Introduction to Data Science

with Python

Alexis Bogroff

May 24, 2022

Presenter

Alexis Bogroff
Lecturer and Mentor in Data Science
at Paris 1 Panthéon-Sorbonne, ESILV,
Openclassrooms, EM-Lyon

Presenter

Alexis Bogroff
Lecturer and Mentor in Data Science
at Paris 1 Panthéon-Sorbonne, ESILV,
Openclassrooms, EM-Lyon

- 4 years Teaching Assistant and lecturer in VBA, Python for finance, SQL, Data Analysis and Data Science
- 9 months Researcher Assistant at Paris 1 Panthéon-Sorbonne within H2020 European Project
- 1 year Data Scientist at Pléiade Asset Management

What does that mean?

- Patterns (correlations)
- Correlation vs Causality
- Continuity (stationarity)
- Examples:
  - Present:
    - Electricity consumption (e.g. Seattle)
    - Missing values
    - Client category
  - Future:
    - Electricity consumption next month (time series)
    - Client clicking add (recommander sys.)
    - Pedestrian and cars trajectories (RL)

Regression

- Linear regression
- Polynomial regression
- SARIMA
- Deep Learning (RNN, LSTM)

Classification

- Logistic regression
- Tree (ensemble models, RF, XGBoost)
- K-NN
- Neural Network (Deep Learning: CNN)

Unsupervised ML

- Clustering (grouping)
- Dimensionality reduction
- Reducing multicolinearity
- Models:
  - K-Means: entropy minimisation principle (min var intra, max var inter)
  - Hierarchical Clustering
  - PCA

General

- Parameters
- Hyperparameters
- Train, cross-validate, test
- Feature importance
- Data Leakage

Fit

- Generalization
- Complexity
- Over/Under-sampling
- Unbalanced Datasets (weights on cost function, SMOTE, Auto-encoder)

Metrics

- Regression
  - RMSE
  - R^2
- Classification
  - Accuracy
  - AUC, ROC Curve
  - Other metrics based on confusion matrix

Transfer Learning, Why?

- Training can be complicated, long and expensive
- Specific but complex (and similar) task (NLP)
- Few samples

What has been learnt?

- Weights value (or centroids)
- Hyperparameters

Deep Learning

- Optimize target objective on long term, intermediate steps on short term:
  - Increase task difficulty gradually
  - Better generalisation: Multi-task learning (RL, learn recognize unrelated objects)
  - Improve Neural Network architecture (genetic algorithms)

Some code examples

- Sklearn simple 4 lines of code
- More advanced Sklearn
- Deep Learning with Pytorch