Experiment No.7

Title: Data normalization and discretization

Batch -B4
Roll no – 16010420133
Exp -7

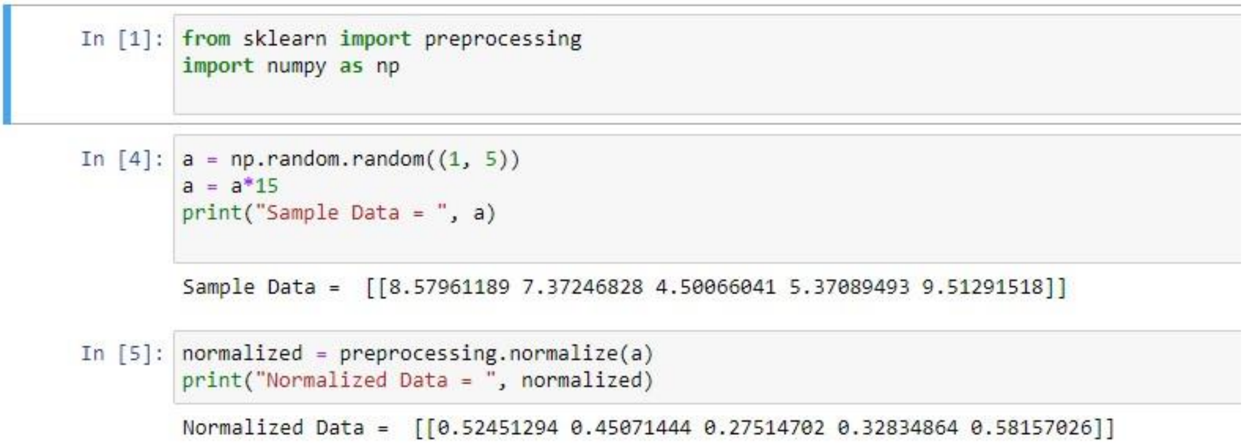
Aim: Data normalization and discretization

Resources needed: Python

Results:

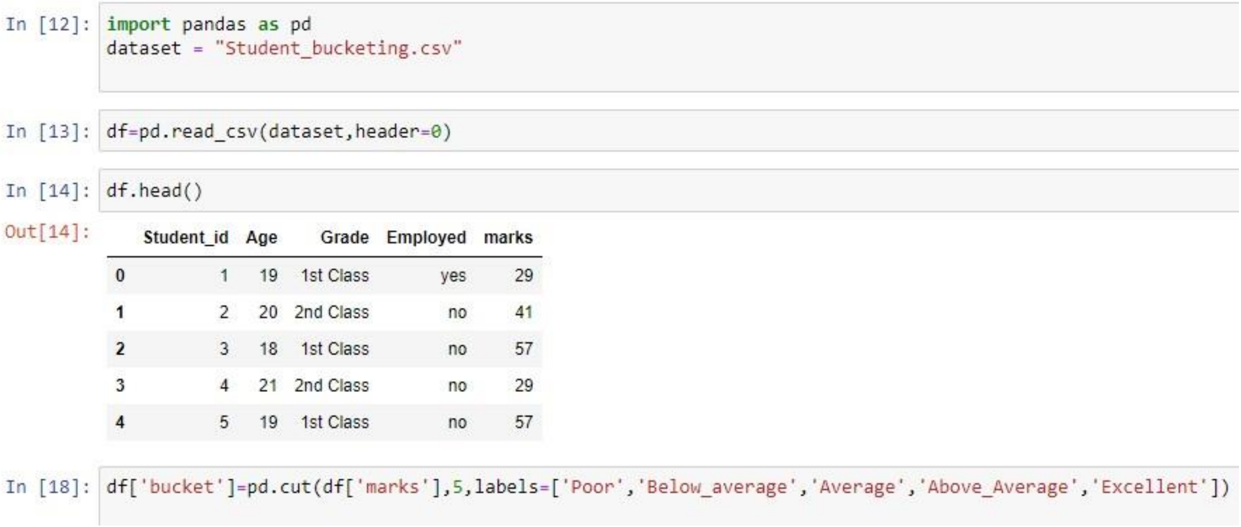
Q1. Data Normalization:

```
 from sklearn import preprocessing
  import numpy as np a =
  np.random.random((1, 5)) a
  = a*15
  print("Data = ", a) normalized =
  preprocessing.normalize(a)
  print("Normalized Data = ", normalized)
```



```
In [1]: from sklearn import preprocessing
        import numpy as np

In [4]: a = np.random.random((1, 5))
        a = a*15
        print("Sample Data = ", a)

        Sample Data =  [[8.57961189 7.37246828 4.50066041 5.37089493 9.51291518]]

In [5]: normalized = preprocessing.normalize(a)
        print("Normalized Data = ", normalized)

        Normalized Data =  [[0.52451294 0.45071444 0.27514702 0.32834864 0.58157026]]
```

Q2. Activity on Data Discretization of continuous data:

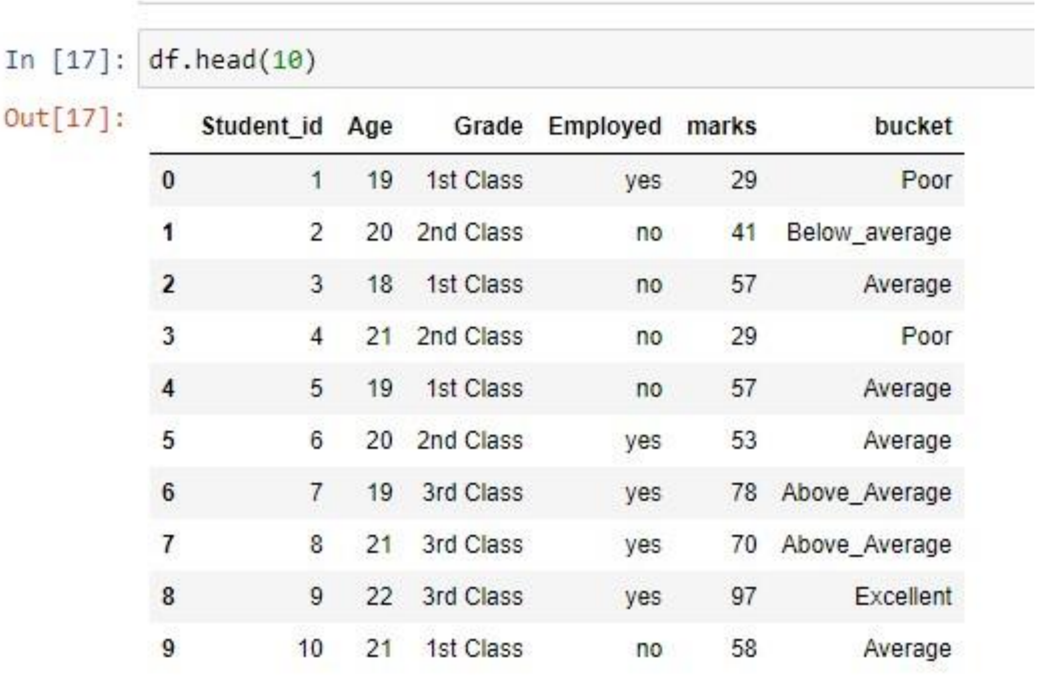


```
In [12]: import pandas as pd
         dataset = "Student_bucketing.csv"

In [13]: df=pd.read_csv(dataset,header=0)

In [14]: df.head()
```

Out[14]:

	Student_id	Age	Grade	Employed	marks
0	1	19	1st Class	yes	29
1	2	20	2nd Class	no	41
2	3	18	1st Class	no	57
3	4	21	2nd Class	no	29
4	5	19	1st Class	no	57

```
In [18]: df['bucket']=pd.cut(df['marks'],5,labels=['Poor','Below_average','Average','Above_Average','Excellent'])
```



```
In [17]: df.head(10)
```

Out[17]:

	Student_id	Age	Grade	Employed	marks	bucket
0	1	19	1st Class	yes	29	Poor
1	2	20	2nd Class	no	41	Below_average
2	3	18	1st Class	no	57	Average
3	4	21	2nd Class	no	29	Poor
4	5	19	1st Class	no	57	Average
5	6	20	2nd Class	yes	53	Average
6	7	19	3rd Class	yes	78	Above_Average
7	8	21	3rd Class	yes	70	Above_Average
8	9	22	3rd Class	yes	97	Excellent
9	10	21	1st Class	no	58	Average

Questions:

1. Explain scikit learn library of python and its use.

Scikit-learn (Sklearn) is the most useful and robust library for machine learning in Python. It provides a selection of efficient tools for machine learning and statistical modeling including classification, regression, clustering and dimensionality reduction via a consistence interface in Python. This library, which is largely written in Python, is built upon NumPy, SciPy and Matplotlib.

Uses:

Supervised Learning algorithms − Almost all the popular supervised learning algorithms, like Linear Regression, Support Vector Machine (SVM), Decision Tree etc., are the part of scikit-learn.

Unsupervised Learning algorithms − On the other hand, it also has all the popular unsupervised learning algorithms from clustering, factor analysis, PCA (Principal Component Analysis) to unsupervised neural networks.

Clustering − This model is used for grouping unlabeled data.

Cross Validation − It is used to check the accuracy of supervised models on unseen data.

Dimensionality Reduction − It is used for reducing the number of attributes in data which can be further used for summarisation, visualisation and feature selection.

Ensemble methods − As name suggest, it is used for combining the predictions of multiple supervised models.

Feature extraction − It is used to extract the features from data to define the attributes in image and text data.

Feature selection − It is used to identify useful attributes to create supervised models.

2. Explain pandas and Numpy in python.

Numpy

- NumPy is a Python library used for working with arrays.
- It also has functions for working in domain of linear algebra, fourier transform, and matrices.
- NumPy was created in 2005 by Travis Oliphant. It is an open source project and you can use it freely.
- NumPy stands for Numerical Python.
- In Python we have lists that serve the purpose of arrays, but they are slow to process.
- NumPy aims to provide an array object that is up to 50x faster than traditional Python lists.
- The array object in NumPy is called ndarray, it provides a lot of supporting functions that make working with ndarray very easy.
- Arrays are very frequently used in data science, where speed and resources are very important.

Pandas

Pandas is built on top of the Numpy package, means Numpy is required for operating the Pandas.

Before Pandas, Python was capable for data preparation, but it only provided limited support for data analysis. So, Pandas came into the picture and enhanced the capabilities of data analysis. It can perform five significant steps required for processing and analysis of data irrespective of the origin of the data, i.e., load, manipulate, prepare, model, and analyze.

Key Features of Pandas

- It has a fast and efficient DataFrame object with the default and customized indexing.
- Used for reshaping and pivoting of the data sets.
- Group by data for aggregations and transformations.
- It is used for data alignment and integration of the missing data.
- Provide the functionality of Time Series.

- Process a variety of data sets in different formats like matrix data, tabular heterogeneous, time series.
- Handle multiple operations of the data sets such as subsetting, slicing, filtering, groupBy, re-ordering, and re-shaping.
- It integrates with the other libraries such as SciPy, and scikit-learn.

Outcomes:

CO 3: Apply the transformations required on data to make it suitable for mining

Conclusion: (Conclusion to be based on the objectives and outcomes achieved)

I have implemented the bucketing and normalization techniques on the dataset.

Grade: AA / AB / BB / BC / CC / CD /DD

Signature of faculty in-charge with date

References:

Books/ Journals/ Websites:

1. Han, Kamber, "Data Mining Concepts and Techniques", Morgan Kaufmann 3nd Edition
2. https://www.educative.io/edpresso/data-normalization-in-python