Projets master MASH

Séance 23 novembre 2015

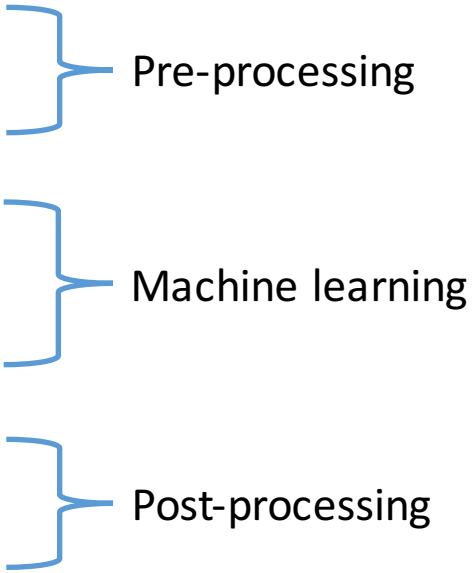
Fajwel Fogel

Aujourd’hui :

- Revue des étapes essentielles pour les projets
- Comment travailler avec des gros jeux de données
- Questions sur les projets

Data analysis: crucial steps

- Identify goal(s)
- Exploratory analysis
- Extract and construct relevant features
- Choose and tune algorithm
- Evaluate and compare
- Refine
- Visualize
- Repeat



Identify your goal(s)

- What is the information you want to retrieve?
- What are the tasks you want to perform?
- What are you optimizing?

- Do you have labels?
- Which kind of method will you use?
- How will you measure the quality of your method?

Machine learning

- Supervised/unsupervised
- Prediction/Explanatory/Causality
- Regression/classification
- Clustering
- Dimension reduction, low-dimensional embedding
- Recommendation, collaborative filtering
- Ranking
- …

Exploratory analysis

- Describe your dataset (size, type of variables…)
- Get univariate stats (averages, quantiles, std, histograms…)
- Visualize your dataset
  - Low dimensional embedding? (e.g. PCA …)
  - Correlations?
  - ...

Extract features

- Identify relevant information for your goal(s)
  - Clean your data
  - Remove useless and non-reliable data
  - Fill/remove missing values...
- Create new variables that will be better suited for your model
  - Create dummies/quantize variables
  - Center variables? Reduce/normalize variables?
  - If you use a linear model you can add interaction terms (product/ratios of variables)
  - Add variables taking into account the temporal dimension
  - Use other representation (e.g. Fourier, wavelets…)
  - Neural nets learn features from data…

The quality of your input data is crucial !!!

Choose and tune algorithm

- Check your goals
  - Supervised/unsupervised
  - Prediction/Explanatory/Causality
  - Regression/classification
  - Clustering
  - Dimension reduction, low-dimensional embedding
  - Recommendation, collaborative filtering
  - Ranking
  - …
- Use standard methods for benchmark
  - implemented in scikit-learn or other ready to use package
  - many references
  - good documentation on how to use them
- Tune algorithms using cross-validation or other relevant technique

Evaluate and compare

- Get relevant metrics for your goal (e.g. prediction error, ROC curves, AUC etc.)
- Compare different methods
  - Very easy with scikit-learn
  - Keep in mind that running-time can vary a lot between different algorithms, even when they aim at solving the same task
- Store the results of all your experiments

Refine

- Combine different methods: e.g. use an unsupervised method for initialization then a supervised scheme.

- Polish results of efficient large-scale methods locally with finer grained data and more sophisticated algorithms.

Visualize

- Find a nice way to present your results
  - plots, histograms, heatmaps…
  - low-dimensional embedding
- Interpretability: with just a bit of knowledge on the context, you can get very valuable and reassuring qualitative insights from your analysis
- Visualization gives you intuition on how to improve previous steps (beware visualization is reliable)

« Big Data »

A few strategies:

1. "Sub-sample" your data
2. Parallelize your task (or serialize it if you only have access to one computer)
3. Reduce the size of your problem using low complexity method, then refine your results

Projets « Challenge Data »

https://challengedata.ens.fr/en/challenges