AiEnsured

Ensuring comprehensive validation of AI systems

An article on

Data Preprocessing

On Fuel efficiency Data Set

Submitted by: Rayapureddi Venkata Sri Sai Subhash

Intern in Machine Learning May 2023-July 2023

(rayapureddi.subhash@testaing.com)

Data Preprocessing:

Data preprocessing is a critical step in the data analysis and machine learning pipeline. It refers to the process of cleaning, transforming, and organizing raw data into a format suitable for further analysis or model training. The quality of the data and the effectiveness of any machine learning algorithm heavily depend on the quality of the preprocessing.

I have taken the fuel efficiency dataset which contains the following attributes 'MPG', 'Cylinders', 'Displacement', 'Horsepower', 'Weight', 'Acceleration', 'Model Year', 'Origin' of automobiles in late-1970s and early 1980s.

We can get the dataset from this below link.

'http://archive.ics.uci.edu/ml/machine-learning-databases/auto-mpg/auto-mpg.data'

In the preprocessing we need to do the Data Cleaning, Data Transformation, Feature Selection, Handling Categorical Data, Data Reduction, Handling Outliers and etc.

Importing the libraires:

For the preprocessing we need to import some libraries as shown in the code snippet mentioned below.

```
import matplotlib.pyplot as plt
import numpy as np
import tensorflow as tf
import pandas as pd
import seaborn as sns
```

Loading the Dataset:

First download and import the dataset using pandas.

```
url = 'http://archive.ics.uci.edu/ml/machine-learning-databases/auto-mpg/auto-mpg.data'
column_names = ['MPG', 'Cylinders', 'Displacement', 'Horsepower', 'Weight','Acceleration', 'Model Year', 'Origin']
dataset = pd.read_csv(url, names=column_names,na_values='?', comment='\t',sep=' ', skipinitialspace=True)
```

By using pandas we can load the dataset.

Cleaning the Dataset:

The dataset contains a few unknown values.

By using some pandas functions we can clean the data.

```
dataset.isna().sum()
```

Dropping the null values in the data.

```
dataset = dataset.dropna()
```

Getting the number of rows and columns in the final dataset.

```
dataset.shape
```

Data Visualization:

Next, let's print a heatmap that shows the correlation between the numerical columns in the dataset.

```
plt.rcParams["figure.figsize"] = [8, 6]
sns.heatmap(dataset.corr())
```