UNIVERSITÄT BIELEFELD
Faculty of Technology

Programming

Applied Machine Learning

Luna Pianesi

Faculty of Technology, Bielefeld University

Recap

Pandas data structures

Series

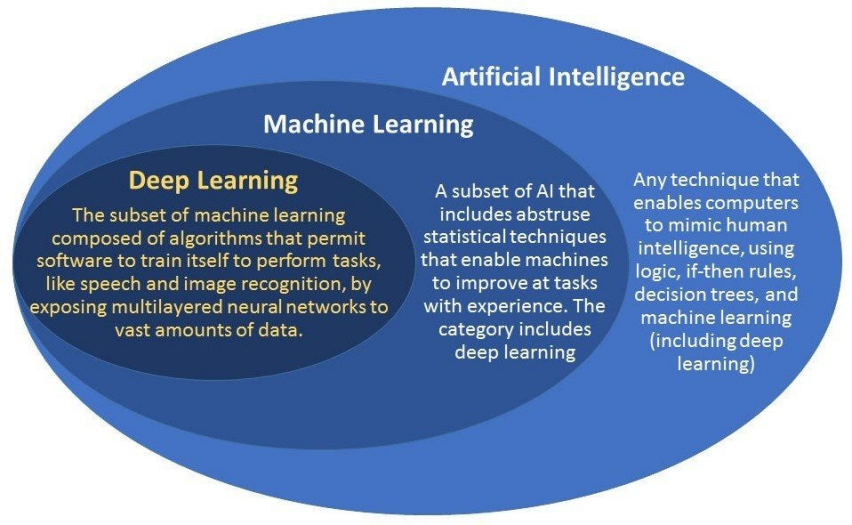
- Container for scalar values
- 1D array
- More powerful than a “1D NumPy array”
- Allows to freely set index
- Size immutable

Data Frame

- Container for Series
- 2D array / table
- Mutability
  - Rows are immutable
  - Allows insertion of new columns


Machine Learning
Scikit-Learn
Applications

What is machine learning?



source: https://medium.com/@clairedigitalogy/what-is-machine-learning-deep-learning-7788604004da

Machine Learning

- Branch of Artificial Intelligence
- Combination of statistics, optimization theory, computer science, information theory, ...
- Different paradigms:

Unsupervised Learning

- Dimensionality reduction
- Clustering

Supervised Learning

- Classification
- Regression

First ML ingredient: data

In machine learning, we use ***data*** to train our models.
Data comes in many different shapes and flavours, but we usually deal with what we can call ***samples***, ***features***, and ***labels***.

First ML ingredient: data

- ***Samples***: they are our data! Every sample can have a number of features, and possibly a label.
- ***Features***: they are measurable properties or characteristics of our data.
- ***Labels***: they are informative tags about the samples of our dataset.

First ML ingredient: data

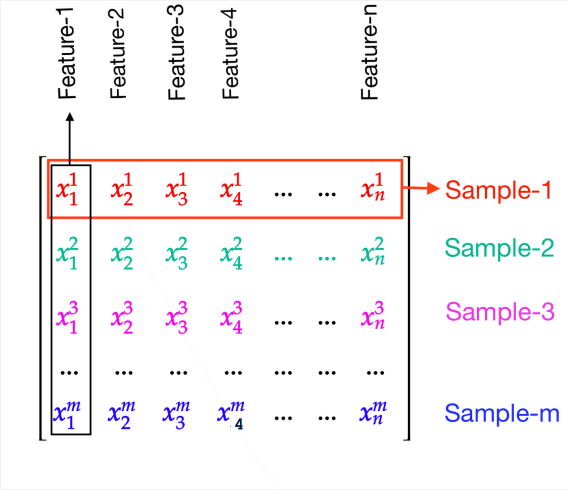
Example: imagine we have a dataset composed of 10,000 samples, where each sample represents a ***person*** in a population.

Every person can be described by some ***features***, which can be, for instance, age, height, eye colour, shoe number, etc..

The ***label*** of each sample can be binary, and can describe whether that person currently lives in Germany or not.

But how can we ***represent*** our dataset to perform machine learning analyses?

Data representation: feature matrix

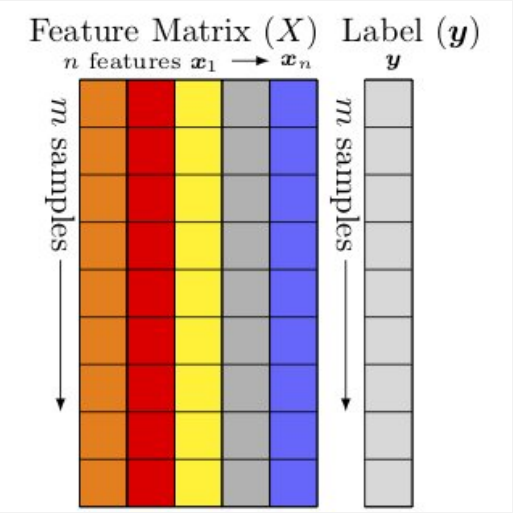


source: https://medium.com/from-the-scratch/deep-learning-deep-guide-for-all-your-matrix-dimensions-and-calculations-415012de1568

Feature representation

- ***Categorical***: can be stored and identified by names or labels (e.g. person's eye colour)
- ***Numerical***: simply numbers (e.g. person's height)

Data representation: label vector



source: https://www.researchgate.net/figure/Feature-matrix-and-label_fig12_331482292

Second ML ingredient: models

In machine learning, we usually deal with ***models***.

A machine learning model is a ***program*** that is used to make ***predictions*** for a given ***data set***.

A machine learning model is built by a ***supervised machine learning algorithm*** and uses computational methods to “learn” information directly from data without relying on a predetermined equation.

source: https://www.mathworks.com/discovery/machine-learning-models.html

Second ML ingredient: models

From the programming perspective, a machine learning model is an ***object*** (stored locally in a file) that has been ***trained*** to recognize certain types of patterns.

You train a model over a set of data, providing it an algorithm that it can use to reason over and learn from those data.

Once you have trained the model, you can use it to reason over data that it hasn't seen before, and make predictions about those data.

source: `https://learn.microsoft.com/en-us/windows/ai/windows-ml/what-is-a-machine-learning-model`

ML paradigms

We will mainly discuss about two machine learning paradigms:

- ***Supervised learning***: you use it when your data is ***labelled***
- ***Unsupervised learning***: you use it when your data is ***unlabelled***

Other paradigms: *semi-supervised learning*, *self-supervised learning*, *reinforcement learning*, **...**

Paradigm vs method vs model

- ***Paradigm***: it is a belief system, a set of assumptions guiding the methods.
- ***Method***: it is a way of doing something, can be also called *algorithm* in this case.
- ***Model***: it is a specific implementation of a method, the output of a certain algorithm applied to some data.

Example: supervised learning is the paradigm, classification is the method, and SVM (support vector machine) is the model

Supervised learning

Supervised learning is a ML paradigm where a model is trained using input data and their respective *labels*.

Supervised learning models are built to create a ***map*** between data and their expected labels.

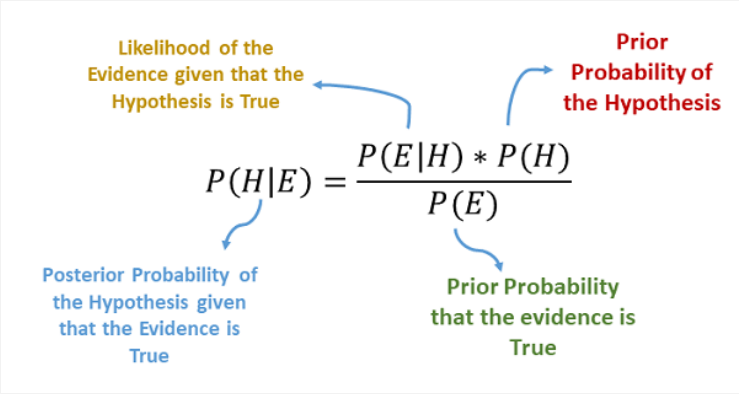
source: https://en.wikipedia.org/wiki/Supervised_learning

Supervised learning methods

Supervised learning can be categorized into:

- ***Classification***: assign data points to pre-existing classes or categories
- ***Regression***: estimate the relationship between a dependent variable and an independent variable

Classification

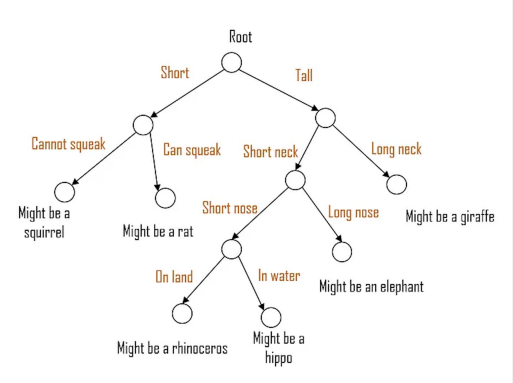


Some methods:

- ***Naive Bayes***
- Decision trees

source: https://www.turing.com/kb/an-introduction-to-naive-bayes-algorithm-for-beginners

Classification



Some methods:

- Naive Bayes
- ***Decision trees***

source: https://medium.com/@jainvidip/understanding-decision-trees-1ba0ef5f6bb4

Regression

Some methods:

- ***Linear regression***: estimates linear relationship between dependent and independent variable
- ***Ridge regression***, ***Lasso regression***: mostly used for regularizing models
- ***Multiple regression***: generalized case of simple linear regression
- ***Multivariate regression***: using several linear regression models at once

Unsupervised learning

Unsupervised learning is another ML paradigm where a model is trained using input data only to learn ***patterns*** within data.

No labels are used in the process of training unsupervised learning models.

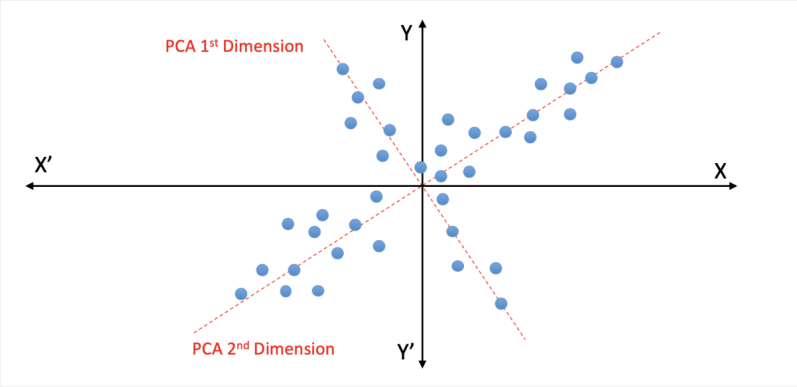
source: https://en.wikipedia.org/wiki/Unsupervised_learning

Unsupervised learning methods

Unsupervised learning can be categorized into:

- ***Dimensionality reduction***: reduce dimensions of high-dimensional data
- ***Clustering***: group objects into clusters containing similar objects

Dimensionality reduction

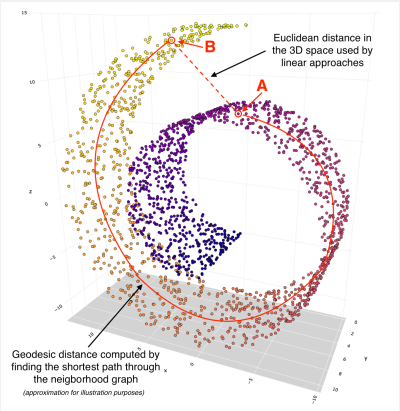


Some methods:

- ***Principal Component Analysis (PCA)***
- Isomap

source: https://medium.com/ds3ucsd/the-objective-of-principal-component-analysis-9f9c540260c3

Dimensionality reduction

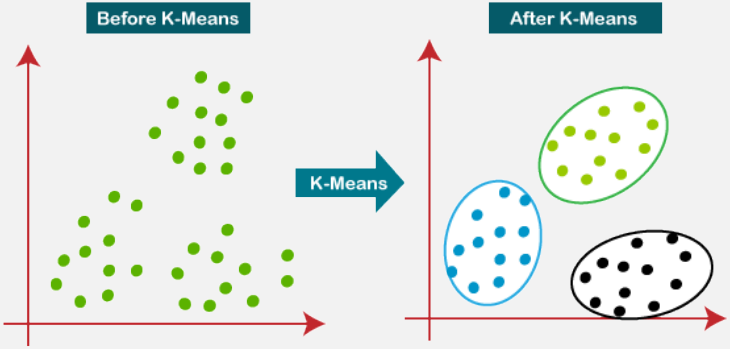


Some methods:

- Principal Component Analysis (PCA)
- ***Isomap***

source: https://www.digitalocean.com/community/tutorials/dimension-reduction-with-isomap#

Clustering

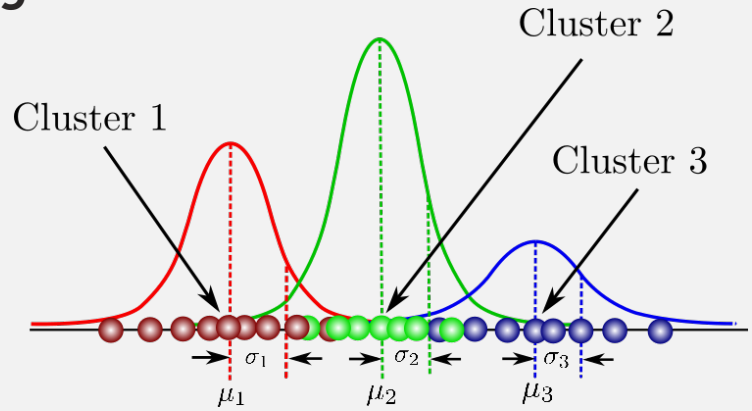


Some methods:

- ***K-means***
- Gaussian Mixture Models (GMM)
- Spectral Clustering

source: https://www.javatpoint.com/k-means-clustering-algorithm-in-machine-learning

Clustering

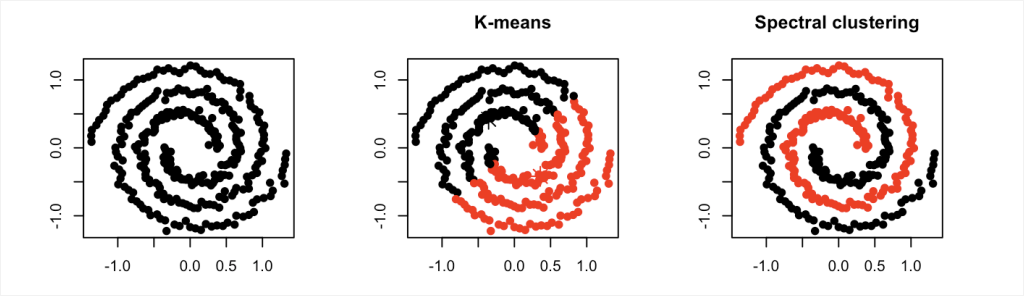


Some methods:

- K-means
- ***Gaussian Mixture Models (GMM)***
- Spectral Clustering

source: https://builtin.com/articles/gaussian-mixture-model

Clustering



Some methods:

- K-means
- Gaussian Mixture Models (GMM)
- ***Spectral Clustering***

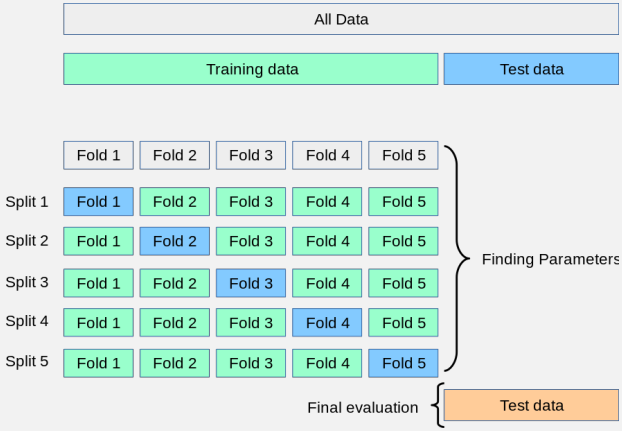
source: https://www.analyticsvidhya.com/blog/2021/05/what-why-and-how-of-spectral-clustering/

Third ML ingredient: carefully following the steps

- Collect and preprocess data ✓
- Choose the model ✓
- ***Train*, *validate*, *test***
  - ***Training***: training a ML model consists in providing a ML algorithm with training data to learn from.
  - ***Validation***: validating a model means to statistically evaluate a model's performance on data that was not used to train it.
  - ***Testing***: testing a model consists in using a trained ML model to make predictions on previously unseen data

How to validate a model? Cross-validation

Cross-validation (CV) is model validation technique for assessing how the results of a trained model will generalize to a test dataset. With CV we can *tune* the parameters of our model.



source: https://scikit-learn.org/1.5/modules/cross_validation.html

Quiz

- Assign the following methods to their categories:
  - Naive Bayes
  - Kmeans
  - PCA
  - Decision Tree
  - Gaussian Mixture Models
  - Isomap
  - Spectral Clustering

- *True* or *false?*
  - Cross validation can only be performed on labeled data
  - Gaussian Mixture Models assumes that data points follow a normal distribution

Quiz

- Assign the following methods to their categories:
  - Naive Bayes — Classification
  - Kmeans — Clustering
  - PCA — Dimensionality red.
  - Decision Tree — Classification
  - Gaussian Mixture Models — Clustering
  - Isomap — Dimensionality red.
  - Spectral Clustering — Clustering

- *True* or *false?*
  - Cross validation can only be performed on labeled data — true
  - Gaussian Mixture Models assumes that data points follow a normal distribution — true


Machine Learning
Scikit-Learn
Applications

Scikit-Learn

Scikit-Learn is a free and open-source machine learning library for Python.

Scikit-learn allows us to easily use models via the “estimator API” (Application Programming Interface).

The estimator API gives us a consistent interface for a wide range of ML applications. The object that learns from the data is called an ***estimator*** (we previously called it model).

source: `https://scikit-learn.org/stable/`

The Estimator API

The advantage of using Scikit-Learn's estimator API is that, for any model we choose, we can follow the same procedure:

1. Choose ***estimator/model***
2. Choose ***hyperparameters***
3. ***Instantiate*** estimator with hyperparameters
4. Call `fit()` to ***train*** the model on a given data set
5. Apply model to ***new data***:
   - Supervised learning: call `predict()`
   - Unsupervised learning: call `transform()` or `predict()` (depending on the estimator)

Quiz

- *True* or *false?*
  - The basic steps are *model*, *fit*, *predict/transform*
  - `LinearRegression.coef_` returns slope and intercept of line
  - Scikit-Learn can generate artificial datasets
  - Scikit-Learn doesn't provide real world data sets
  - transformers uses the `predict()` to transform data.
- Explain the function of the following estimators:
  - `OneHotEncoder`
  - `ColumnTransformer`
  - `DictVectorize`
  - `CountVectorizer`

Quiz

- *True* or *false?*
  - The basic steps are *model*, *fit*, *predict/transform* — true
  - `LinearRegression.coef_` returns slope and intercept of line — false
  - Scikit-Learn can generate artificial datasets — true
  - Scikit-Learn doesn't provide real world data sets — false
  - transformers uses the `predict()` to transform data. — false
- Explain the function of the following estimators:
  - `OneHotEncoder` Transforms one categorical feature with *n* possible values into *n* binary features
  - `ColumnTransformer` Transforms all columns of a `DataFrame`
  - `DictVectorize` Transforms `dict` with categorical variables into numeric features
  - `CountVectorizer` Tokenizes strings and constructs word count frequency matrix


Machine Learning
Scikit-Learn
Applications

Applications

See Jupyter Notebook!

Quiz

- In which order does function `train_test_split` return test/train data?
  - `Xtrain, Ytrain, Xtest, Ytrain`
  - `Xtest, Ytest, Xtrain, Ytrain`
  - `Xtrain, Xtest, Ytrain, Ytest`
  - `Xtest, Xtrain, Ytest, Ytrain`
- What data is stored in
  - `digits.images`
  - `digits.data`
  - `digits.target`

Quiz

- In which order does function `train_test_split` return test/train data?
  - `Xtrain, Ytrain, Xtest, Ytrain`
  - `Xtest, Ytest, Xtrain, Ytrain`
  - `Xtrain, Xtest, Ytrain, Ytest` ✔
  - `Xtest, Xtrain, Ytest, Ytrain`
- What data is stored in
  - `digits.images` bitmap data of all images
  - `digits.data` feature matrix
  - `digits.target` labels (ground truth digits)

Recap

Summary

- Machine Learning
  - Supervised learning
    - Classification
    - Regression
  - Unsupervised learning
    - Dimensionality reduction
    - Clustering
- Scikit-Learn
  - Estimator API
- Applications
  - Handwritten digits dataset
  - Text comparison

What comes next?

- Have a look at the Jupyter Notebook of this lecture
- Further reading about Pandas: Chapter 5 of the “Python Data Science Handbook”:
  `https://jakevdp.github.io/PythonDataScienceHandbook/`
- Have a look at the in-depth analyses that are provided in the handbook