

CS106 Basics of machine learning & applications

1. Read the "Employee_list" dataset.

a) Use pandas to read Name, Age and Profession, and Salary of Employee list with Name as index.

b) Sort data by age, and if age is same then display values based on sorted salary.

c) Display minimum, maximum, and average salary for each profession (Hint: use groupby function).

d) Find median, mode, skewness and kurtosis for salary of employees.

e) In the given list of employees, the owner wants to have the numeric values in the profession for easy counting of

employees. He wants to put a code instead of profession. Write a python code to replace the profession like

engineer as 0, doctor as 1 and teacher as 2.

Code:

```
1. import pandas as pd
2. import numpy as np
3. from sklearn.preprocessing import LabelEncoder

4. emp= pd.read_csv("Employee_list.csv")
5. emp.set_index("Name",inplace=True)
6. print(emp.head())
7. emp_sorted=emp.sort_values(by=["Age","Salary"])
8. print(emp_sorted)
9. salary_stats=emp.groupby("Profession")["Salary"].agg(["min","max","mean"])
10. print(salary_stats)
11. print("Median:",emp["Salary"].median())
12. print("Mode:",emp["Salary"].mode()[0])
13. print("Skewness:",emp["Salary"].skew())
14. print("Kurtosis:",emp["Salary"].kurt())
15. le=LabelEncoder()
16. encoded_profession=le.fit_transform(emp["Profession"])
17. print(encoded_profession)
```

Output:

```
...      Sno.  Age Profession   Salary  Empid
Name
Rahul      1   38  Engineer  86567.0    15
Vipul      2   29   Doctor  77298.0     9
Saurav     3   33   Doctor  81302.0    11
Niyaz      4   39  Teacher  30456.0     6
Franklin   5   28  Engineer   NaN     21
      Sno.  Age Profession   Salary  Empid
Name
Shashank   8   28  Teacher  45000.0    31
Franklin   5   28  Engineer   NaN     21
Vipul      2   29   Doctor  77298.0     9
Priya     10   29  Teacher  78600.0    10
Meetesh    7   29  Engineer   NaN     23
Saurav     3   33   Doctor  81302.0    11
Niroja     6   34  Engineer  79000.0    12
Rahul      1   38  Engineer  86567.0    15
Niyaz      4   39  Teacher  30456.0     6
Meenal    11   41  Engineer  55324.0     8
Chauhan     9   41   Doctor  73249.0    44
      min      max      mean
Profession
Doctor    73249.0  81302.0  77283.000000
Engineer  55324.0  86567.0  73630.333333
Teacher   30456.0  78600.0  51352.000000
Median: 77298.0
Mode: 30456.0
Skewness: -1.1205193424544682
Kurtosis: 0.024126103574920954
[1 0 0 2 1 1 1 2 0 2 1]
```

2. Read the “attainment.csv” dataset.

a) Parse the data from the CSV file using pandas and print the header. Note that the file uses '?' as the entry to

represent missing data.

b) Project leader wants to know the total number of rows which are having atleast 1 missing values. Display

those rows.

c) Project leader wants to know the missing values in complete dataset as well as in individual column. He is not

interested to process those columns which are containing missing values more than 50%. Write a program to

provide the filtered data frame and save the data in ‘filtered.csv’.

d) The data is in pre-processing phase. The leader wants to complete the data. But he is confused that he should

use mean or median to fill the missing values. Kindly identify the correct statistics (consider skewness) and

print the complete data.

Code:

```
1. import pandas as pd
2. import numpy as np

3. att=pd.read_csv("attainment.csv",na_values="?")
4. print(att.head())
5. rows_with_missing=att[att.isnull().any(axis=1)]
6. print("Rows with missing data:",rows_with_missing.shape[0])
7. print(rows_with_missing)
8. missing_percent=att.isnull().mean()*100
9. print(missing_percent)
10. filtered=att.loc[:,missing_percent<=50]
11. filtered.to_csv("filtered.csv",index=False)
12. print(filtered.head())
13. completed_att = filtered.copy()

14. for col in completed_att.select_dtypes(include=np.number):
15. skewness = completed_att[col].skew()

16. if abs(skewness) < 1:
17. completed_att[col].fillna(completed_att[col].mean())
18. else:
19. completed_att[col].fillna(completed_att[col].median())

20. print("Completed Dataset:")
21. print(completed_att.head())
```

Output:

	210	2015	F	master's	10.4	12.0	7.2	4.1	23.2
	211	2016	F	master's	11.2	12.3	6.3	6.3	28.8
...	212	2017	F	master's	10.5	11.8	6.8	5.0	25.8
	213	2018	F	master's	10.7	12.6	6.2	3.8	29.9

		Pacific Islander	American Indian/Alaska Native	Two or more races
0		NaN	NaN	NaN
1		NaN	NaN	NaN
2		NaN	NaN	NaN
3		NaN	NaN	NaN
4		NaN	NaN	NaN
..	
209		NaN	NaN	7.5
210		NaN	NaN	10.2
211		NaN	NaN	8.2
212		NaN	NaN	5.4
213		NaN	NaN	NaN

[125 rows x 11 columns]

```

Year          0.000000
Sex           0.000000
Min degree    0.000000
Total         0.934579
White         0.000000
Black         0.000000
Hispanic      4.672897
Asian        21.495327
Pacific Islander 56.542056
American Indian/Alaska Native 38.785047
Two or more races 26.635514
dtype: float64

```

```

    Year Sex   Min degree   Total   White   Black   Hispanic   Asian \
...  0  1920   A   high school   NaN    22.0    6.3         NaN    NaN
    1  1940   A   high school  38.1    41.2   12.3         NaN    NaN
    2  1950   A   high school  52.8    56.3   23.6         NaN    NaN
    3  1960   A   high school  60.7    63.7   38.6         NaN    NaN
    4  1970   A   high school  75.4    77.8   58.4         NaN    NaN

    American Indian/Alaska Native   Two or more races
    0                             NaN         NaN
    1                             NaN         NaN
    2                             NaN         NaN
    3                             NaN         NaN
    4                             NaN         NaN
Completed Dataset:
    Year Sex   Min degree   Total   White   Black   Hispanic   Asian \
    0  1920   A   high school   NaN    22.0    6.3         NaN    NaN
    1  1940   A   high school  38.1    41.2   12.3         NaN    NaN
    2  1950   A   high school  52.8    56.3   23.6         NaN    NaN
    3  1960   A   high school  60.7    63.7   38.6         NaN    NaN
    4  1970   A   high school  75.4    77.8   58.4         NaN    NaN

    American Indian/Alaska Native   Two or more races
    0                             NaN         NaN
    1                             NaN         NaN
    2                             NaN         NaN
    3                             NaN         NaN
    4                             NaN         NaN

```

3. Read the “wine” dataset

a) A wine factory owner having the data of chemical composition of different wines. However, in his sheet the title of column is missed. Please put the column title as 'wine_class', 'Alcohol', 'Malic acid', 'Ash', 'Alcalinity of ash', 'Magnesium', 'Total phenols', 'Flavanoids', 'Nonflavanoid phenols', 'Proanthocyanins', 'Color intensity', 'Hue', 'OD280/OD315 of diluted wines', 'Proline'

b) The owner wants to scale the value of alcohol in 0 to 1. Write a code to achieve this using min-max normalization.

c) Write a code to see the distribution of normalized alcohol and assess whether it is normal curve or not.

d) The owner now decided to apply z-score normalization. Use the initial data frame and apply z- score normalization to the alcohol and Malic acid. Plot the normalized Malic Acid graph.

Code:

1. import pandas as pd
2. import matplotlib.pyplot as plt

3. cols = [
4. 'wine_class', 'Alcohol', 'Malic acid', 'Ash', 'Alcalinity of ash', 'Magnesium',
5. 'Total phenols', 'Flavanoids', 'Nonflavanoid phenols', 'Proanthocyanins',
6. 'Color intensity', 'Hue', 'OD280/OD315 of diluted wines', 'Proline'
7.]
8. wine = pd.read_csv("wine.csv", header=None, names=cols)
9. print(wine.head())
10. wine["Alcohol_minmax"] = (wine["Alcohol"] - wine["Alcohol"].min()) /
 (wine["Alcohol"].max() - wine["Alcohol"].min())
11. print(wine[["Alcohol", "Alcohol_minmax"]].head())
12. wine["Alcohol_minmax"].hist(bins=20)
13. plt.title("Distribution of Normalized Alcohol")
14. plt.xlabel("Normalized Alcohol")
15. plt.ylabel("Frequency")
16. plt.show()
17. wine["Alcohol_z"] = (wine["Alcohol"] - wine["Alcohol"].mean()) /
 wine["Alcohol"].std()
18. wine["Malic_acid_z"] = (wine["Malic acid"] - wine["Malic acid"].mean()) /
 wine["Malic acid"].std()
19. wine["Malic_acid_z"].plot(kind="line", title="Z-score Normalized Malic Acid")
20. plt.xlabel("Index")
21. plt.ylabel("Z-score Malic Acid")
22. plt.show()

Output:

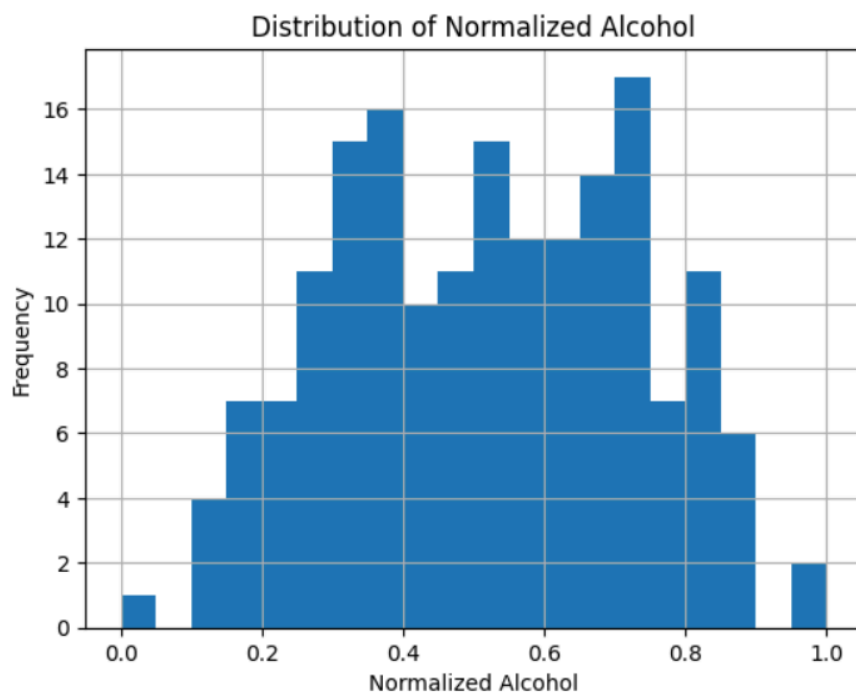
plt.show()

```
... wine_class Alcohol Malic acid Ash Alcalinity of ash Magnesium \
0 1 14.23 1.71 2.43 15.6 127
1 1 13.20 1.78 2.14 11.2 100
2 1 13.16 2.36 2.67 18.6 101
3 1 14.37 1.95 2.50 16.8 113
4 1 13.24 2.59 2.87 21.0 118

Total phenols Flavanoids Nonflavanoid phenols Proanthocyanins \
0 2.80 3.06 0.28 2.29
1 2.65 2.76 0.26 1.28
2 2.80 3.24 0.30 2.81
3 3.85 3.49 0.24 2.18
4 2.80 2.69 0.39 1.82

Color intensity Hue OD280/OD315 of diluted wines Proline
0 5.64 1.04 3.92 1065
1 4.38 1.05 3.40 1050
2 5.68 1.03 3.17 1185
3 7.80 0.86 3.45 1480
4 4.32 1.04 2.93 735

Alcohol Alcohol_minmax
0 14.23 0.842105
1 13.20 0.571053
2 13.16 0.560526
3 14.37 0.878947
4 13.24 0.581579
```



...

