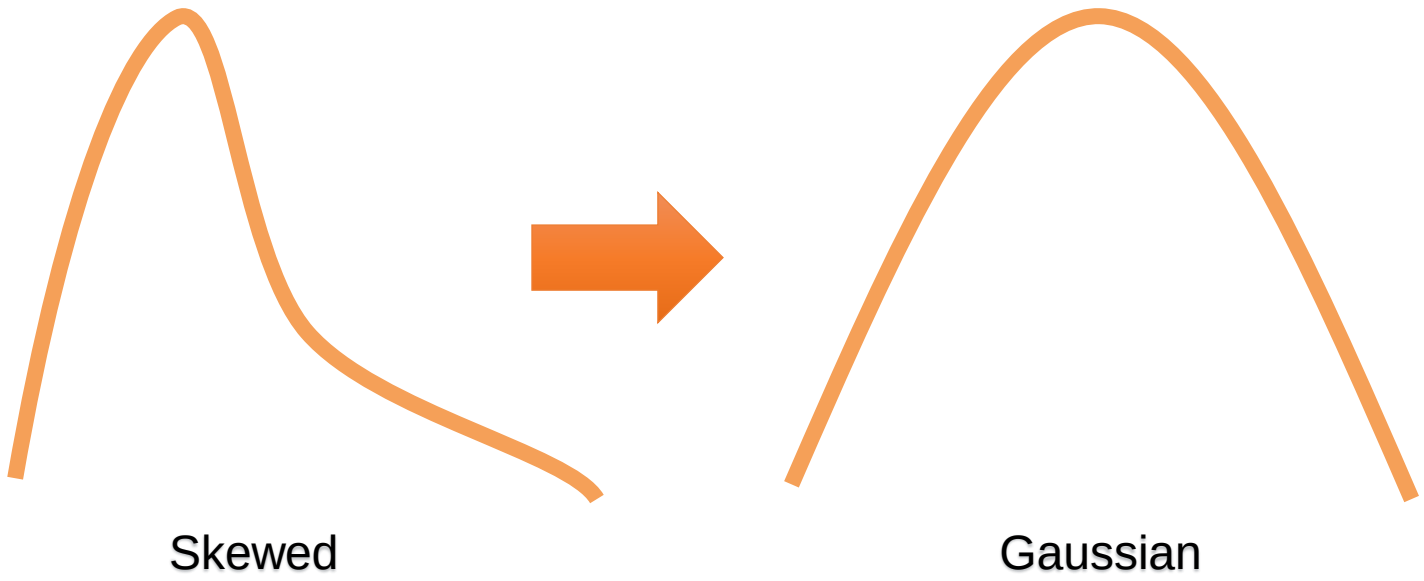
Variable Transformation

Variable transformation



Variable transformation

- Logarithmic
- Power
- Box-Cox
- Yeo-Johnson

Linear Model Assumptions

1. **Linearity**: The mean values of the outcome variable for each increment of the predictor(s) lie along a straight line. There is a linear relationship between predictors and target.
2. **Normally distributed errors**: the residuals (ε_i) are random, normally distributed with a mean of 0.
3. **Homoscedasticity**: At each level of the predictor variable(s), the variance of the residual terms should be constant.

When the assumptions are met

- The coefficients and parameters of the regression equation are said to be unbiased.

- The model is a good fit for the data.

When the assumptions are not met

We can’t fully trust the predictions of the model.

What can we do?

Sometimes, we can correct the failure in the assumptions by transforming the variables prior to the analysis.

This would improve the performance and reliability of the models.

Which variables can we transform?

We can transform the target variable itself when its distribution is skewed.

Transforming the predictor variables, very often helps meet the model assumptions when the raw data does not.

Which variables can we transform?

We can transform the target variable itself when its distribution is skewed.

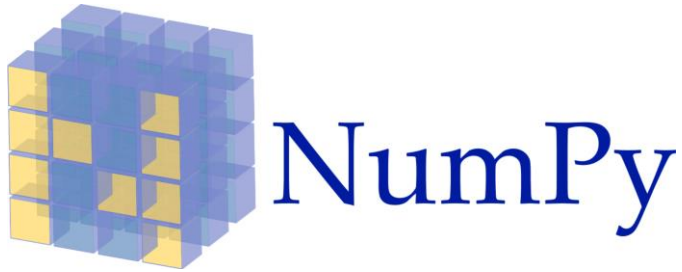
Transforming the predictor variables, very often helps meet the model assumptions when the raw data does not.

When should we transform variables?

- Variable transformations are usually applied when we analyze data through linear statistical tests like ANOVA and when training linear regression models.

- In other words, there is no need to transform variables when training non-linear models like decision tree based algorithms, nearest neighbors, or neuronal networks

Variable transformation in Python



Feature-Engine

Content

For each lecture:

- Presentation and video
- Accompanying Jupyter notebook
  - Implementation in **numpy-scipy**
  - Implementation in **sklearn**
  - Implementation in **Feature-engine**

Train In Data

THANK YOU

www.trainindata.com