COGNITIOR

WE MAKE MACHINES ALIVE

Problem Statement

"Analyse various outcomes of School and Visualize using Data Visualization Technique"

Presented by:

Tarunkumar J. Barapatre

INFORMATION TECHNOLOGY, NAGPUR

Session 2018 - 2019

Contents

Problem Statement

Analysis various school outcomes in Tennessee using pandas. Suppose you are a public school administrator. Some schools in your state of Tennessee are performing below average academically. Your super intendent, under pressure from frustrated parents and voters, approached you with the task of understanding why these schools are under performing.

To improve school performance, you need to learn more about these schools and their students, just as a business needs to understand its own strengths and weaknesses and its customers. Though you is eager to build an impressive explanatory model, you know the importance of conducting preliminary research to prevent possible pitfalls or blind spots. Thus, you engages in a thorough exploratory analysis, which includes: a lit review, data collection, descriptive and inferential statistics, and data visualization.

Objectives

- Understanding why these schools are under-performing ?

- To improve school performance, you need to learn more about these schools and their students.

Phase – 1 Data Collection

- Here is a data of every public school in middle Tennessee. The data also includes various demographic, school faculty, and income variables. You need to convert the data into useful information
  - Read the data in pandas data frame
  - Describe the data to find more details

Step_1:- import matplotlib.pyplot, seaborn, pandas, numpy, os packages

Step_2:- Help of os package to chance the directory

Step_3:- Put and display the data set in with the help of pandas package with file name i.e. dataset = pd.read_csv('middle_tn_schools.csv')

Step_4:- pd.DataFrame.describe(dataset) use this command to show the selected data to describe.

Phase – 1

- **Describe the data to find more details**
  - data=pd.DataFrame.describe(dataset) data
- It can describe the dataset in which the following categories can show

	school_rating	size	reduced_lunch	state_percentile_16	state_percentile_15
count	347.000000	347.000000	347.000000	347.000000	341.000000
mean	2.968300	699.472622	50.279539	58.801729	58.249267
std	1.690377	400.598636	25.480236	32.540747	32.702630
min	0.000000	53.000000	2.000000	0.200000	0.600000
25%	2.000000	420.500000	30.000000	30.950000	27.100000
50%	3.000000	595.000000	51.000000	66.400000	65.800000
75%	4.000000	851.000000	71.500000	88.000000	88.600000
max	5.000000	2314.000000	98.000000	99.800000	99.800000

Phase – 2 Group Data by School Rating

- Isolates ‘reduced_lunch’ and groups the data by ‘school_rating’ using pandas groupby method

Step_1:- Write a this command
dataset.groupby(['school_rating']).count() and show the grouping of school rating as well as show the relation between other heading with reduced lunch time

Step_2:- Using new variable for represent like dataset_2

```
In [7]: dataset_2 = dataset.groupby(['school_rating']).size()

In [8]: dataset_2

Out[8]: school_rating
        0.0    43
        1.0    40
        2.0    44
        3.0    56
        4.0    86
        5.0    78
        dtype: int64
```

Phase – 2

Grouped the data by school rating

school_rating	
0.0	43
1.0	40
2.0	44
3.0	56
4.0	86
5.0	78

Phase – 3 Correlation Analysis

Find the correlation between ‘reduced_lunch’ and ‘school_rating’. The values in the correlation matrix table will be between -1 and 1.

Step_1:- Create a new data frame select from original dataset and to give the new variable name

Step_2:- Using this command **dataset_3= dataset[['reduced_lunch', 'school_rating']]** and this is work for data slicing

```
dataset_3
```

	reduced_lunch	school_rating
0	10.0	5.0
1	71.0	2.0
2	43.0	4.0
3	91.0	0.0
4	26.0	4.0
5	48.0	4.0
6	58.0	4.0
7	16.0	5.0
8	21.0	4.0
9	50.0	3.0
10	75.0	0.0
11	67.0	3.0
12	72.0	3.0
13	25.0	5.0
14	25.0	4.0

Phase – 3

Step_3:- Find out correletion matrix table in between ‘reduced_lunch’ and‘school_rating’ then select index location from dataset_3

Step_4:- Enter this command **x=dataset_3.iloc[0:,:].values** and x variable is show array from data

```
In [82]: x=dataset_3.iloc[0:,:].values

In [83]: x

Out[83]: array([[10.,  5.],
                [71.,  2.],
                [43.,  4.],
                [91.,  0.],
                [26.,  4.],
                [48.,  4.],
                [58.,  4.],
                [16.,  5.],
                [21.,  4.],
                [50.,  3.],
                [75.,  0.],
                [67.,  3.],
                [72.,  3.],
                [25.,  5.],
                [25.,  4.],
                [30.,  4.],
                [66.,  2.],
                [63.,  3.],
                [61.,  4.],
```

Phase – 3

Step_5:-from sklearn.preprocessing import StandardScaler

sc_x = StandardScaler()

x = sc_x.fit_transform(x) using this command x variable data array convert into a standardization from and x variable show in negative to positive number

```
In [91]: x

Out[91]: array([[-1.58309773,  1.20365718],
                [ 0.81437166, -0.57365789],
                [-0.28610609,  0.61121882],
                [ 1.6004272 , -1.7585346 ],
                [-0.9542533 ,  0.61121882],
                [-0.08959221,  0.61121882],
                [ 0.30343556,  0.61121882],
                [-1.34728107,  1.20365718],
                [-1.15076719,  0.61121882],
                [-0.01098666,  0.01878047],
                [ 0.97158277, -1.7585346 ],
                [ 0.65716055,  0.01878047],
                [ 0.85367444,  0.01878047],
                [-0.99355608,  1.20365718],
                [-0.99355608,  0.61121882],
                [-0.79704219,  0.61121882],
                [ 0.61785778, -0.57365789],
                [ 0.49994945,  0.01878047],
                [ 0.42134389,  0.61121882],
                [ 1.32530776, -1.16609624],
```

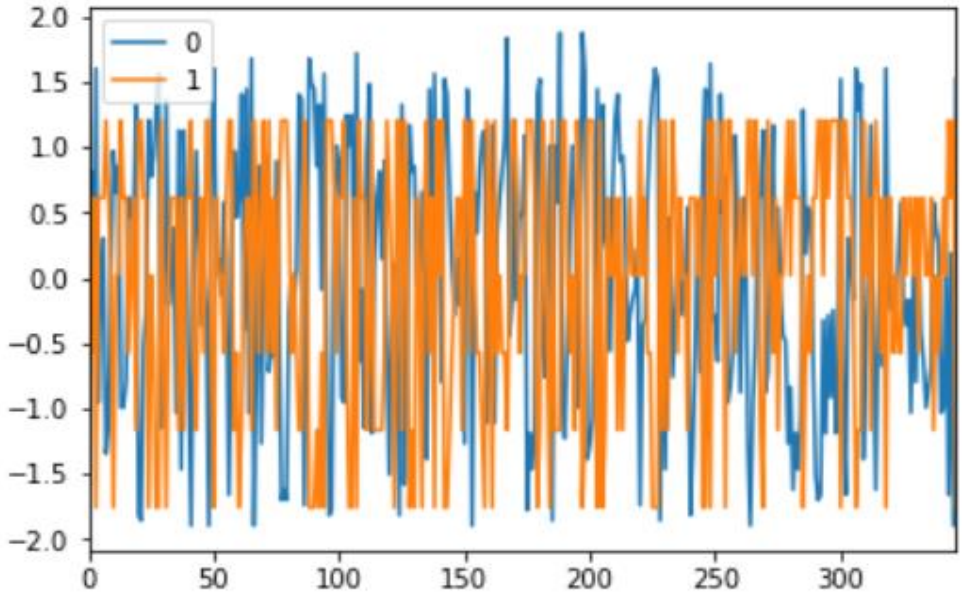
Phase – 3

Step_6:- data=pd.DataFrame(x) using this command array dataset converting into data frame with including new variable.

Step_7:- data.plot() this command to show data frame in graphical from

```
In [95]: data.plot()

Out[95]: <matplotlib.axes._subplots.AxesSubplot at 0x1e5bf767da0>
```



Phase 4 – Scatter Plot

- Find the relationship between school_rating and reduced_lunch, Plot a graph with the two variables on a scatter plot.

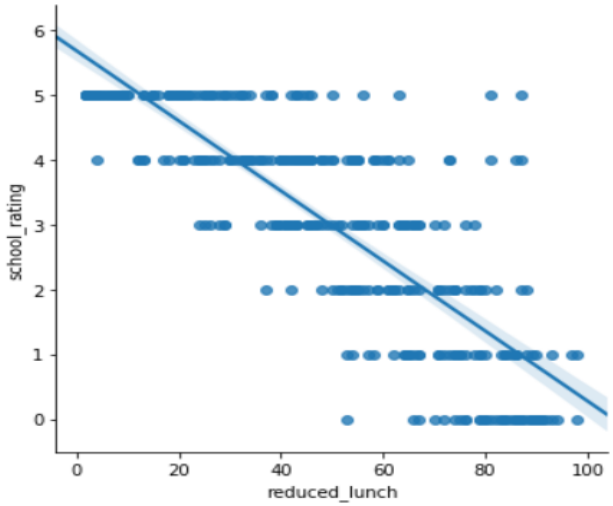
Step_1:- import matplotlib.pyplot as plt

import seaborn as sns import this packages.

Step_2:- plt.scatter(data=dataset, x='reduced_lunch', y='school_rating') using this command to show the scatter relation between Reduced lunch and School rating

```
sns.lmplot(data=dataset, x='reduced_lunch', y='school_rating', fit_reg=True)
```

```
<seaborn.axisgrid.FacetGrid at 0x2a332f87a90>
```

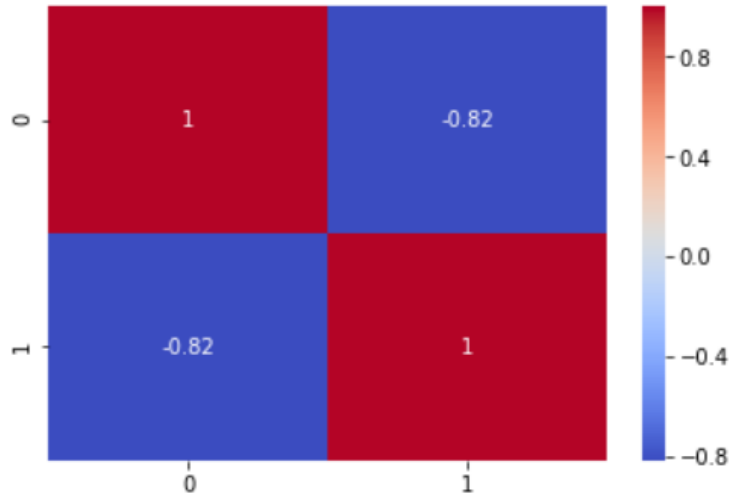


Phase 5 – Correlation Matrix

- The darker the colors, the stronger the correlation (positive or negative) between those two variables. Draw a graph of correlation matrix having all important fields of dataframe.

Step_1:- **sns.heatmap(data.corr(), annot=True)** this command show directly matrix fields of data set with help of standardization

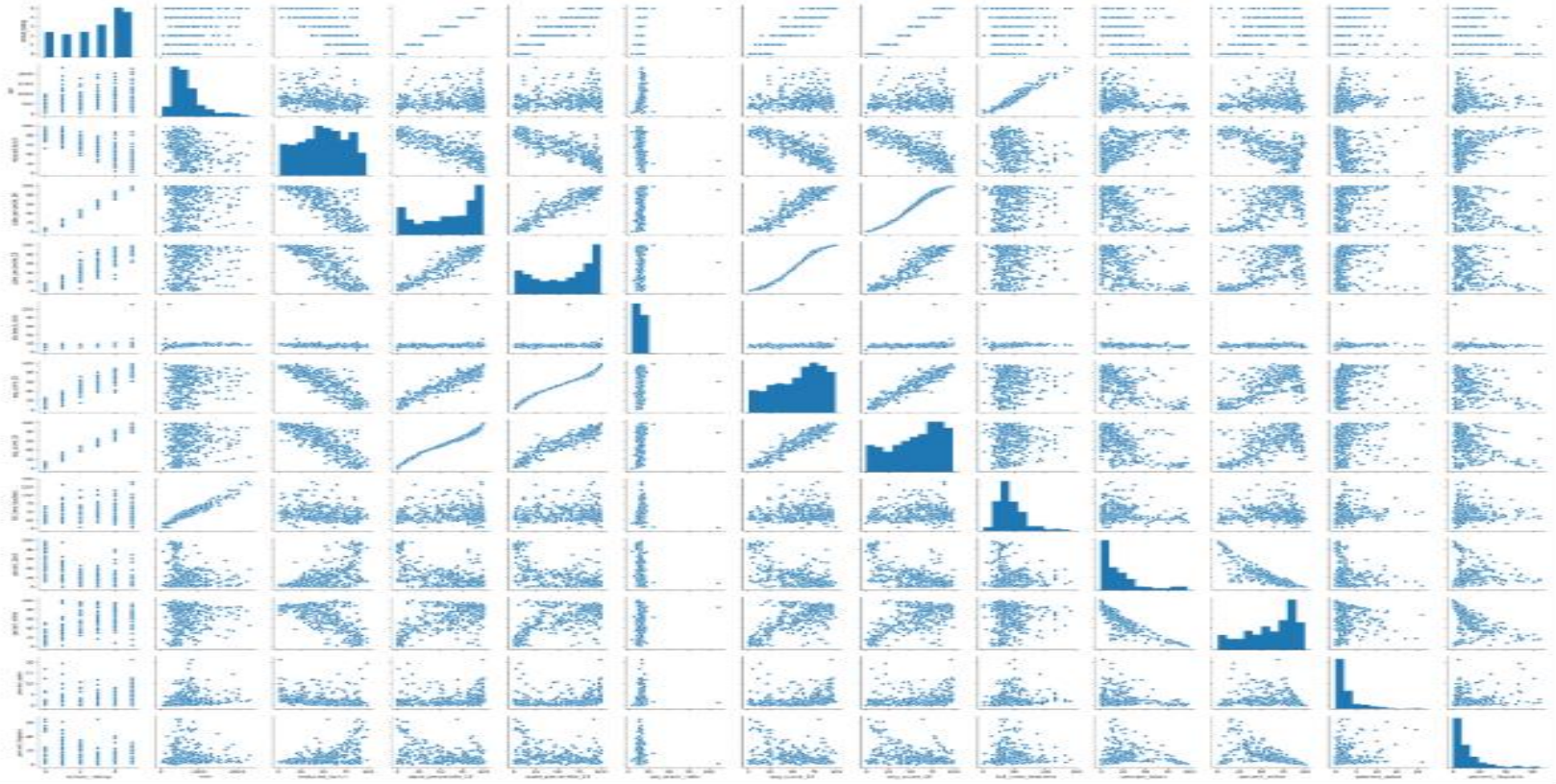
```
In [26]: sns.heatmap(data.corr(),cmap='coolwarm', annot=True)
Out[26]: <matplotlib.axes._subplots.AxesSubplot at 0x2a3339b9748>
```



Phase - 5

Step_2:-sns.pairplot(dataset) using this command to view graph of correlation matrix having all important fields of dataframe.

```
In [29]: sns.pairplot(dataset)
Out[29]: <seaborn.axisgrid.PairGrid at 0x2a33fac7ac8>
```



Conclusion

Conclusion :

- As per the analysis we found that reduced_food create a grea importance in Performance.
- Reduced_food is a dependent variable on school rating.

THANK YOU