EXPERIMENT NO. 5

Aim: Implementation of Data Discretization and Visualization

Software Used: Java/C/Python

Theory: Data discretization refers to a method of converting a huge number of data values into smaller ones so that the evaluation and management of data become easy. In other words, data discretization is a method of converting attributes values of continuous data into a finite set of intervals with minimum data loss. There are two forms of data discretization first is supervised discretization, and the second is unsupervised discretization. Supervised discretization refers to a method in which the class data is used. Unsupervised discretization refers to a method depending upon the way which operation proceeds. It means it works on the top-down splitting strategy and bottom-up merging strategy.

Example

Suppose we have an attribute of Age with the given values

Age	1,5,9,4,7,11,14,17,13,18, 19,31,33,36,42,44,46,70,74,78,77

Table before Discretization

Attribute	Age	Age	Age	Age
	1,5,4,9,7	11,14,17,13,18,19	31,33,36,42,44,46	70,74,77,78
After Discretization	Child	Young	Mature	Old

Some techniques of data discretization:

Histogram analysis

Histogram refers to a plot used to represent the underlying frequency distribution of a continuous data set. Histogram assists the data inspection for data distribution. For example, Outliers, skewness representation, normal distribution representation, etc.

Binning

Binning refers to a data smoothing technique that helps to group a huge number of continuous values into smaller values. For data discretization and the development of idea hierarchy, this technique can also be used.

Cluster Analysis

Cluster analysis is a form of data discretization. A clustering algorithm is executed by dividing the values of x numbers into clusters to isolate a computational feature of x.

Data discretization using decision tree analysis

Data discretization refers to a decision tree analysis in which a top-down slicing technique is used. It is done through a supervised procedure. In a numeric attribute discretization, first, you need to select the attribute that has the least entropy, and then you need to run it with the help of a recursive process. The recursive process divides it into various discretized disjoint intervals, from top to bottom, using the same splitting criterion.

Data discretization using correlation analysis

Discretizing data by linear regression technique, you can get the best neighbouring interval, and then the large intervals are combined to develop a larger overlap to form the final 20 overlapping intervals. It is a supervised procedure.

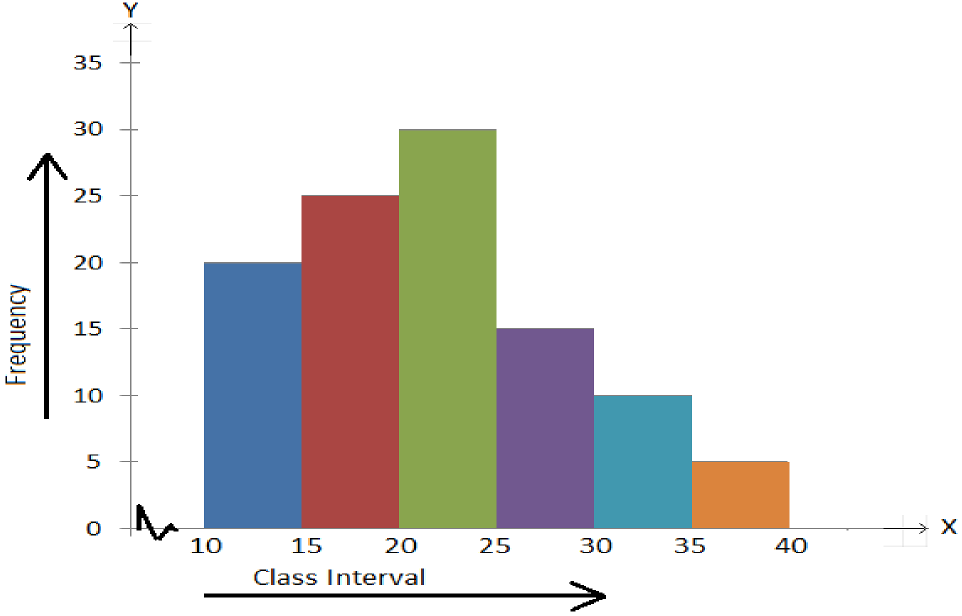
Data visualization

Data visualization is actually a set of data points and information that are represented graphically to make it easy and quick for user to understand. Data visualization is good if it has a clear meaning, purpose, and is very easy to interpret, without requiring context. Tools of data visualization provide an accessible way to see and understand trends, outliers, and patterns in data by using visual effects or elements such as a chart, graphs, and maps.

Data Visualization Techniques:

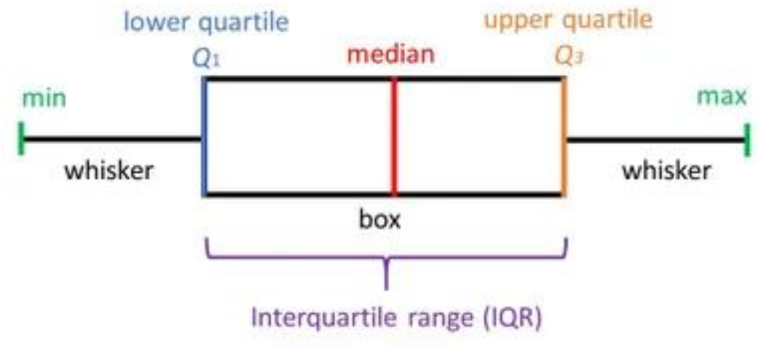
- **Histogram**

  A histogram is a graphical display of data using bars of different heights. In a histogram, each bar groups numbers into ranges. Taller bars show that more data falls in that range. A histogram displays the shape and spread of continuous sample data.



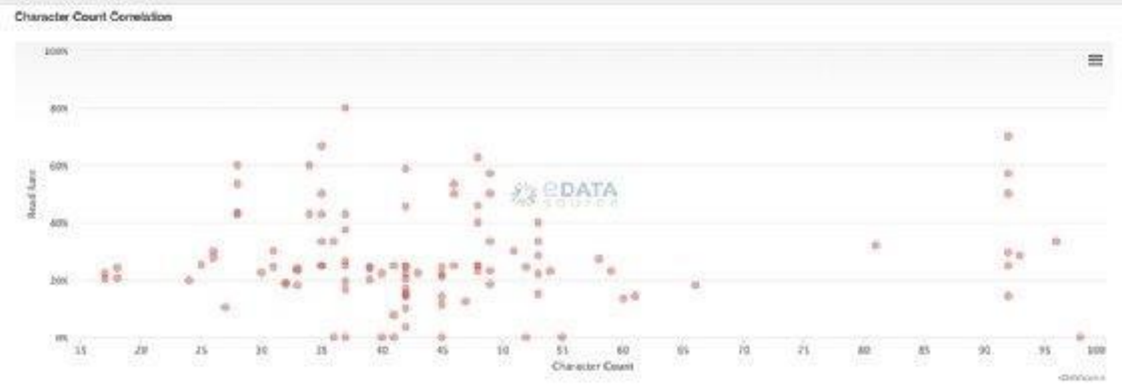
- **Boxplots**

  A box plot is a graph that gives you a good indication of how the values in the data are spread out. Although box plots may seem primitive in comparison to a histogram or density plot, they have the advantage of taking up less space, which is useful when comparing distributions between many groups or datasets.



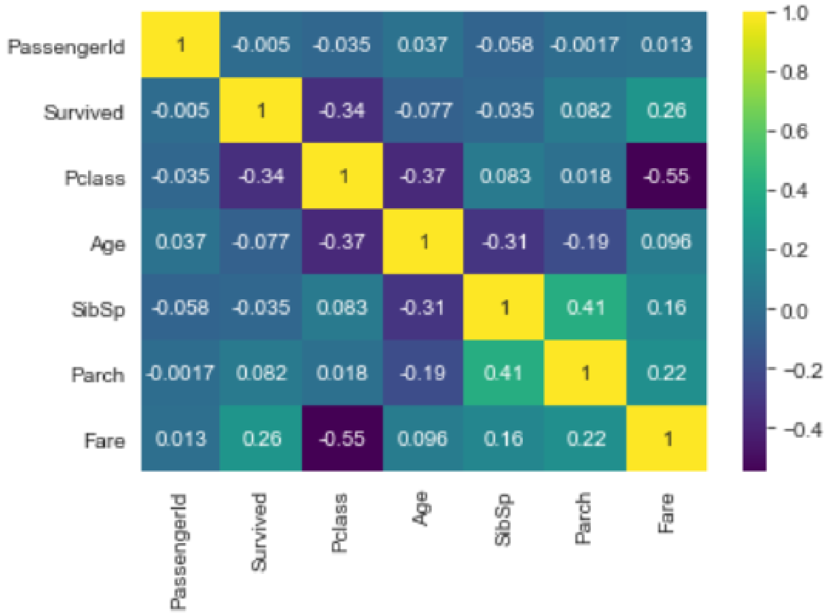
- **Scatter plots**

  Scatter plots are useful to display the relative density of two dimensions of data. Well-designed ones quantify and correlate complex sets of data in an easy-to-read manner. Often, these charts are used to discover trends and data, as much as they are to visualize the data.



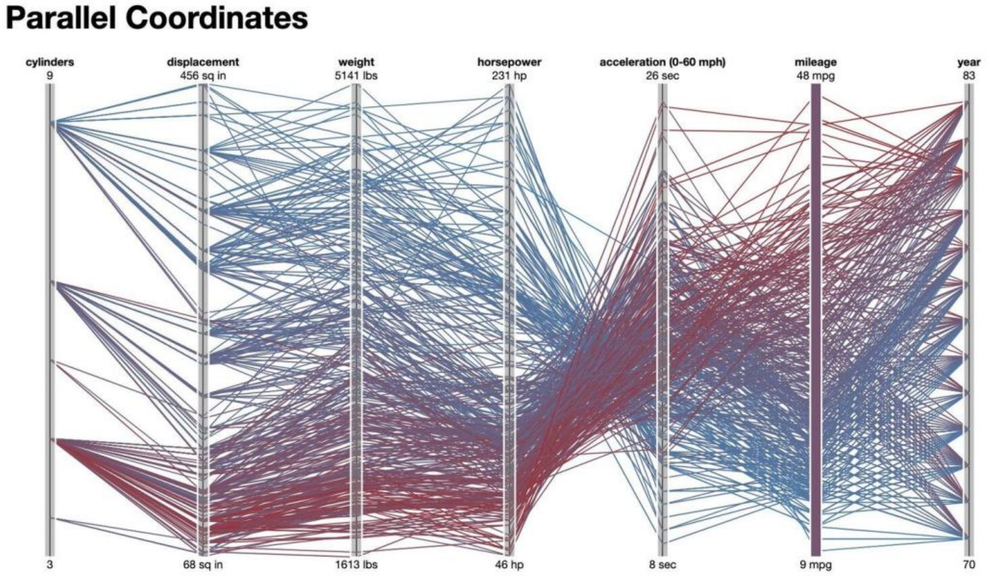
- **Matrix plots**

  These are the special types of plots that use two-dimensional matrix data for visualization. It is difficult to analyze and generate patterns from matrix data because of its large dimensions. So, this makes the process easier by providing color coding to matrix data.



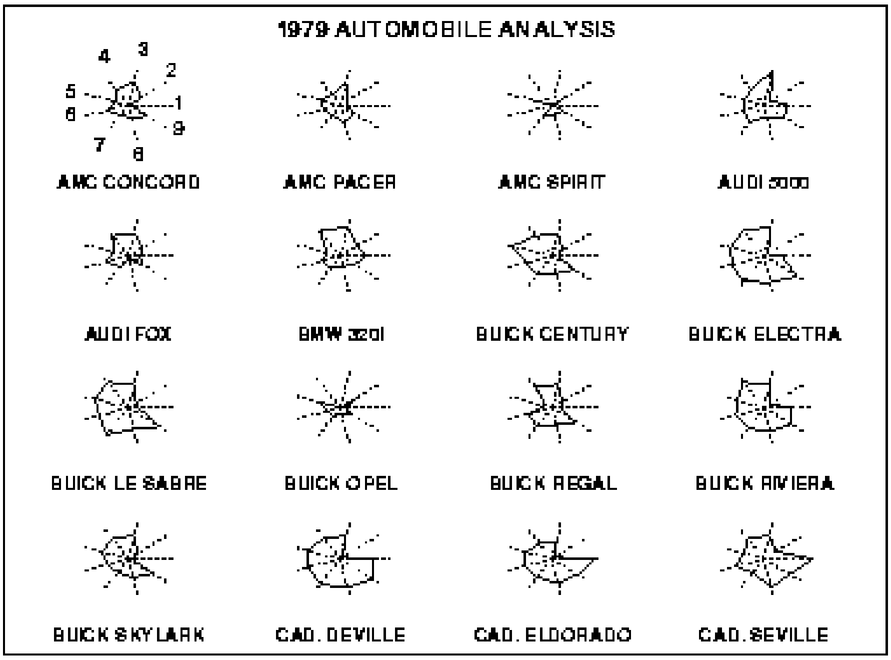
- **Parallel Coordinates**

  Parallel coordinates is a visualization technique used to plot individual data elements across many performance measures. Each of the measures corresponds to a vertical axis and each data element is displayed as a series of connected points along the measure/axes.



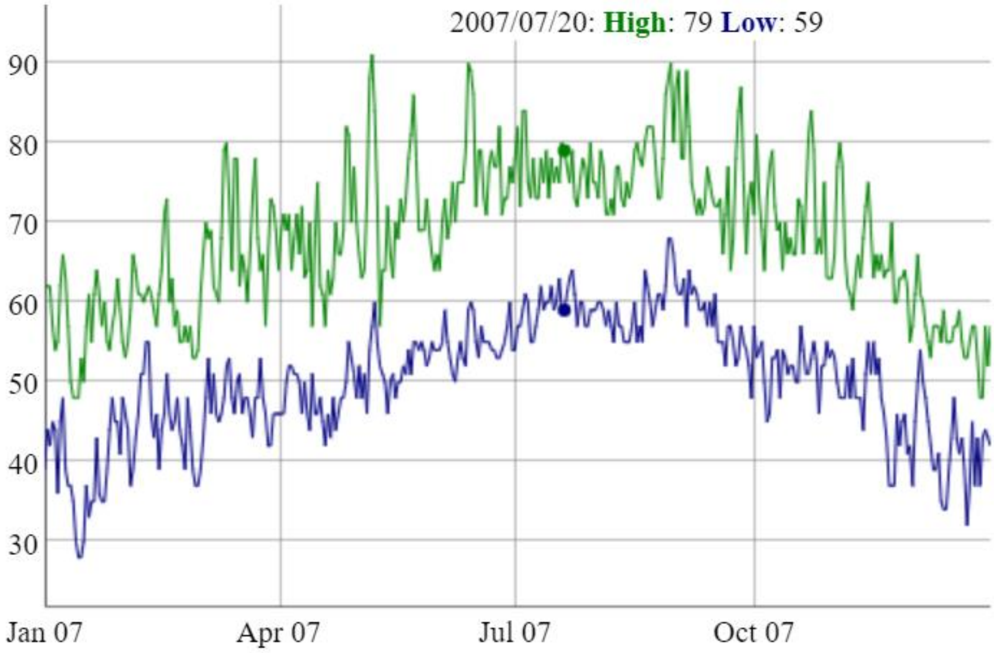
- **Star plots**

  The star plot (Chambers 1983) is a method of displaying multivariate data. Each star represents a single observation. Typically, star plots are generated in a multi-plot format with many stars on each page and each star representing one observation.Star plots are used to examine the relative values for a single data point.



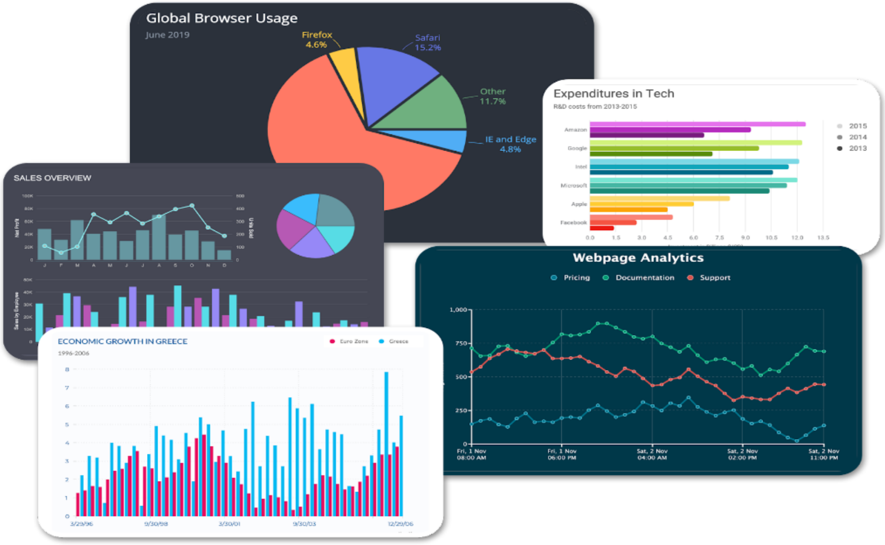
- **Dygraphs**

  dygraphs is an open source JavaScript library that produces produces interactive, zoomable charts of time series. It is designed to display dense data sets and enable users to explore and interpret them.



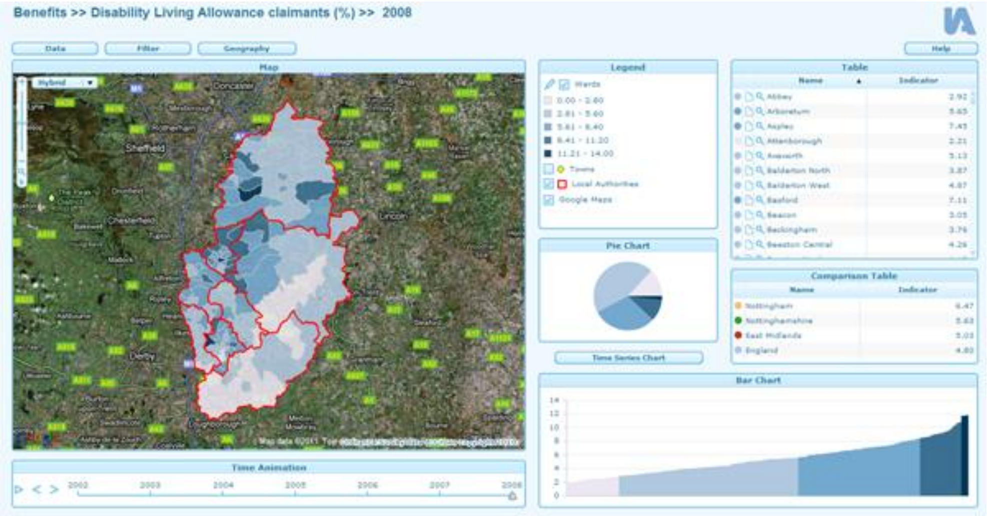
- **Zing chart**

JavaScript Charts in one powerful declarative library | ZingChart See what ZingChart's 50+ built-in chart types & modules can do for your data visualization projects. Create animated & interactive charts with hundreds of thousands of data records using the ZingChart JavaScript charting library.



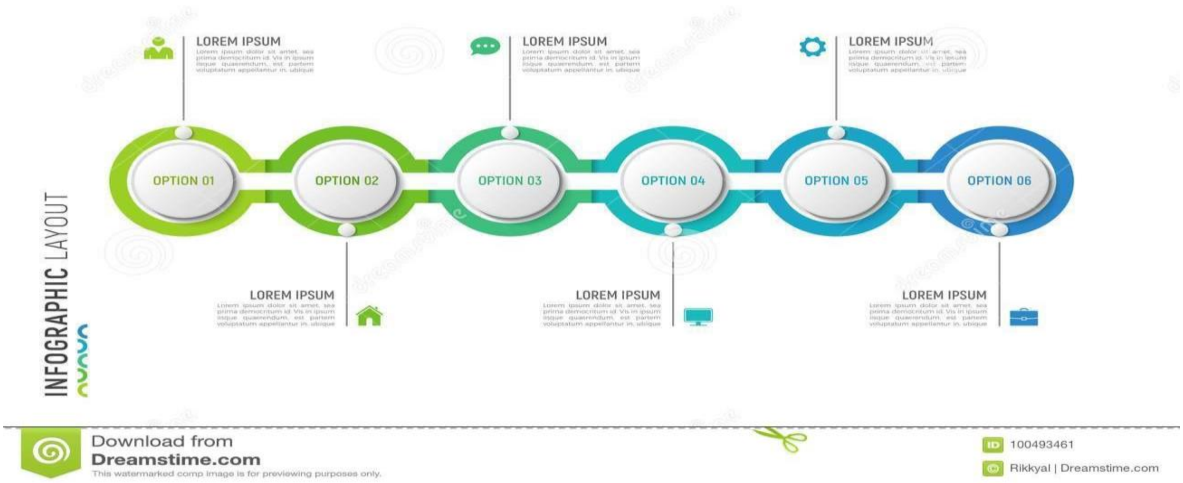
- **Instant Atlas**

The InstantAtlas team prepare and manage large statistical indicator data sets and deliver community information systems, local observatories and knowledge hub websites for clients as fully managed services. so that you can build your own services and sites using ArcGIS Online and WordPress.



- **Timeline**

A timeline is a great data visualization technique when you wish to show data in a chronological order and highlighting those important points in time. To create a Timeline, simply layout your data points along a PowerPoint shape, and mark the data off to visually see your overall project.



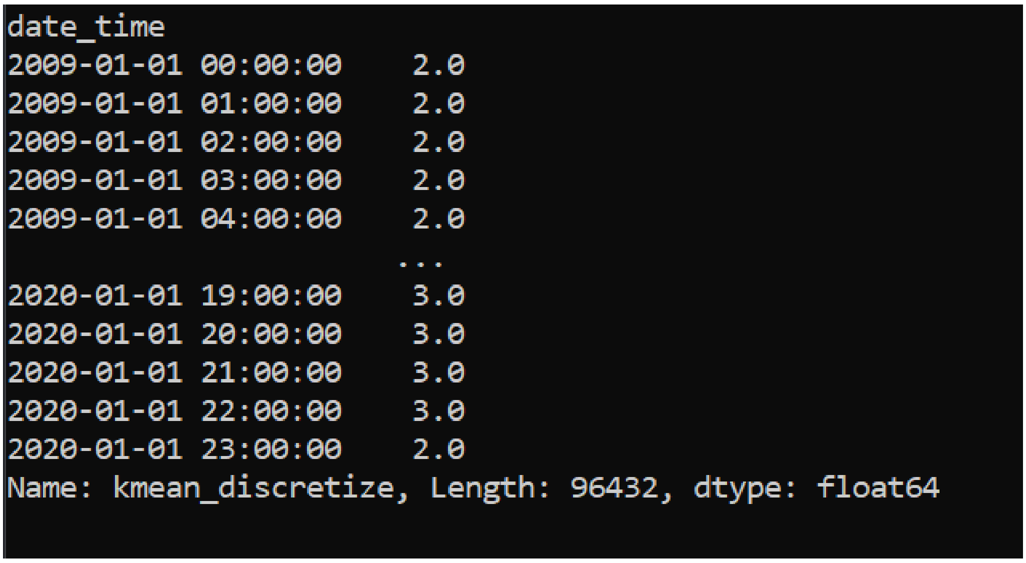
PROGRAM:

```
import sys
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt

import seaborn as sns
from statsmodels.tsa.arima_model import ARIMA
from statsmodels.tsa.stattools import adfuller, acf, pacf
from statsmodels.graphics.tsaplots import plot_acf, plot_pacf
from sklearn.preprocessing import KBinsDiscretizer

weather_df = pd.read_csv('./bombay.csv', parse_dates=['date_time'], index_col='date_time')
weather_condition =
(weather_df.sunHour.value_counts()/(weather_df.sunHour.value_counts().sum()))*100
weather_condition.plot.bar(figsize=(16,9))
plt.xlabel('Weather Conditions')
plt.ylabel('Percent')
def kmean_discretize(data, n):
   kmeans = KBinsDiscretizer(n_bins=n, encode='ordinal', strategy='kmeans')
   return kmeans.fit_transform(data.values.reshape(-1, 1)).flatten()
weather_df['kmean_discretize'] = kmean_discretize(weather_df['tempC'], 5)
print(weather_df['kmean_discretize'])
weather_df.plot(subplots=True, figsize=(20,12))
plt.show()
```

Output:



```
date_time
2009-01-01 00:00:00    2.0
2009-01-01 01:00:00    2.0
2009-01-01 02:00:00    2.0
2009-01-01 03:00:00    2.0
2009-01-01 04:00:00    2.0
                       ...
2020-01-01 19:00:00    3.0
2020-01-01 20:00:00    3.0
2020-01-01 21:00:00    3.0
2020-01-01 22:00:00    3.0
2020-01-01 23:00:00    2.0
Name: kmean_discretize, Length: 96432, dtype: float64
```

Figure 1 kmean discretize

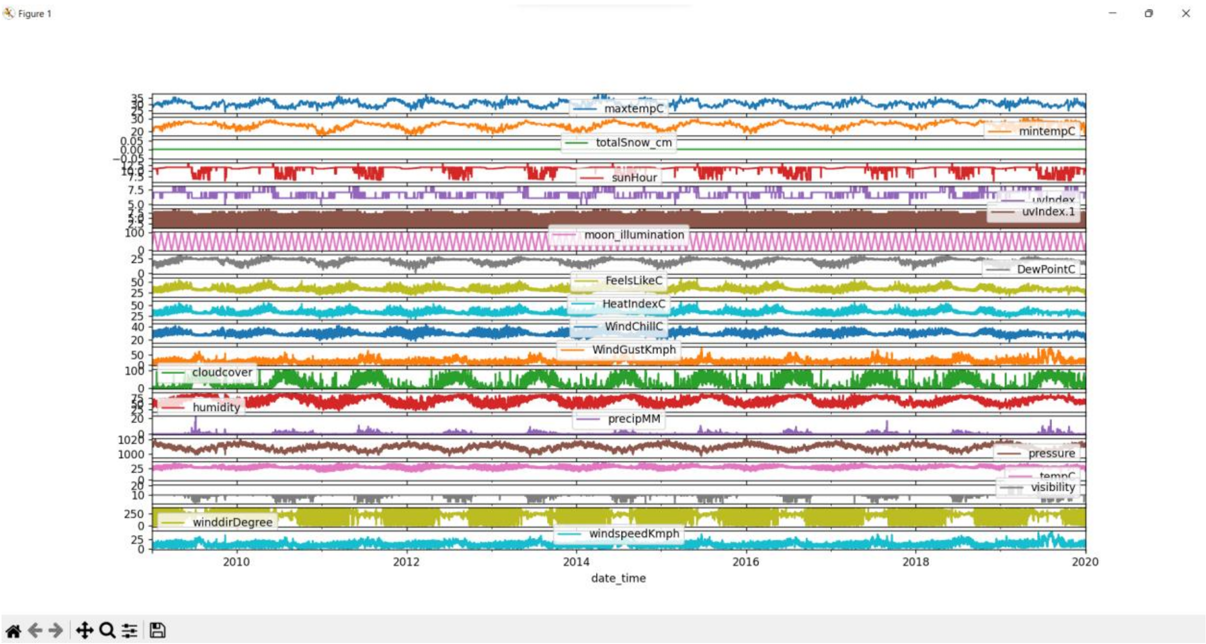


Figure 2 Bombay Weather data plot graph

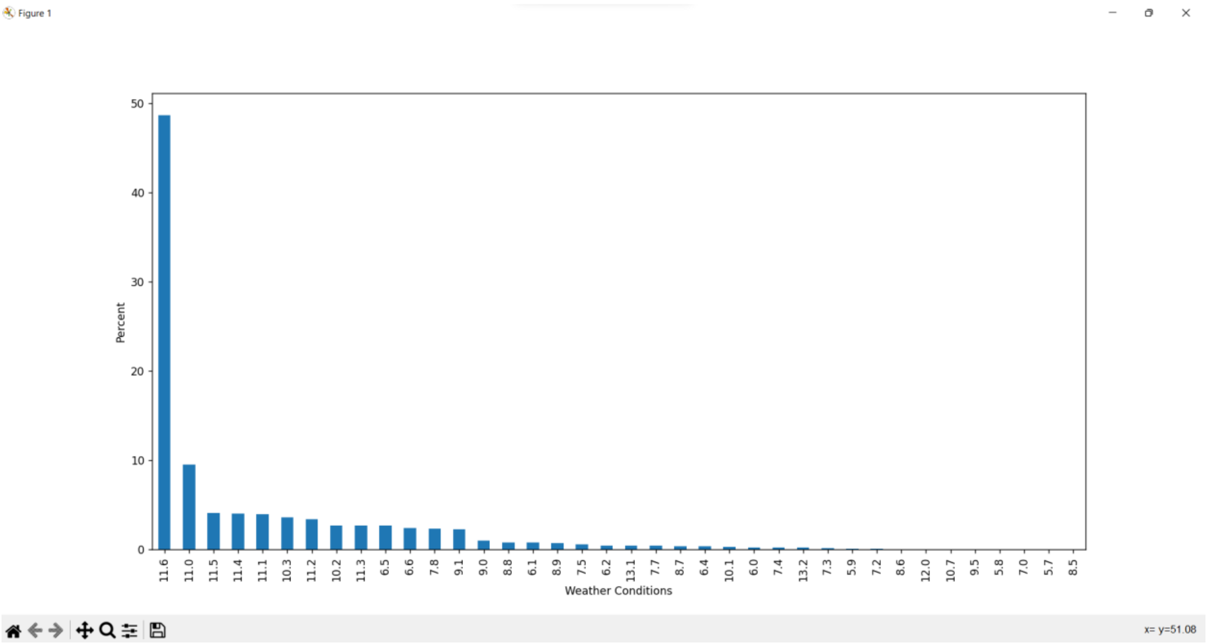


Figure 3 Bombay Weather condition output

CONCLUSION:

Thus we implemented Data Discretization and Visualization

SIGN AND REMARK

R1 (3 M)	R2 (3 M)	R3 (3 M)	R4 (3 M)	R5 (3 M)	Total	Sign

DATE