Artificial Intelligence

Zeeshan Abbas (Visiting Lecturer)

zeeshanabbas5@hotmail.com

TR/AI at main · ZeeshanAbbas/TR (github.com)

Outline

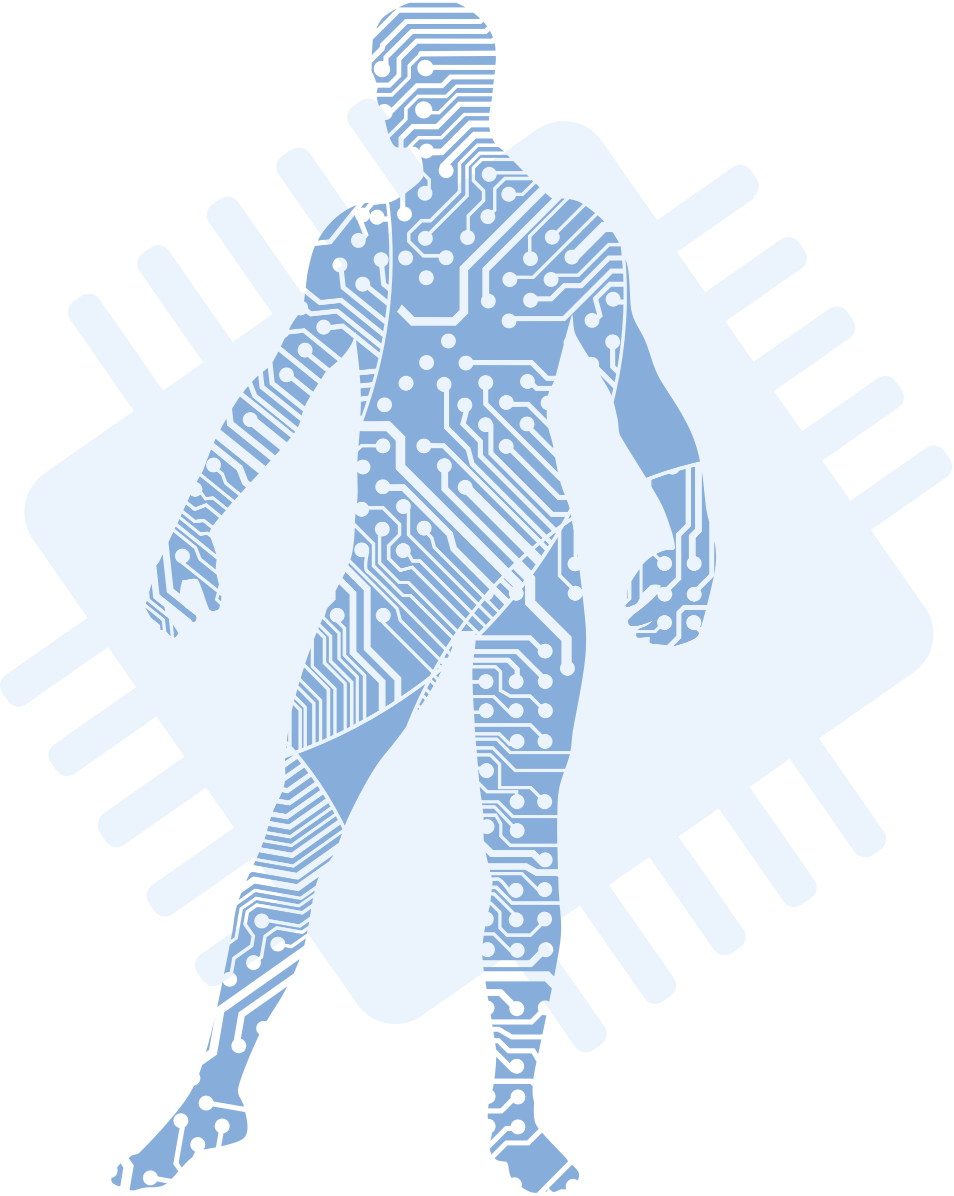
01 **Introduction to ML**

02 **Data Collection**

03 **Data Preprocessing**

- **Binarization**
- **Mean removal**
- **Scaling**
- **Normalization**

04 **Label Encoding**



Machine Learning

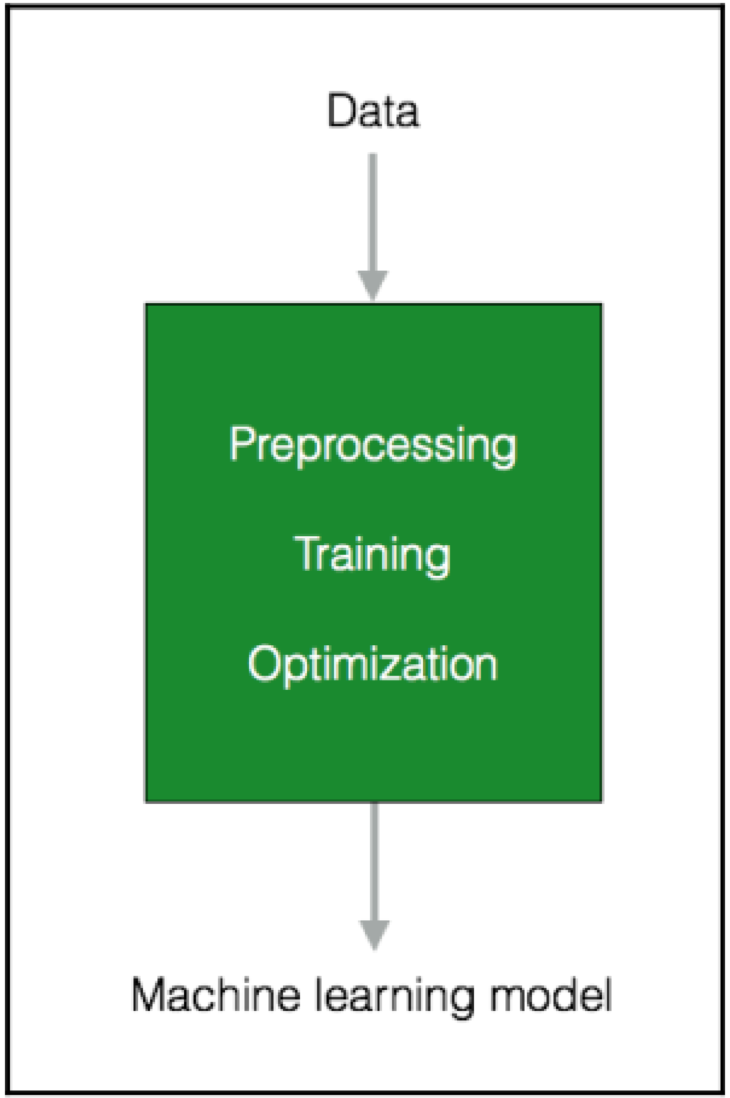
“The Science and engineering of making Intelligent machine, especially intelligent computer programs is called Machine learning.”

John McCarthy

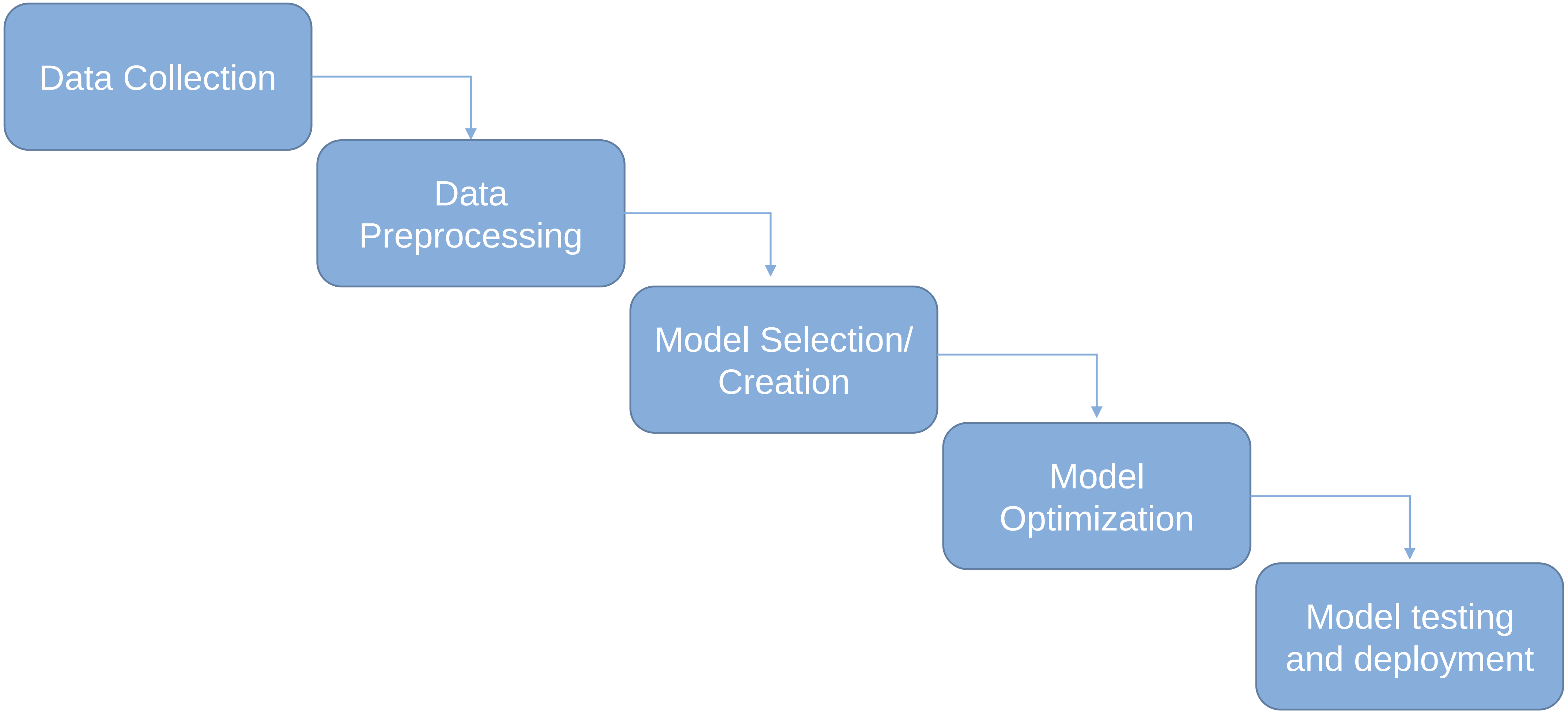
Machine Learning

- **Machine learning** is the most popular form of AI out there.
- **Machine learning** is about developing learning models, that can learn from data.
- These **learning models,** is used to perform predictions on unknown data.
- One of the main constraints here is that these programs are limited to the power of the data.
- If the dataset is small, then the learning models would be limited as well.
- Let's see what a typical machine learning system looks like:

Machine Learning



Machine Learning Model Implementation



Data Collection

Data Preprocessing

Model Selection/ Creation

Model Optimization

Model testing and deployment

Machine Learning Model Implementation

Data Collection:

- **Sklearn Data sets:**

Python Package which contain huge amount of datasets.

- **UCI Data Repository:**

https://archive-beta.ics.uci.edu/

- **Kaggle Data Repository:**

https://kaggle.com/

Machine Learning Model Implementation

Data Preprocessing:

It is also some times called data wrangling, data munging etc. Why its important?

- Machine learning algorithms expect data to be formatted in a certain way before they start the training process.
- We deal with a lot of raw data in the real world.
- In order to prepare the data for ingestion by machine learning algorithms, we have to preprocess it and convert it into the right format.

Machine Learning Model Implementation

Data Preprocessing:

There are several different preprocessing techniques, we discuss the following techniques:

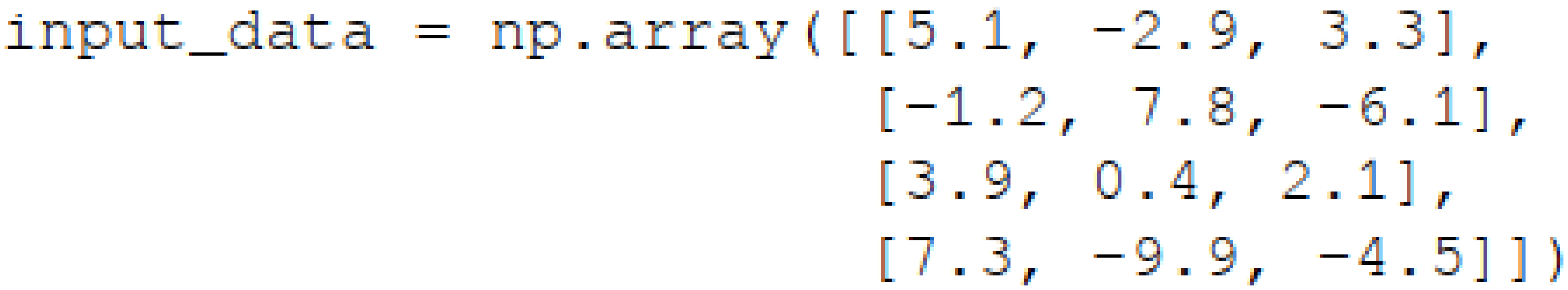
- Binarization
- Mean removal
- Scaling
- Normalization

Machine Learning Model Implementation

Binarization:

This process is used when we want to convert our numerical values into Boolean values.

e.g.



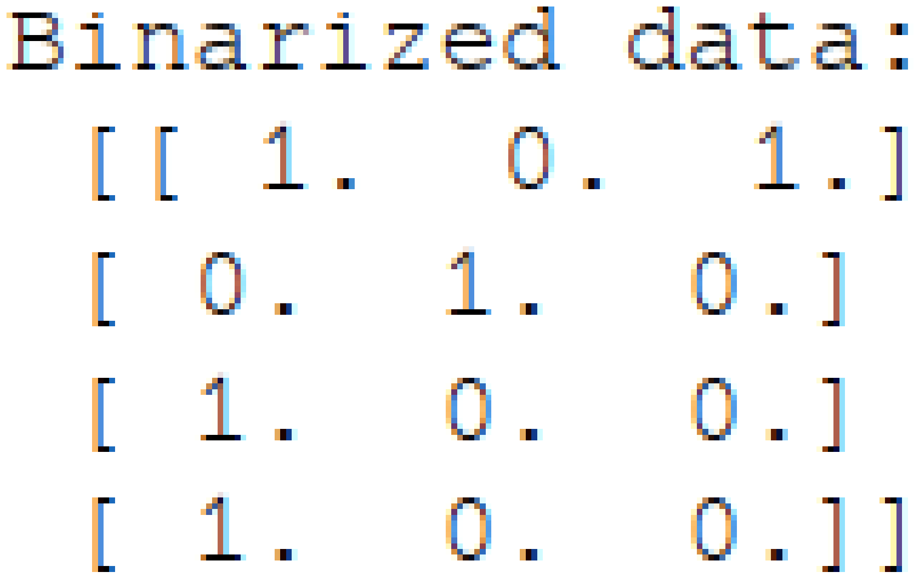
```
input_data = np.array([[5.1, -2.9, 3.3],
                       [-1.2, 7.8, -6.1],
                       [3.9, 0.4, 2.1],
                       [7.3, -9.9, -4.5]])
```

Let's use an inbuilt method to binarize input data using 2.1 as the **threshold value**.

Machine Learning Model Implementation

Binarization:

```
data_binarized =
preprocessing.Binarizer(threshold=2.1).transform(input_data)
```



```
Binarized data:
 [[ 1.   0.   1.]
 [ 0.   1.   0.]
 [ 1.   0.   0.]
 [ 1.   0.   0.]]
```

Machine Learning Model Implementation

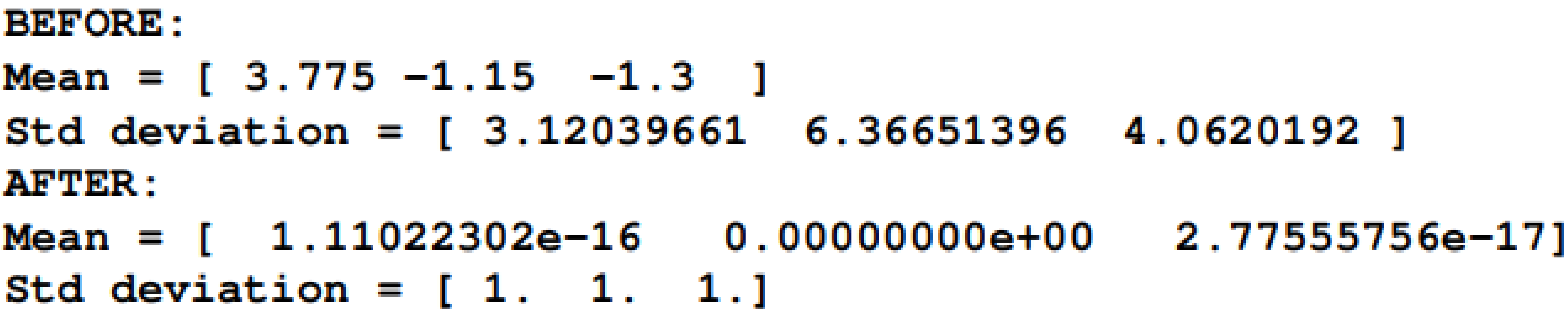
Mean removal:

- Removing the mean is a common preprocessing technique used in machine learning.
- It's usually useful to remove the mean from our feature vector, so that each feature is centered towards zero.
- We do this in order to remove bias from the features in our feature vector.

Machine Learning Model Implementation

Mean removal:

- Removing the mean is a common preprocessing technique used in machine learning.



```
BEFORE:
Mean = [ 3.775 -1.15  -1.3   ]
Std deviation = [ 3.12039661  6.36651396  4.0620192 ]
AFTER:
Mean = [  1.11022302e-16   0.00000000e+00   2.77555756e-17]
Std deviation = [ 1.  1.  1.]
```

As seen from the values obtained, the mean value is very close to 0 and standard deviation is 1.

Machine Learning Model Implementation

Scaling:

- In our feature vector, the value of each feature can vary between many random values.
- So it becomes important to scale those features so that it is a level playing field for the machine learning algorithm to train on.
- We don't want any feature to be artificially large or small just because of the nature of the measurements.

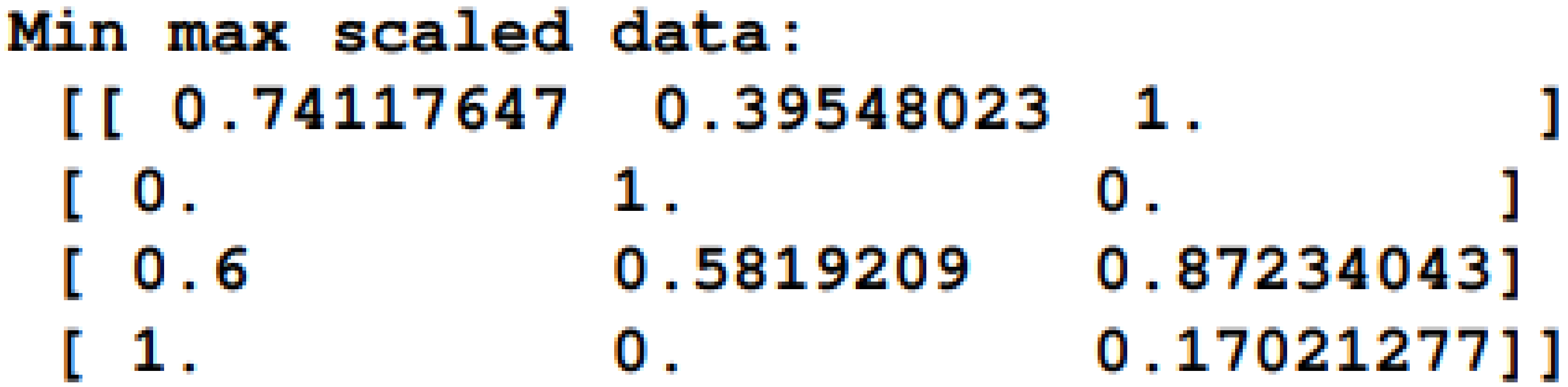
Machine Learning Model Implementation

Scaling:

- In our feature vector, the value of each feature can vary between many random values.
- So it becomes important to scale those features so that it is a level playing field for the machine learning algorithm to train on.
- We don't want any feature to be artificially large or small just because of the nature of the measurements.

Machine Learning Model Implementation

Scaling:



```
Min max scaled data:
 [[ 0.74117647  0.39548023  1.        ]
 [ 0.          1.          0.        ]
 [ 0.6         0.5819209   0.87234043]
 [ 1.          0.          0.17021277]]
```

Each row is scaled so that the maximum value is 1 and all the other values are relative to this value.

Machine Learning Model Implementation

Normalization:

- We use the process of normalization to modify the values in the feature vector so that we can measure them on a common scale.
- In machine learning, we use many different forms of normalization.
- Some of the most common forms of normalization aim to modify the values so that they sum up to 1

Machine Learning Model Implementation

Normalization:

- **L1 Normalization:**
  - L1 normalization, refers to Least Absolute Deviations, works by making sure that the sum of absolute values is 1 in each row.
- **L2 Normalization:**
  - L2 normalization, refers to least squares, works by making sure that the sum of squares is 1.

Machine Learning Model Implementation

Normalization:

- In general, **L1 normalization** technique is considered more robust than L2 normalization technique.
- **L1 normalization** technique is robust because it is resistant to outliers in the data.
- A lot of times, data tends to contain outliers and we cannot do anything about it.

Machine Learning Model Implementation

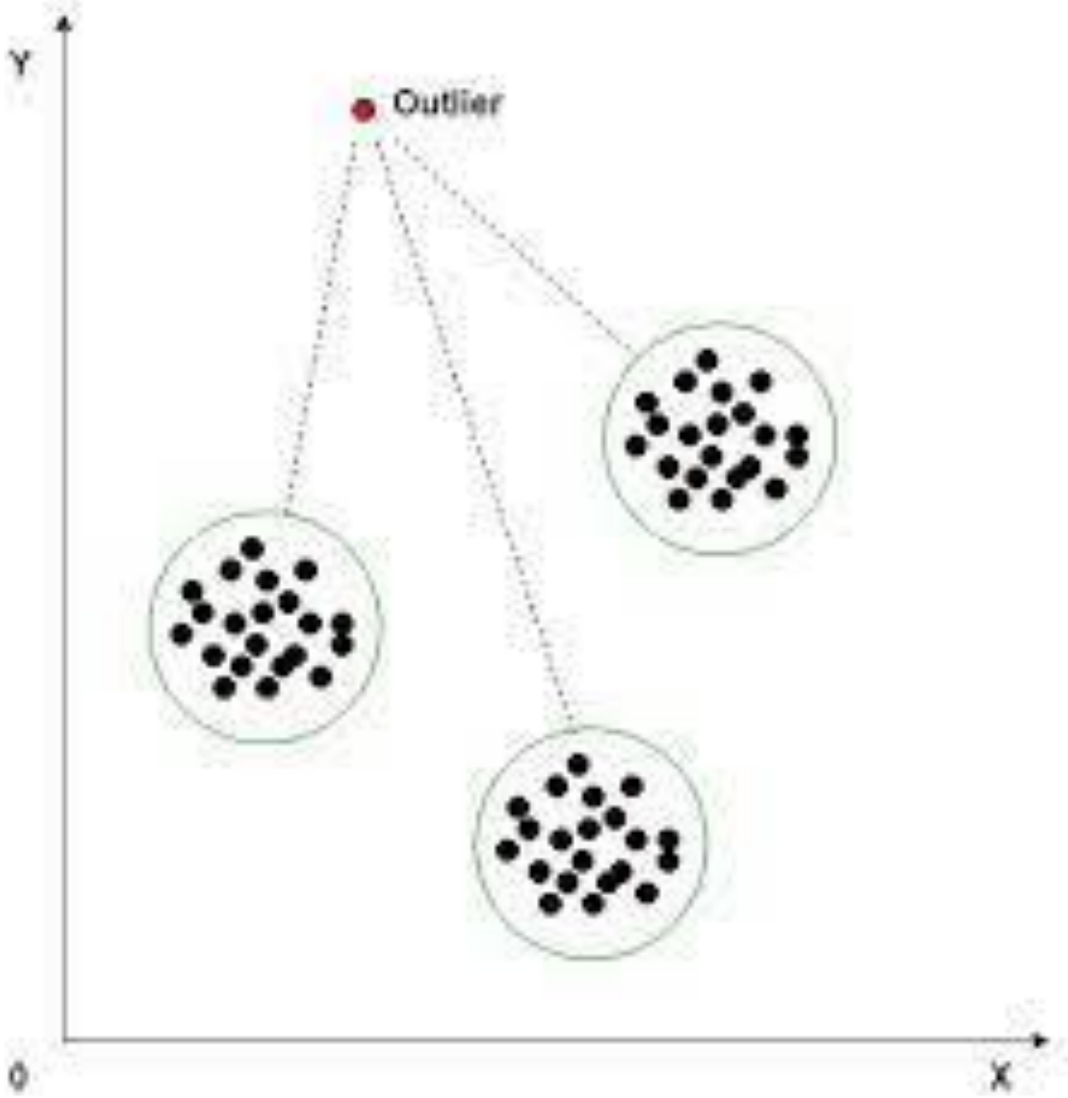
Normalization:

- We usually want to use techniques that can safely and effectively ignore them during the calculations.
- If we are solving a problem where outliers are important, then maybe L2 normalization becomes a better choice.

Machine Learning Model Implementation

Outliers:

- Outliers are those data points that are significantly different from the rest of the dataset.
- They are often abnormal observations that skew the data distribution, and arise due to inconsistent data entry, or erroneous observations.



Machine Learning Model Implementation

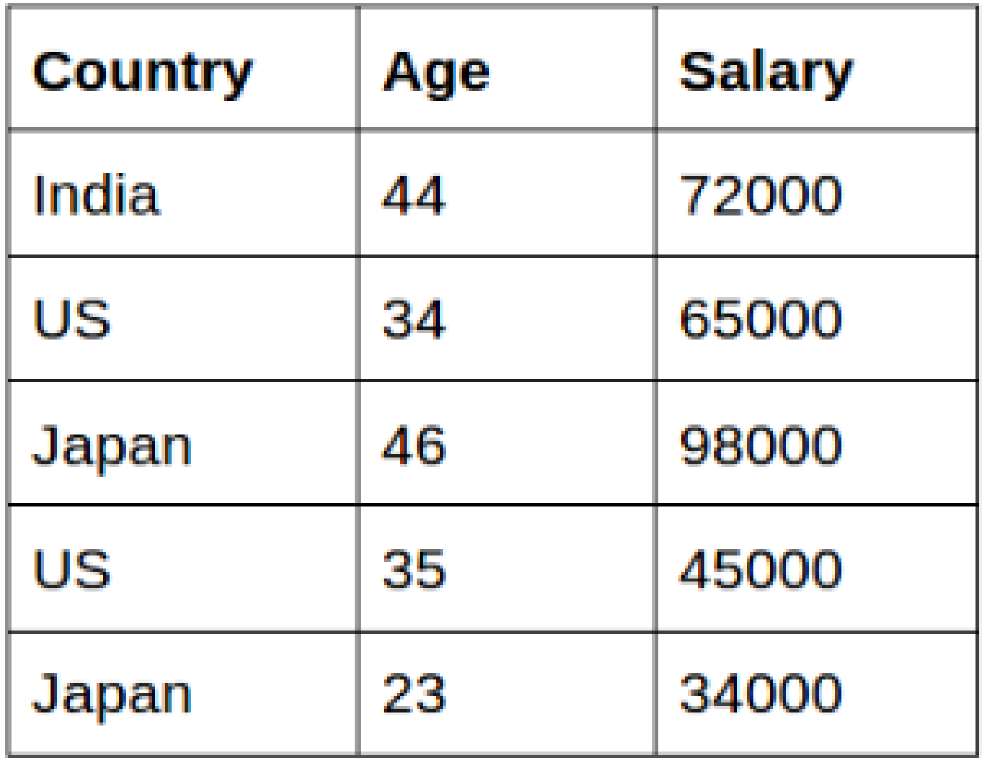
Label Encoding:

Label Encoding is a popular encoding technique for handling categorical variables. In this technique, each label is assigned a unique integer based on alphabetical ordering.

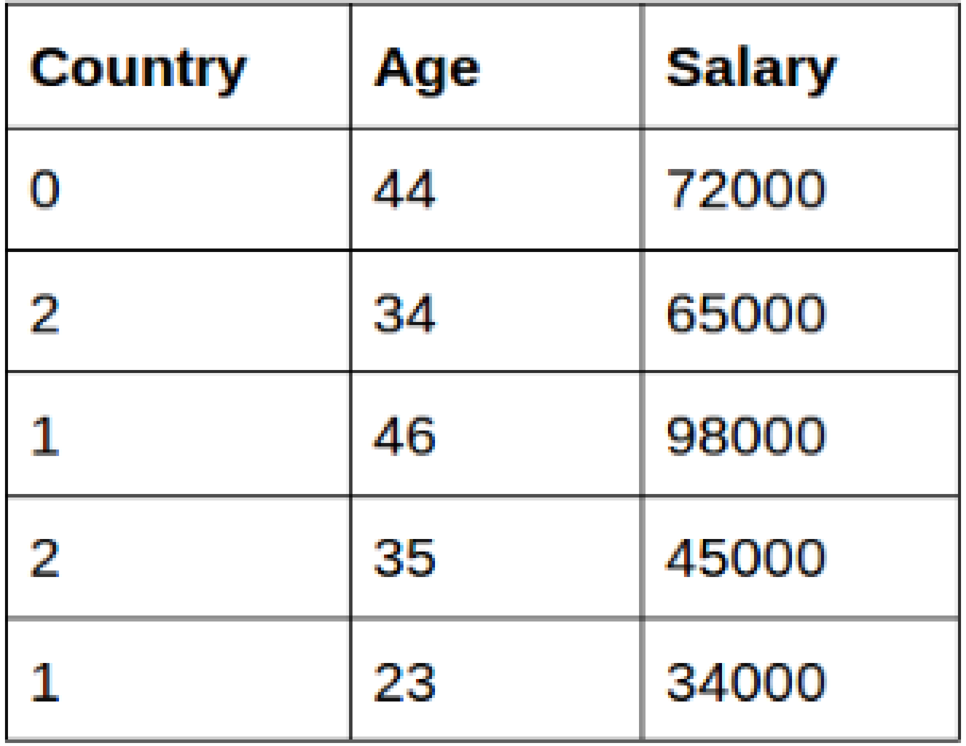
- When we perform classification, we usually deal with a lot of label data.
- Most machine learning algorithms especially **sklearn** expect them to be numbered.
- Label encoding is a process of transforming word labels into numerical form.

Machine Learning Model Implementation

Label Encoding:



Country	Age	Salary
India	44	72000
US	34	65000
Japan	46	98000
US	35	45000
Japan	23	34000



Country	Age	Salary
0	44	72000
2	34	65000
1	46	98000
2	35	45000
1	23	34000

Machine Learning Model Implementation

Label Encoding:

- ❖ In the above scenario, the Country names do not have an order or rank. But, when label encoding is performed, the country names are ranked based on the alphabet. Due to this, there is a very high probability that the model captures the relationship between countries such as India < Japan < and the US.