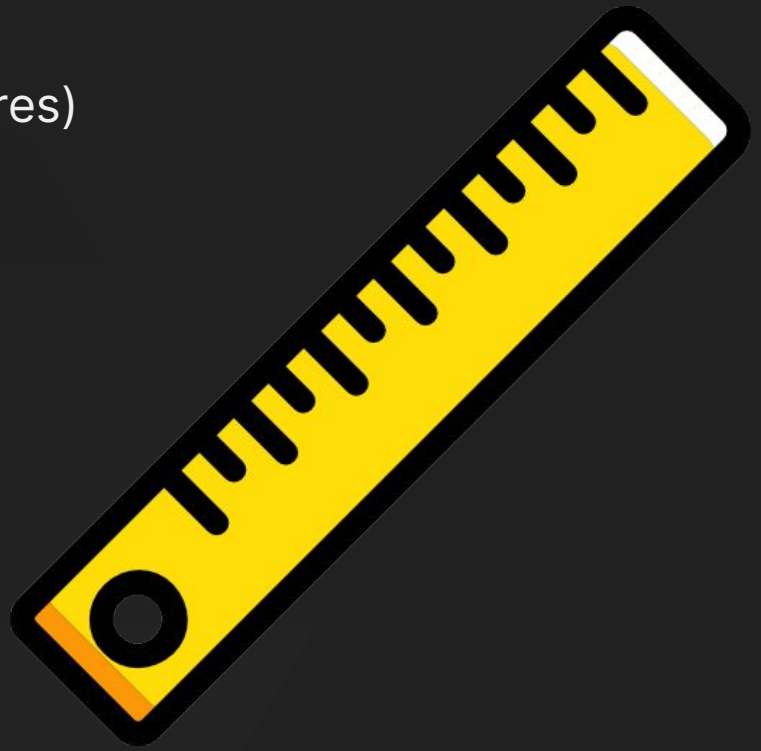

Strive
SCHOOL

/ Numerical Encoding

Numerical Features

/ Numerical features are:

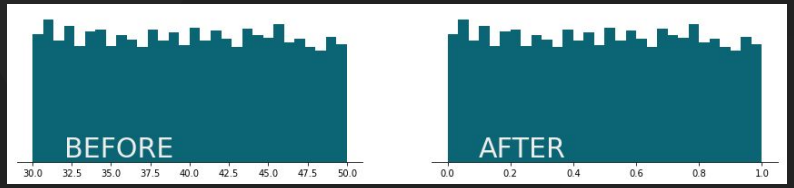
- Discrete numbers (aka ordinal features)
  - Example: Age of the person.
- Continuous numbers
  - Example: Height of the person.
  - Example: Weight of the person.



Sklearn methods

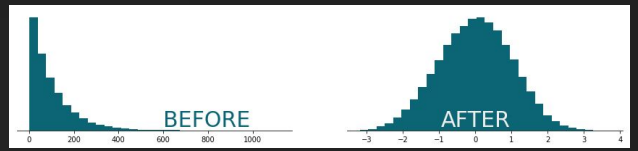
A) Scaling

- Min-max scaling `MinMaxScaler()`
- Max-abs scaling `MaxAbsScaler()`
- Standard scaling `StandardScaler()`
- Robust scaling `RobustScaler()`



B) Normalization

- Manually
  - Logarithm `np.log(1+x)`
  - Square root `np.sqrt(x+2/3)`
- `PowerTransformer()`
  - Box-Cox
  - Yeo-Johnson
- `QuantileTransformer()`
  - (aka GaussRank)



Other Sklearn methods

C) Create groups

- Binarize data `Binarizer()`
  - Set feature values to 0 or 1 according to a threshold.
- Create bins `KBinsDiscretizer()`
  - Bin continuous data into intervals.

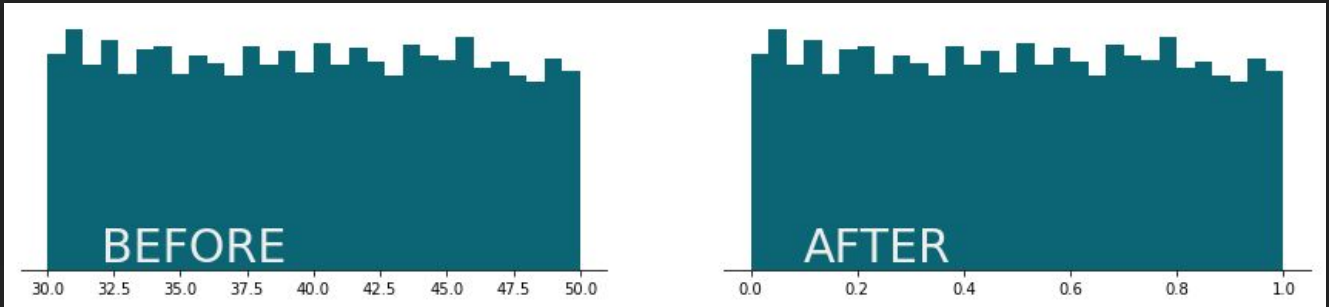
D) Create more features

- PolynomialFeatures()
  - Generate polynomial and interaction features.

This is useful for linear models only

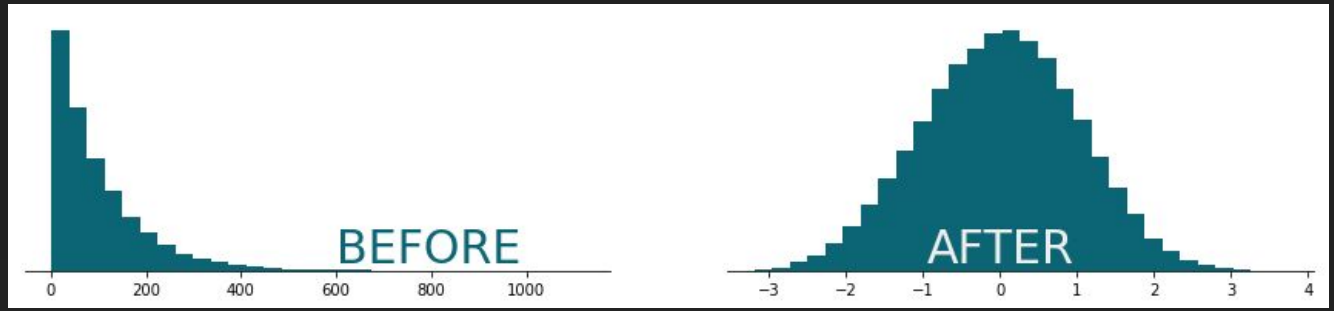
/ A) Scaling

Transforming your data so that it fits within a specific scale, like 0-100 or 0-1.

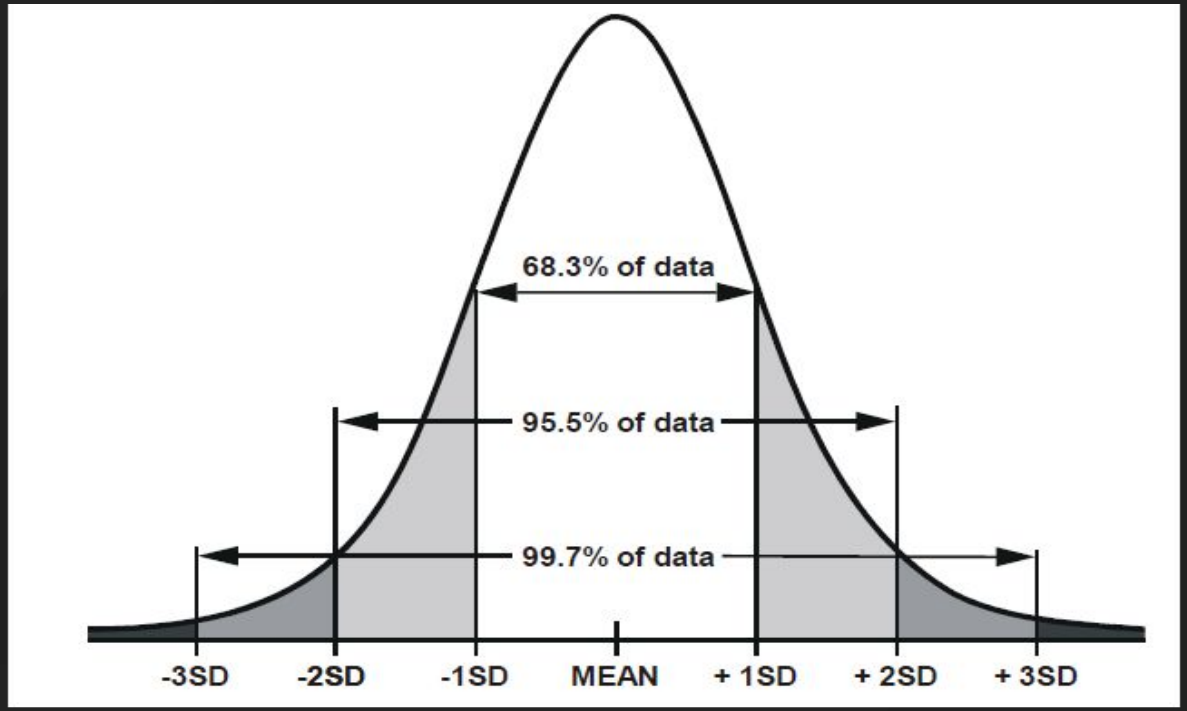


/ B) Normalization

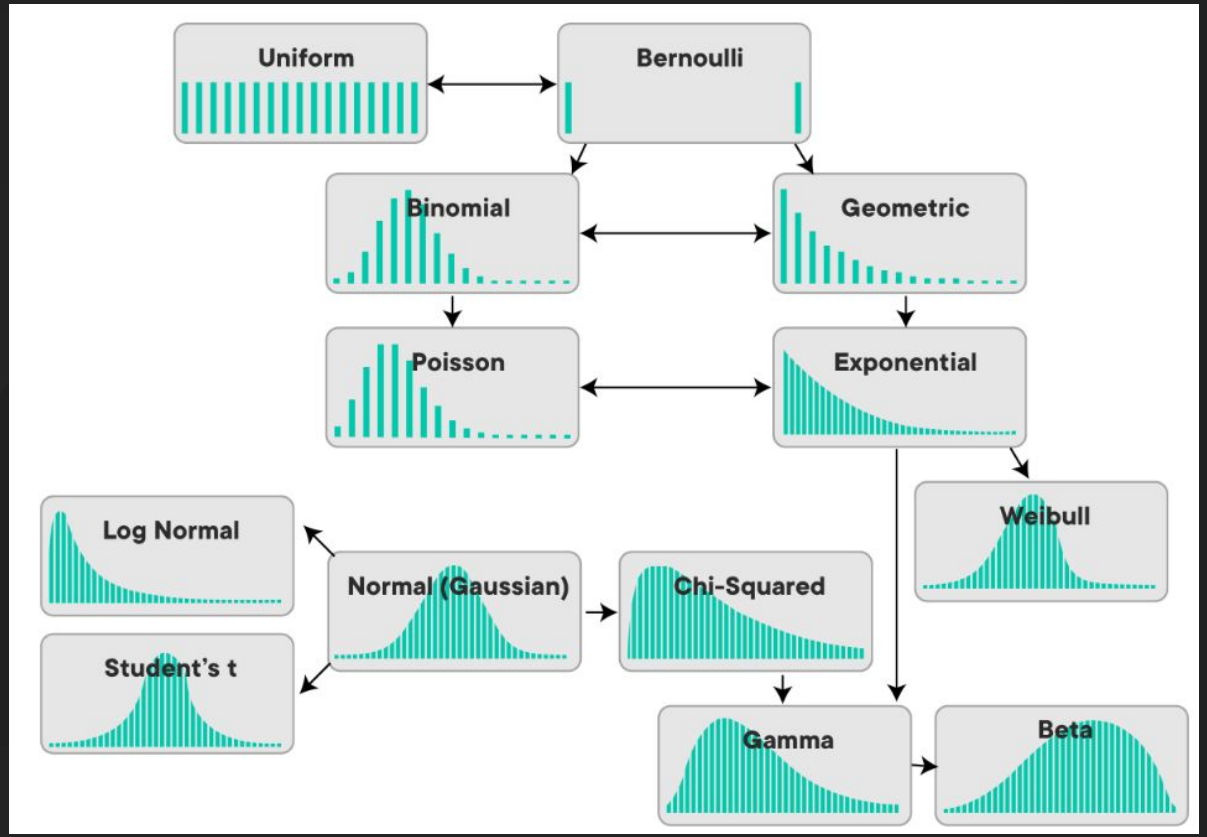
Changing the shape of the distribution to a Normal distribution ("bell curve")



Normal distribution (aka Gaussian distribution)



Other types of distributions

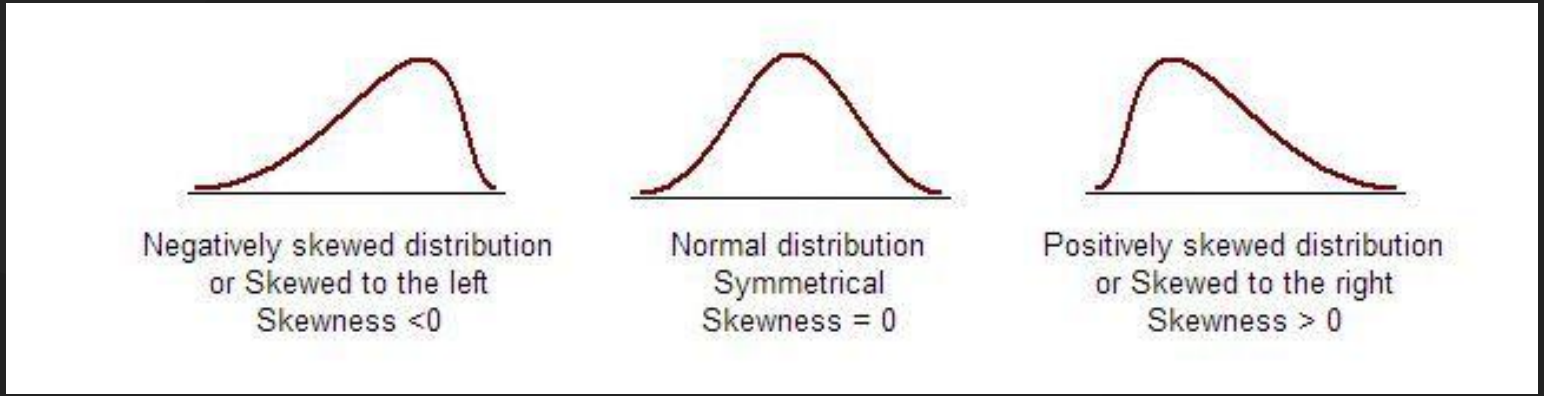


Skewness

`pandas.skew()`

/ A number to determine the asymmetry of the distribution.

/ Normal distribution have skewness = 0

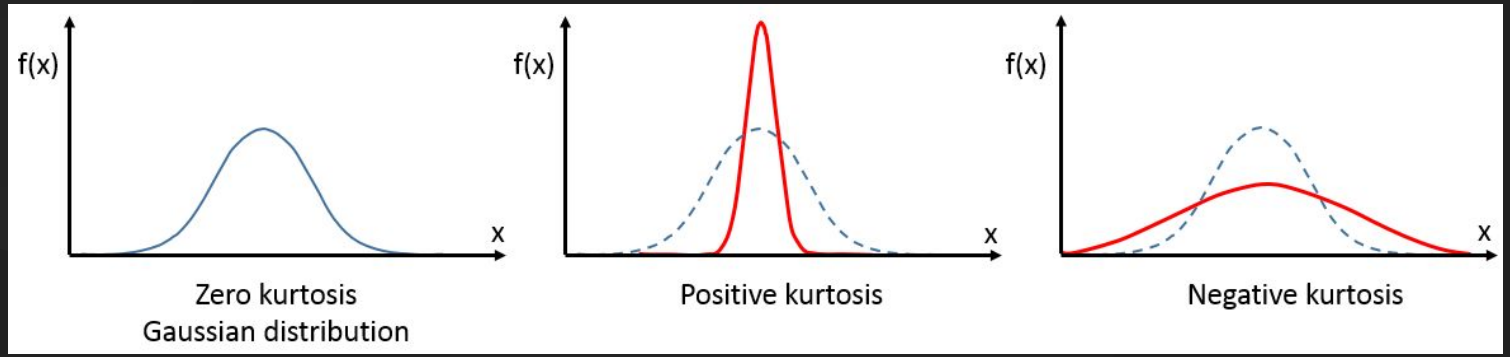


Kurtosis

`pandas.kurt()`

/ from Greek: κυρτός meaning "curved, arching" is a measure of the "tailedness" of the distribution.

/ Normal distribution have kurtosis = 0

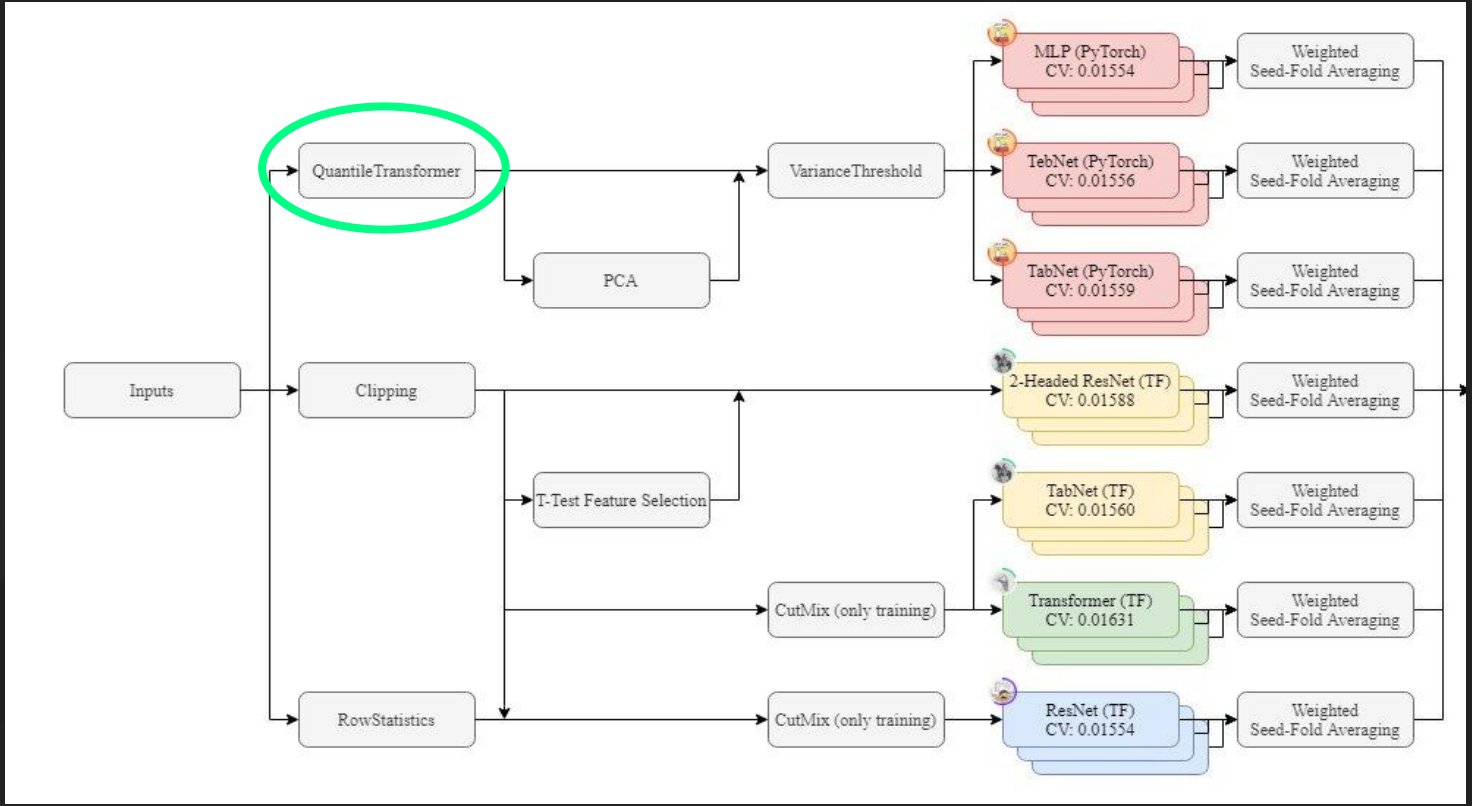


/ Preprocessing of numerical feats

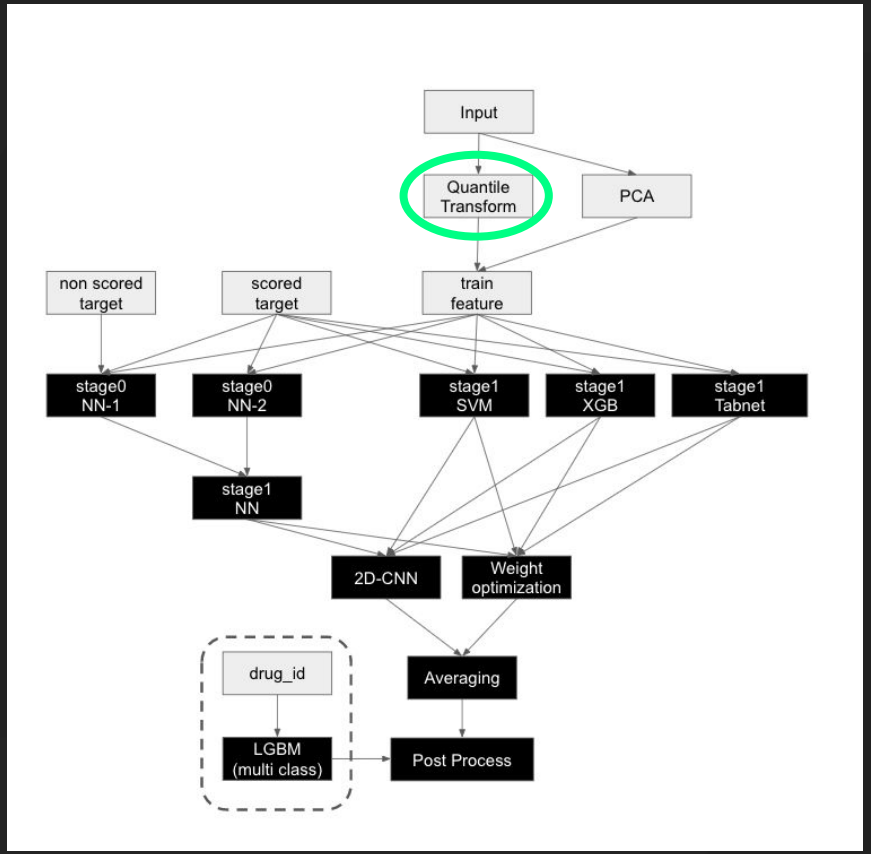
The rule of thumb is:

- Tree models: Does not need anything
- Other models: Scaling or Normalization

Kaggle example



Kaggle example



/ Q&A

What are your doubts?


Strive
SCHOOL