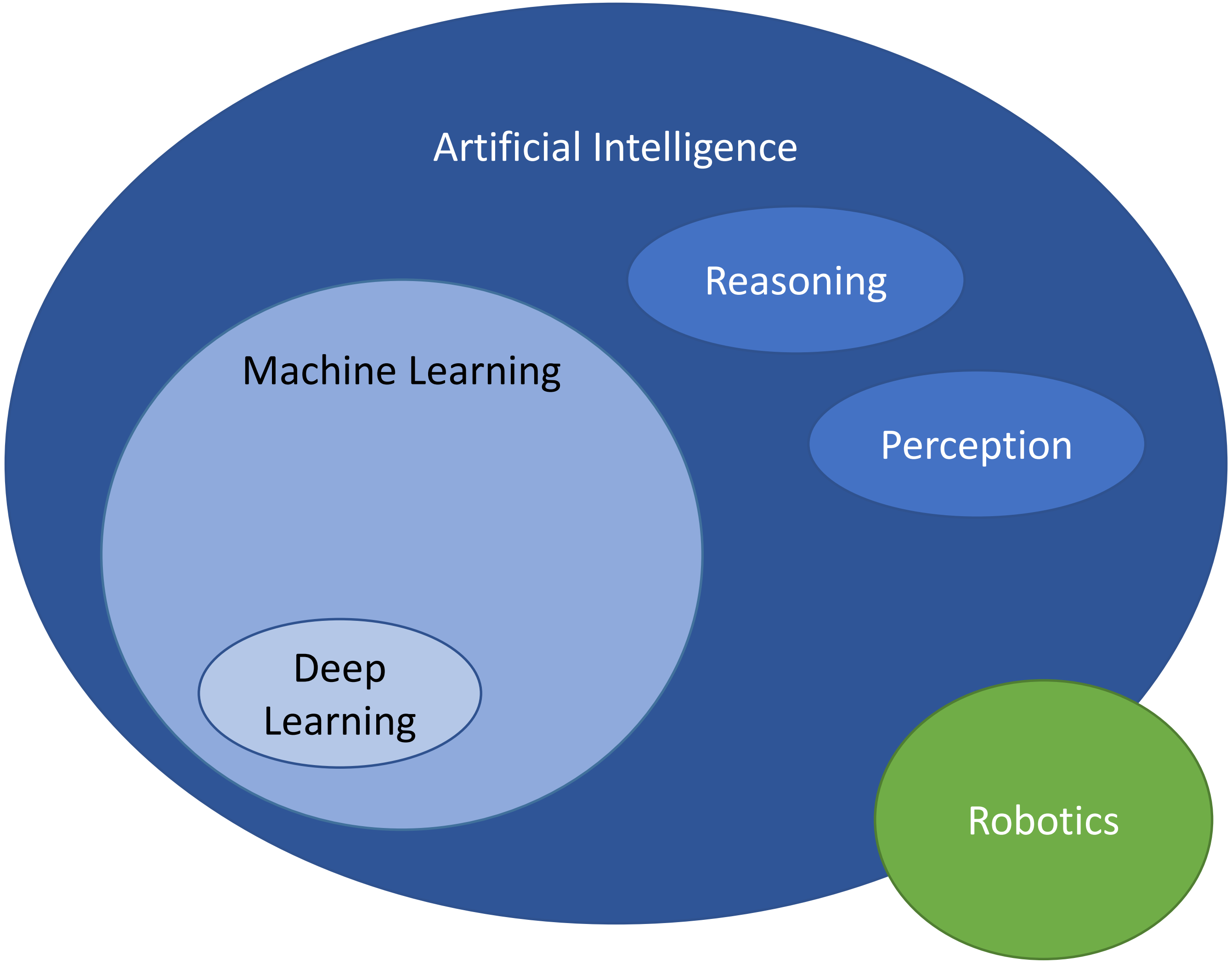
Machine Learning

1. Introduction

Nicolas Gartner

Artificial Intelligence



Artificial Intelligence

Reasoning

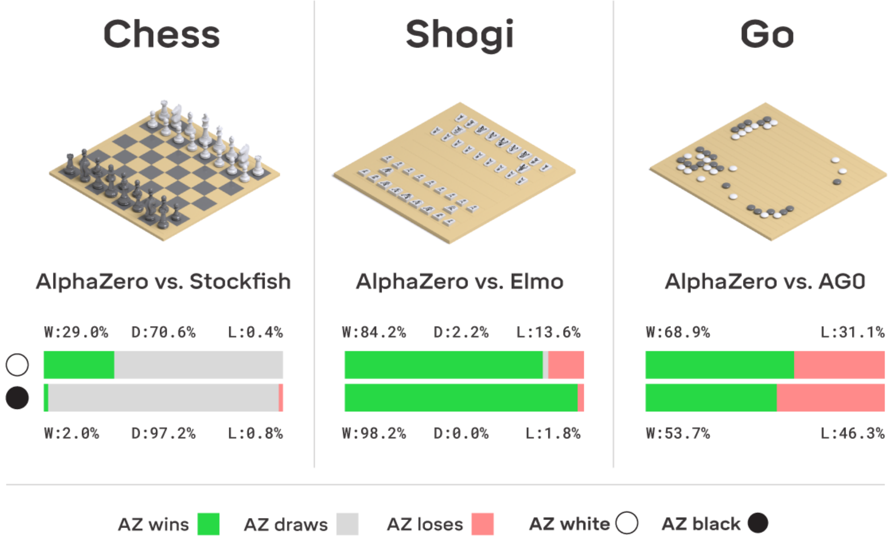
Machine Learning

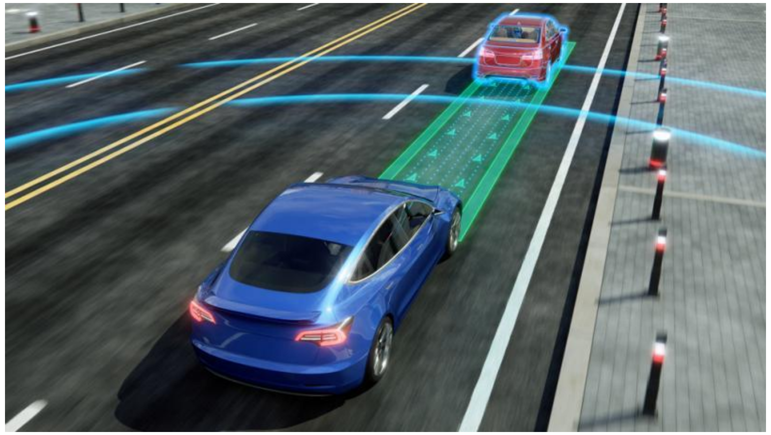
Perception

Deep Learning

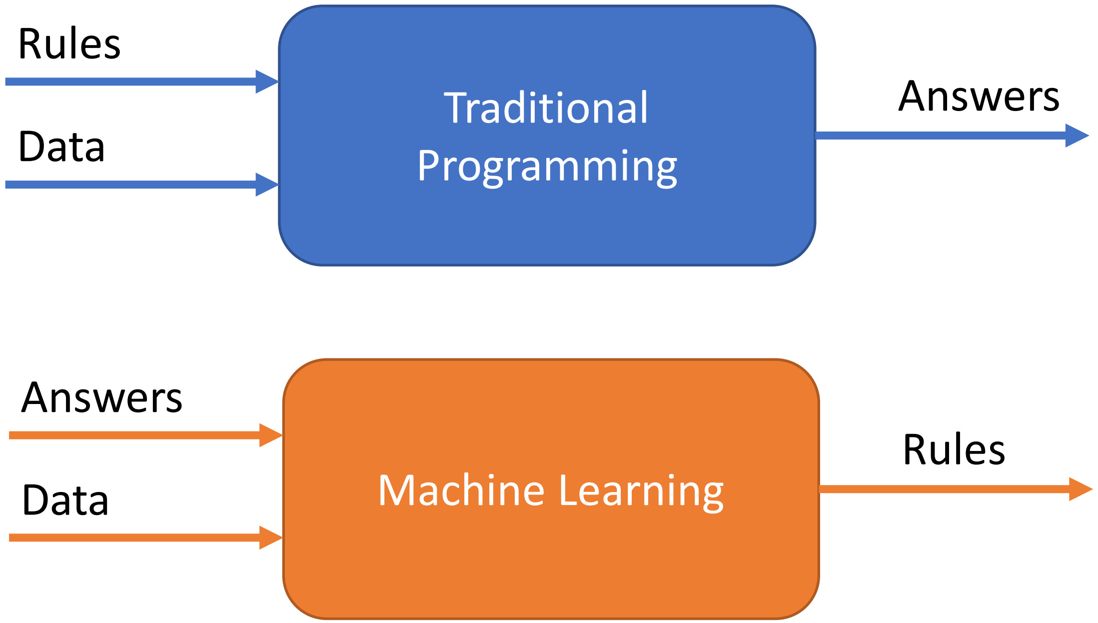
Robotics

AlphaZero





Traditional Programming vs Machine Learning



Rules

Data

Traditional Programming

Answers

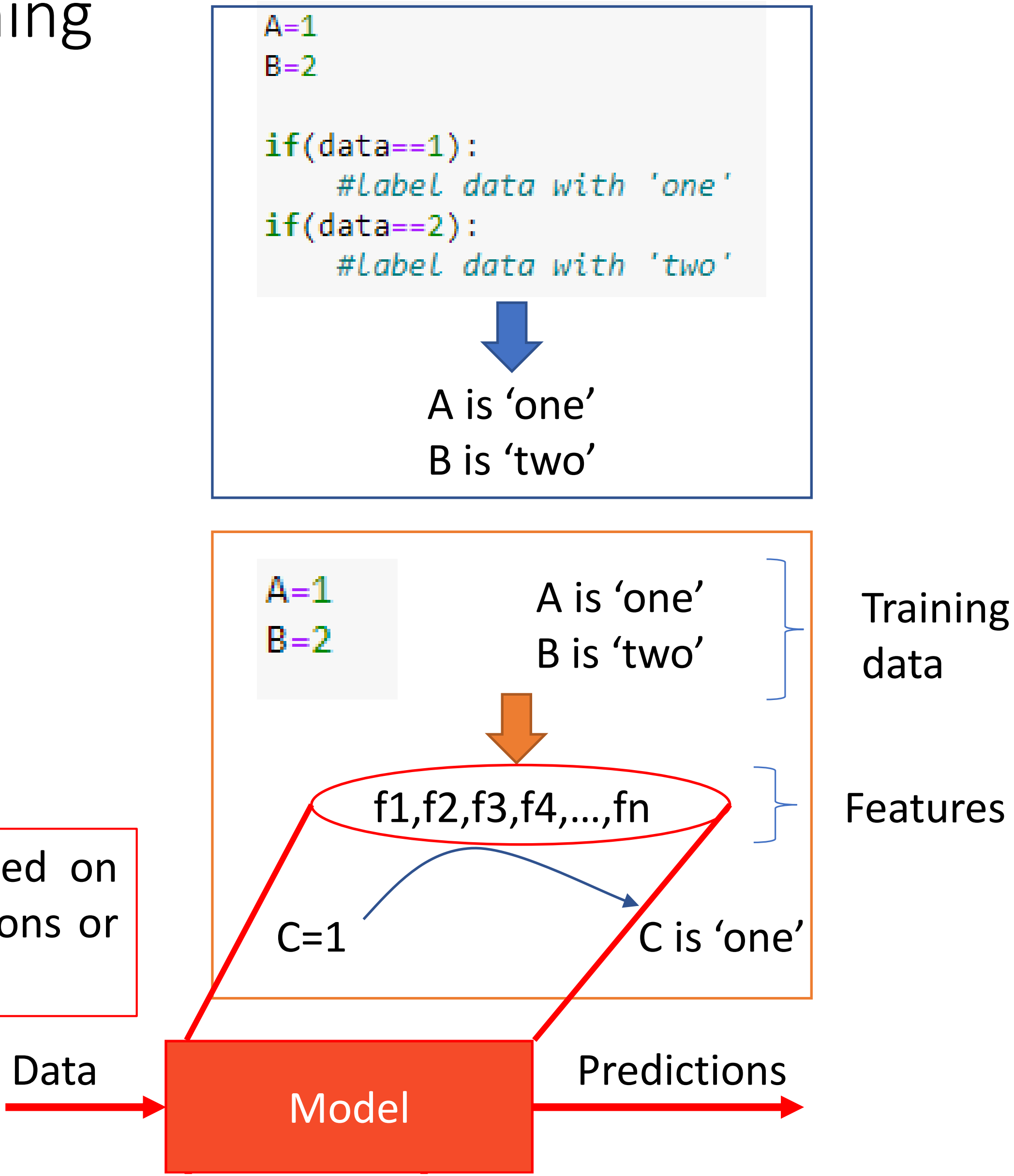
Answers

Data

Machine Learning

Rules

Machine learning algorithms build a mathematical model based on sample data, known as training data, in order to make predictions or decisions without being explicitly programmed to do so



```
A=1
B=2

if(data==1):
    #label data with 'one'
if(data==2):
    #label data with 'two'
```

A is ‘one’
B is ‘two’

```
A=1
B=2
```

A is ‘one’
B is ‘two’

Training data

f1,f2,f3,f4,…,fn

Features

C=1

C is ‘one’

Data

Model

Predictions

A simple example : guessing linear functions

X (input)	-2	-1	0	1	2	3	4
Y (output)	-5	-3	-1	1	3	5	?

What is the relationship between input and output ?

$$Y = 2X - 1$$

How did your brain figure that out ?

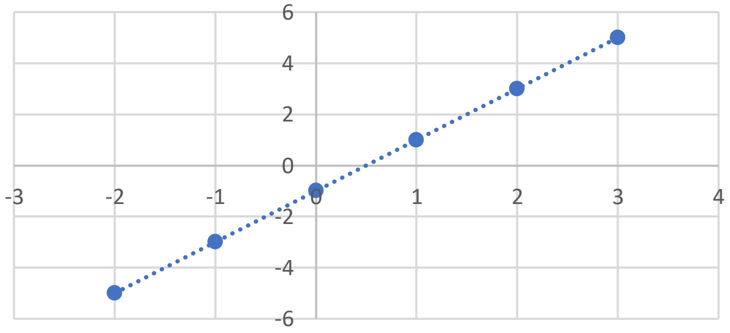
The Machine Learning process

- Define a feature that would guess a linear function using linear regression

$$Y = \alpha_1 X - \alpha_0$$

With α_1 and α_2 the values we want to guess

Same as : Add a trend line to a chart



What is the next number ?

Linear regression model example

The answer of our model

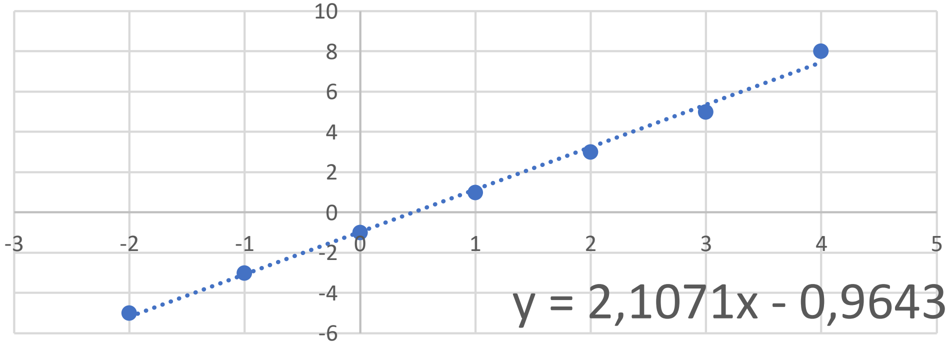
X (input)	-2	-1	0	1	2	3	4
Y (output)	-5	-3	-1	1	3	5	7

7 would likely be the answer of our model … but !

- It might be something very close to 7 (6.99998)
- This is a prediction. What if that was not the right result ?

A new Dataset → New model

X (input)	-2	-1	0	1	2	3	4
Y (output)	-5	-3	-1	1	3	5	8



A new model :
Every new data input can change the model

Here the model is the best answer to the input dataset and allows to make predictions with a certain reliability

Rock, Paper, Scissors : a very advanced example

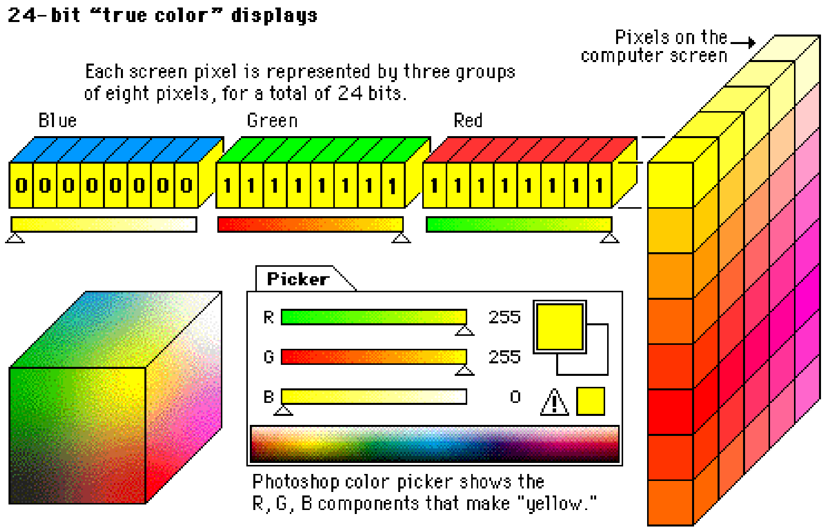
Rock

Paper

Cissor

https://www.tensorflow.org/datasets/catalog/rock_paper_scissors Laurence Moroney

How numerical images are made ?



For each pixel of your picture, values saved:

- Red (0-255)
- Green (0-255)
- Blue (0-255)

The mix defines the color displayed

Why 0-255 ?

- Binary : 11111111 (8 bit)

$$255 = 1 * 128 + 1 * 64 + 1 * 32 + 1 * 16 + 1 * 8 + 1 * 4 + 1 * 2 + 1$$

$$255 = 1 * 2^7 + 1 * 2^6 + 1 * 2^5 + 1 * 2^4 + 1 * 2^3 + 1 * 2^2 + 1 * 2^1 + 1 * 2^0$$

- Decimal : 255

$$255 = 2 * 100 + 5 * 10 + 5$$

$$255 = 2 * 10^2 + 5 * 10^1 + 5 * 10^0$$

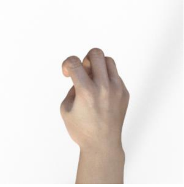
- Hexadecimal : FF (0 1 2 3 4 5 6 7 8 9 A B C D E F)

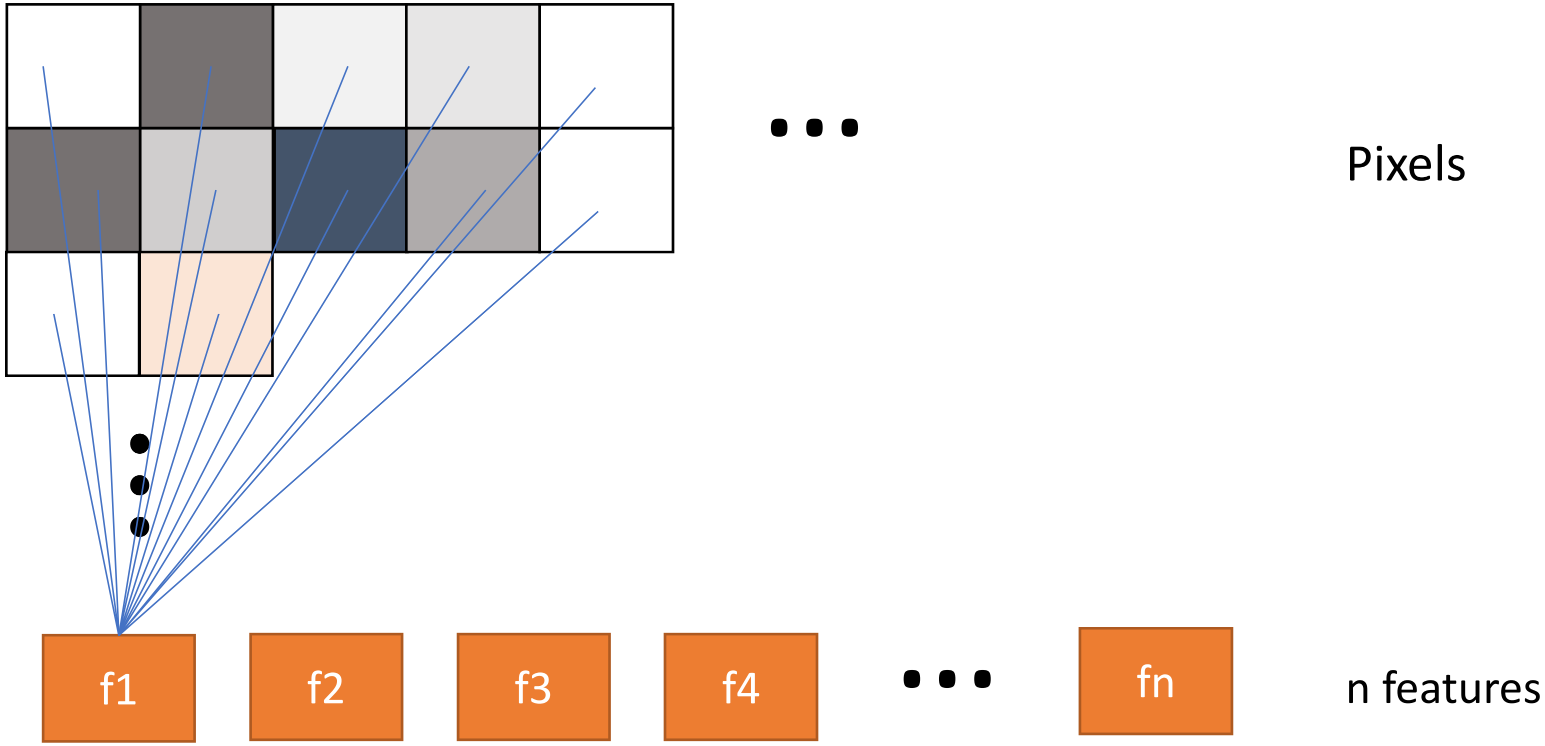
$$FF = F(15) * 16^1 + F(15) * 16^0$$

How to deal with that much data ?
4608 x 3456 (24 bits)

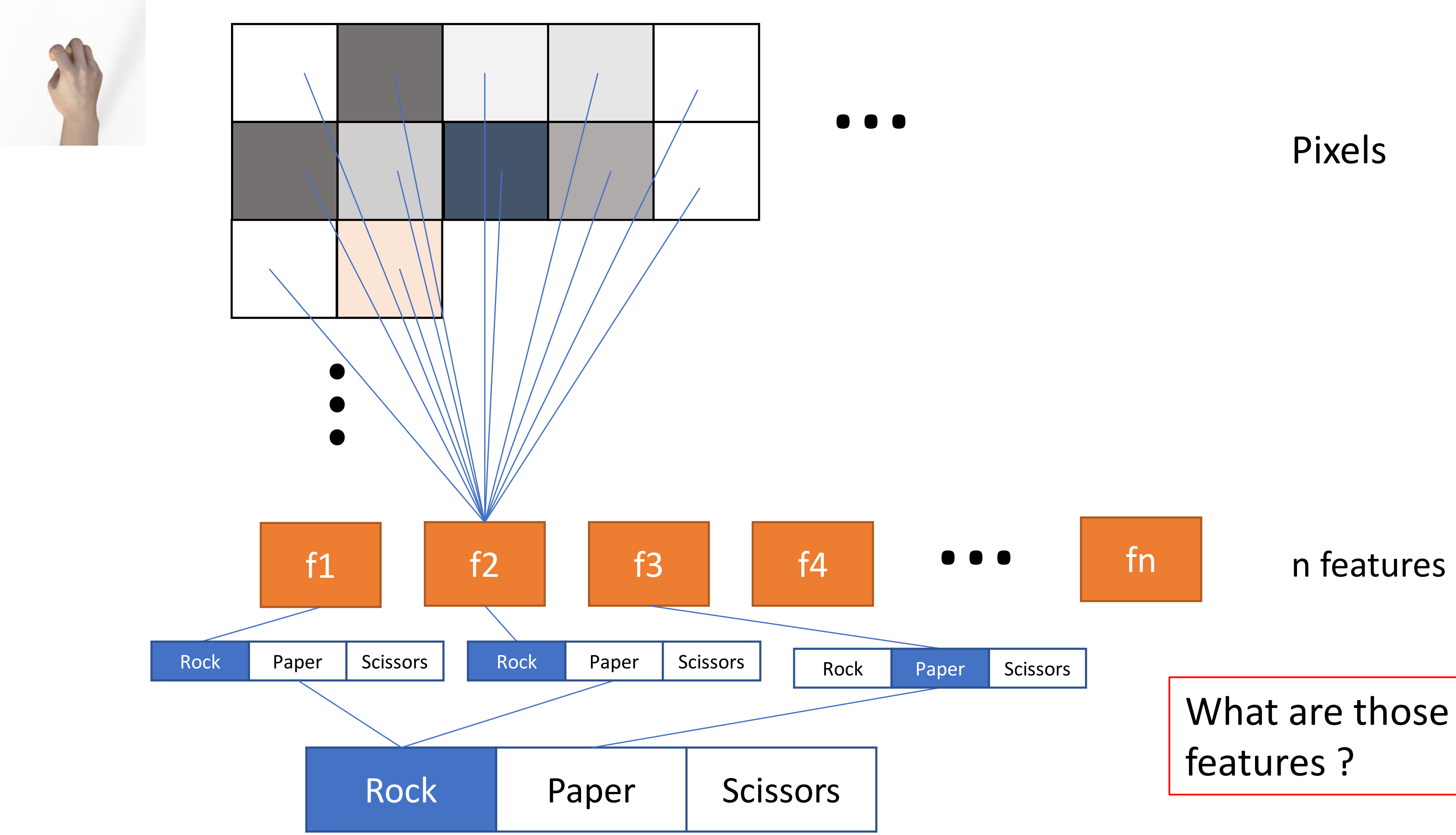
Images are very big amout of data

Machine Learning for the Rock, Paper, Scissors problem





Machine Learning for the Rock, Paper, Scissors problem



What are those features ?

Convolution feature

A convolution is a filter that passes over an image, processes it, and extracts the important features.

Grayscale pictures

Convolution filter matrix

-1	0	1
-2	0	2
-1	0	1

Vertical lines are highlighted

255	118	242
118	208	0
255	166	49

Current pixel value is 208.

Displayed with neighbor pixel around

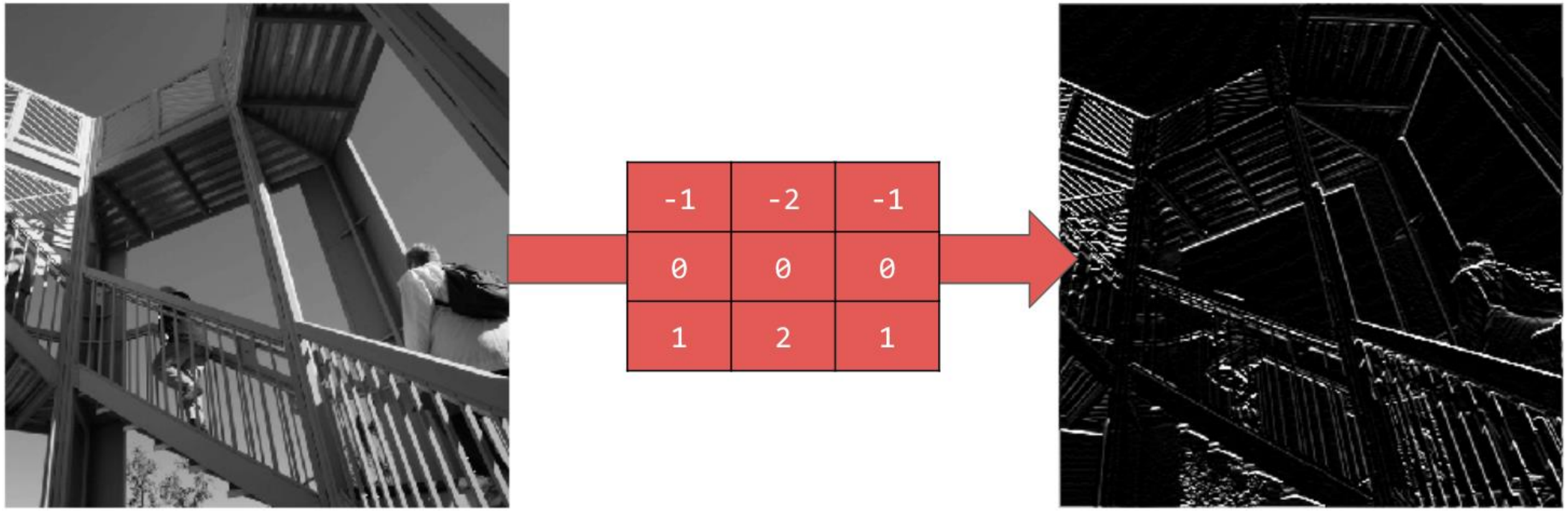
Convolution is changing the 208 pixel value to :

$$-1 * 255 + 0 * 118 + 1 * 242 + (-2) * 118 + 0 * 208 + 2 * 0 + (-1) * 255 + 0 * 166 + 1 * 49 = -455\ (0)$$

Convolution allows to highlight features in the image

Convolution feature

Another example of convolution filter :

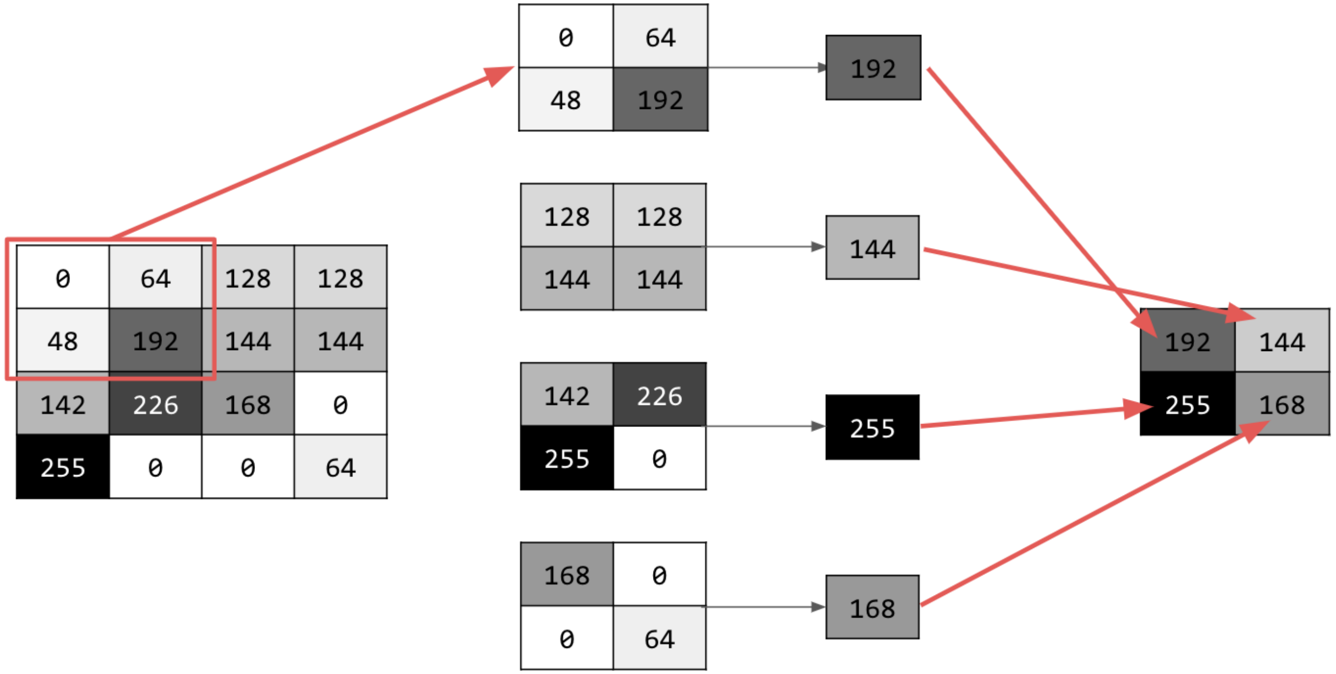


-1	-2	-1
0	0	0
1	2	1

This one highlights the horizontal lines

Pooling feature

Pooling layers provide an approach to down sampling feature maps by summarizing the presence of features in patches of the feature map



0	64	128	128
48	192	144	144
142	226	168	0
255	0	0	64

0	64
48	192

192

128	128
144	144

144

142	226
255	0

255

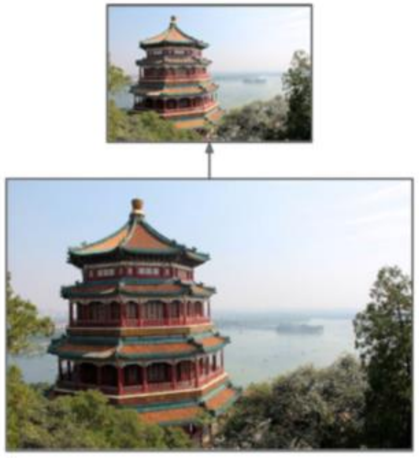
168	0
0	64

168

192	144
255	168

Example of a “maximum Pooling”

A new and smaller image is created, keeping most important elements

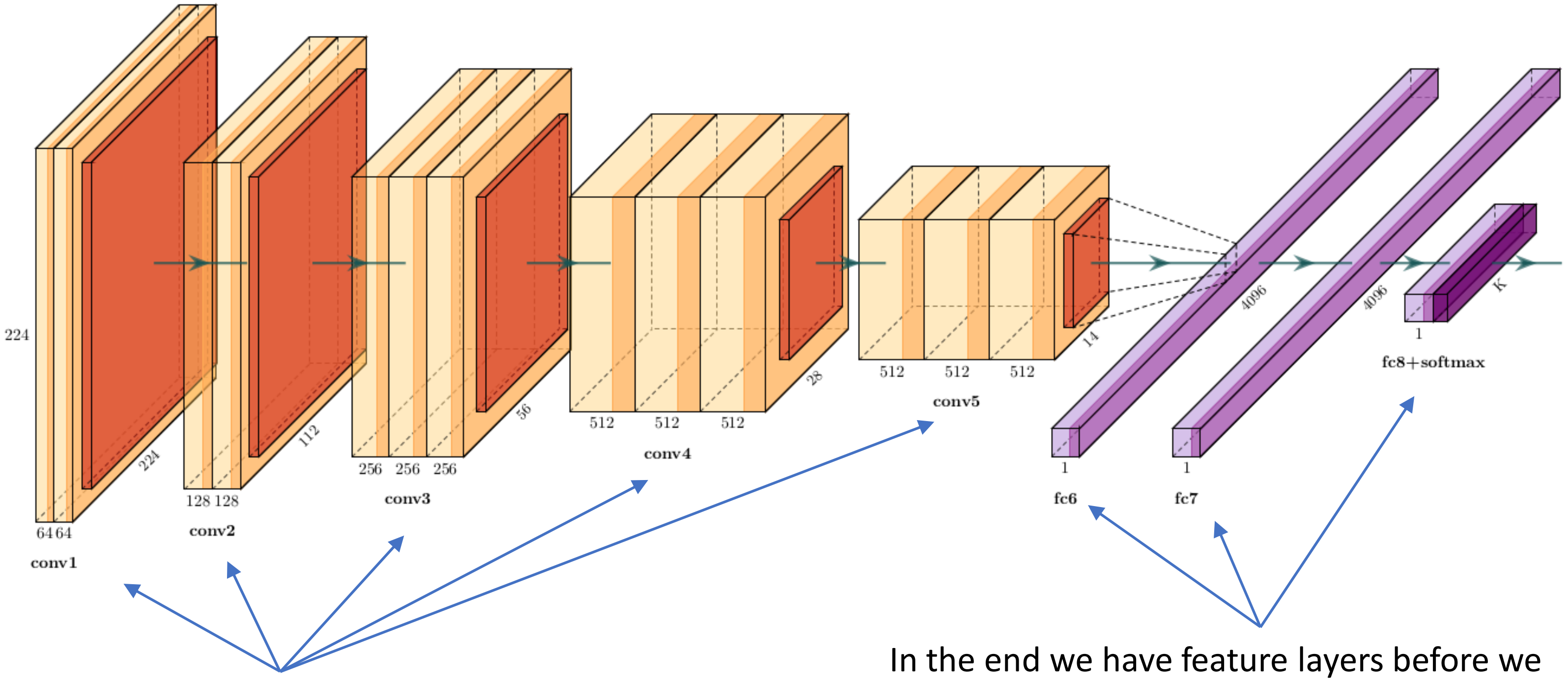


How to make the machine learn ?

Build a succession of randomly initialized features that would optimize themselves according to the objective using numerical methods

Conv. matrix

?	?	?
?	?	?
?	?	?

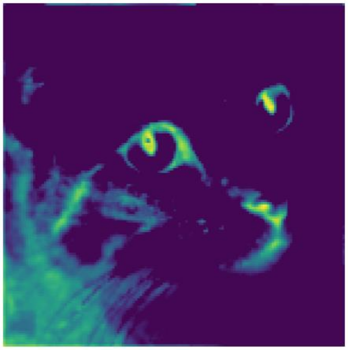


Each convolution layer is <u>quite</u> randomly initialized

In the end we have feature layers before we finally get the result

An example of Neural Network

Feature map



Each features (convolution, pooling) gets optimized -> resulting filters can be looked at

The machine learning process

		Examples
Problem definition	Understand and describe the problem that must be solved	How to make a Rock, paper, Scissor game ?
Analyze and/or gather data	Understand the information available that will be given as input and organize it	A collection of hand images
Prepare data	Discover and expose the structure of the dataset	Decide that you will use pixel data
Evaluate algorithms	Develop an automated test framework and baseline accuracy from which to improve	Define that you will use convolution and pooling and that you want the best accuracy in guessing rock, paper or scissors. Make it learn
Improve model / results	Use the developed algorithms to improve according to accuracy or to make guesses	Run the machine learning algorithms over the dataset and make predictions

The machine learning process

Tools

Problem definition

- Understand and describe the problem that must be solved

Analyze and/or gather data

- Understand the information available that will be given as input and organize it

Prepare data

- Discover and expose the structure of the dataset

Pandas, numpy, matplotlib, iPython, Excel

Evaluate algorithms

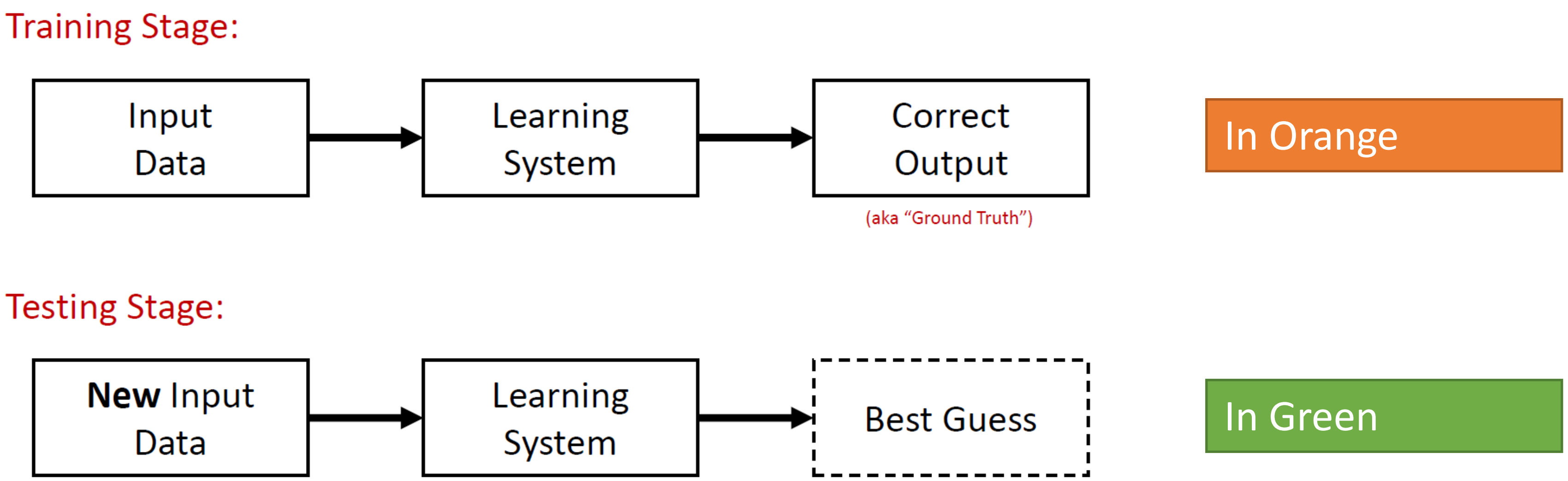
- Develop an automated test framework and baseline accuracy from which to improve

Improve model / results

- Use the developed algorithms to improve according to accuracy or to make guesses

Scikit-learn, Theano, Caffe, Tensorflow (Google), Torch – pyTorch (Facebook)

Stages of a machine learning algorithm



Training Stage:

Input Data → Learning System → Correct Output

(aka "Ground Truth")

In Orange

Testing Stage:

New Input Data → Learning System → Best Guess

In Green

Problem definition → Analyze and/or gather data → Prepare data → Evaluate algorithms → Improve model / results

Machine learning: problem setting

In general, a learning problem is :

- a set of n samples of data
- tries to predict properties of unknown data

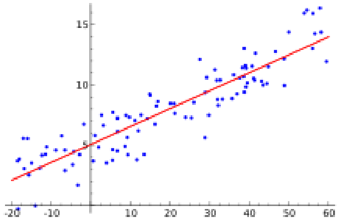
Learning problems falls into two main categories:

Supervised learning

Data comes with additional attributes that we want to predict

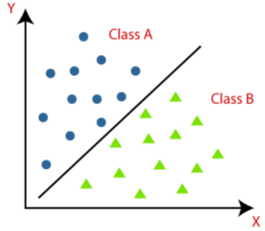
Regression

Desired output consists of one or more continuous variables



Classification

Samples belong to different classes and we want to learn from already labeled data how to predict the class of unlabeled data

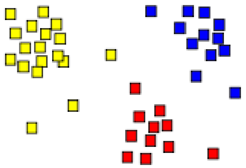


Unsupervised learning

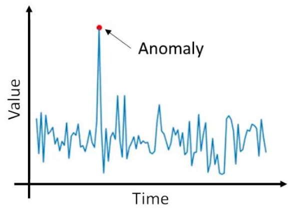
Training data consists of a set of input vectors x without any corresponding target value

Examples : (1) Clustering, (2) Anomaly detection, (3) Unsupervised Neural Networks, and (4) Approaches for learning latent variable

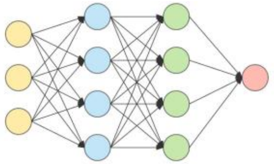
(1)



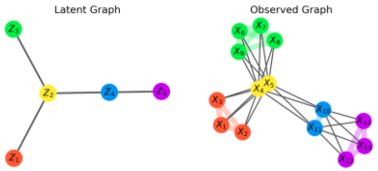
(2)



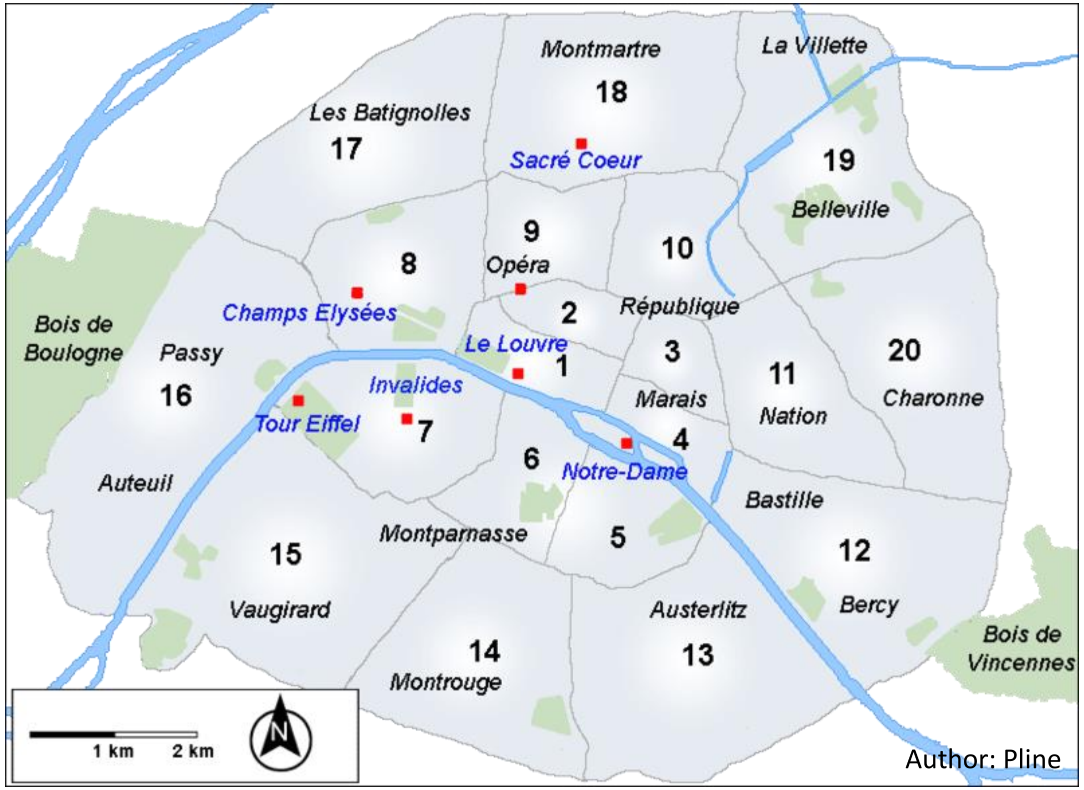
(3)



(4)



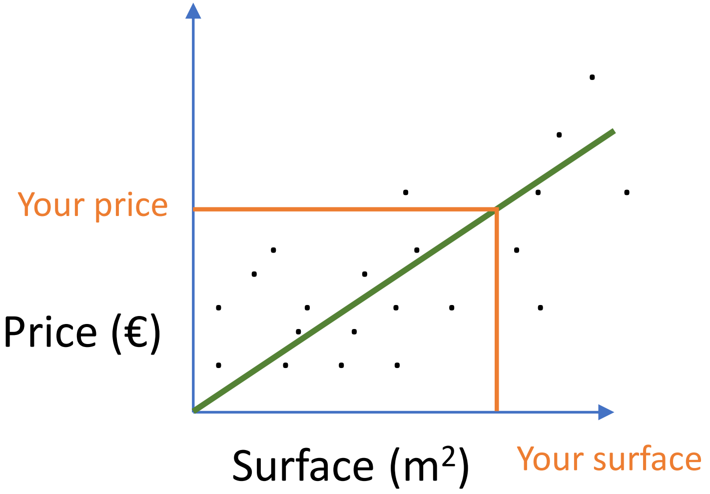
Regression problems



Author: Pline

Objective: Sell your nice apartment in Paris, but you don't know the price to set.

You have gathered some data of selling prices according to the square meters

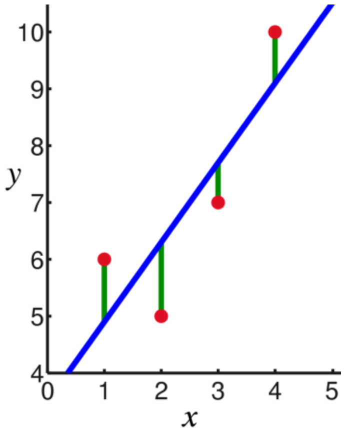


Linear regression is the way to solve this problem

Consider additionally the distance to Eiffel tower -> multiple linear regression

Regression problems

Linear regression (or curve fitting)



In linear regression, the observations (**red**) are assumed to be the result of random deviations (**green**) from an underlying relationship (**blue**) between a dependent variable (*y*) and an independent variable (*x*).

Examples of methods:

- Least square algorithms: a method where the sum of the squares of the residuals made in the results of every single equation is minimized.
- Bayesian linear regression: an approach to linear regression in which the statistical analysis is undertaken within the context of Bayesian inference

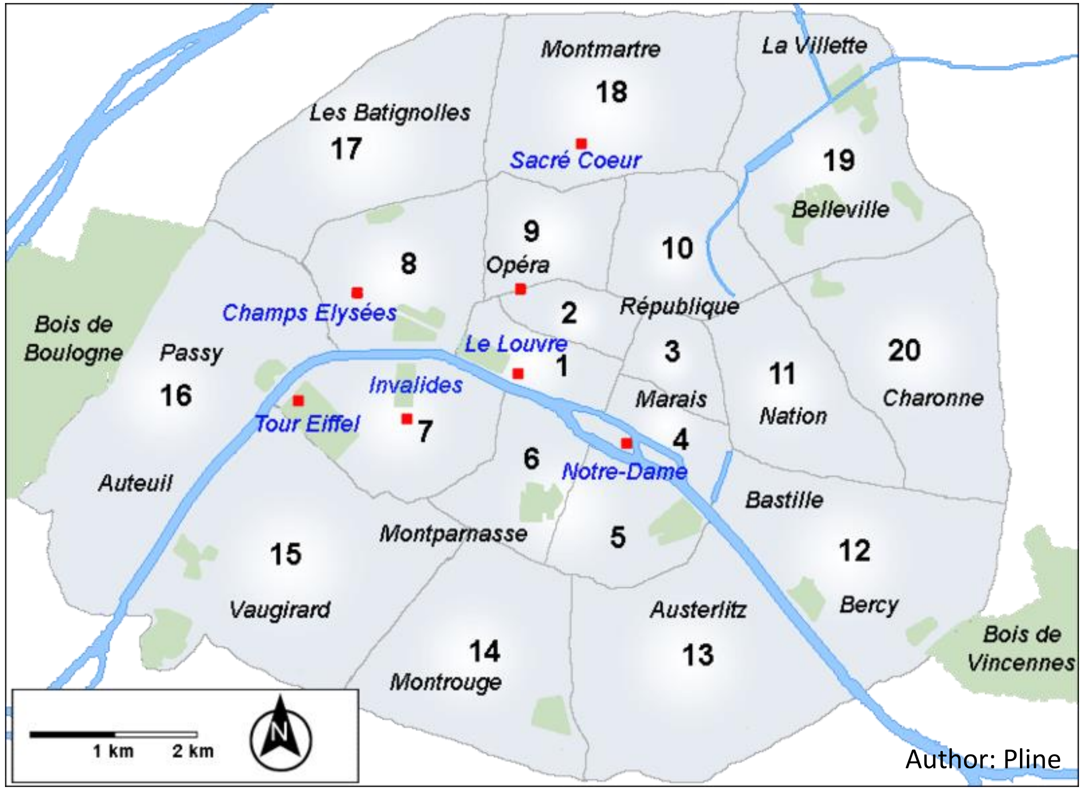
$$Posterior\ distribution = \frac{prior\ distribution\ \times likelihood}{model\ evidence}$$

Ex. : maximum likelihood or maximum a posteriori estimation

This can also be made with multiple variables (and input x would become a n-dimensional vector)

$prior\ distribution$: initial set of parameters (things that you want to learn)
likelihood: similarity of the considered sample, from which you want to compute something, to the prior samples (to the prior distribution) considered, from which your algorithm has learned.
$model\ evidence$: represents how well it seems that the model is correct
posterior distribution: the new set of parameters

Classification problems



Author: Pline

The apartment is on sale, now you want to buy a new one. Unfortunately, you cannot find any that matches your expectations at the moment.

Information gathered about the <u>precedent</u> apartments sold and the neighborhoods:

- Price, Square meter
- Restaurants nearby
- Shopping nearby
- Noise isolation
- Charges
- Metro station nearby
- Separate them into different category : “Good looking and cheap”, “Good looking but expensive”, “Maybe worse visiting”, “Too expensive”, “Bad”

Classification would help you to analyze new on-sale apartments and put them in the categories you have defined

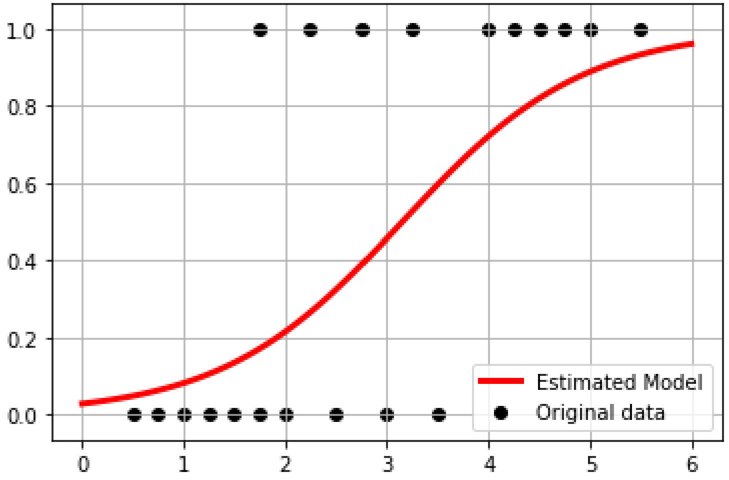
Classification problems

Binary classification :

- Only two possible choices of labels.

  Examples: A tumor is malignant or not, a student passes or not, a cat is on the picture or not, etc.

- Logistic regression is a very common method used



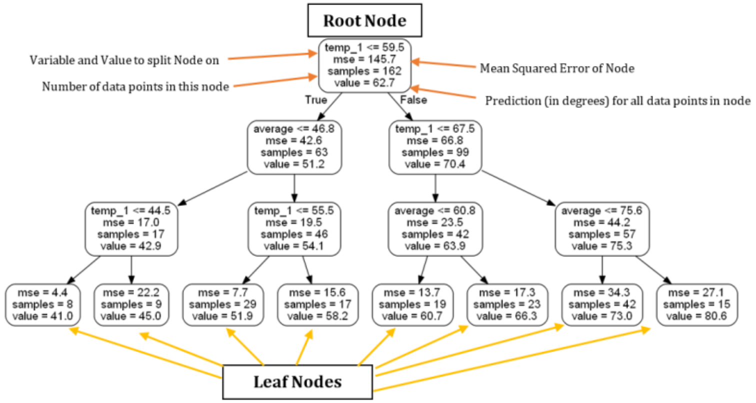
Multi-class classification :

- Multiple choices of labels.

  Examples: divide pictures depending on the animals on them, classify patients according to their supposed disease, classify the nearby elements from cameras to help drones know what is around, etc.

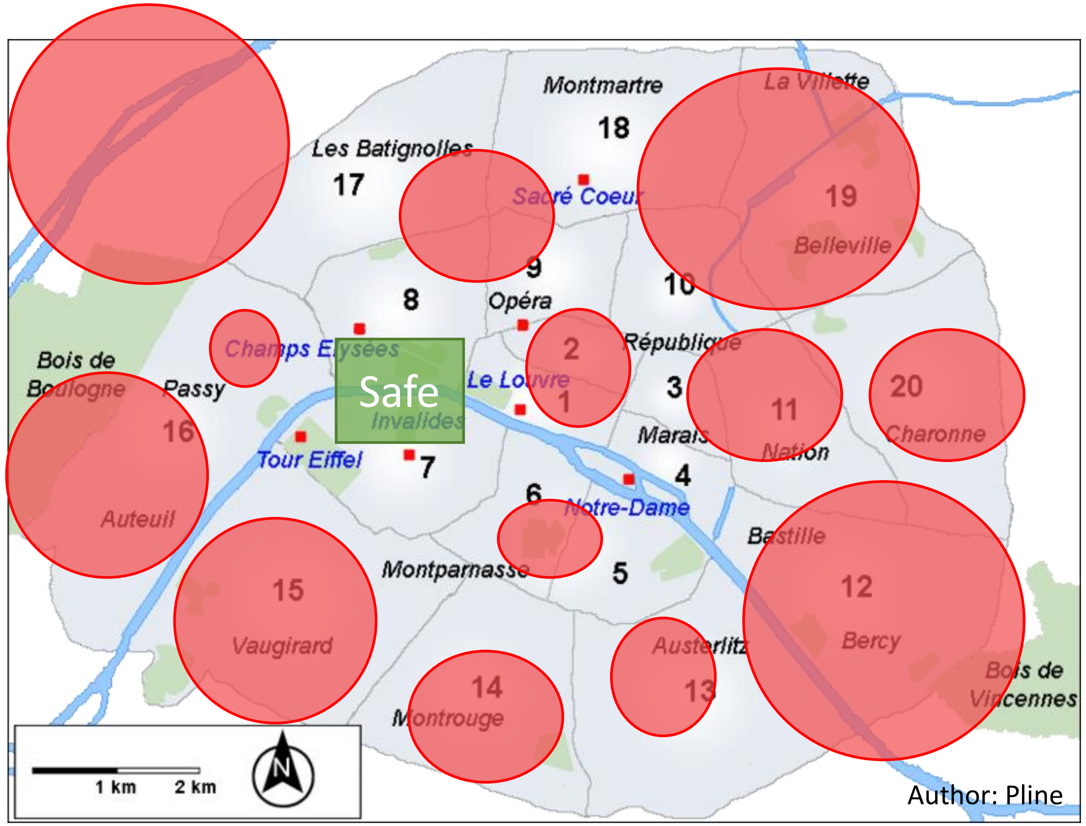
- Decision Trees or Random forests are very common methods for this type of problem

Decision Tree = just one tree



Random Forest = A forest of trees

Clustering problem



It’s the Covid-19 outbreak. The apartment in Paris is sold. You are “homeless”, but rich and you want to find a hotel far away from any *cluster* of ill people.

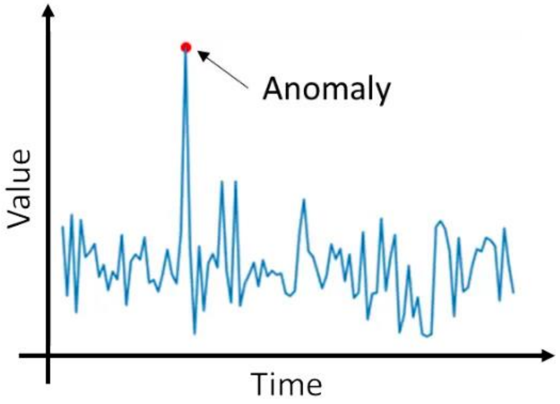
Data available:

- Ill person name
- Address

You will try to find the clusters of ill people and then decide to take the hotel the furthest of the main clusters

Anomaly detection problem

- Paris didn't fit your hypochondriac way of life
- Welcome to Tokyo
- The new problem is seismic activity
- You bought a seismometer, and you want to program it:
  - Track the activity on mobile
  - Get a message when a seismic activity is detected

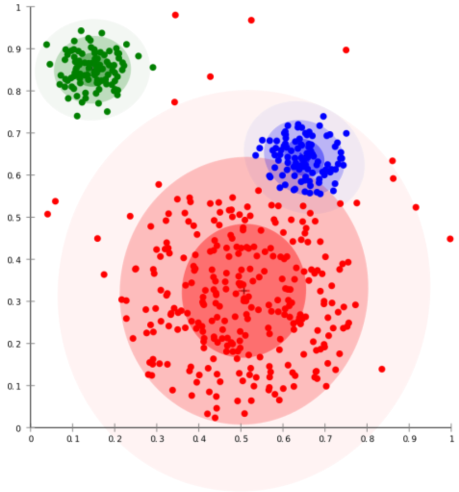


- This is a "point anomaly"
- Another example: your bank detecting a credit card (CC) transaction of 1 billion €
- Other anomaly types:
  - Conditional anomaly – CC transaction attempt much higher than the credit limit
  - Collective anomaly – CC transaction occurring in two countries at the same time

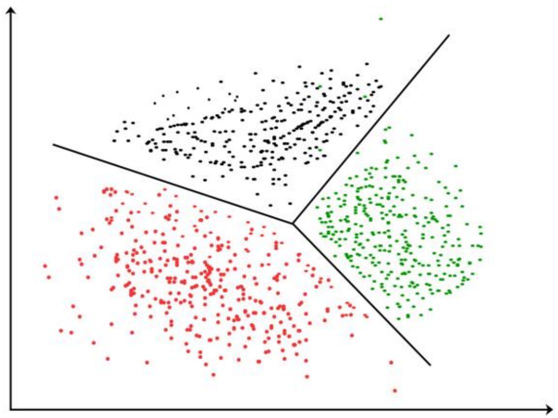
Unsupervised learning problems

Clustering:

- Different types of algorithms: some are based on density, some are distributed-based, some are connectivity-based and some are based on graph theory.
- Looks like classification but it is not, because you do not know beforehand what clusters will be made. (No corresponding target value)



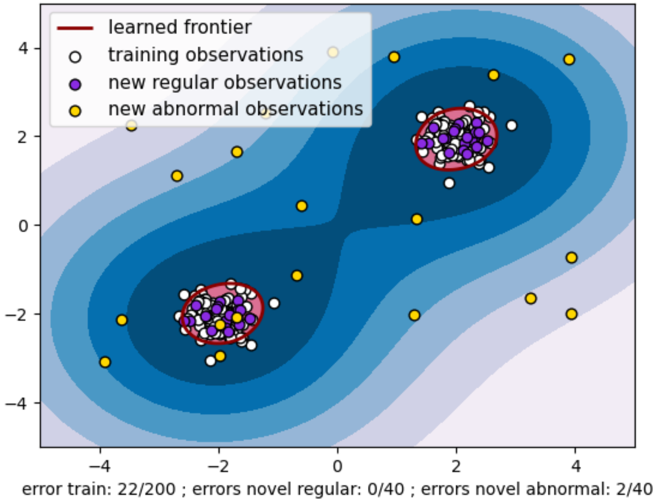
Distribution-based clustering



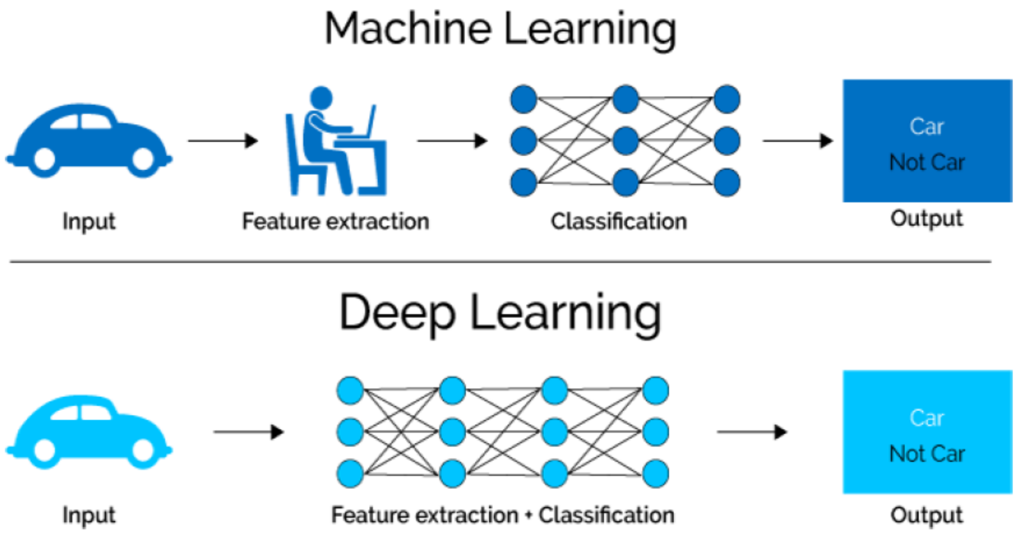
K-means clustering

Anomaly detection:

- Allows to decide whether a new observation belongs to the same distribution as existing observations (it is an *inlier*), or should be considered as different (it is an *outlier*)
- Two types of detection:
    - Outlier detection: The training data contains outliers which are defined as observations that are far from the others.
    - Novelty detection: The training data is not polluted by outliers and we are interested in detecting whether a **new** observation is an outlier

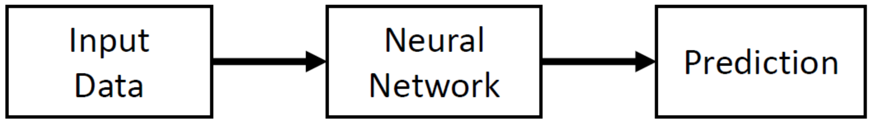


Deep Learning (Unsupervised Neural Network)



How neural networks learn: Backpropagation

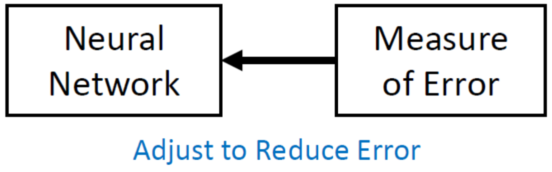
Forward Pass:


Input Data

Neural Network

Prediction

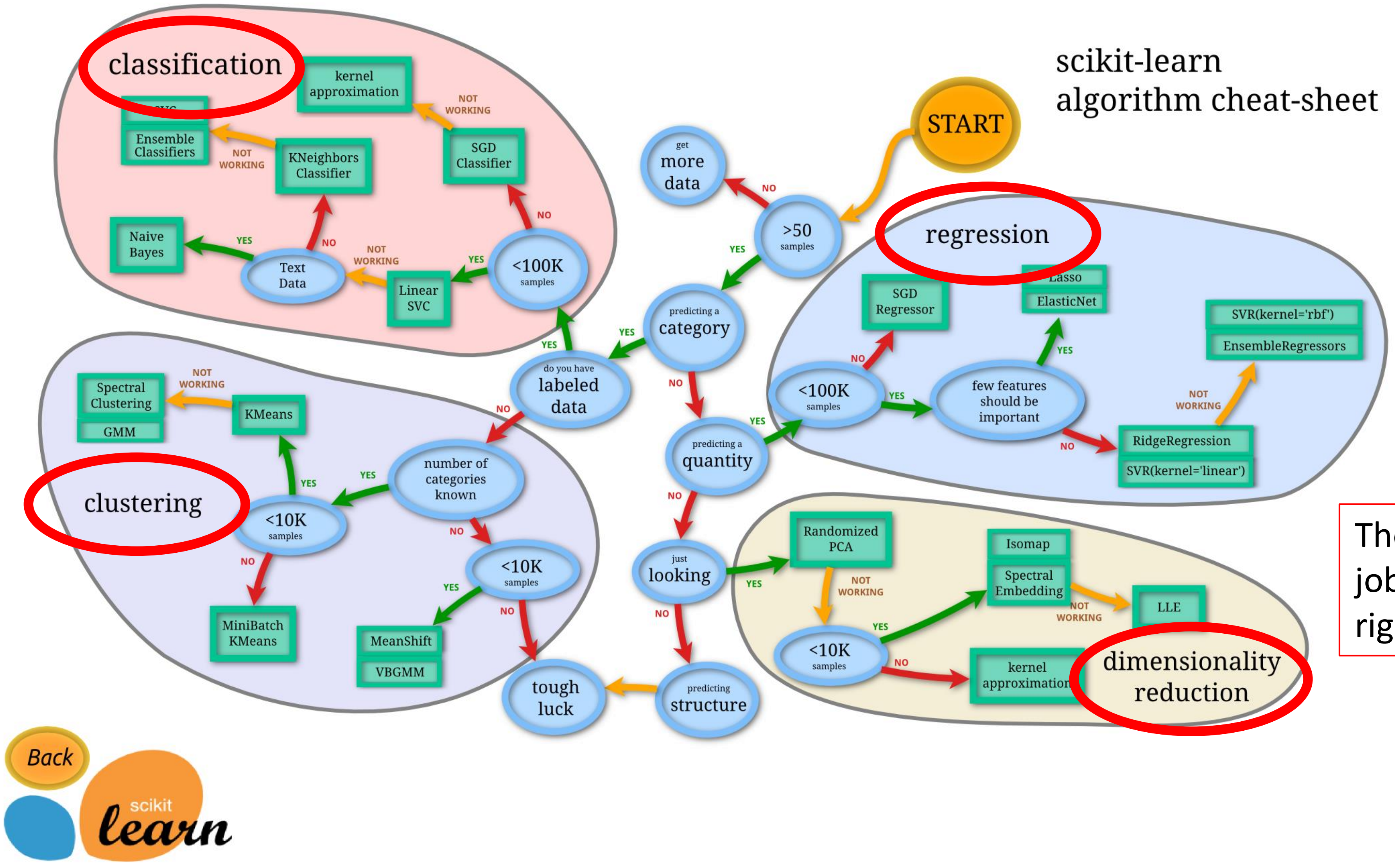
Backward Pass (aka Backpropagation):


Neural Network

Measure of Error

Adjust to Reduce Error

Choosing the right estimator



The biggest part of your job will be to choose the right tool to use

Limitations of machine learning

1. Results hangs on data

Machine Learning requires sufficiently big data sets to train on, and these should be inclusive/unbiased. The quality of the data immediately impacts the quality of the result.

2. Time and computational ressources

Enough time and resources are required to let the algorithms learn and develop to fulfill their purpose with enough accuracy and relevancy. This can mean additional requirements of computer power.

3. Interpretation of results

Results generated after the algorithm has learned have to be carefully interpreted. The algorithms are not 100% accurate and just give you their best guess based on what they know.

4. Algorithms don’t collaborate

Machine Learning algorithms are trained on particular tasks and cannot be transposed to one other. What the machine has learned is not “teachable” to other AI algorithm, nor hardly understandable for humans. Transfer learning is an actual research topic and is not solved today.

What will be done in that course

1. Simple linear regression
2. Multiple and multivariate linear regression
3. Classification problems
4. Unsupervised learning problems
5. Machine learning project (about 3 sessions in class)
    - Learn the use of git (optional to use)
    - Teams of 5 (selected by the teachers – depending on the answer of a small questionnaire)
    - You will be able to suggest teammates
    - Projects proposal will be given after the 4th session (1st or 2nd February)
    - Some work is expected from you between the sessions
    - Project results will be presented during last session

        10 min presentation of your achievement

        Assessment

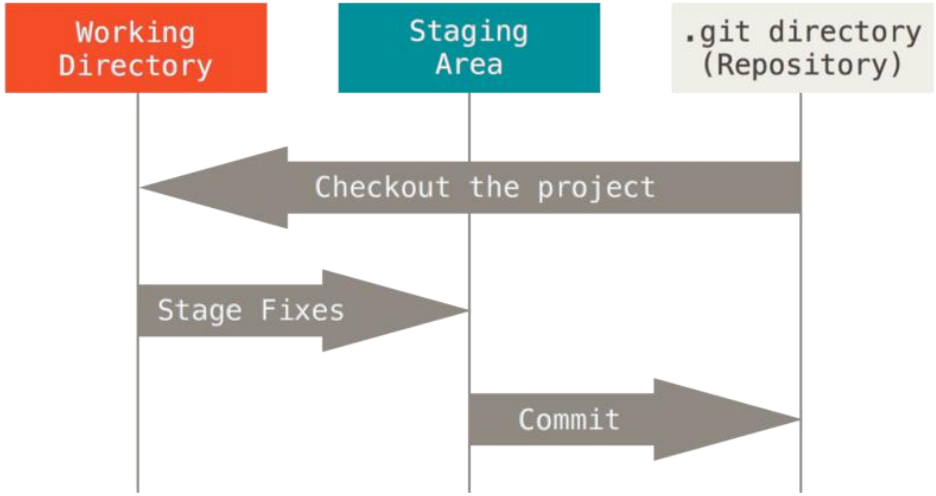
Course assessment

- 50 % of your grade depends on individual evaluation:
    - 5 sessions will start with a quick 5-15 min. test (Session 2, 3, 4, 5 & 7)
    - Do not be late: Tests are automated on Brightspace
    - Quick questions of your understanding of the topic
    - Some questions might require some quick coding so you should have a new Jupyter Notebook file opened and ready to work with.
    - Every material will be available

- The other 50% of your grade depends on the project
    - Results
    - Quality of the presentation

Git

- A Version Control System (VCS): allows you to keep track of the different version of a particular folder on your computer
- Part of the Linux and Android core, available on Windows and Mac
- Works locally (on your computer) and remotely (on servers)
- Saves snapshots of the folder state -> “commit”. Commits get a 40 hexadecimal SHA-1 hash so none can be lost
- Has 3 working stages:

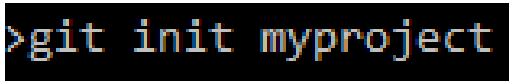


Saves locally versions of your work

Basic Workflow: Files modified Files staged Files committed

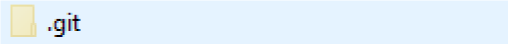
Initialize Git on your local computer

- Initialize a project (create a new file called myproject that would be tracked using git)

  Into Terminal / Command shell 

```
>git init myproject
```

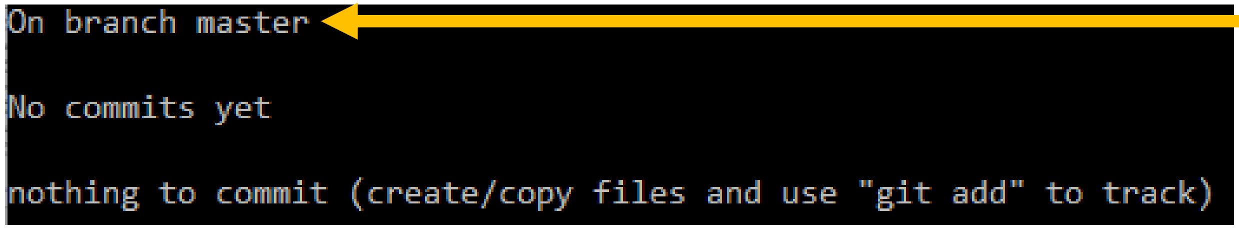
- Inside the new file: Empty file with a .git (hidden file) containing the data of the project and the versions



- You can check the status of git using:

```
>git status
```

- For an empty project it should show:



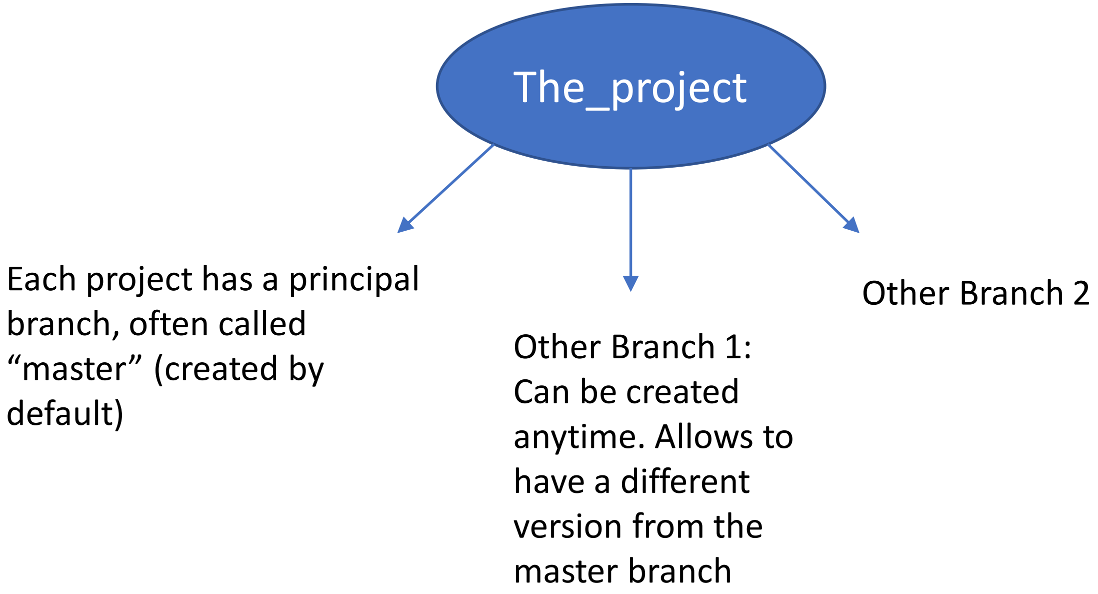
```
On branch master

No commits yet

nothing to commit (create/copy files and use "git add" to track)
```

What is a branch ?

Git branch



The_project

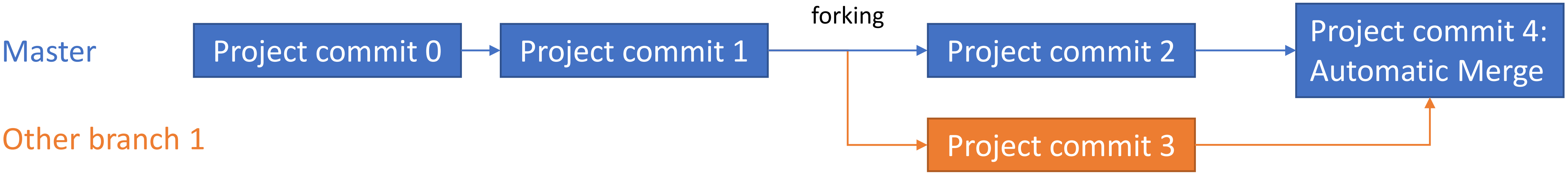
Each project has a principal branch, often called “master” (created by default)

Other Branch 1: Can be created anytime. Allows to have a different version from the master branch

Other Branch 2

Branch work as follows:

Git often allows automatic merging



Master

Other branch 1

Project commit 0

Project commit 1

forking

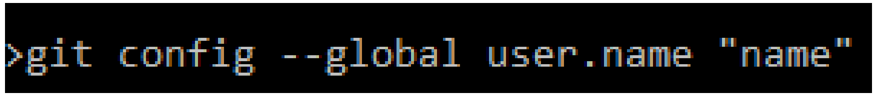
Project commit 2

Project commit 3

Project commit 4: Automatic Merge

git remote repository

- Git allows you to use remote repository and to copy data safely between those
- Set-up git credentials:



```
>git config --global user.name "name"
```

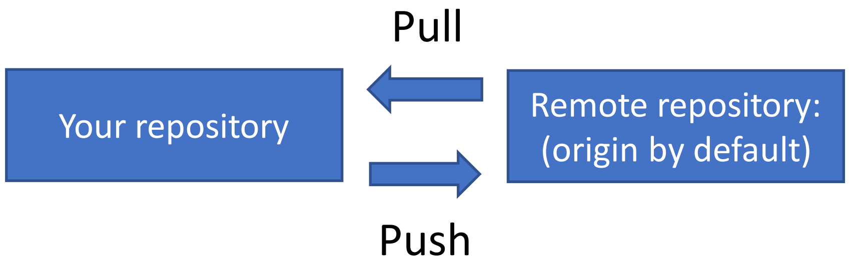
```
git config --global user.email "email_adress@..."
```

- Define the remote repository location

```
git remote add origin https://github.com/person/reponame
```

Name of the remote

Location of the remote



Pull

Your repository

Remote repository:
(origin by default)

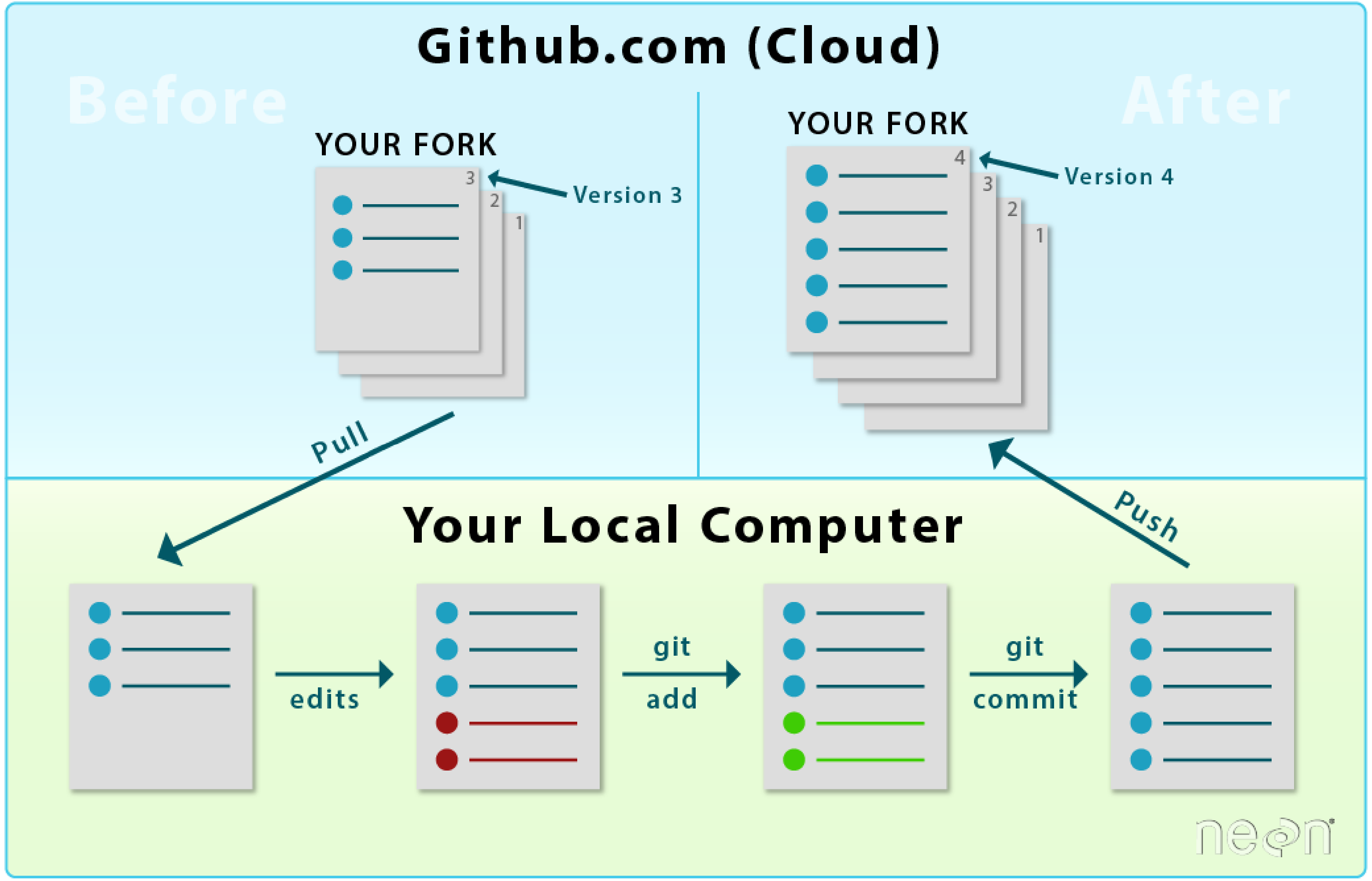
Push

- Clone an existing repository

```
git clone https://github.com/person/reponame
```

Please clone the course repository: https://github.com/ngartner/MSc_DMDS_MachineLearning/

Saving your work using git



Add your work to the status staged

- Will be considered for the next commit
- `git add .`

Write/Save a commit

- Commits are saved version of your code/project

```
git commit -m "Description"
```

Specify that you give a description

The course repository

- Branch master of https://github.com/ngartner/MSc_DMDS_MachineLearning/ will be regularly updated
- You can make your branch and save your notes or exercises

```
git branch testing

git checkout testing
```

Branch name

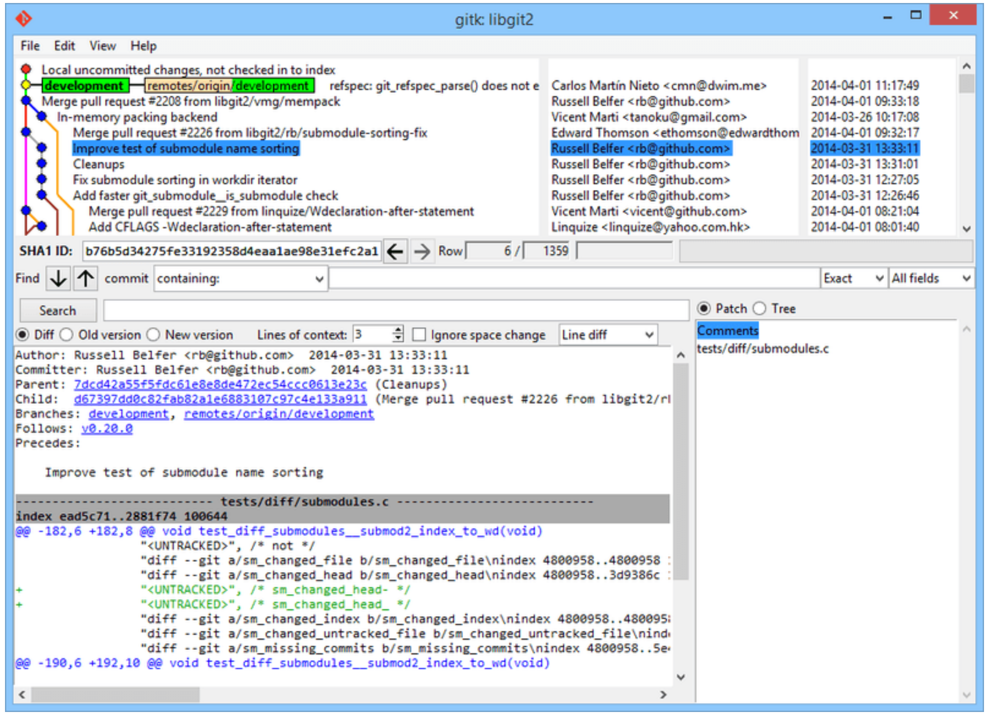
Branch: creates the branch

Checkout: switch to another branch

Be careful: You will not be able to push on branch master

- A cool tool to visualize the evolution of your projects: gitk

```
gitk
```



```
Local uncommitted changes, not checked in to index
development  remotes/origin/development  refspec: git_refspec_parse() does not e   Carlos Martín Nieto <cmn@dwim.me>           2014-04-01 11:17:49
Merge pull request #2208 from libgit2/vmg/mempack                                   Russell Belfer <rb@github.com>              2014-04-01 09:33:18
 In-memory packing backend                                                          Vicent Marti <tanoku@gmail.com>             2014-03-26 10:17:08
  Merge pull request #2226 from libgit2/rb/submodule-sorting-fix                    Edward Thomson <ethomson@edwardthom         2014-04-01 09:32:17
  Improve test of submodule name sorting                                            Russell Belfer <rb@github.com>              2014-03-31 13:33:11
  Cleanups                                                                          Russell Belfer <rb@github.com>              2014-03-31 13:31:01
  Fix submodule sorting in workdir iterator                                         Russell Belfer <rb@github.com>              2014-03-31 12:27:05
  Add faster git_submodule__is_submodule check                                      Russell Belfer <rb@github.com>              2014-03-31 12:26:46
   Merge pull request #2229 from linquize/Wdeclaration-after-statement              Vicent Marti <vicent@github.com>            2014-04-01 08:21:04
   Add CFLAGS -Wdeclaration-after-statement                                         Linquize <linquize@yahoo.com.hk>            2014-04-01 08:01:40
```

SHA1 ID: b76b5d34275fe33192358d4eaa1ae98e31efc2a1 Row 6 / 1359

Lines of context: 3

```
Author: Russell Belfer <rb@github.com>  2014-03-31 13:33:11
Committer: Russell Belfer <rb@github.com>  2014-03-31 13:33:11
Parent: 7dcd42a55f5fdc61e8e8de472ec54ccc0613e23c (Cleanups)
Child:  d67397dd0c82fab82a1e6883107c97c4e133a911 (Merge pull request #2226 from libgit2/rl
Branches: development, remotes/origin/development
Follows: v0.20.0
Precedes:

    Improve test of submodule name sorting

-------------------------- tests/diff/submodules.c --------------------------
index ead5c71..2881f74 100644
@@ -182,6 +182,8 @@ void test_diff_submodules__submod2_index_to_wd(void)
                "<UNTRACKED>", /* not */
                "diff --git a/sm_changed_file b/sm_changed_file\nindex 4800958..4800958
                "diff --git a/sm_changed_head b/sm_changed_head\nindex 4800958..3d9386c
+               "<UNTRACKED>", /* sm_changed_head- */
+               "<UNTRACKED>", /* sm_changed_head_ */
                "diff --git a/sm_changed_index b/sm_changed_index\nindex 4800958..480095
                "diff --git a/sm_changed_untracked_file b/sm_changed_untracked_file\nind
                "diff --git a/sm_missing_commits b/sm_missing_commits\nindex 4800958..5e
@@ -190,6 +192,10 @@ void test_diff_submodules__submod2_index_to_wd(void)
```

```
tests/diff/submodules.c
```

A little bit of practice with a git exercise
(see Git/Git_exercise.pdf)

Videos

- Between some of the sessions, videos to be seen for the next session will be given
- These videos will be discussed during 10-20 min. during the next session.
- They deal with different aspect of machine learning that we do not have time to bring up in class and should
- For next class, please watch: https://youtu.be/aKf6pB4p06E

Towards Transparency in AI:
Methods and Challenges

Timnit Gebru, Google AI

Recap - Important vocabulary from today

Features

Generic name for attributes, predictors, regressors, or independent variables.

Model

output by algorithms and are comprised of model data and a prediction algorithm.

Regression problem (supervised learning)

A problem where the desired output consists of one or more continuous variables

Classification problem (supervised learning)

A problem where samples have to be classified into different classes

Training dataset

a set of examples used to fit the parameters / your model

Label

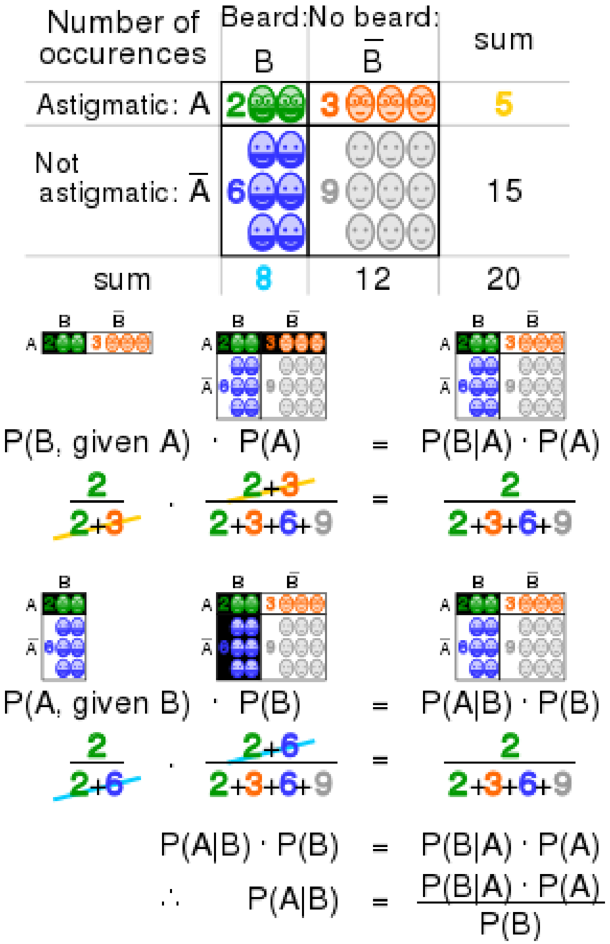
The thing we are predicting in classification, the answer you get after training your model.
Ex: Photo 1 of the dataset has the label ‘car’. The label could be the kind of animal shown in a picture, the meaning of an audio clip, or just about anything.

Unsupervised Learning

Unsupervised Learning is a machine learning technique in which the users do not need to supervise the model. Instead, it allows the model to work on its own to discover patterns and information that was previously undetected

Annex

Bayes theorem



Number of occurences	Beard: B	No beard: $\bar{B}$	sum
Astigmatic: A	2	3	5
Not astigmatic: $\bar{A}$	6	9	15
sum	8	12	20

$$P(B, \text{given } A) \cdot P(A) = P(B|A) \cdot P(A)$$

$$\frac{2}{\cancel{2+3}} \cdot \frac{\cancel{2+3}}{2+3+6+9} = \frac{2}{2+3+6+9}$$

$$P(A, \text{given } B) \cdot P(B) = P(A|B) \cdot P(B)$$

$$\frac{2}{\cancel{2+6}} \cdot \frac{\cancel{2+6}}{2+3+6+9} = \frac{2}{2+3+6+9}$$

$$P(A|B) \cdot P(B) = P(B|A) \cdot P(A)$$

$$\therefore \quad P(A|B) = \frac{P(B|A) \cdot P(A)}{P(B)}$$

$$P(A \mid B) = \frac{P(B \mid A)P(A)}{P(B)}$$

A & B	are events
$P(A)$	is the probability that A occurs
$P(A \mid B)$	is the likehood of A occurring given B occurs, called conditional probability