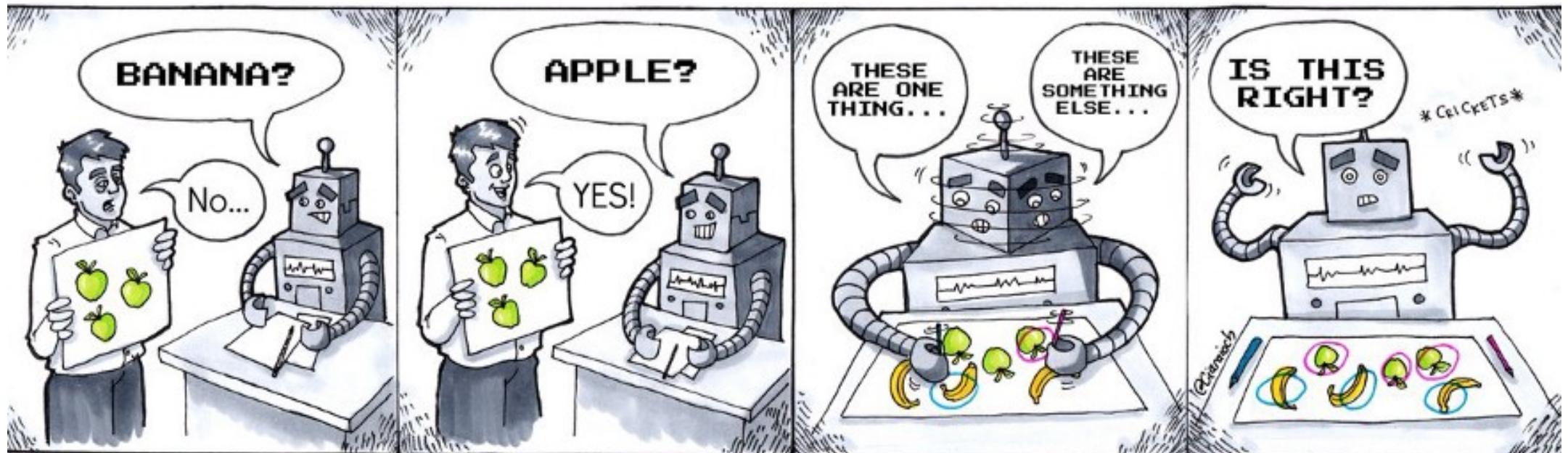


No time & money



Simple problem
to solve

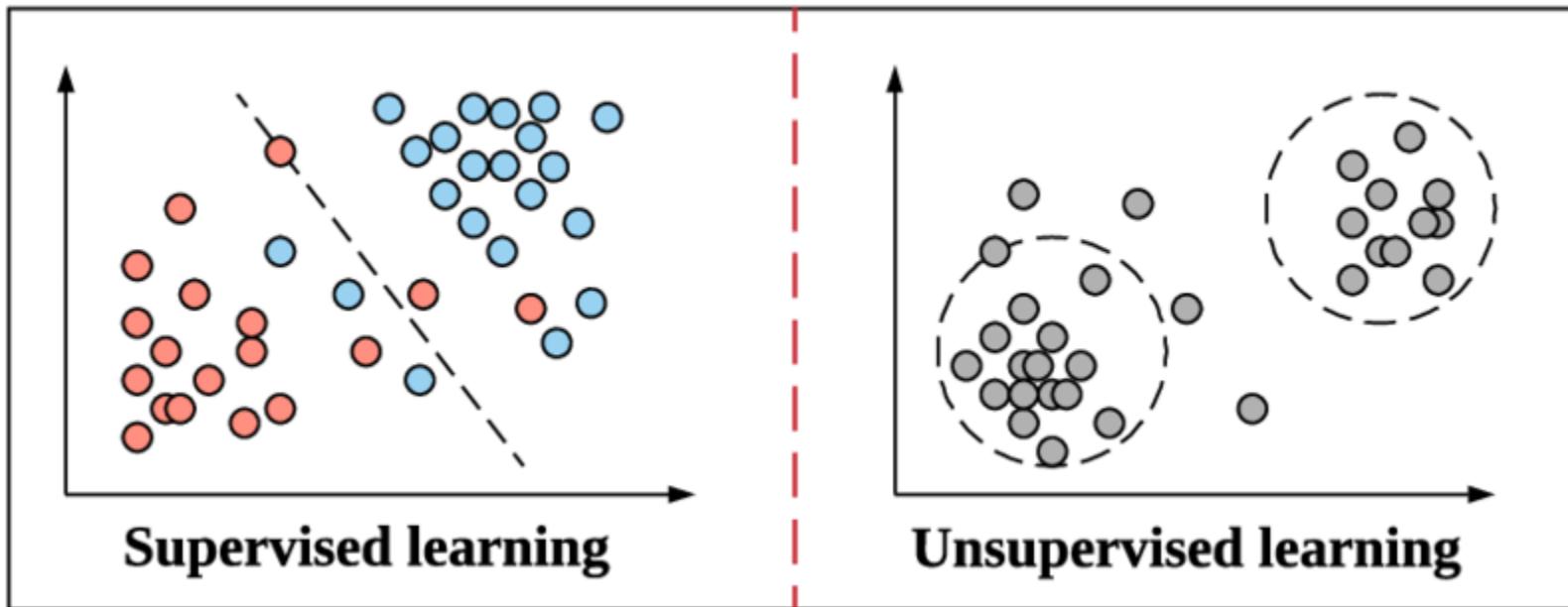
Types of Machine Learning



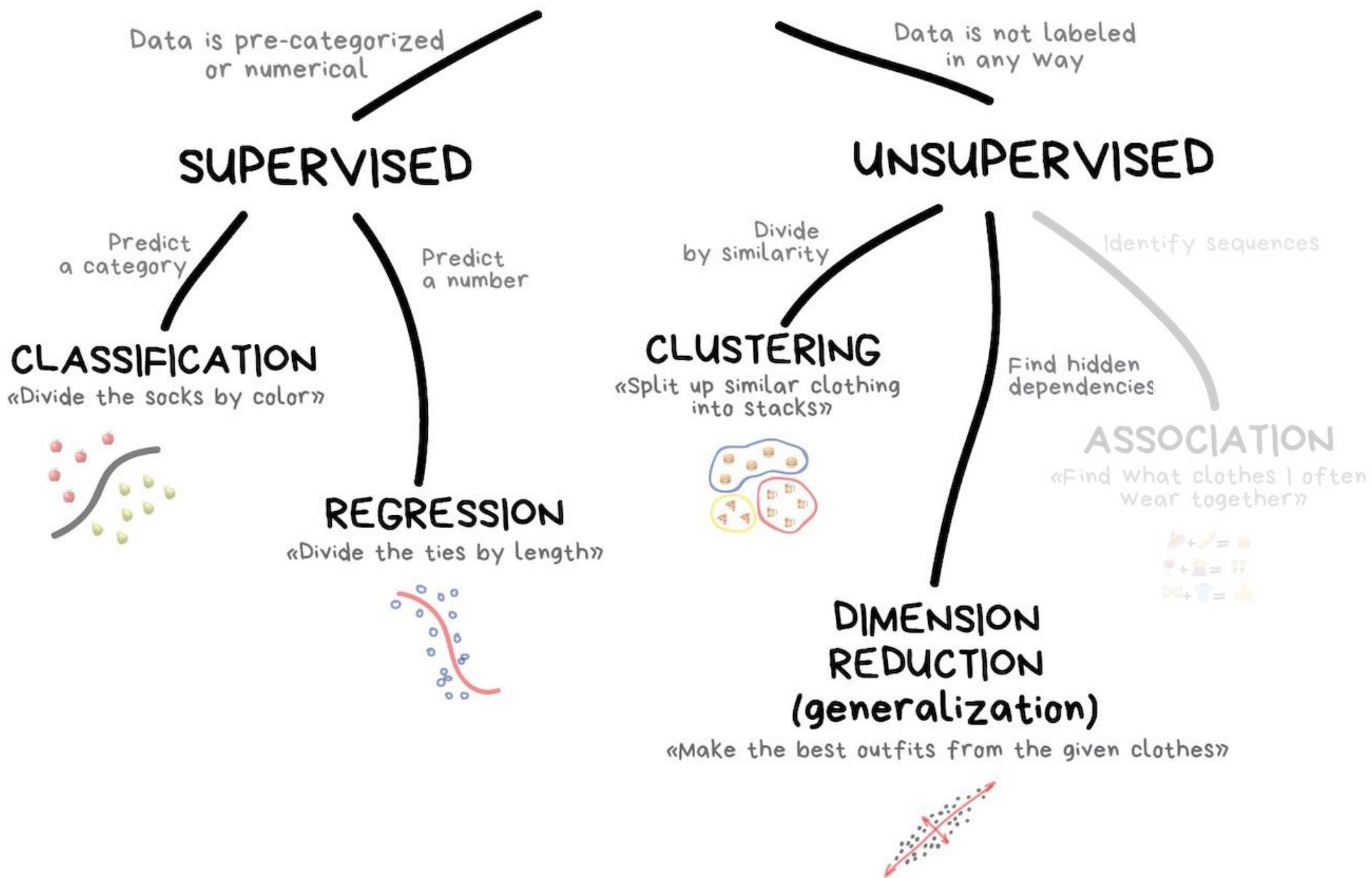
Supervised Learning

Unsupervised Learning

Types of Machine Learning



CLASSICAL MACHINE LEARNING



Main Types of Machine Learning

- **Unsupervised ML:** Find patterns & trends in the data, without any prior knowledge.
- **Supervised ML:** Given a dataset and its corresponding desired output, determine the best Algorithm to predict the output from the data.
- **(Reinforcement Learning):** Given a set of rules, develop a strategy to maximize a score

Tools for Machine Learning

- Python
- Anaconda
 - Numpy
 - Pandas
 - Matplotlib & Seaborn
 - Scikit Learn
 - Tensorflow
- Jupyter (Lab)





“

Data Handling

Data Handling

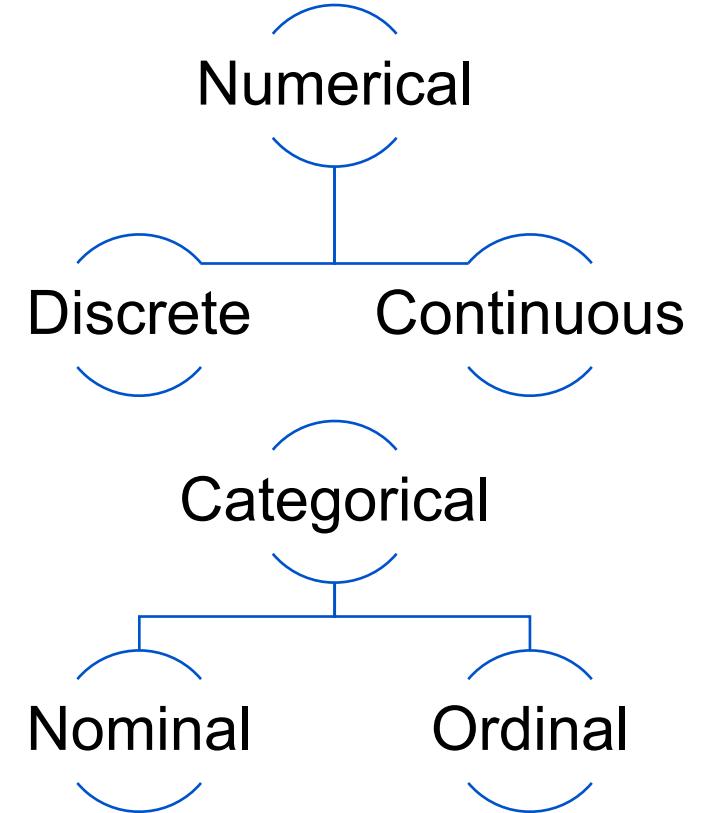
Common Tasks

- Identifying types of data
- Data cleaning
- Standardization
- Data Augmentation
- Visualization



Data Types

- **Numerical:** Numbers (duh)
 - Discrete or continuous
- **Ordinal:** Different states with a defined order
 - T-Shirt size: S < M < L
 - Low, medium, high
- **Nominal:** Multiple states without order
 - T-Shirt color
 - Gender



Data cleaning

Missing values

- Strategies for handling missing values:
 - Ignore (ಠ_ಠ)
 - Remove (losing statistical power)
 - Default values (e.g. 0)
 - Interpolate (e.g. mean, max)

BuildingArea	YearBuilt	CouncilArea
NaN	1981.0	NaN
133.0	1995.0	NaN
NaN	1997.0	NaN
157.0	1920.0	NaN
112.0	1920.0	NaN

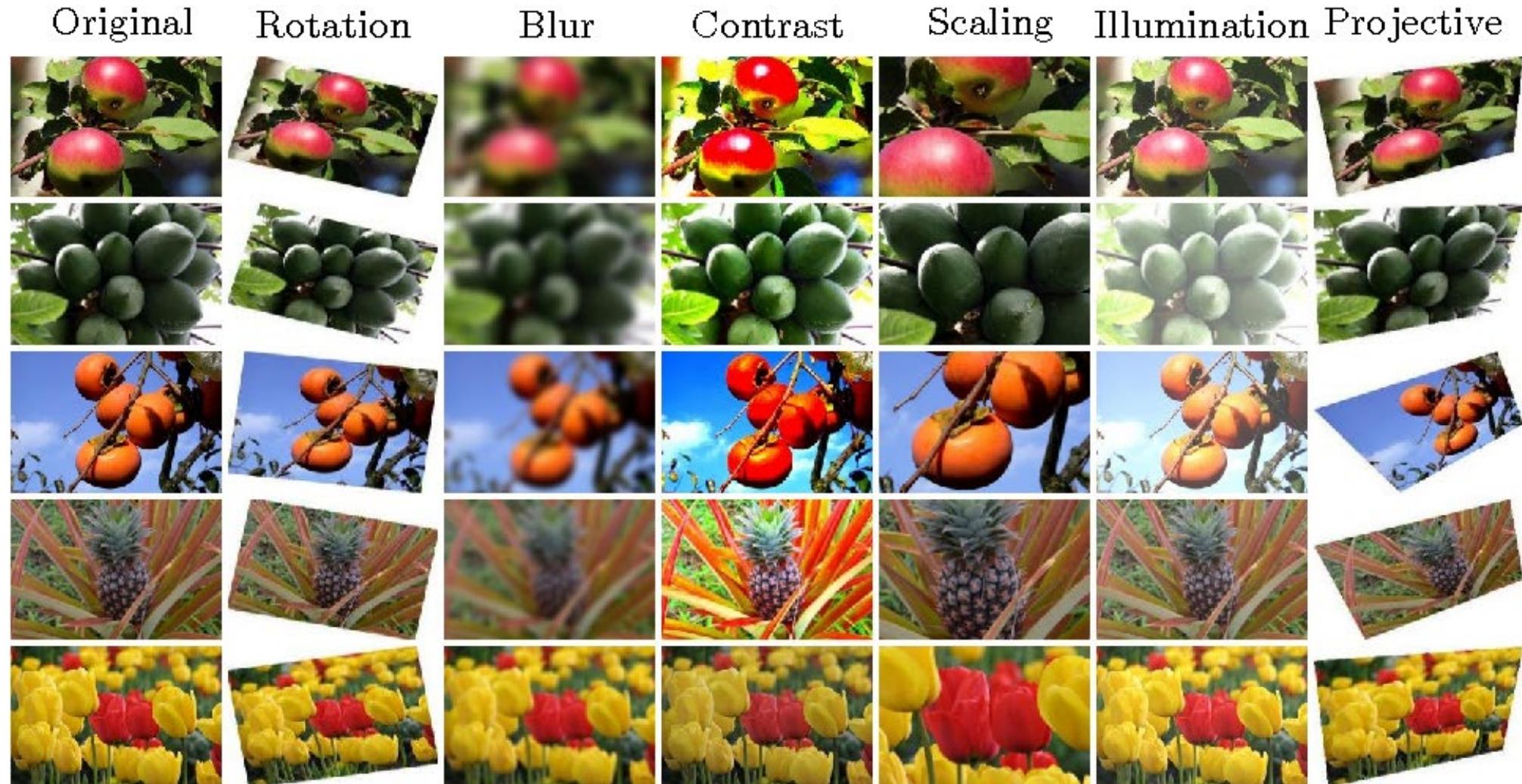
Standardization

- **Problem:**
 - Features with a large scale are interpreted as having more weight (e.g. grams vs. kg).
 - Features with a large variance are interpreted as more informative.
- **Idea:** Scale features to same mean and variance:

$$\text{standard}(x_i) = \frac{x_i - \text{mean}(x)}{\text{stdev}(x)}$$

- Implemented by: `sklearn.preprocessing.StandardScaler`

Data Augmentation



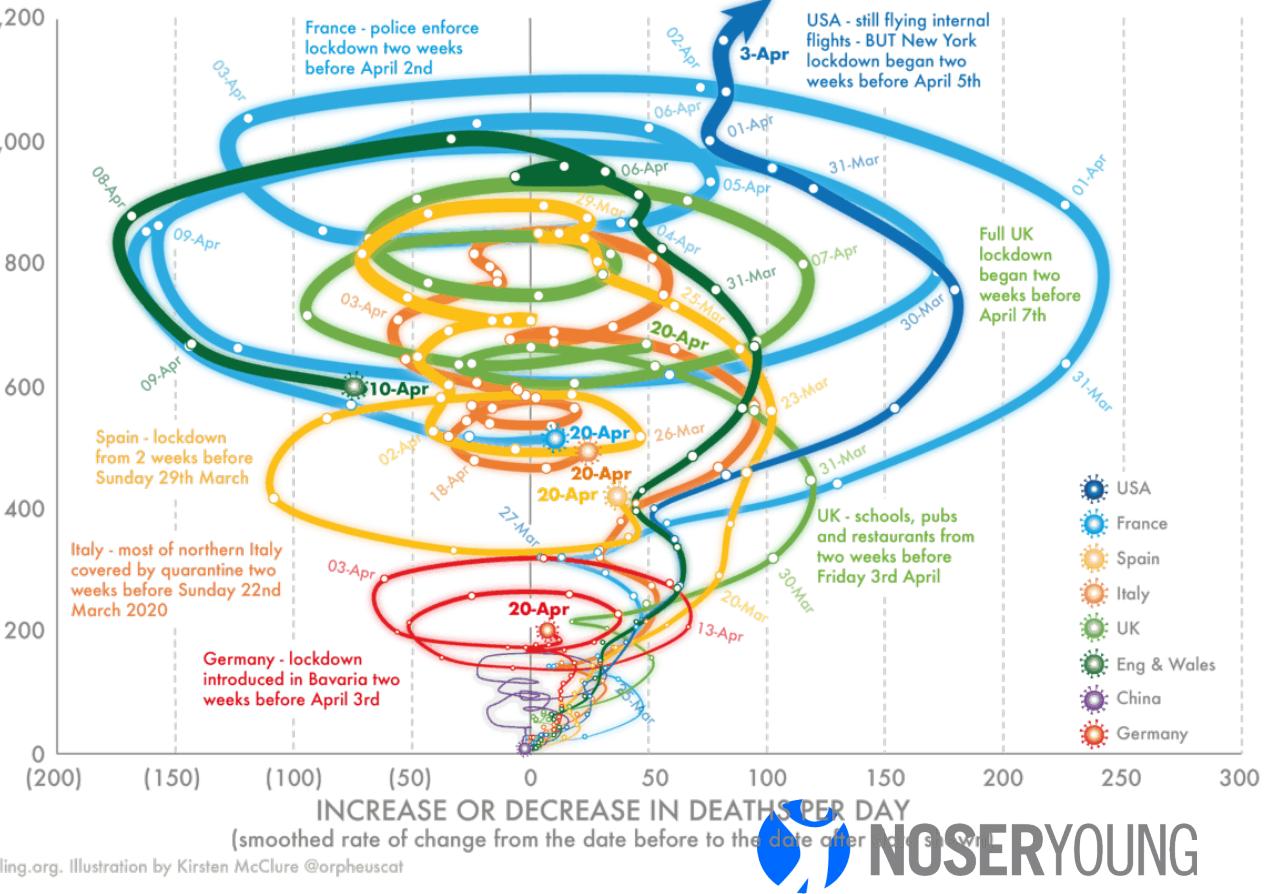
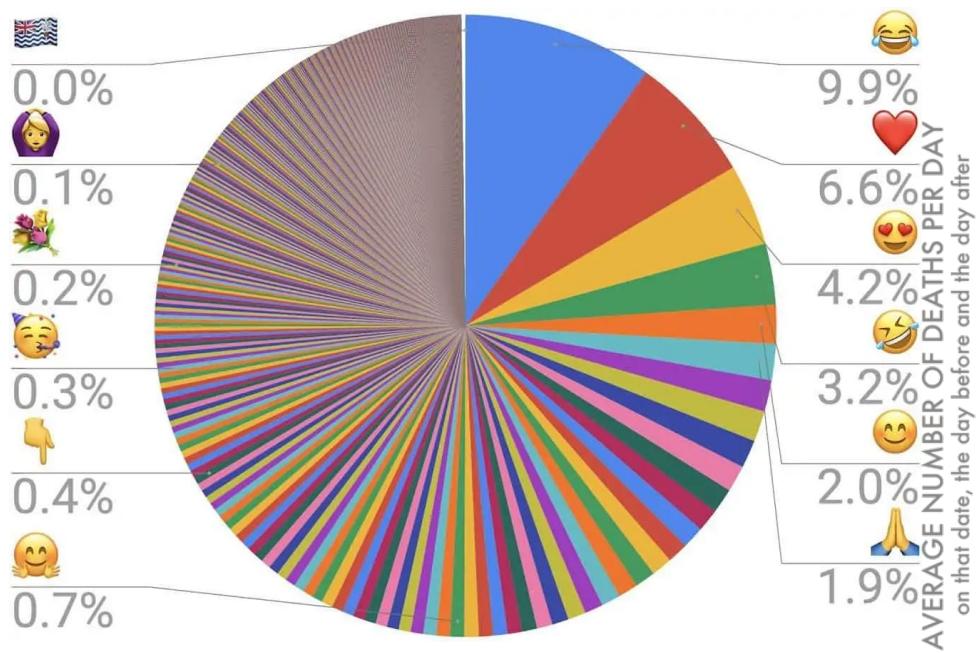
Data Augmentation

- **Idea: Modify data to augment the dataset**
- Used to make a training set more robust by introducing more variation in the Dataset
- **Improving model prediction accuracy** by increasing generalizability and overall increasing the size of the training dataset.

Data-Visualization

Data-Visualization

How NOT to do it

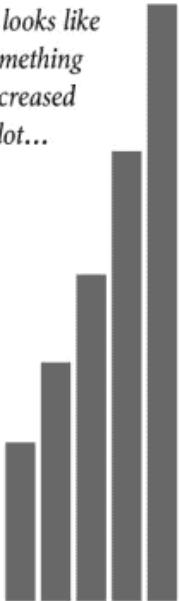


Data-Visualization

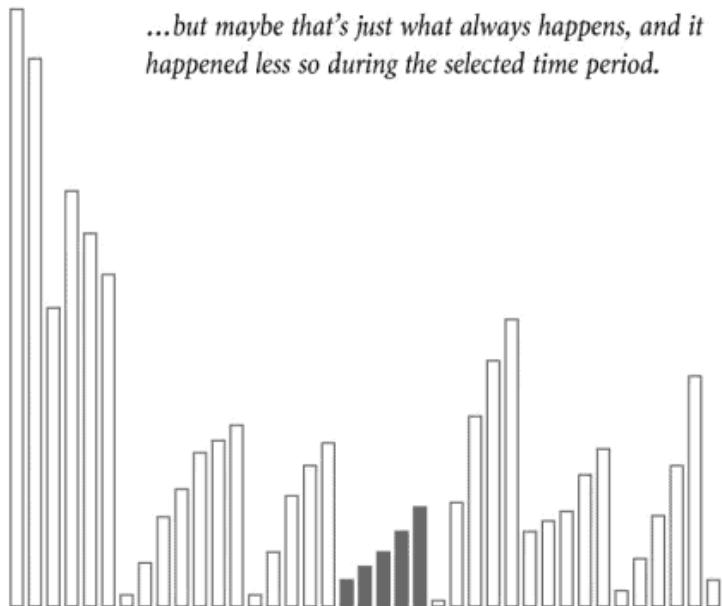
How NOT to do it

LIMITED SCOPE

It looks like something increased a lot...

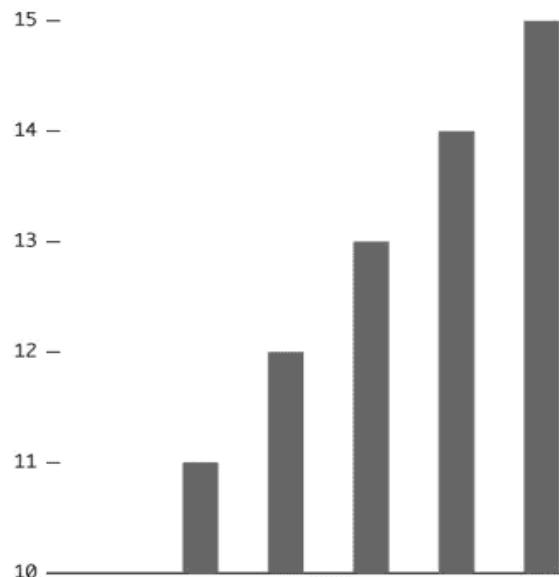


...but maybe that's just what always happens, and it happened less so during the selected time period.

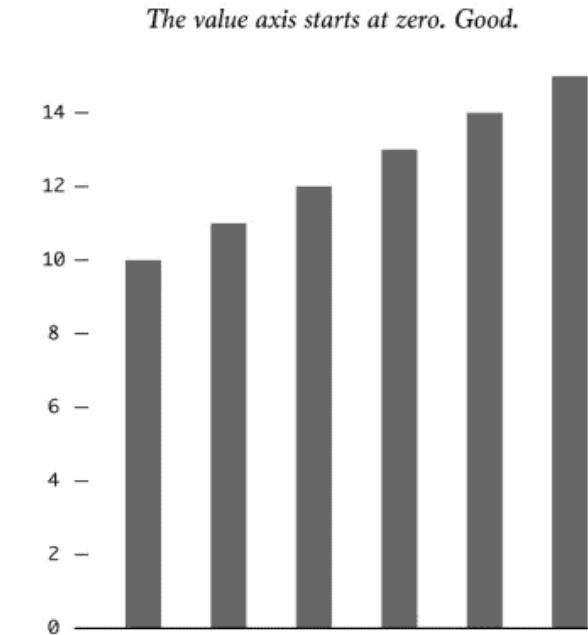


TRUNCATED AXIS

The value axis starts at ten. Liar, liar, pants on fire.

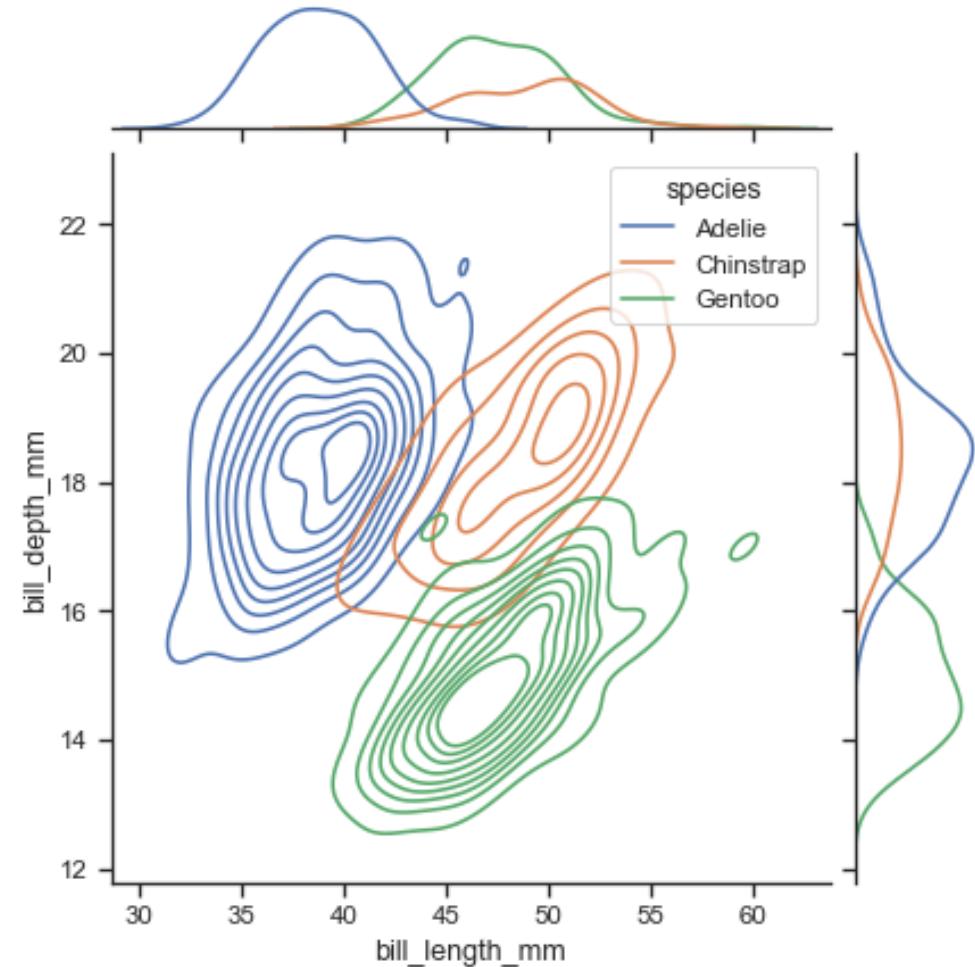


The value axis starts at zero. Good.



Data-Visualization

- **Visualizing your data should be the first and last thing you do!**
- Communicating results is a difficult but important part of Data Science
- The more complex the data the more important good and accurate data visualization becomes

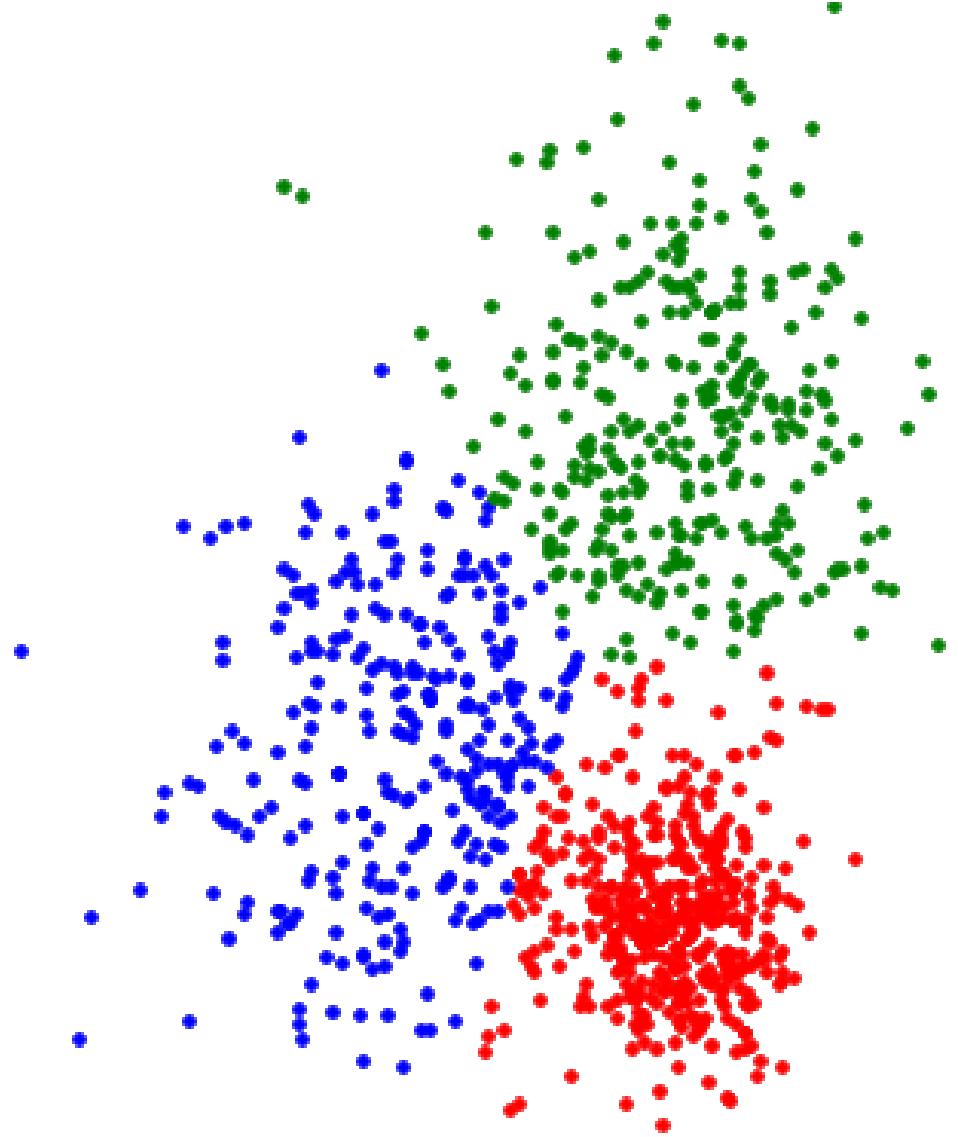


Hands-On

Part 2

Explore the Dataset “melb_data.csv” of the Melbourne Housing Market

1. Clean the dataset
2. Standardize the data
3. Think about how you would augment this dataset
4. Visualize and present an aspect of the dataset you find interesting

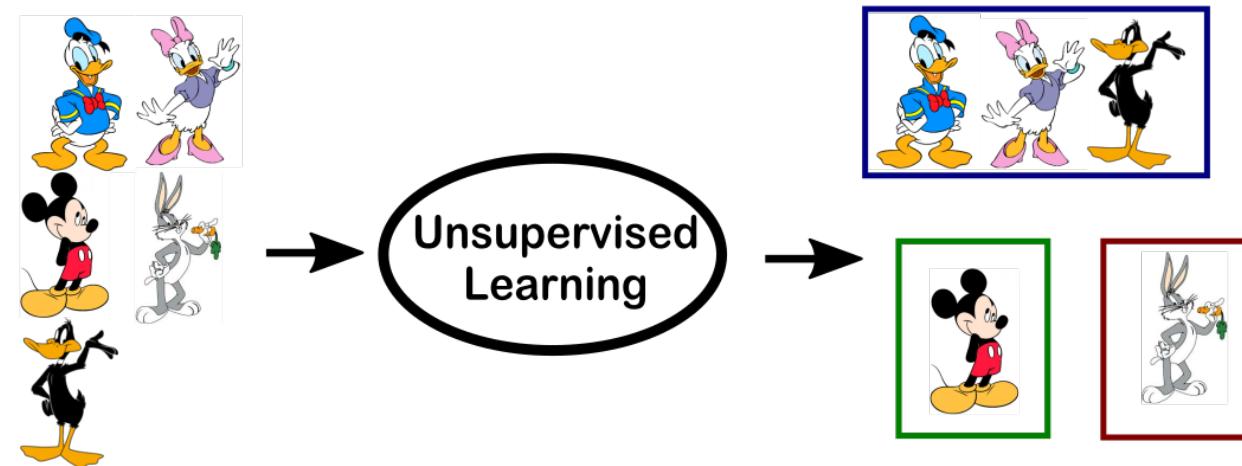


”

Unsupervised Machine Learning

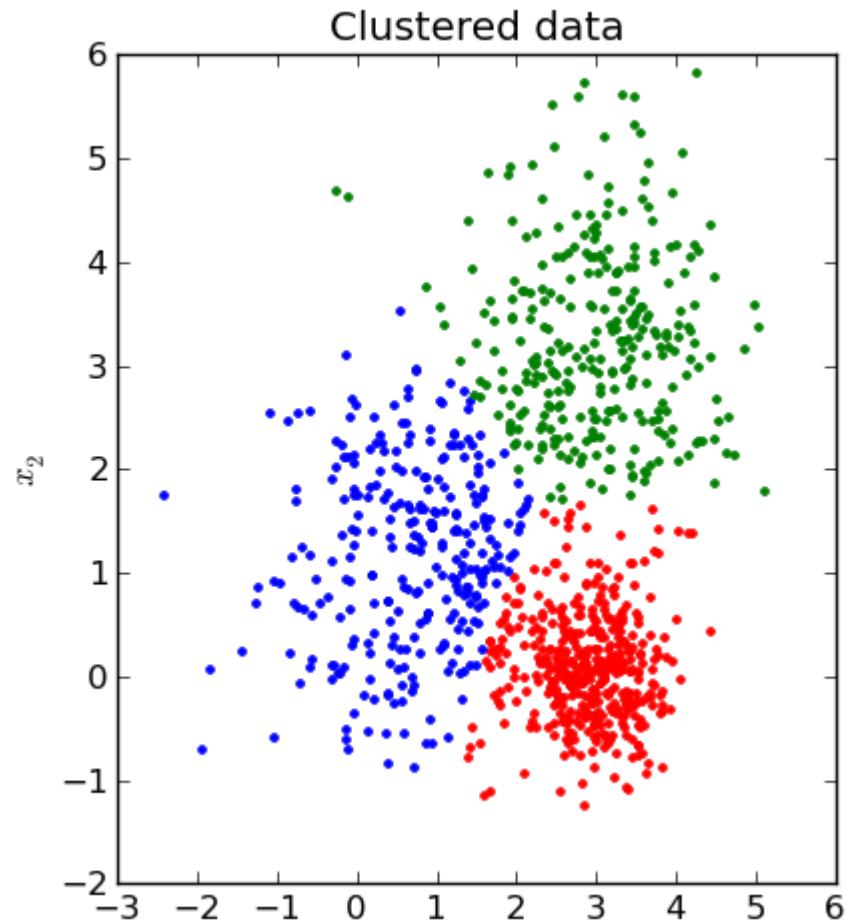
Unsupervised ML

- Idea: Find patterns & trends in the data, without any prior knowledge
- These patterns may give us new insights into our data
- Main Types:
 - Clustering
 - Dimensionality reduction



Clustering

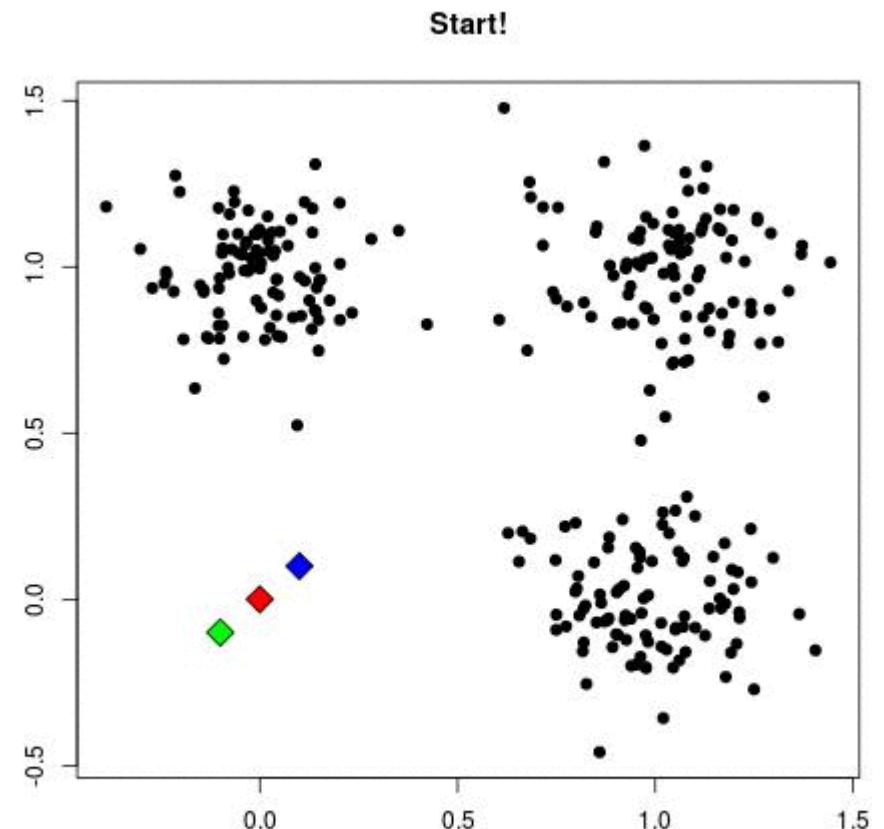
- Group datapoints into «close» groups
→ Works on some measure of similarity / distance
- Applications:
 - Customer segmentation
 - Recommender Systems
 - Anomaly Detection



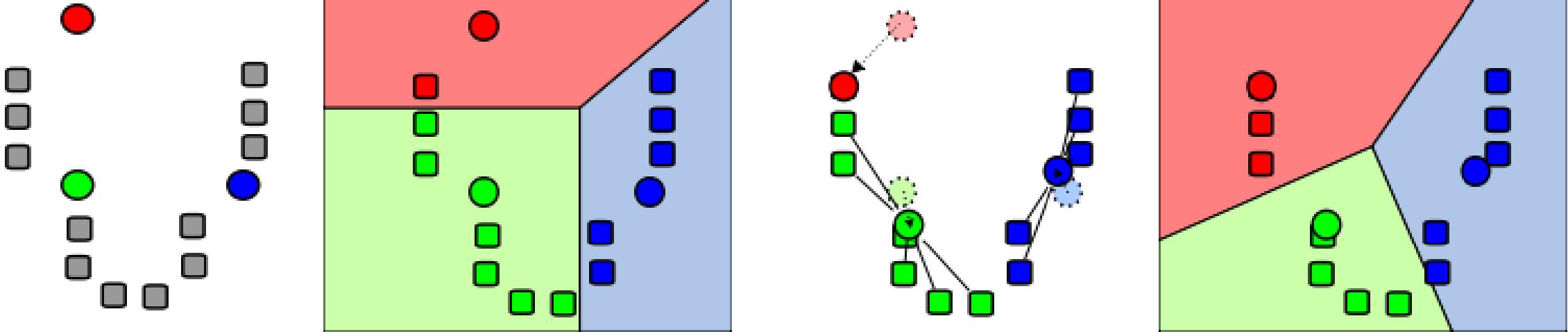
K-Means

Algorithm

1. Initialize k random centroids
2. do{
 1. re-assign all points to closest centroid
 2. Recalculate centroid}
- } while #reassignments > 0

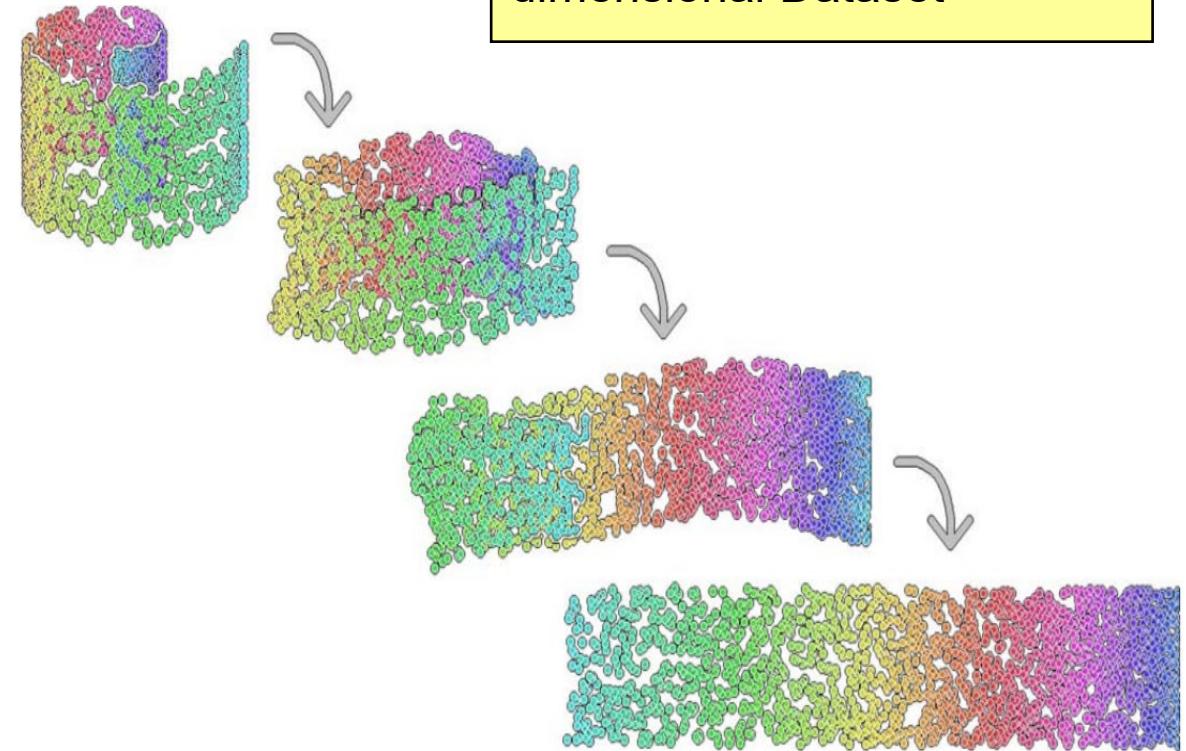


K-Means



Dimensionality Reduction

- **Idea:** represent a high-dimensional dataset in lower dimensions, while preserving local structures
- **Uses:**
 - **Data visualisation**
 - **Denoising**



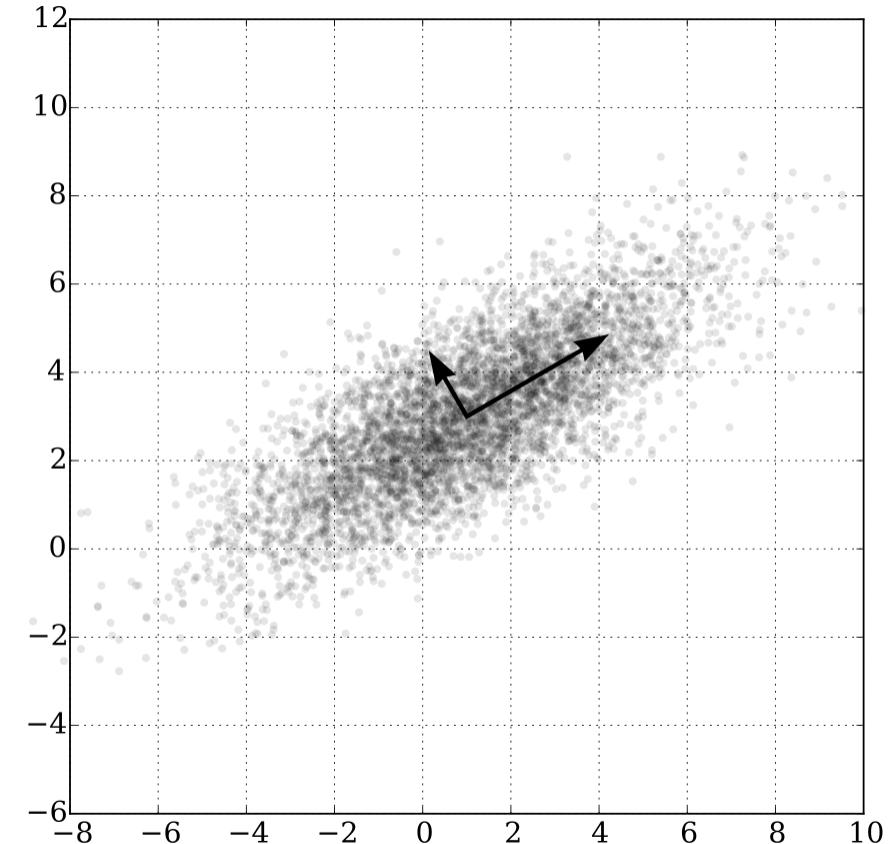
Dimension in Data-Science / Mathematics:

One axis or column of a dataset. E.g. a Dataset with 10 Columns is a 10 dimensional Dataset

Principal Component Analysis

For dimensionality reduction

- **Idea:** Find the **principal components** that best describe the variations in the data
 - “Intuitive” explanation of PC:
Shift the coordinate system such that you can discard one or more axis without loosing much information
→ PC is the main axis of variance
- Implemented in `sklearn.decomposition.PCA`



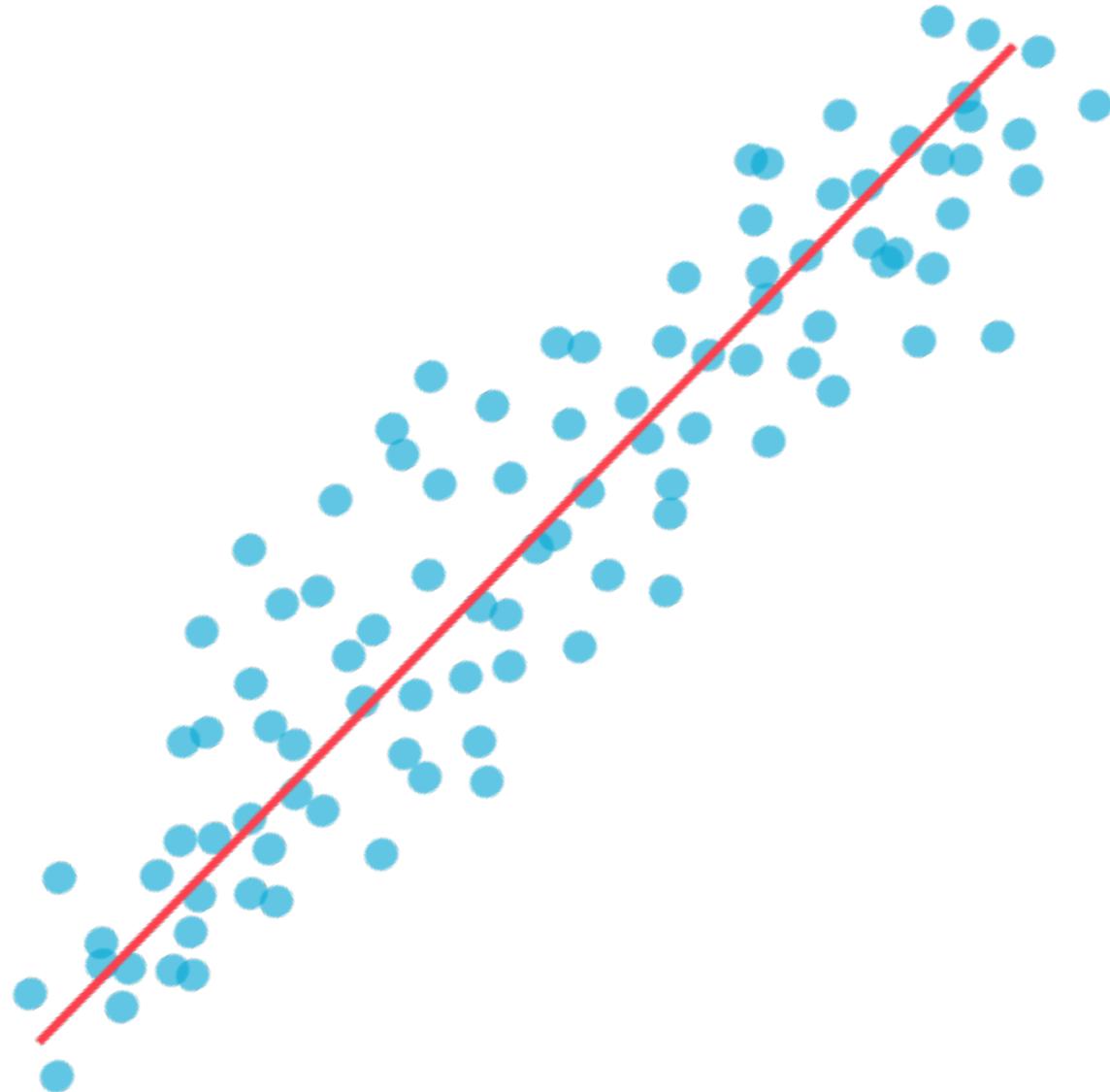
Additional Information Unsupervised Learning

- **kMeans:** <https://www.youtube.com/watch?v=mfqmoUN-Cuw>
- **PCA:** <https://www.youtube.com/watch?v=UVHneBUBW0>
 - Math: <https://www.youtube.com/watch?v=PFDu9oVAE-g>

Hands-On

Part 3

1. Implement the K-Means Algorithm for a set of random 2D datapoints (use the ‘`sklearn make_blobs`’ function to get a random dataset with underlying clusters
 1. Visualize your results (Bonus: can you animate the graph to show each iteration of the algorithm?)
2. Think about how you could use your implementation to categorize a new (previously unknown) datapoint.
 1. Bonus: Implement your idea and visualize the result

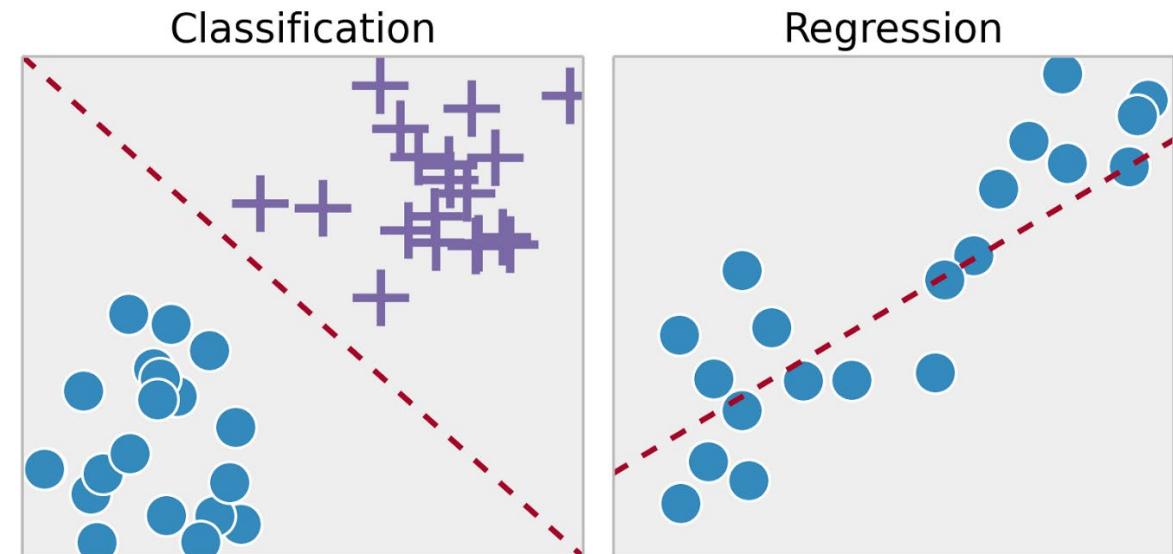


”

Supervised Machine Learning

Supervised ML

- **Idea:** given a dataset and its corresponding desired output, determine the best algorithm & parameters to predict the output from the data
- **Use cases:**
 - Classification
 - Regression / Prediction
- **Common Methods:**
 - Decision Trees
 - Support Vector Machines
 - **Neural Networks**

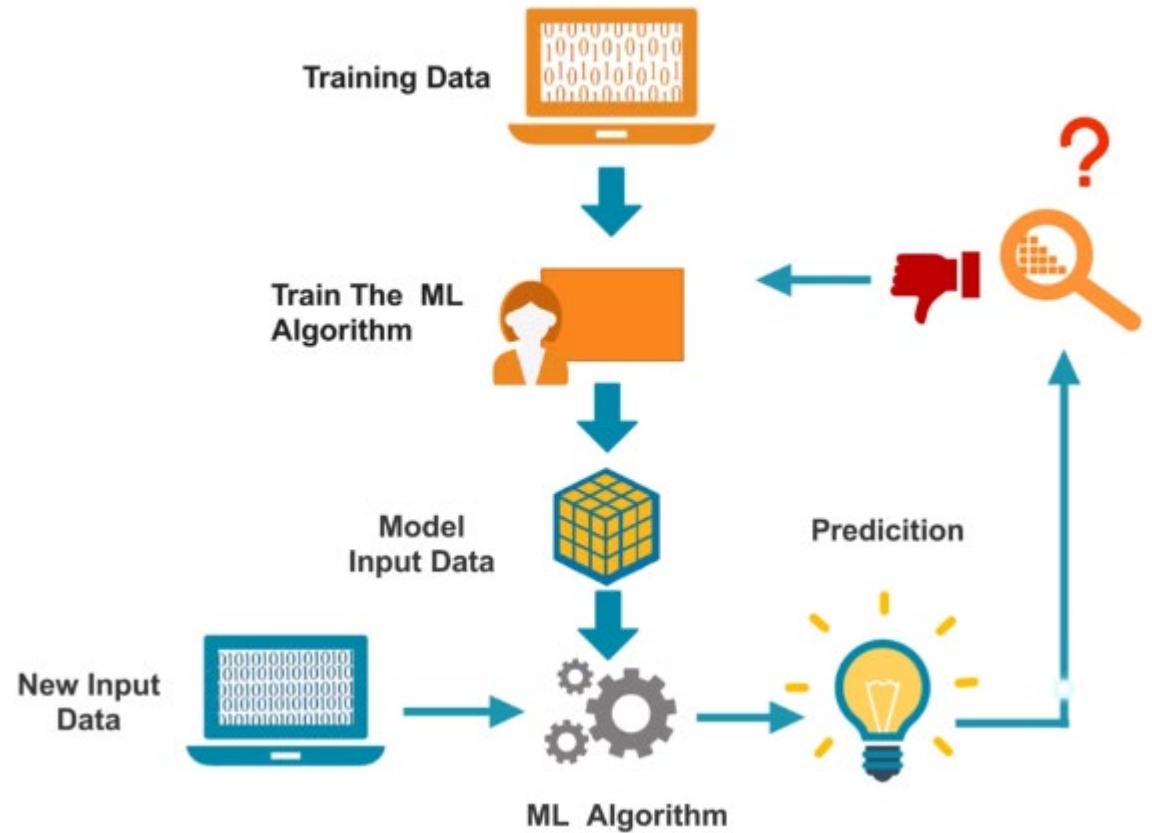


How does “supervision” work?

- **Intuition:** 1 Algorithm **builds models**, 1 Algorithm **scores the models** and chooses the best.
 - The builder Algorithm tweaks the best model and repeats
- **Building:** Depends on specific method, generally tweaking of model parameters.
- **Scoring:** Calculate a predefined **cost function / score**
 - E.g. Sum of Squared Errors (SSE)
- How Machines Learn [CGPGrey]

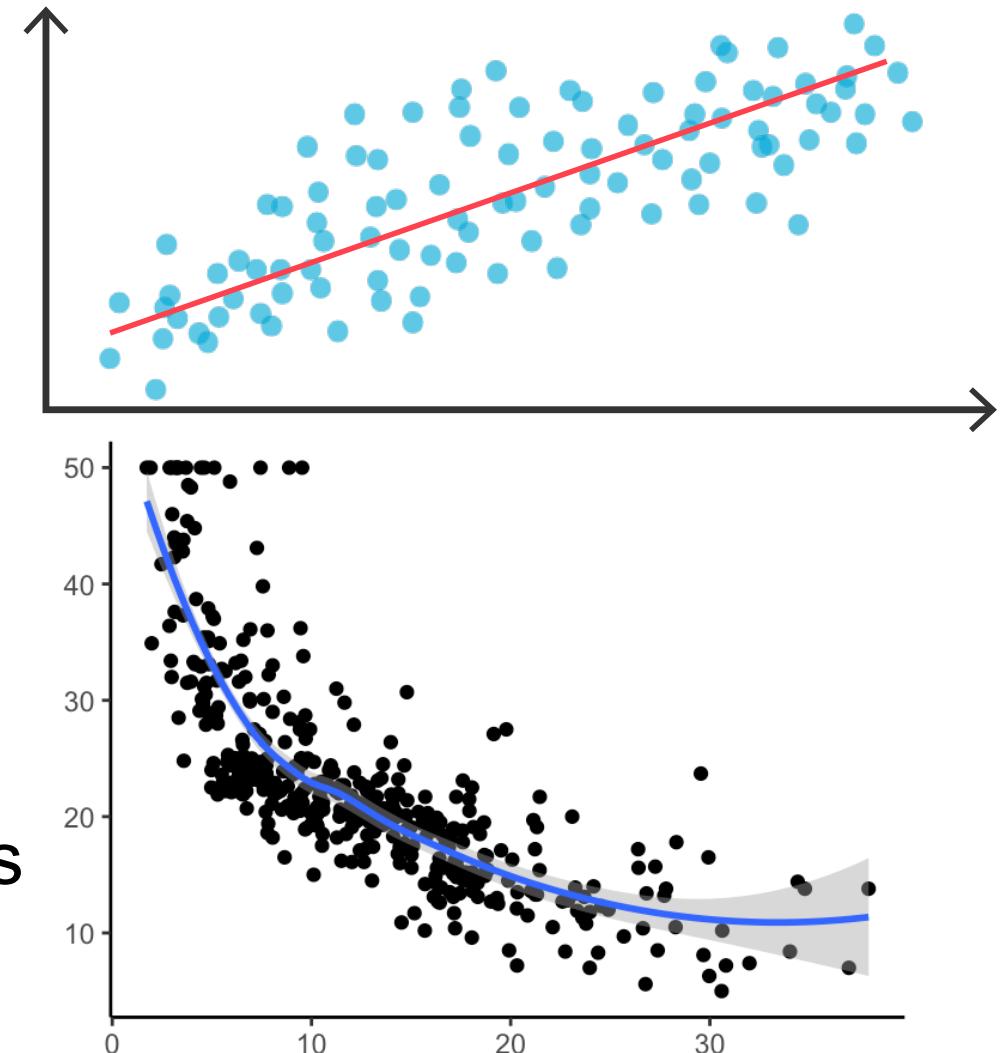
Typical Supervised ML Workflow

1. Define Goal
2. Get Data
3. Prepare Data
4. Create & Train A Model
5. Evaluate & Improve
6. Make Predictions



Regression

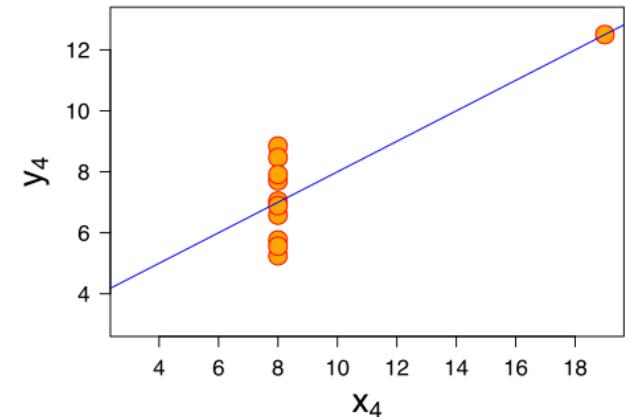
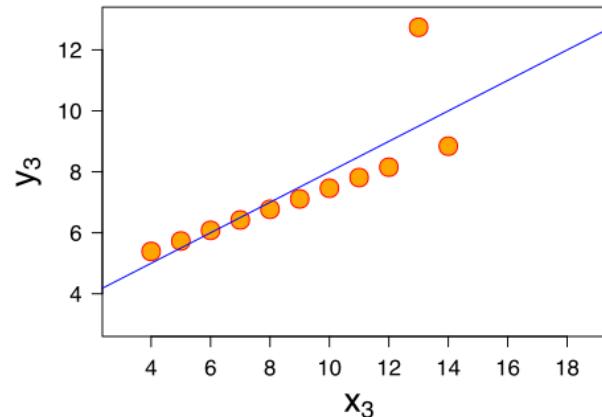
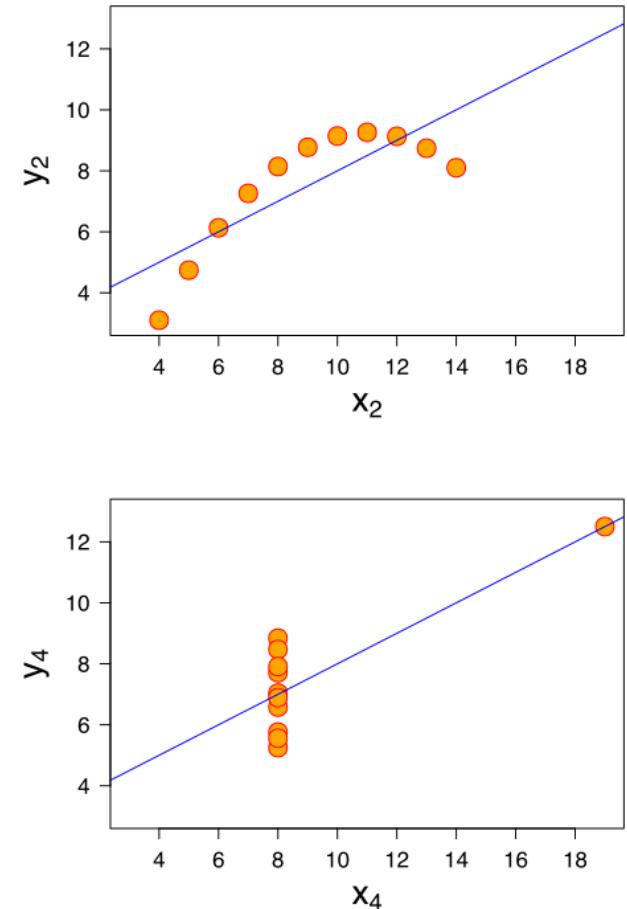
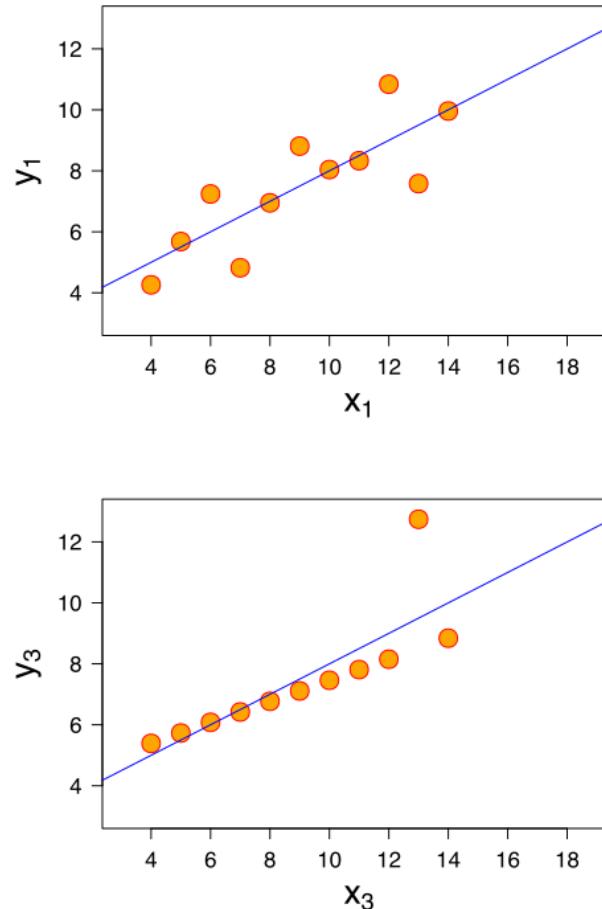
- **Idea:** Given a set of input variables, predict a numerical output variable
- **Use cases:**
 - Prediction in Marketing, Medicine, Finance etc.
- **Method:**
 - Find line that minimizes error between prediction and real values



NOSERYOUNG

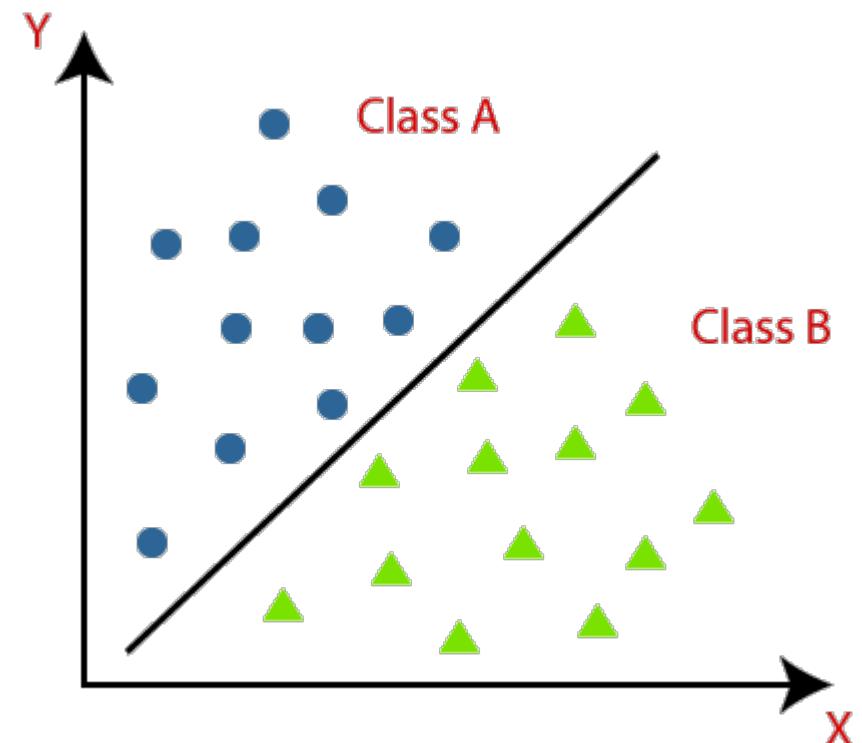
Regression Problems

- Can be susceptible to outliers (this can be overcome)
- Can be susceptible to over- / underfitting



Classification

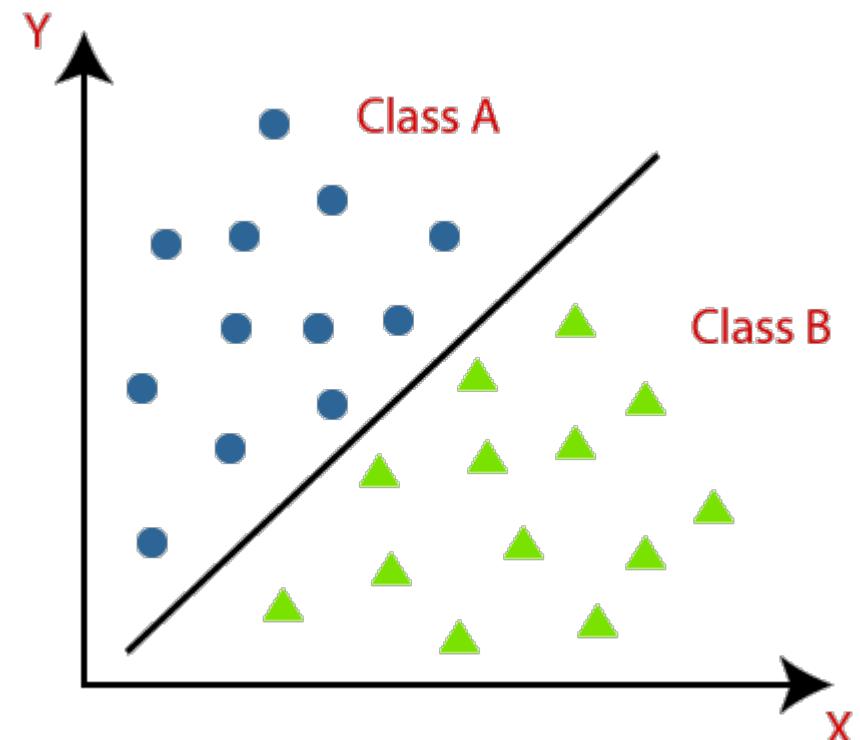
- **Idea:** given a set of input variables, predict a class for each datapoint
- **Use cases:**
 - Image-/ Speech-recognition
 - Medical prognosis
 - Customer Segmentation
 - Spam detection



Classification

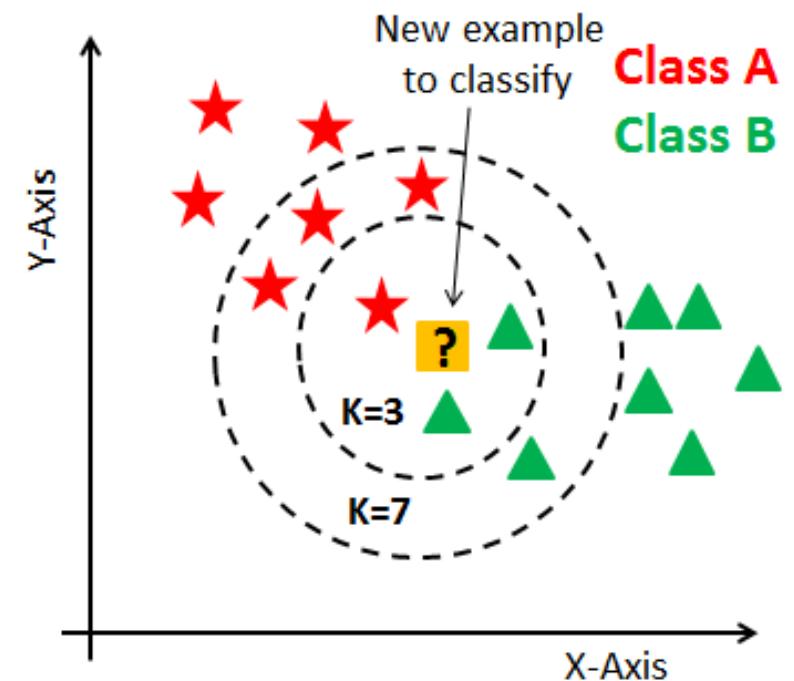
Method

- In general, the aim is to minimize the number and severity of wrong assignments
- **Selection of classification Algorithms:**
 - K-nearest neighbours
 - Decision Trees
 - Support Vector Machines
 - Neural Networks



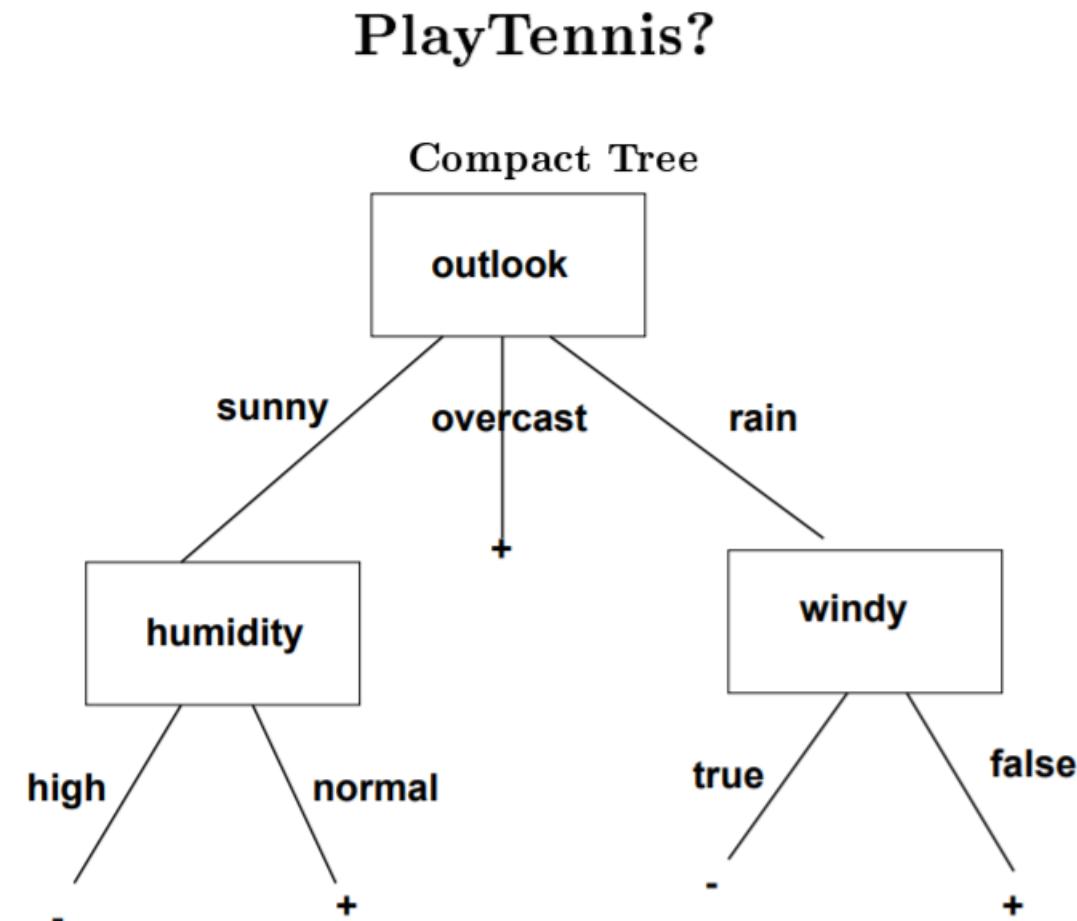
K-nearest-neighbour

- **Idea:** classify a point as the majority class of its k nearest neighbours.
- **Pros:** no training, simple, easy to incorporate more data
- **Con:** cannot handle very large, very high dimensional or imbalanced datasets. Sensitive to outliers



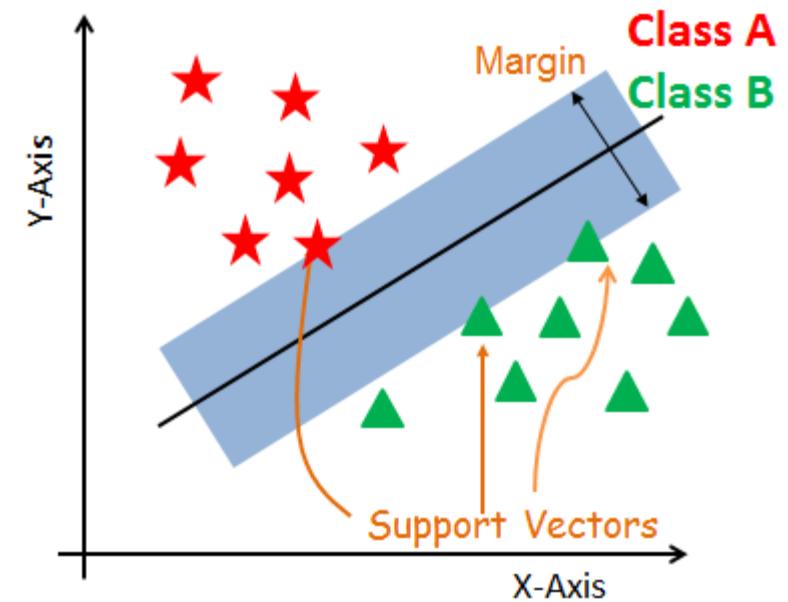
Decision Trees

- **Idea:** find a combination of binary decisions that best classify all datapoints into output classes
- **Pros:** less preprocessing required
Easy to interpret and may give additional insights
- **Con:** unstable results → ensamble methods (random forest)



Support Vector Machines

- **Idea:** find the hyperplane (Line in 2D) that best separates the classes (largest margin)
- Pros: handles highdimensional data well
- Cons: problems with noisy / overlapping data.



Problems

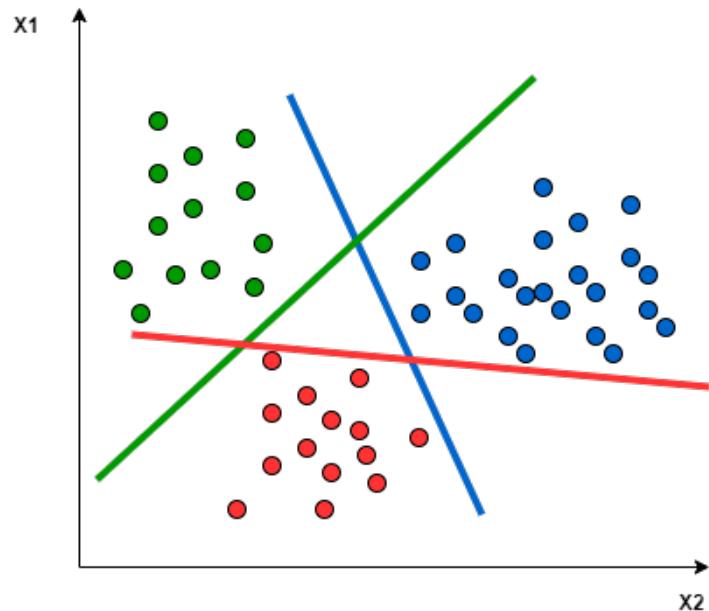
Poor class separation



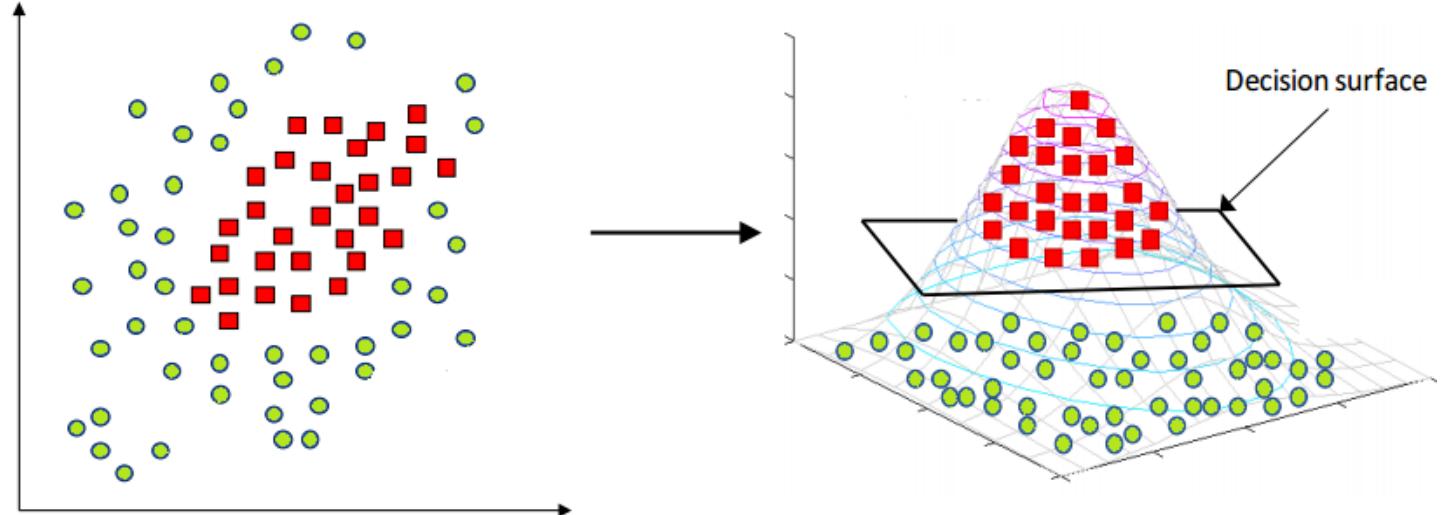
Problems

Advanced

Multi-Class Classification



Feature Transformation



Hands-On

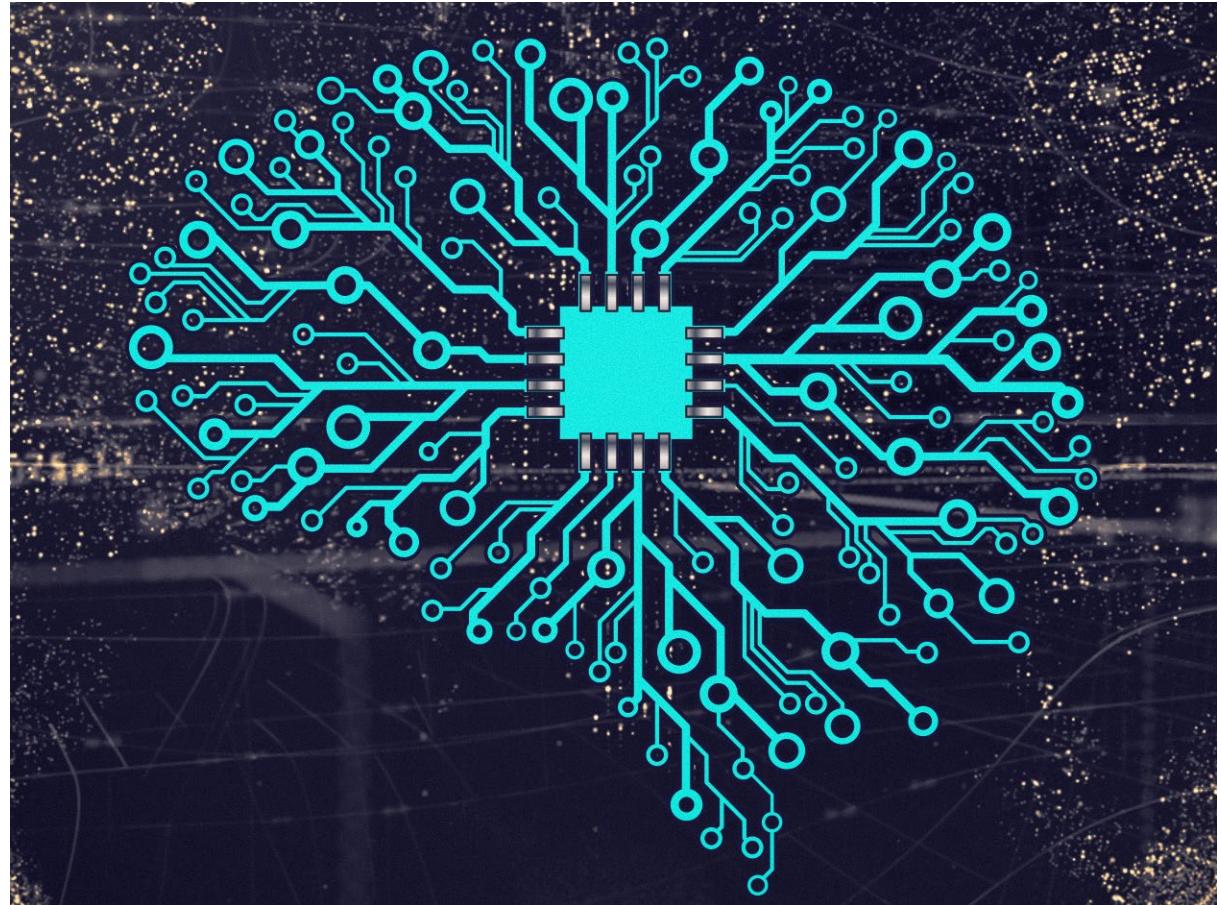
Part 4

1. Implement the KNN Algorithm for a set of 2D datapoints (use the ‘`sklearn make_blobs`’ function to get a random dataset with underlying clusters)
2. Use `seaborn.lmplot` to perform a linear regression
3. Use `sklearn.svm.SVC` to **train** a SVM classifier and plot the data with your trained classifier
4. Use your trained classifier to predict the classes of new datapoints.

Neural Networks

What are they not?

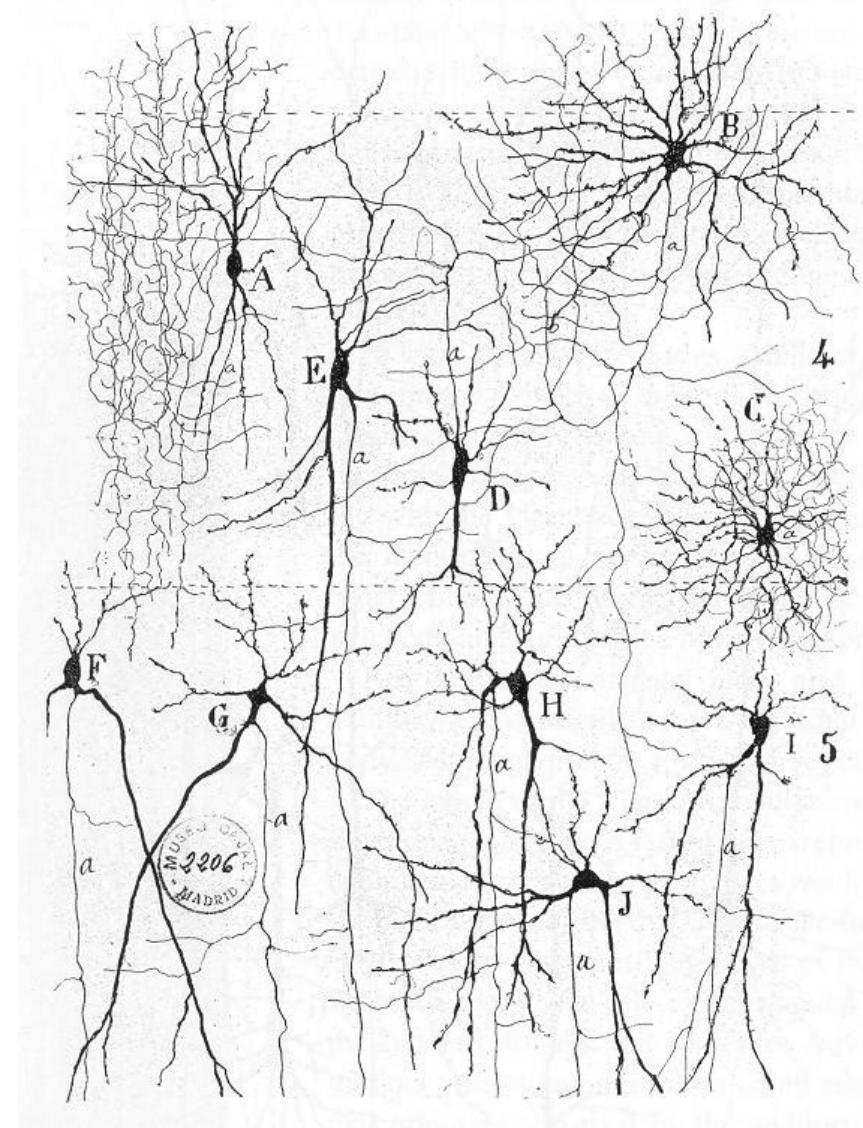
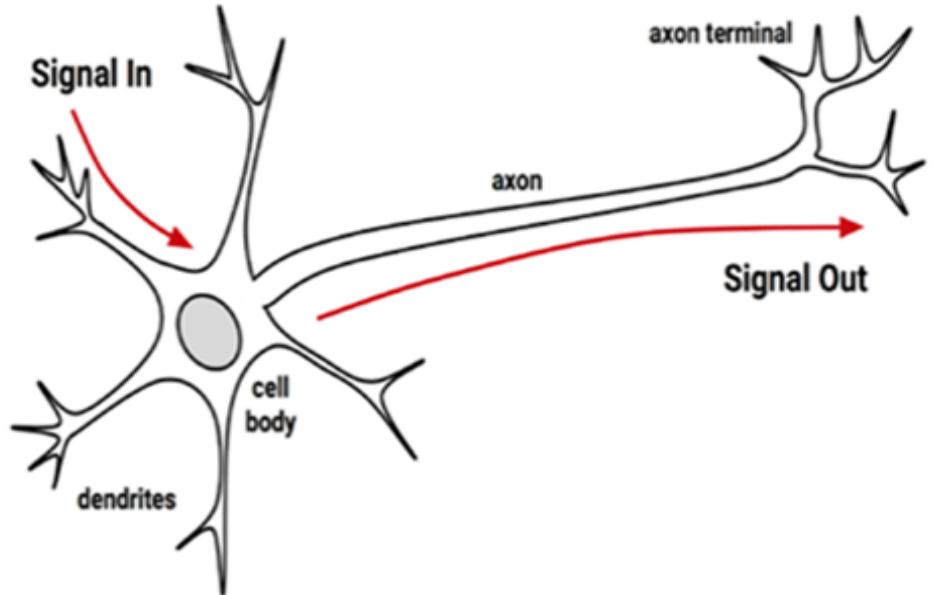
- Artificial Brains
- Super Computers
- Flexible
- “Intelligent”?



Neural Networks

What are they?

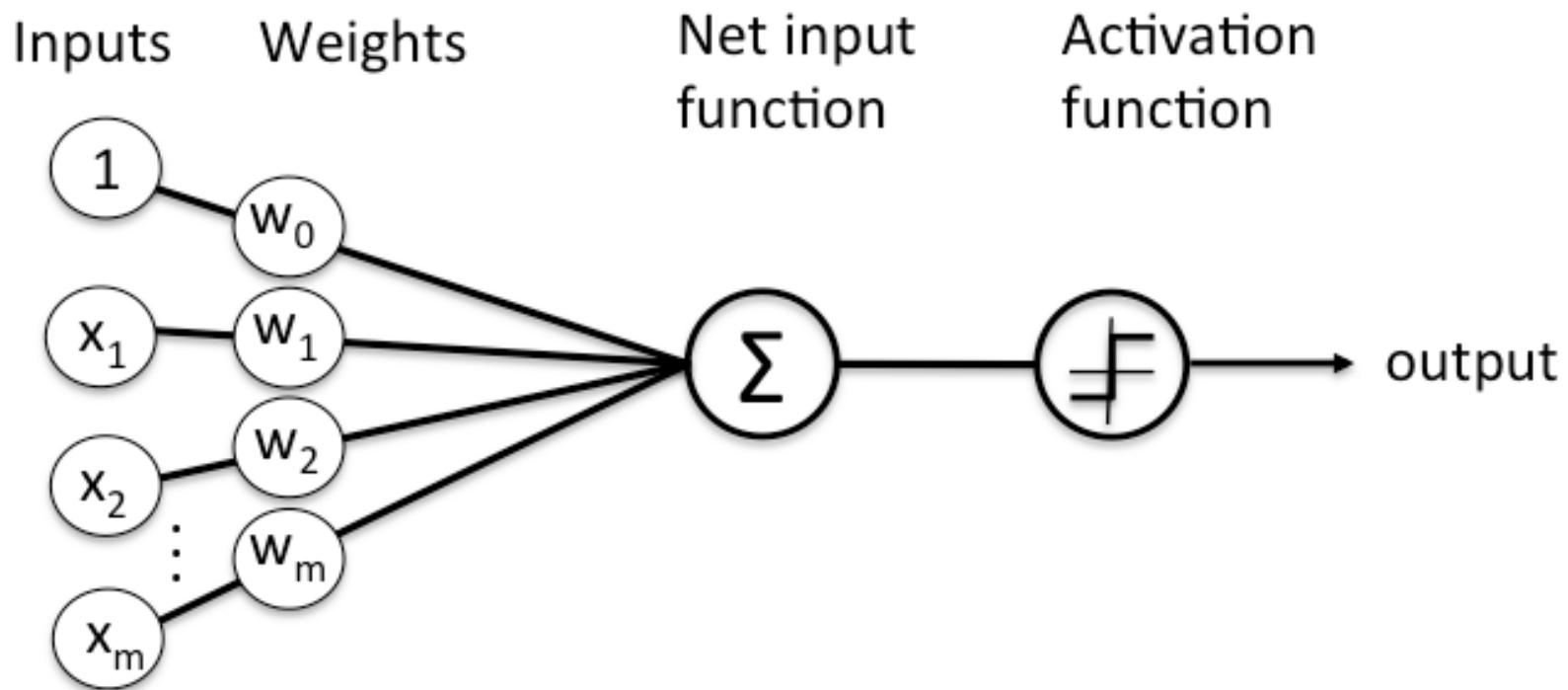
- Inspired by biological “neural networks” aka brains.



Neural Networks

What are they?

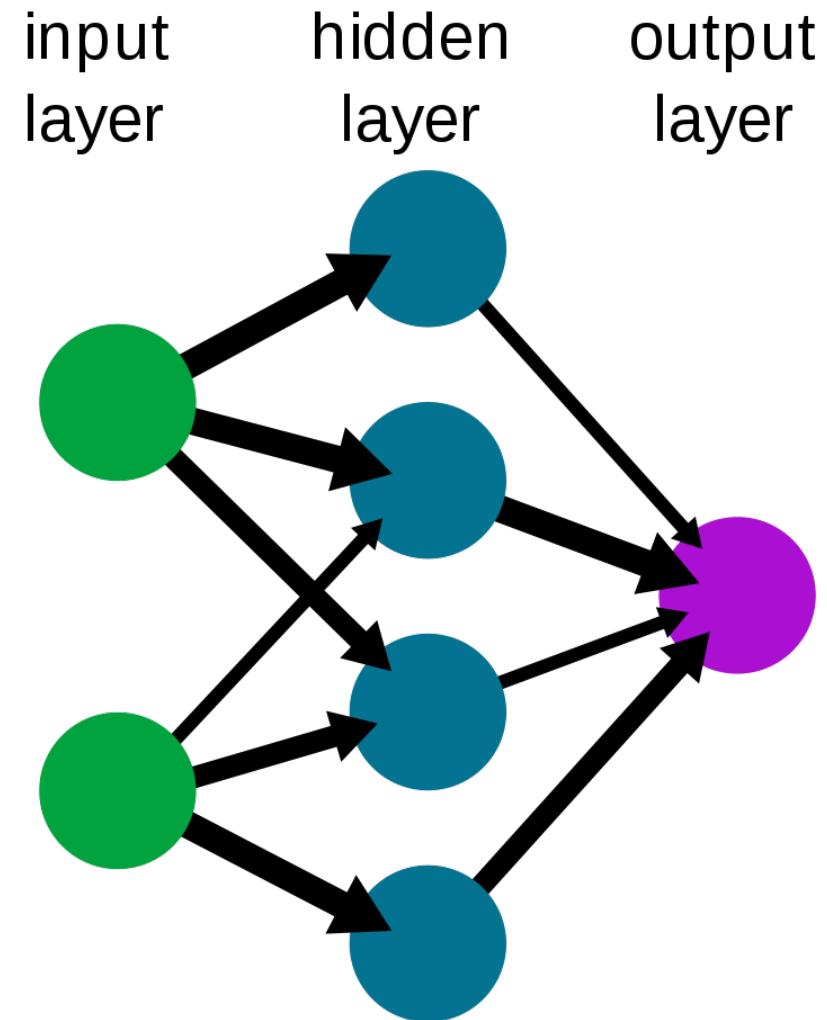
- A network of logical modules or **Nodes**



Neural Networks

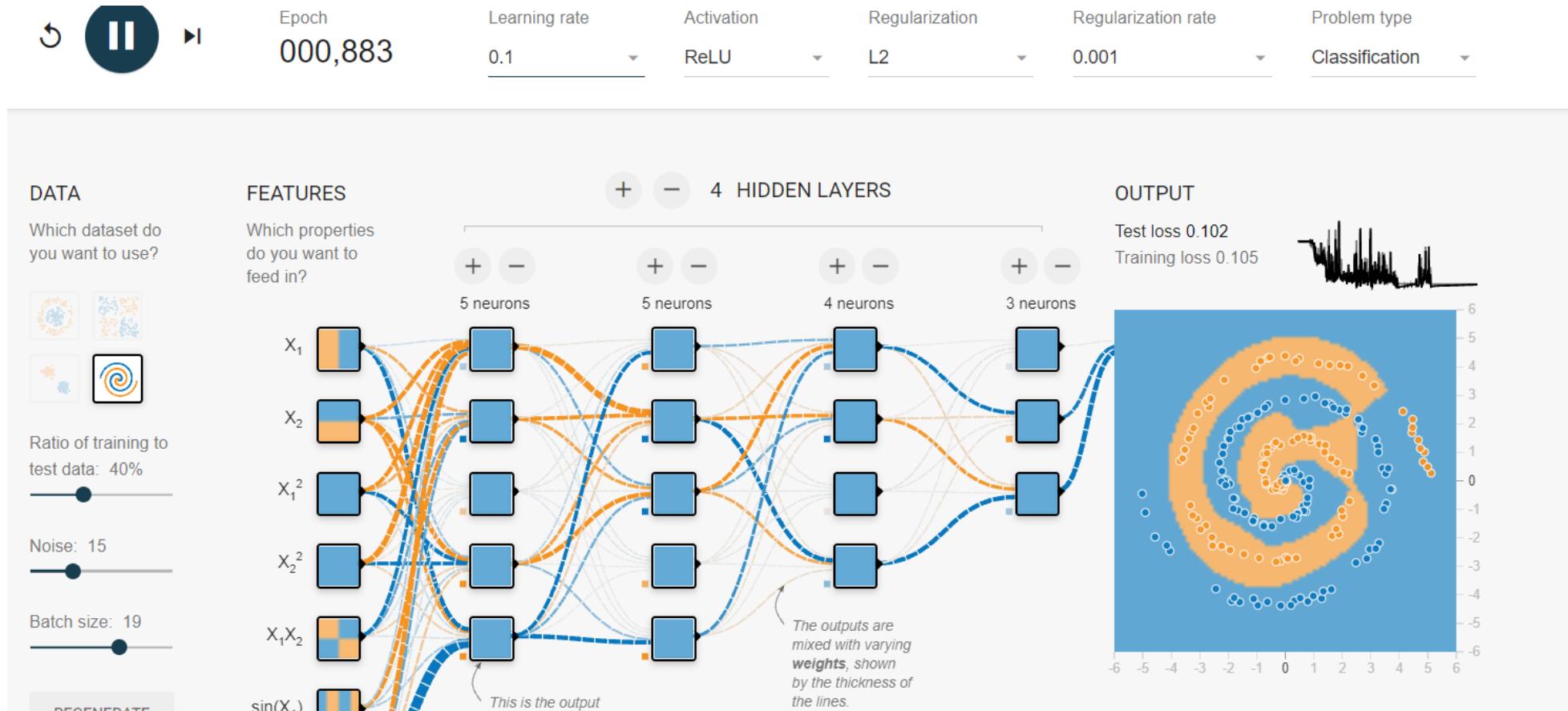
What are they?

- A network of logical modules or **Nodes**, generally arranged in multiple (hidden) layers with an input and output layer
- The **type, arrangement and connections** between the modules lead to complex **emergent patterns**.
- A Neural network is **trained by tuning the connections and parameters** of each node



Neural Networks

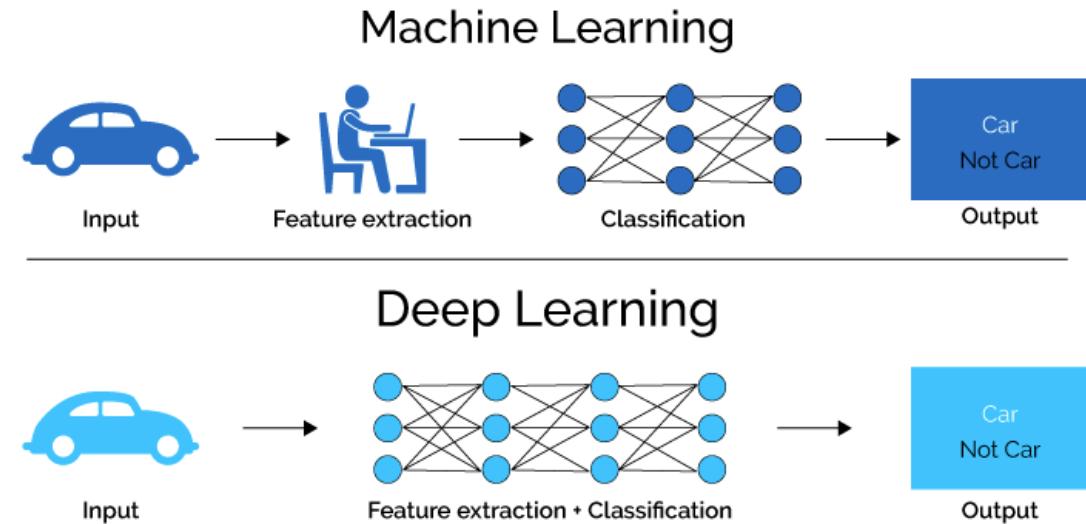
Try it out yourselves



<https://playground.tensorflow.org/>

Deep Neural Nets

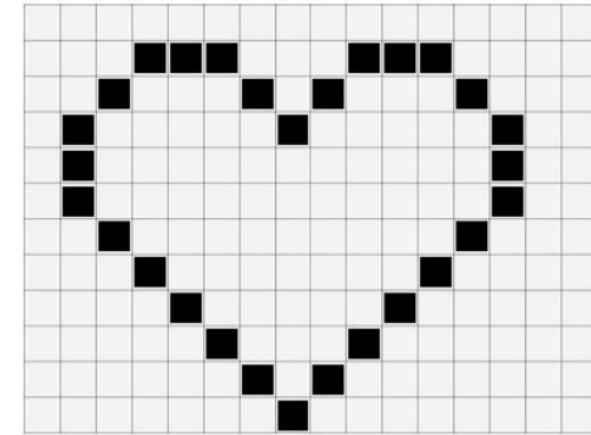
- A subset of machine learning algorithms, which extract their own training features from “raw” data
 - ➔ Requires less human intervention
 - ➔ Requires much more data



Convolutional Neural Net (CNN)

Example of Deep Neural Nets

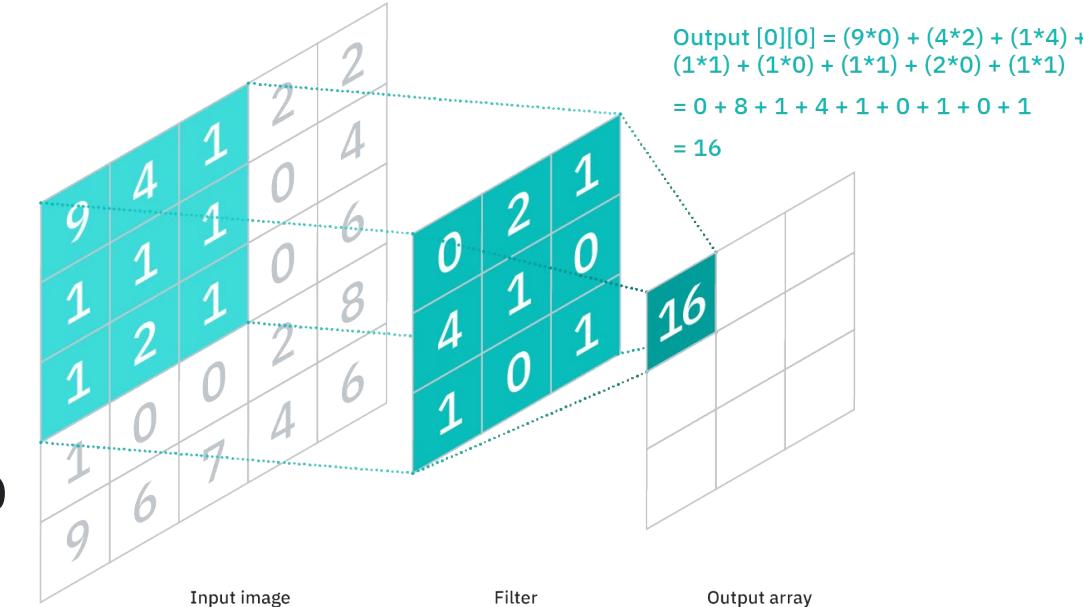
- **Idea:** instead of datapoints use whole matrices (such as images) as input.


$$\begin{bmatrix} 0000000000000000 \\ 0001110001110000 \\ 0010001010001000 \\ 0100000100000100 \\ 0100000000000100 \\ 0100000000000000 \\ 0100000000000000 \\ 0010000000000000 \\ 0001000000000000 \\ 0000100000100000 \\ 0000010001000000 \\ 0000000101000000 \\ 0000000100000000 \end{bmatrix}$$

Convolutional Neural Net (CNN)

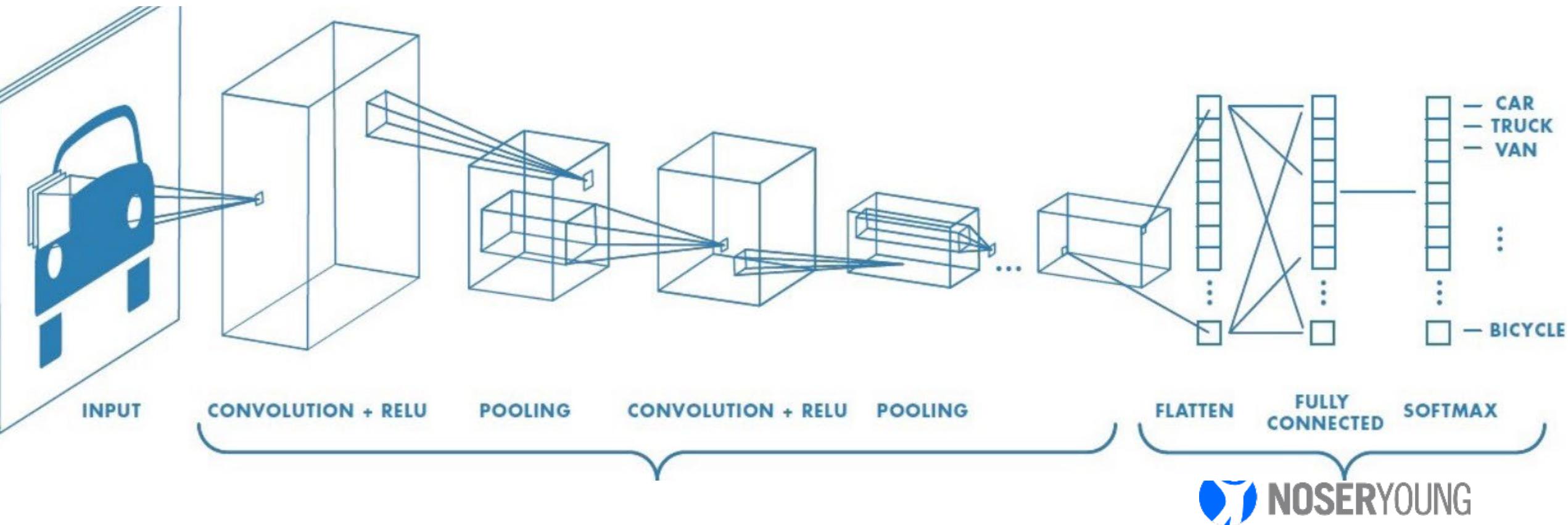
Example of Deep Neural Nets

- **Idea:** instead of datapoints use whole arrays/matrices (such as images) as input.
- Use **Convolution** to extract features from matrix
- **Convolution:** application of a filter to an input. The output is used in subsequent layers
- <https://poloclub.github.io/cnn-explainer/>



Convolutional Neural Net (CNN)

Example of Deep Neural Nets



Neural Networks

Pros & Cons

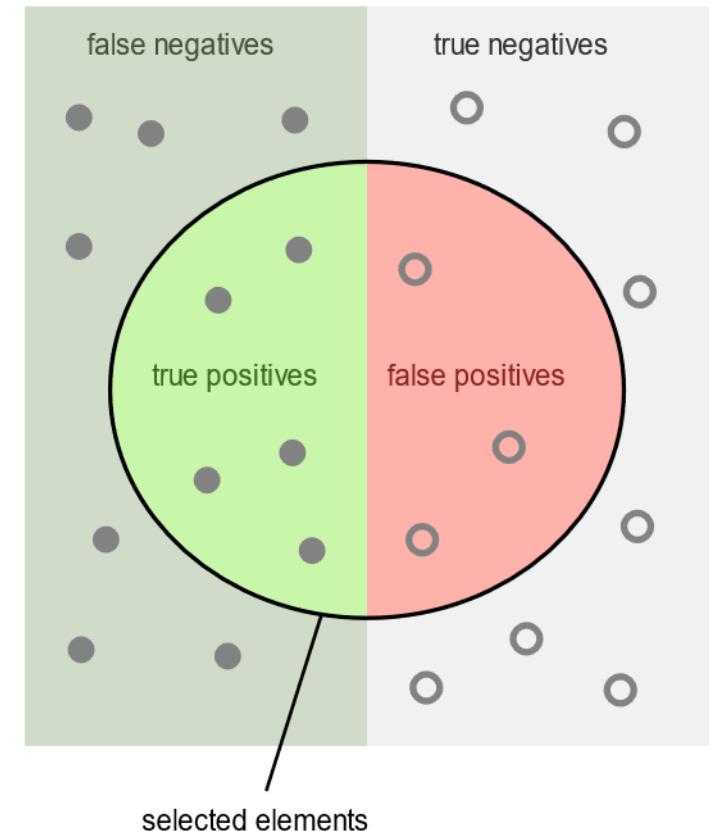
- **Pros:**
 - Extremely flexible
 - Handle complex big datasets (the more data the better)
 - Once trained, predictions are fast
- **Cons:**
 - Model is a Black Box
 - Expensive training, inefficient with CPUs



Metrics

What are they & why should we care?

- **Problem:** We want to quantify the quality & performance of our model
- **Idea:** Predict with the model and compare the model predictions to the true values.
- These metrics can be used to make automatic decisions (e.g. in training, model-selection)



Confusion matrix / error matrix

For Binary Classifier

		Predicted condition	
		Positive (PP)	Negative (PN)
		Total population $= P + N$	
Actual condition	Positive (P)	True positive (TP), hit	False negative (FN), type II error, miss, underestimation
	Negative (N)	False positive (FP), type I error, false alarm, overestimation	True negative (TN), correct rejection

Derived Metrics

Example of binary classifier (positive vs. negative)

- **Recall** = $\frac{TP}{TP + FN}$
 - Probability of detecting cases
- **Precision** = $\frac{TP}{TP + FP}$
 - Probability of correct positive prediction
- **Specificity** = $\frac{TN}{TN + FP}$
 - Probability of correct negative prediction
- **Accuracy** = $\frac{TP + TN}{TP + TN + FP + FN}$
 - Probability of predicting correctly

		Predicted condition	
		Positive (PP)	Negative (PN)
		Total population = P + N	
Actual condition	Positive (P)	True positive (TP), hit	False negative (FN), type II error, miss, underestimation
	Negative (N)	False positive (FP), type I error, false alarm, overestimation	True negative (TN), correct rejection

Challenges in ML applications

Limited Data (Underfitting)

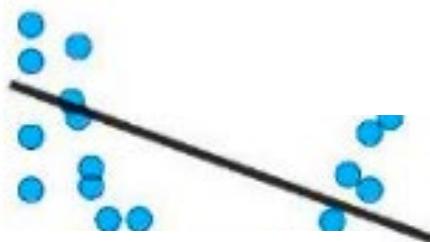
- **Curse of dimensionality:** With increasing dimensionality of the data, much more datapoints are needed to reliably fit a model
- Additionally; the complexer the model, the more data you need!

→ You need **LOTS** of data to train state of the art ML models
→ Obtaining, storing & handling this data is a key challenge of “Big Data” applications

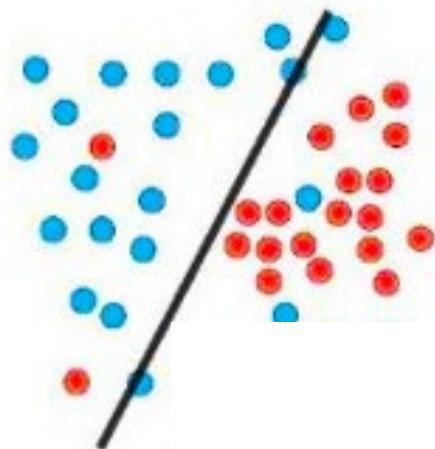
Challenges in ML applications

Underfitting

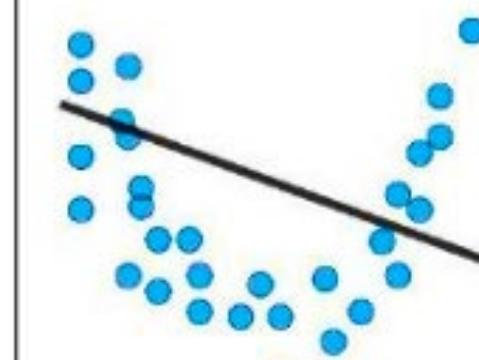
Regression



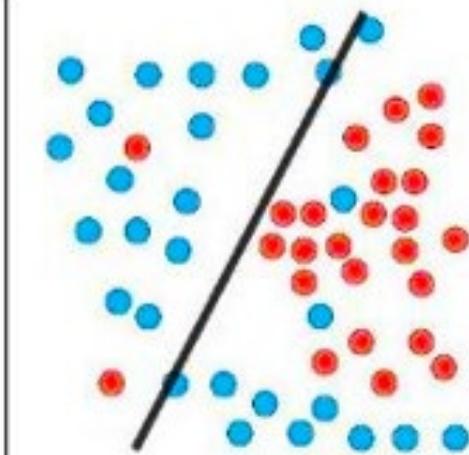
Classification



Regression



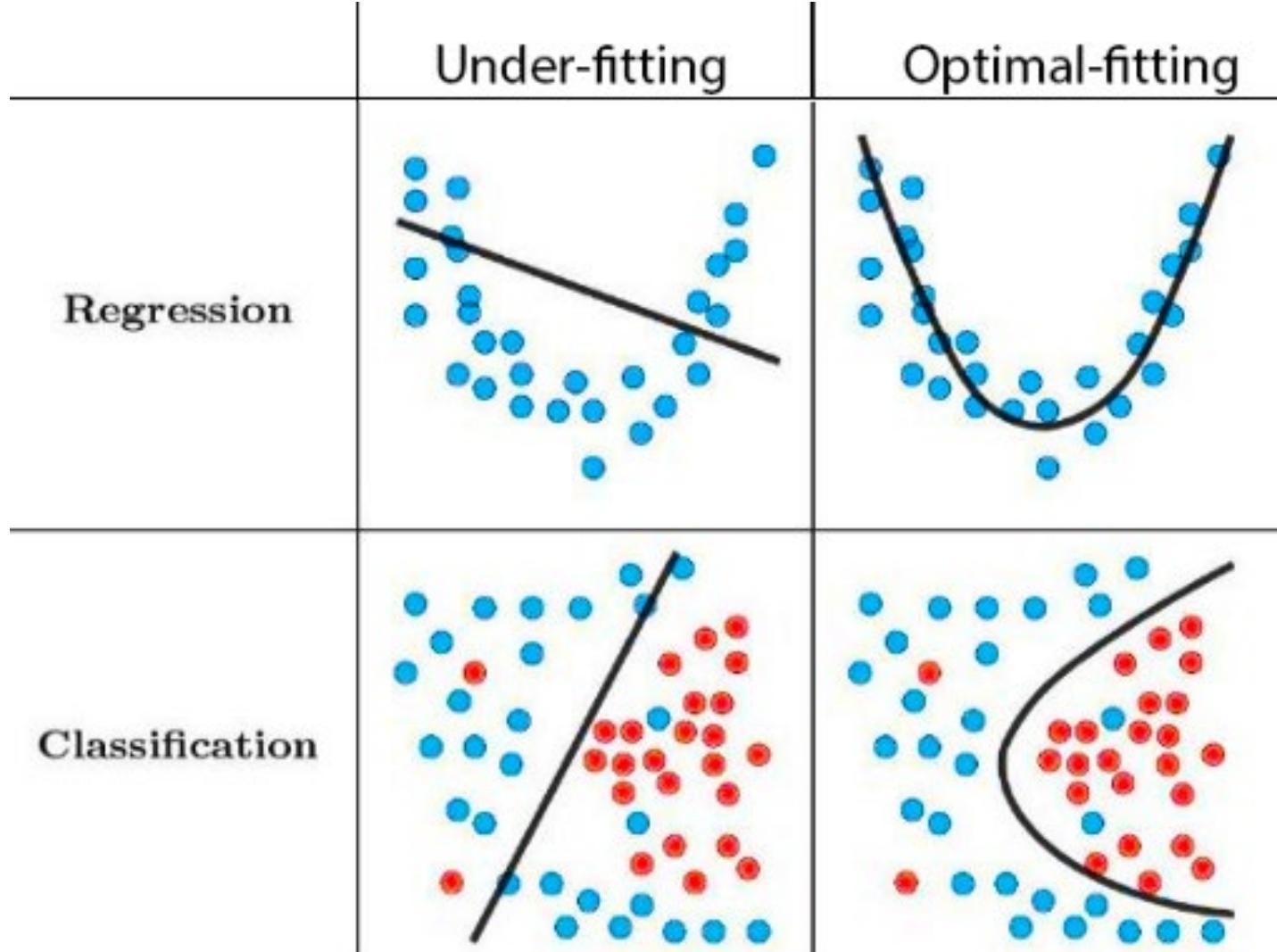
Classification



NOSERYOUNG

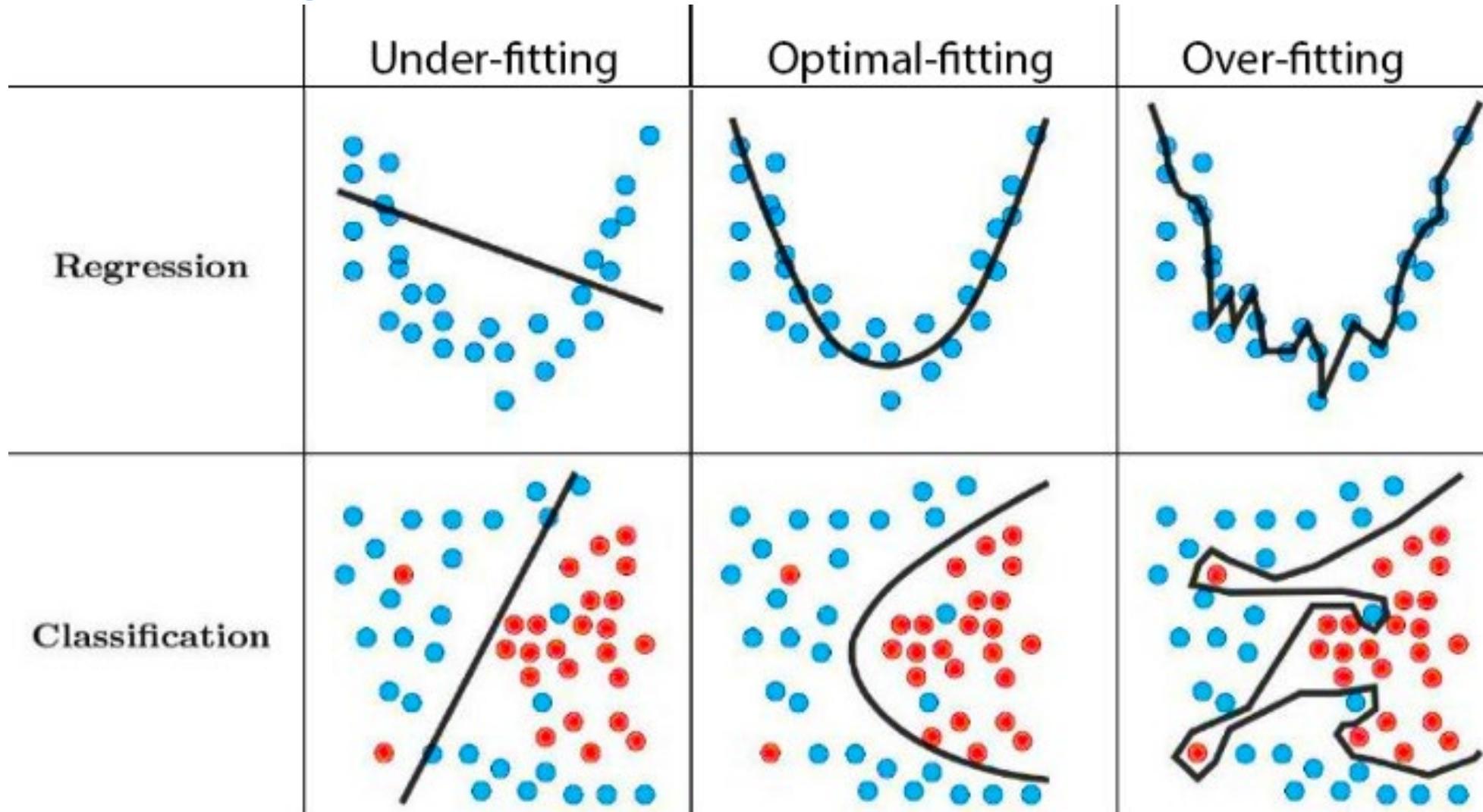
Challenges in ML applications

Overfitting



Challenges in ML applications

Overfitting



ERYOUNG

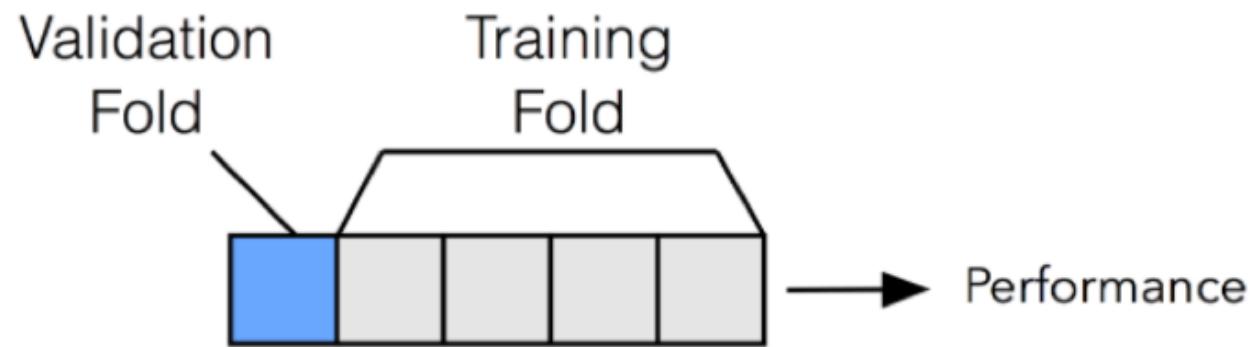
Challenges in ML applications

Overfitting

- Creating a model that fits the training-data **too well!**
 - I.e. it learns patterns that aren't there (Noise)
- Fixes:
 - **Removing noise** from the dataset (e.g. by dimensionality reduction)
 - Manually **limiting the allowable complexity** of the model
 - **Punishing overcomplicated models** by including a complexity metric in the cost function
 - Using test sets or crossvalidation to evaluate your model

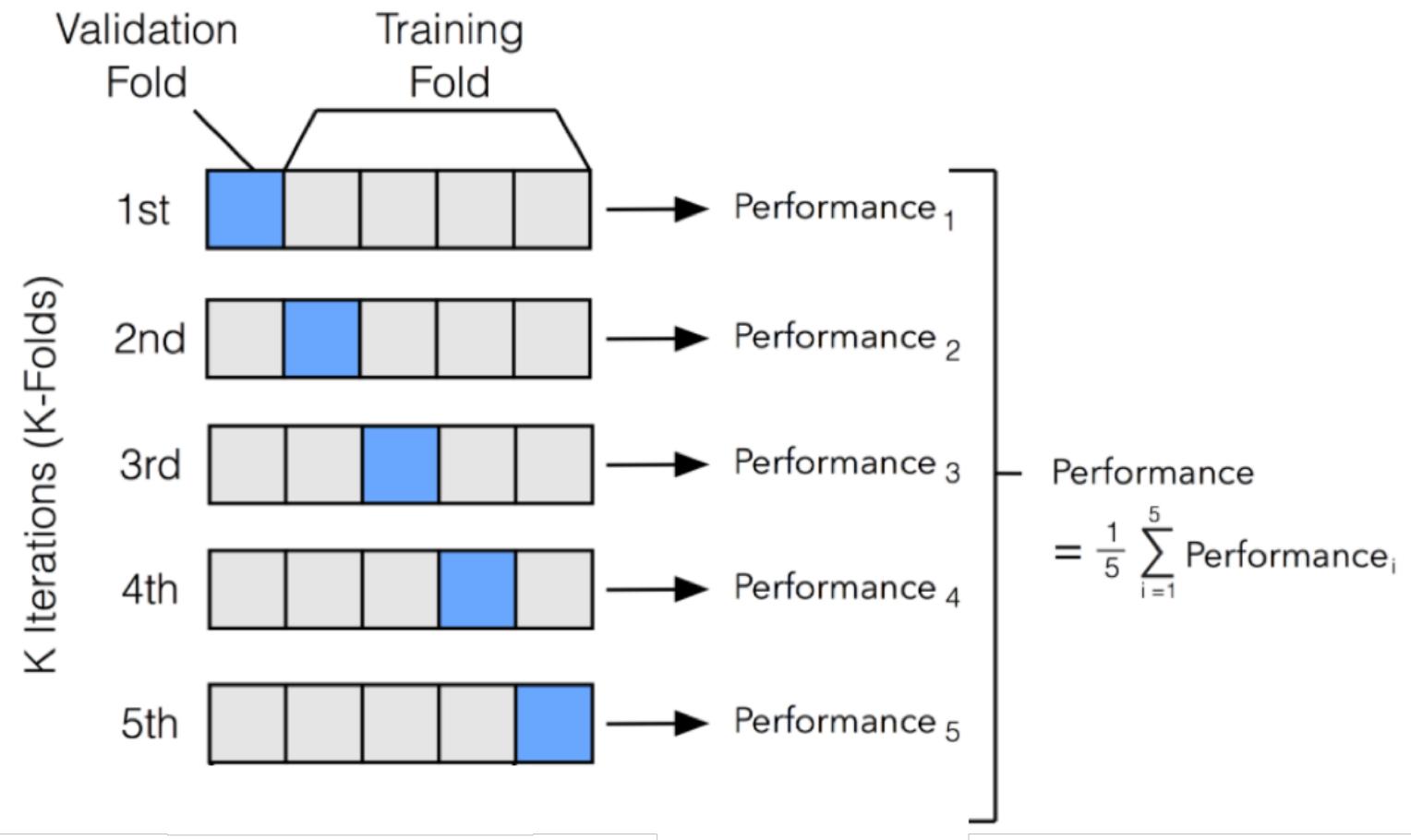
Cross Validation

- **Problem:** selecting a model based on training scores leads to overfitting
- **Idea:** Keep a **subset of the data to validate the model.** Choose the best model based on performance on the validation set
- **This data is never used in training!**



K-fold Cross Validation

- **Problem:** we loose valuable data to the test set
 - **Idea:** train and evaluate multiple times, with different splits, average results
- => Every datapoint



Personendaten im Kontext von Big Data

- «Alle Angaben, die sich auf eine bestimmte oder bestimmbare Person beziehen» (Art. 3 Bst. a DSG)
- **Problem:** De-Anonymisierung bzw. Re-Identifikation
 - <https://amiunique.org/fp>
 - Set von Einkäufe mit gleicher Kreditkarte
 - Sets von Webrequests von gleicher IP Adresse

Personendaten im Kontext von Big Data

- «Besonders schützenswerte Daten»:
 - Weitreichende Einschränkungen für Gebrauch solcher Daten
- Quantität, Qualität und geplanter Gebrauch spielen eine Rolle

Schutzmaßnahmen

1. Vertraulichkeit

- Authentifizierung und Authorisierung
- **Trennungsgebot:** gewährleisten, dass zu unterschiedlichen Zwecken erhobene Daten getrennt verarbeitet werden können

2. Integrität

- Weitergabe-, Verarbeitungs- und Dokumentationskontrolle
- Resilienz von Systemen/ Diensten

3. Regelmäßige Evaluierung der Maßnahmen

Project Work

- Choose a Dataset and a question you want to answer
 - Classification
 - Clustering
 - Prediction
- Choose a fitting ML method and metric to solve your problem
- Incorporate techniques such as data pre-processing, cross-validation and data visualization in your solution.
- **Present your solutions**

Further Reading / Watching

- Basics:
 - <https://www.youtube.com/watch?v=h0e2HAPTGF4>
 - https://www.youtube.com/playlist?list=PL8dPuuaLjXtO65LeD2p4_Sb5XQ51par_b
- More Math:
https://www.youtube.com/playlist?list=PLZHQBObOWTQDNU6R1_67000Dx_ZCJB-3pi
- Industry Trends and Opinions
<https://towardsdatascience.com/>
- More In-Depth Tutorial in GoogleColab
<https://developers.google.com/machine-learning/crash-course>