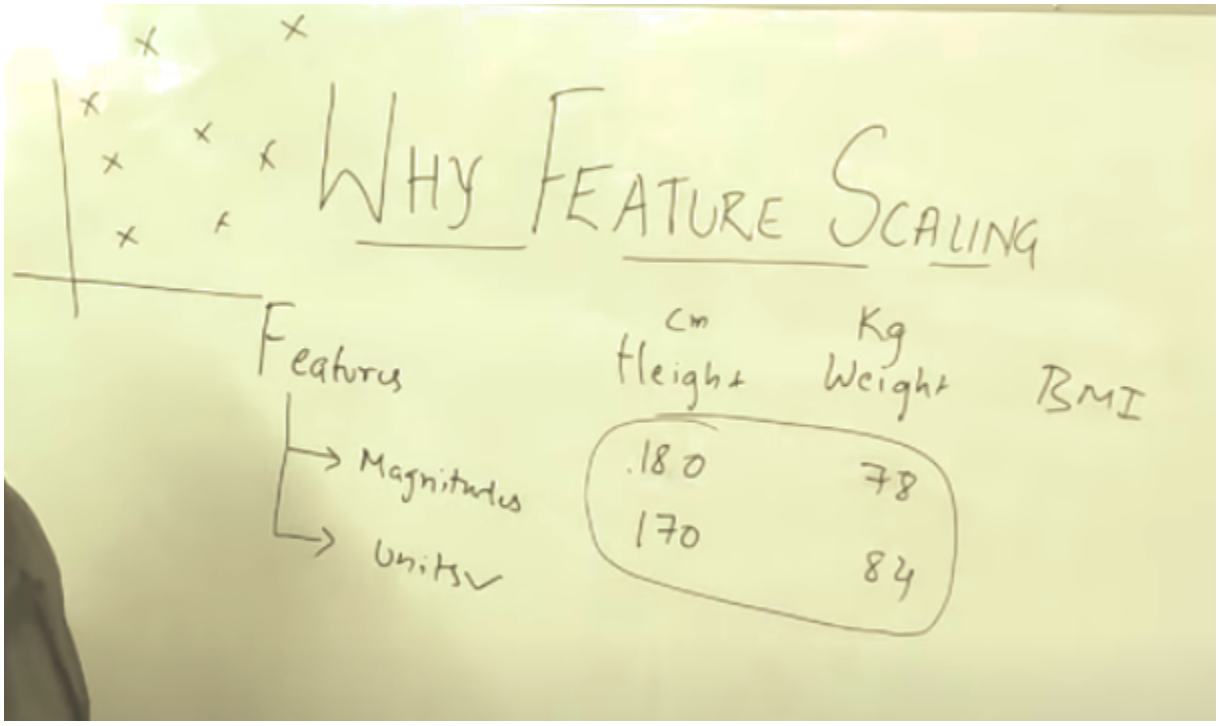
Feature Scaling SKlearn

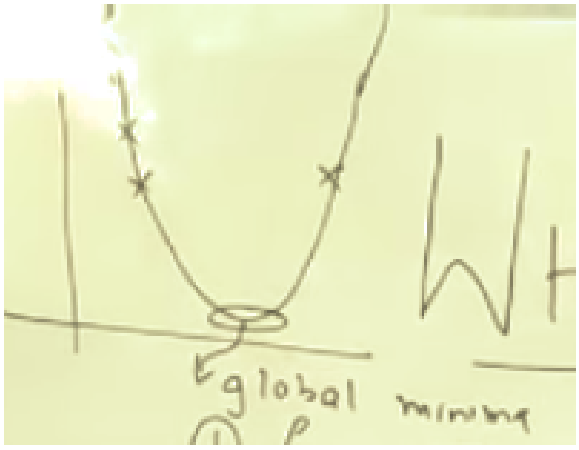



- Given features like height and weight we want to predict body mass index.
- With our independent features we have two things, magnitude and units
- Height magnitude - 180
- Height units - centimeters
- Weight magnitude - 78
- Weight units - kilograms
- If we take the magnitude values as is and apply a machine learning algorithm such as K nearest neighbor, we have a problem. KNN works off of euclidean distance, therefore the different units can not be plotted on the same graph. They are not of the same type, so their magnitudes must be scaled down to an equal playing field. We must apply a scaling technique.

WHAT ALGORITHMS NEED SCALING?

- Linear regression
- In linear regression we want to minimize our cost function aka the squared error function. A high cost function means that our prediction was really off, hence, the focus on minimizing the cost function, thereby providing an accurate prediction model.

- The coefficients are created on the basis of Ordinary Least Squared (OLS) and sometimes gradient descent. The values of the coefficients should be able to converge at the global minima. If we perform scaling it may happen in such a way where the randomized coefficients may be already very near to the global minima. If we do not scale then the random initial point may be very far away.
- If we scale down the value then convergence will happen quickly.




- K Means Clustering K Nearest Neighbor
- In these algorithms we use euclidean distance, so if we have independent variables of different units like kg, and cm it will throw off our classifications. With scaling it will allow our algorithm to be run much faster.

WHEN SHOULD WE NOT APPLY FEATURE SCALING?

- Decision tree, random forest, XG boost (ensemble techniques)
- These are based off of probability and entropy, scaling does not impact here because we are not basing it off distances, or gradient descent, ols.

STANDARDIZATION VS NORMALIZATION

- Unit and Magnitude and the two most important properties.
- With age the unit is the number of years, the magnitude is the value.
- Each and every feature there must be units and magnitude.
- If there are many features we are subject to different unit types. Therefore it is necessary for the data to be scaled down to be equivalent.
- Normalization helps to scale down the feature between 0 - 1.
- Standardization helps to scale down a feature based on standard normal distribution. The mean is usually 0 and the standard deviation is usually 1.

NORMALIZATION

## Normalization (min-max Normalization) ¶

In this approach we will scale down the values of the features between 0 to 1.

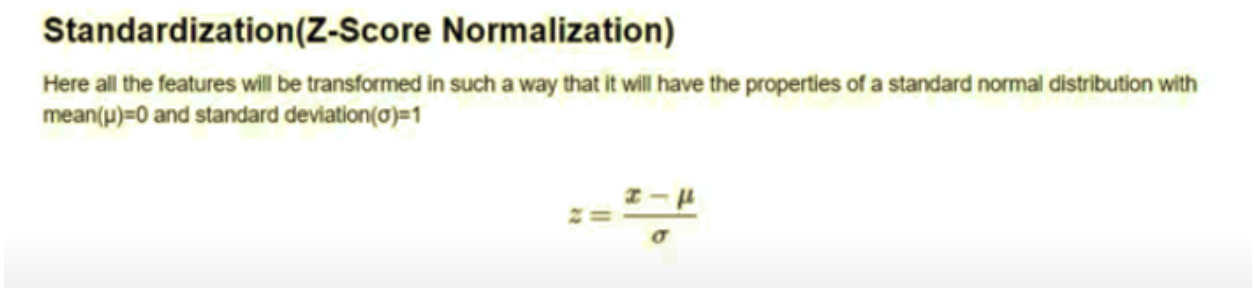
$$X_{norm} = \frac{X - X_{min}}{X_{max} - X_{min}}$$

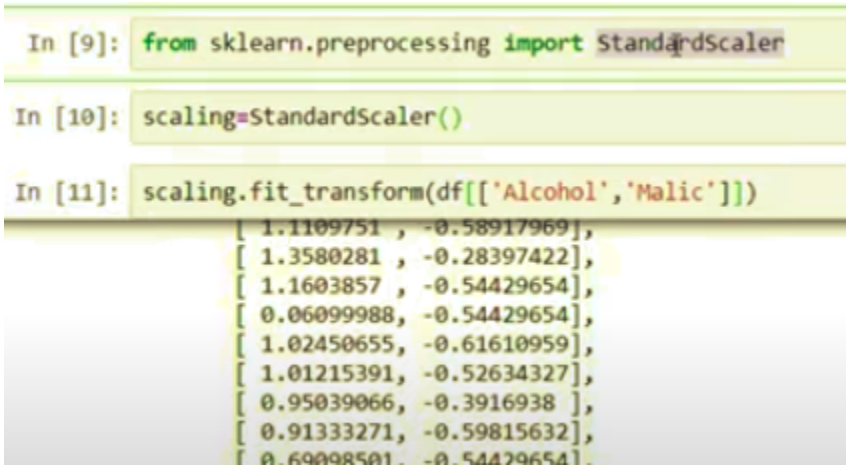
| | Class | Alcohol | Malic |
|---|---|---|---|
| 0 | 1 | 14.23 | 1.71 |
| 1 | 1 | 13.20 | 1.78 |
| 2 | 1 | 13.16 | 2.36 |
| 3 | 1 | 14.37 | 1.95 |
| 4 | 1 | 13.24 | 2.59 |

```
In [4]: from sklearn.preprocessing import MinMaxScaler

In [5]: scaling=MinMaxScaler()

In [8]: 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],
```

- The values are completely different. Therefore I pass in the min max scaler.

STANDARDIZATION




- Mean is equal to zero, and standard deviation equal to 1, which we pass in any given number.
- In this scenario we get a bell curve back.

WHEN TO USE STANDARDIZATION (STANDARDSCALER) OR NORMALIZATION (MINMAXSCALER)

- Most of the ml algorithms that involve euclidean distance, or for algorithms that have gradient descent involved- you are looking to identify a certain point, and by scaling down values it allows a global minima or relationship to be calculated much faster and with less computation power.

- Decision tree it ensemble methods it is not mandatory. If you keep the value high or low it is still creating branches based off of probability and entropy.
- In Krish's personal experience standardscaler has performed better than minmax scaler.
- For most of the deep learning techniques you must scale down your values between 0 and 1. Therefore making min max scaler the go to. Example of images with pixels between 0-255 pixels should be scaled to 0-1.