Standardization vs Normalization - Feature Scaling

- Unit - (years, cms, inches)
- Magnitude - (25 years, 6 cms, 12 inches)
- If you have many features you become subject to different units and magnitudes.

  Therefore it is important to provide scaled data for algorithms.
- The two most common techniques that are used are normalization, and standardization.
- **Normalization** - helps to scale down features between 0-1
- **Standardization -** will help to scale down data based on standard normal distribution.

  Mean is usually 0, and the standard deviation is usually 1.

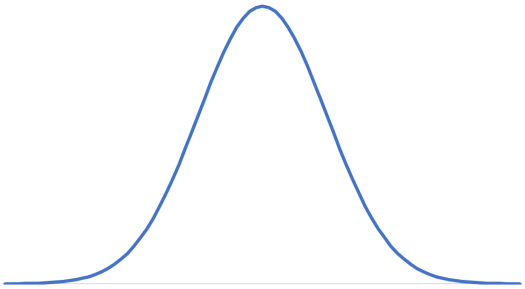
	Class	Alcohol	Malic
0	1	14.23	1.71
1	1	13.20	1.78
2	1	13.16	2.36
3	1	14.37	1.95
4	1	13.24	2.59

- Given the data frame above there are different magnitudes therefore we apply

  MinMaxScaler. This changes all features to be within the range of 0-1.

```
scaling.fit_transform(df[['Alcohol','Malic']])
[0.64473684, 0.18379447],
[0.35      , 0.61067194],
[0.7       , 0.49802372],
[0.47894737, 0.5       ],
[0.50789474, 0.53557312],
[0.72368421, 0.39920949],
[0.71052632, 0.71541502],
[0.63684211, 0.58498024],
[0.47105263, 0.51976285],
[0.67105263, 0.36363636],
[0.62368421, 0.76284585],
[0.30789474, 0.45256917],
```

- With standardization all features will be transformed in a way that it will have the properties of a standard normal distribution with mean = 0, and std = 1 .



```
from sklearn.preprocessing import StandardScaler

scaling=StandardScaler()

scaling.fit_transform(df[['Alcohol','Malic']])
[ 1.1109751 , -0.58917969],
[ 1.3580281 , -0.28397422],
[ 1.1603857 , -0.54429654],
[ 0.06099988, -0.54429654],
[ 1.02450655, -0.61610959],
[ 1.01215391, -0.52634327],
[ 0.95039066, -0.3916938 ],
[ 0.91333271, -0.59815632],
[ 0.69098501, -0.54429654],
[ 1.50625989, -0.57122643],
[ 0.35746347, -0.32885738],
[ 0.88862741, -0.81359548],
[-0.77898029, -1.25345042],
```

WHEN TO USE STANDARD NORMALIZATION & MINMAXSCALER?

- When using ML technique and algorithms that utilize euclidean distance where gradient descent is involved, scaling becomes necessary in (KNN, KnearestNeighbor, K Means Clustering, Deep learning, Artificial Network, Linear Regression, Logistic Regression)
- Scaling is unnecessary in boosting techniques such as values do not affect on an entropy basis. - (Decision tree, Random Forest, XGBoost)
- In Krish's experience, normalization has provided better scores than minmaxscaler.
- Based on article…
- Normalization is good to use when the distribution of data does not follow a gaussian distribution. It can be useful in algorithms that do not assume any distribution of the data like k nearest neighbors.
- Standardization can be helpful in cases where the data follows a gaussian distribution. Though this does not have to be necessarily true since standardization does not have a bounding range. So even if there are outliers in the data, they will not be affected by standardization.
- To conclude you can always start by fitting a model to raw, normalized, and standardized data and compare the performance for best results.