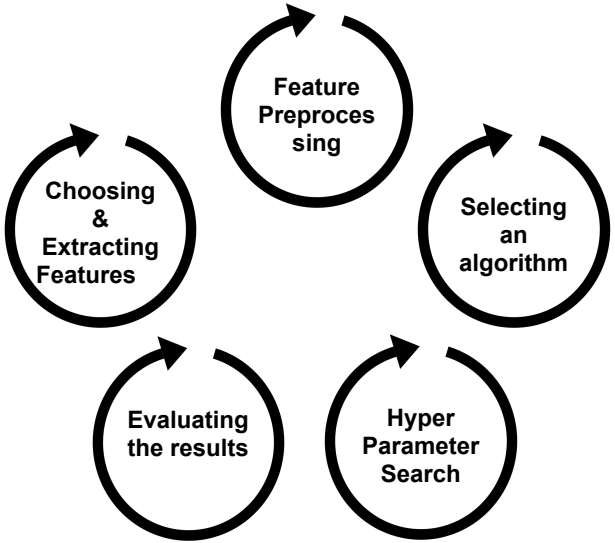
Data Preprocesing Framework

December 2019

Lei Guang

Context

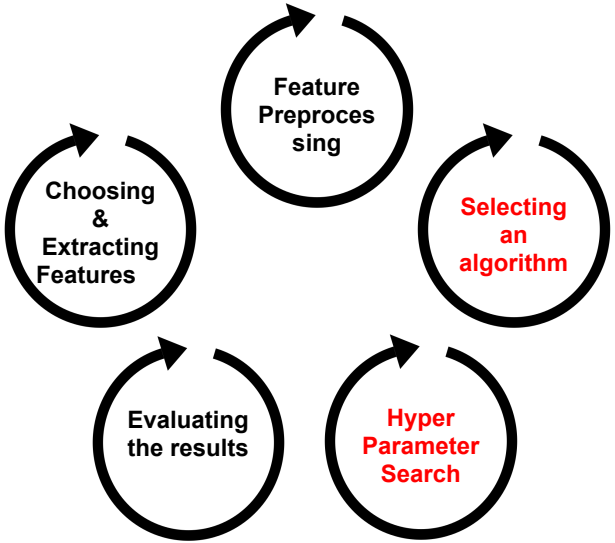
Data Science is a **highly iterative process**. There is no single recipe that can be applied to all problems (**no free lunch theorem**). Therefore, a **multiplicity of experiments** must be done before we can come up with a final predictive model.



Because the underlying distribution of data may change over time, once a model is developed, it **has an expiry date and must therefore be retrained periodically** or **continue learning** (Third order of iteration)

Auto-ml

Last year, we have focussed on building our **proprietary auto-ml framework**, able to speed up our development cycle by automating two steps from the modeling process (Algorithm selection & HPS).



The **auto-modeler** is able to identify the best performing algorithm on a given dataset while performing hyper parameter search and other necessary transformation (rebalancing, calibration, stacking & blending, model evaluation, etc.)

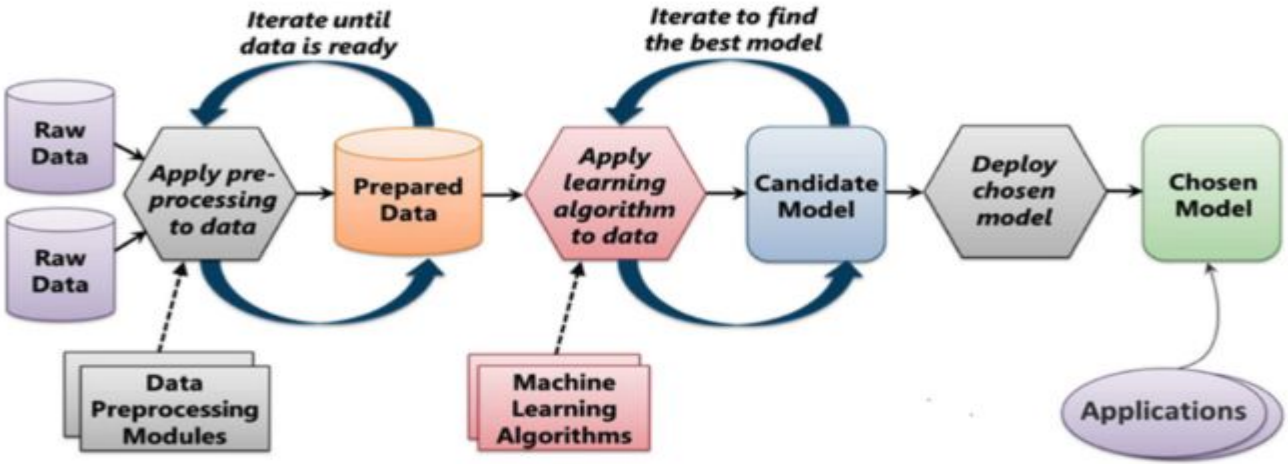
Data Preprocessing

What is data preprocessing?

- A process to transform raw data collected from real life into better quality and understandable format for machine learning algorithms.

Data Preprocessing in Machine Learning Process

An essential and iterative step in machine learning process



From "Introduction to Microsoft Azure" by David Chappell

Why Data Preprocessing?

Data in real word is dirty and noisy

- Incomplete: missing values, lack attributes of interest
  - Example: occupation=”N/A”
- Noisy: contains errors or outliers
  - Example: age= -20
- Inconsistency:
  - Example:
    - Age = 80, Birthday = ‘1/1/2000’
    - Ratings: was 1, 2, 3, … NOW ‘A’, ‘B’, ‘C’

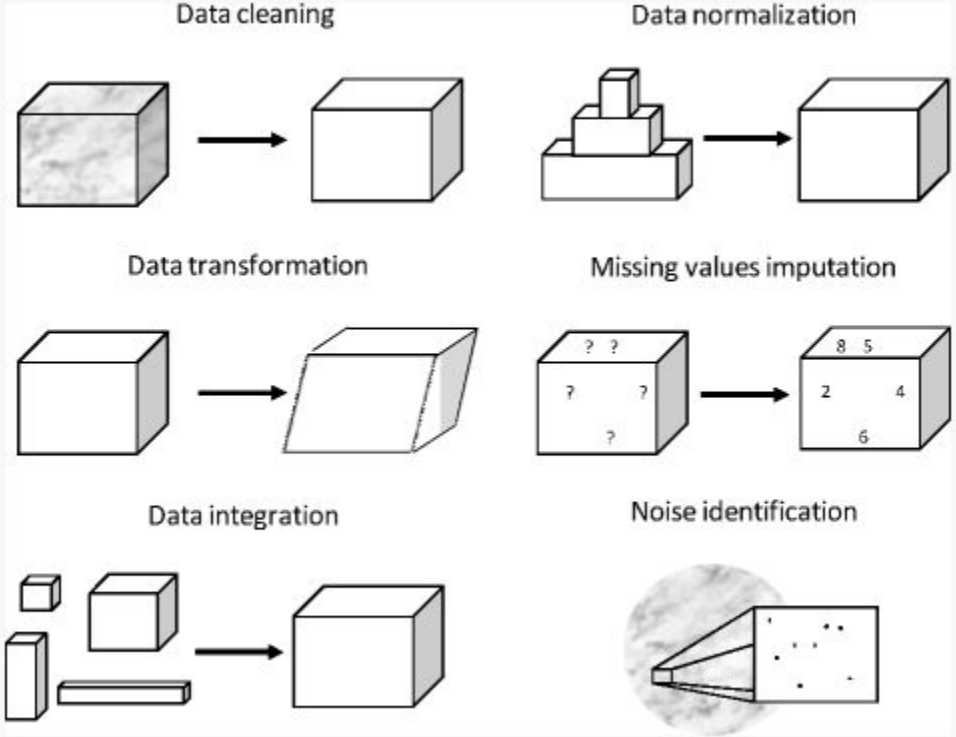
Why Data Preprocessing?

Garbage In, Garbage Out

- Make the raw data ready for learning
- Give hints facilitating the learning process
- Improve ML algo performance

Data Preprocessing Methods

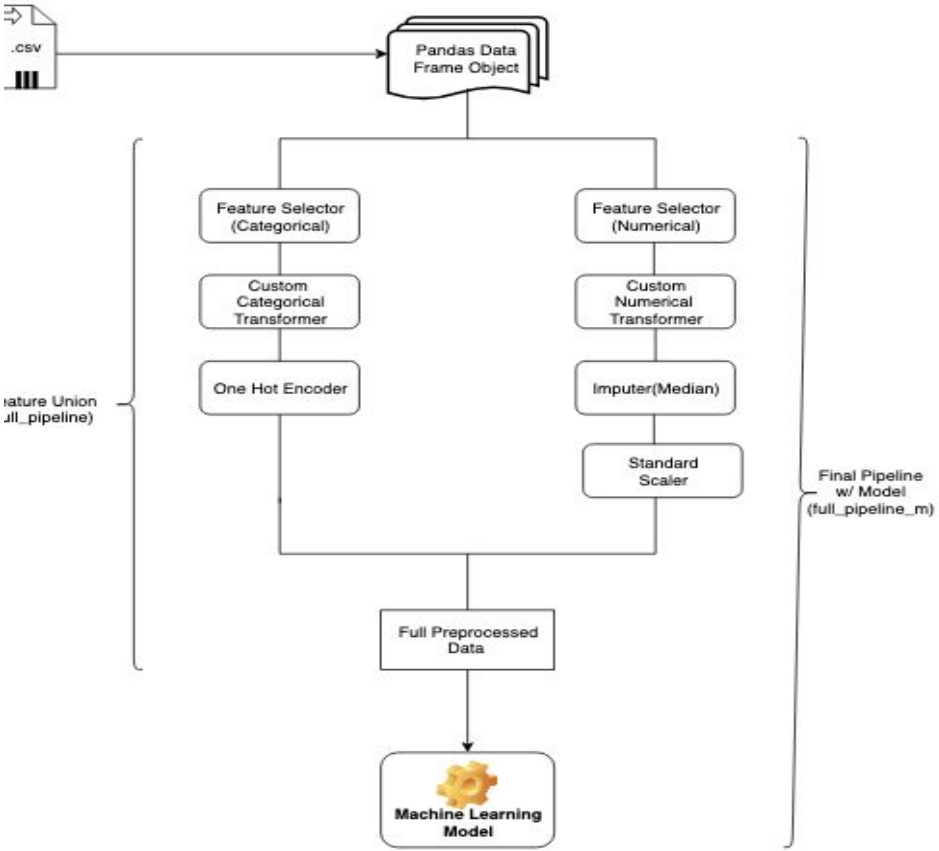
- Data Cleaning
- Data Normalization
- Data Transformation
- Data Imputation
- Data Integration
- Noise Treatment
- Dimensionality Reduction
- Feature Selection
- …….



What We Have Done

- Automate data preprocess with

  Pipeline and **Transformers**



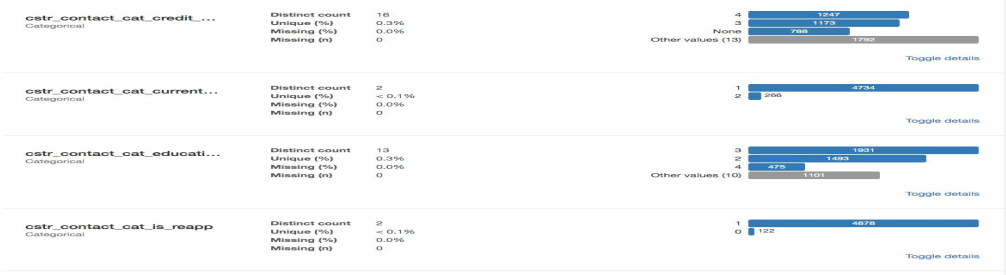
What We Have Done

- Implement **advanced** data preprocessing methods
  - **Now**: More sophisticated methods to fill missing values,
    - e.g., iterative imputation and complex imputation pipeline based on missing value types/percentages
  - Implement advanced transformers for feature selection/dimensionality reduction,
    - e.g., AutoEncoders and Entity Embedding

- Implement **configurable** pipelines
  - Use **Config** file to setup preprocessing steps instead of hard-coding (by adding blocks of transformers).
- Implement **rule-based** pipelines

What We Have Done

- Add **descriptive** statistics of the raw data and preprocessed data
  - Add descriptive stats, e.g., profiling data before and after preprocessing



  - Save preprocessing settings settings along with models and results (iterate model improvement)

Benefits of Our Team Work

- Allow **fast** iteration and **good quality** data analysis and experiments (less time on actual coding problems but focus on data science problems)

- Standard preprocessing pipelines allow us to **compare** different iteration of experiments (why experiment #1 is better than #2)

- Improve the **performance** of our predictive models

- **Faster** model production and model upgrade