Lab 1: Data Pre-processing

Introduction:

This lab report details the implementation and results of various data pre-processing techniques, including data cleaning, normalization, data binning, discretization, and feature selection.

1.1 Data Cleaning:

Datasets

ID	Name	Age	Department	Salary
1	John	28	HR	50000
2	Jane	35	Finance	60000
3	Emily		HR	55000
4	Michael	40	Human Resources	
5	Sarah	29	IT	52000
6	David	50	Finance	75000
7	Laura	38	H.R.	68000
8	Robert	32	HR	57000
9	Linda	45	IT	62000
10	James	30	HR	51000

Implementation Code:

```
import pandas as pd
print("Kishor Lab-1.1")
df = pd.read_csv('employee_data.csv')
print("Initial Data:\n", df.head())

df['Age'] = df['Age'].fillna(df['Age'].mean())
df['Salary'] = df['Salary'].fillna(df['Salary'].mean())

df['Department'] = df['Department'].str.strip().replace({
    'Human Resources': 'HR',
    'H.R.': 'HR',
    'hr': 'HR'
})
print("\nCleaned Data:\n", df.head())
```

Output SnapShot:

```
Kishor Lab-1.1
Initial Data:
    ID     Name   Age       Department   Salary
0   1      John  28.0               HR  50000.0
1   2      Jane  35.0          Finance  60000.0
2   3     Emily   NaN               HR  55000.0
3   4   Michael  40.0  Human Resources      NaN
4   5     Sarah  29.0               IT  52000.0

Cleaned Data:
    ID     Name   Age Department   Salary
0   1      John  28.0         HR  50000.0
1   2      Jane  35.0    Finance  60000.0
2   3     Emily  35.7         HR  55000.0
3   4   Michael  40.0         HR  58100.0
4   5     Sarah  29.0         IT  52000.0
```

1.2 Normalization

Datasets

StudentID	Math	Science	English
1	78	65	80
2	88	75	85
3	60	50	55
4	90	78	92
5	55	48	58
6	83	72	88
7	71	66	79
8	64	52	70
9	88	80	90
10	76	68	82

Implementation Code

```
import pandas as pd
print("Kishor Lab-1.2")
from sklearn.preprocessing import MinMaxScaler
df = pd.read_csv('student_scores.csv')
print("Initial Data:\n", df.head())
scaler = MinMaxScaler()
df[['Math', 'Science', 'English']] = scaler.fit_transform(df[['Math', 'Science', 'English']])
print("\nNormalized Scores:\n", df.head())
```

Output Snapshot

```
Kishor Lab-1.2
Initial Data:
    StudentID  Math  Science  English
0           1    78       65       80
1           2    88       75       85
2           3    60       50       55
3           4    90       78       92
4           5    55       48       58

Normalized Scores:
    StudentID      Math  Science   English
0           1  0.657143  0.53125  0.675676
1           2  0.942857  0.84375  0.810811
2           3  0.142857  0.06250  0.000000
3           4  1.000000  0.93750  1.000000
4           5  0.000000  0.00000  0.081081
```

1.3 Data Binning

DataSets

CustomerID	Age
1	25
2	42
3	36
4	53
5	28
6	47
7	31
8	50
9	22
10	60

Implementation Code:

```python
import pandas as pd
print("Kishor Lab-1.3")
df = pd.read_csv('customer_ages.csv')
print("Initial Data:\n", df.head())
bins = [18, 30, 50, 100]
labels = ['Young', 'Middle-aged', 'Senior']
df['Group'] = pd.cut(df['Age'], bins=bins, labels=labels, right=False)
print("\nData after Binning:\n", df.head())
dist = df['Group'].value_counts()
print("\nAge Group Distribution:\n", dist)
```

Output SnapShot:

```
Kishor Lab-1.3
Initial Data:
    CustomerID  Age
0            1   25
1            2   42
2            3   36
3            4   53
4            5   28

Data after Binning:
    CustomerID  Age         Group
0            1   25         Young
1            2   42  Middle-aged
2            3   36  Middle-aged
3            4   53        Senior
4            5   28         Young

Age Group Distribution:
 Group
Middle-aged    7
Young          5
Senior         3
Name: count, dtype: int64
```

1.4 Discretization

DataSets

Month	Sales
January	15000
February	18000
March	12000
April	30000
May	22000
June	5000
July	8000
August	25000
September	10000
October	20000

Implementation Code:

```python
import pandas as pd
print("Kishor Lab-1.4")
df = pd.read_csv('sales_data.csv')
print("Initial Data:\n", df.head())
bins = [0, 5000, 20000, float('inf')]
labels = ['Low', 'Medium', 'High']
df['Category'] = pd.cut(df['Sales'], bins=bins, labels=labels)
print("\nData after Discretization:\n", df.head())
dist = df['Category'].value_counts()
print("\nSales Category Distribution:\n", dist)
```

Output SnapShot

```
Kishor Lab-1.4
Initial Data:
       Month  Sales
0    January  15000
1   February  18000
2      March  12000
3      April  30000
4        May  22000

Data after Discretization:
       Month  Sales Category
0    January  15000   Medium
1   February  18000   Medium
2      March  12000   Medium
3      April  30000     High
4        May  22000     High

Sales Category Distribution:
 Category
Medium    7
High      4
Low       1
Name: count, dtype: int64
```

1.5 Feature Selection

DataSets

PatientID	Age	BloodPressure	Cholesterol	Glucose	HeartRate	Disease
1	45	130	180	95	70	1
2	50	140	200	105	75	1
3	60	150	240	120	80	1
4	40	120	170	90	65	0
5	35	110	160	85	60	0
6	55	145	210	115	78	1
7	42	135	190	100	72	0
8	38	115	150	80	68	0
9	47	125	170	95	70	1
10	53	140	210	110	76	1

Implementation Code

```
import pandas as pd
from sklearn.feature_selection import SelectKBest, chi2
print("Kishor Lab-1.5")
df = pd.read_csv('medical_data.csv')
print("Initial Data:\n", df.head())

X = df.drop(columns=['Disease'])
y = df['Disease']

sel = SelectKBest(score_func=chi2, k=3)
sel.fit(X, y)

top = X.columns[sel.get_support()]
print("\nTop 3 Features for Predicting Disease:\n", top.tolist())
```

Output SnapShot

```
Kishor Lab-1.5
Initial Data:
    PatientID  Age  BloodPressure  Cholesterol  Glucose  HeartRate  Disease
0          1   45            130          180       95         70        1
1          2   50            140          200      105         75        1
2          3   60            150          240      120         80        1
3          4   40            120          170       90         65        0
4          5   35            110          160       85         60        0

Top 3 Features for Predicting Disease:
 ['Age', 'Cholesterol', 'Glucose']
```