Data Exploration Analysis

Team: tf pandas

Mingjun Xie 30971586 mx4n19@soton.ac.uk

Jie Zhou 31384579 jz5n19@soton.ac.uk

Zixuan Cai 31303528 zc1g19@soton.ac.uk

Chenguang Zhu 31369251 cz1g19@soton.ac.uk

ABSTRACT

The dataset we are going to explore is cardiovascular disease dataset (https://www.kaggle.com/sulianova/cardiovascular-disease-dataset) with the basic information below:

# id	ID number		
# age	Age	Objective Feature	int (days)
# gender	Gender	Objective Feature	1 - women, 2 - men
# height	Height	Objective Feature	int (cm)
# weight	Weight	Objective Feature	float (kg)
# ap_hi	Systolic blood pressure	Examination Feature	int
# ap_lo	Diastolic blood pressure	Examination Feature	int
# cholesterol	Cholesterol	Examination Feature	1: normal, 2: above normal, 3: well above normal
# gluc	Glucose	Examination Feature	1: normal, 2: above normal, 3: well above normal
# smoke	Smoking	Subjective Feature	binary
# alco	Alcohol intake	Subjective Feature	binary
# active	Physical activity	Subjective Feature	binary
# cardio	Presence or absence of cardiovascular disease	Target Variable	binary

We will first develop the exploratory data analysis (EDA) to get an intuition and important information about the data sets. Then the main task for our project is to use these 70000 data to predict whether a testee faced with cardiovascular disease.

1 Browse the Data Structure

The first thing we need to do is to have a glance of the data, we use info(), head(), and describe() to print the profile of the data, The output is shown below.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 70000 entries, 0 to 69999
Data columns (total 13 columns):
id              70000 non-null int64
age             70000 non-null int64
gender          70000 non-null int64
height          70000 non-null int64
weight          70000 non-null float64
ap_hi           70000 non-null int64
ap_lo           70000 non-null int64
cholesterol     70000 non-null int64
gluc            70000 non-null int64
smoke           70000 non-null int64
alco            70000 non-null int64
active          70000 non-null int64
cardio          70000 non-null int64
dtypes: float64(1), int64(12)
memory usage: 6.9 MB
```

Figure 1: info()

	id	age	gender	height	weight	ap_hi	ap_lo	cholesterol	gluc	smoke	alco	active	cardio
0	0	18393	2	168	62.0	110	80	1	1	0	0	1	0
1	1	20228	1	156	85.0	140	90	3	1	0	0	1	1
2	2	18857	1	165	64.0	130	70	3	1	0	0	0	1
3	3	17623	2	169	82.0	150	100	1	1	0	0	1	1
4	4	17474	1	156	56.0	100	60	1	1	0	0	0	0

Figure 2: head()

	id	age	gender	height	weight	ap_hi	ap_lo	cholesterol	gluc	smoke	alco	active	cardio
count	70000.000000	70000.000000	70000.000000	70000.000000	70000.000000	70000.000000	70000.000000	70000.000000	70000.000000	70000.000000	70000.000000	70000.000000	70000.000000
mean	49972.419900	19468.865814	1.349571	164.359229	74.205690	128.817286	96.630414	1.366871	1.226457	0.088129	0.053771	0.803729	0.499700
std	28851.302323	2467.251667	0.476838	8.210126	14.395757	154.011419	188.472530	0.680250	0.572270	0.283484	0.225568	0.397179	0.500003
min	0.000000	10798.000000	1.000000	55.000000	10.000000	-150.000000	-70.000000	1.000000	1.000000	0.000000	0.000000	0.000000	0.000000
25%	25006.750000	17664.000000	1.000000	159.000000	65.000000	120.000000	80.000000	1.000000	1.000000	0.000000	0.000000	1.000000	0.000000
50%	50001.500000	19703.000000	1.000000	165.000000	72.000000	120.000000	80.000000	1.000000	1.000000	0.000000	0.000000	1.000000	0.000000
75%	74889.250000	21327.000000	2.000000	170.000000	82.000000	140.000000	90.000000	2.000000	1.000000	0.000000	0.000000	1.000000	1.000000
max	99999.000000	23713.000000	2.000000	250.000000	200.000000	16020.000000	11000.000000	3.000000	3.000000	1.000000	1.000000	1.000000	1.000000

Figure 3: describe()

From the Figure 1 and 3, we can see that the dataset consists of 70 000 records of patients' data, 11 features and a target and there is no missing data and text data. As shown in Figure 2, there are some outliers in the data. The systolic blood pressure (ap_hi) and diastolic blood pressure (ap_lo) have negative number, their maximum numbers are also abnormal since the blood pressure can't go such high like this. Additionally, the minimum number in weight is too small to be an adult's weight. We will check these outliers and handle it later.

Then we make the histograms about each attribute, we find that the attributes have very different scales and some are tail heavy.

2 Split the training and testing dataset

After taking a quick glance at the data set, we need to create a test set before we do further work to discover and visualize the data and gain insights. If we look at the whole dataset without splitting the test set, we may stumble upon some impressive pattern of the whole dataset which may lead us to make some bias decision. Also, using the whole data as training set, the estimation error will be too optimistic which will cause that our model will be an biased model and has a poor generalization performance.

There are many ways to create a teat set. For example, we can use hash to help us split. But hash method has a problem that the real proportion of training and test set may not strictly follow the setting. As the result showed in Figure 4, we set the proportion of test set is 0.2, but the number of incidences in test set is 14243 but not 14000.

```
train_set_hash, test_set_hash = split_train_test_by_id(cardio_dataset, 0.2, "id")
test_set_hash.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 14243 entries, 4 to 69999
Data columns (total 13 columns):
id              14243 non-null int64
age             14243 non-null int64
gender          14243 non-null int64
height          14243 non-null int64
weight          14243 non-null float64
ap_hi           14243 non-null int64
ap_lo           14243 non-null int64
cholesterol     14243 non-null int64
gluc            14243 non-null int64
smoke           14243 non-null int64
alco            14243 non-null int64
active          14243 non-null int64
cardio          14243 non-null int64
dtypes: float64(1), int64(12)
memory usage: 1.5 MB
```

Figure 4: Splitting using hash

For hash cannot meet our demand, we can use functions in Scikit-Learn to split dataset. So we choose to use function train_test_split which can solve the problem above.

However, function train_test_split is a purely random splitting function. After searching some relevant material in internet and asking friends learning medicine, we know that smoking is a very important attribute to predict someone has cardiovascular disease or not. So we need to ensure that the test set is representative of the various categories of incomes in the whole dataset. To achieve that goal, we can use function StratifiedShuffleSplit. The test set has the same proportion of each categories in attribute smoke as the whole dataset. The result is displayed in Figure 5.

```
print("proportion in test set:\n",cardio_test_set['smoke'].value_counts()/len(cardio_test_set),"\n")
print("proportion in whole dataset:\n",cardio_dataset['smoke'].value_counts()/len(cardio_dataset))
```

```
proportion in test set:
 0    0.911857
1    0.088143
Name: smoke, dtype: float64

proportion in whole dataset:
 0    0.911871
1    0.088129
Name: smoke, dtype: float64
```

Figure 5: Splitting using StratifiedShuffleSplit among 'smoke'

3 Correlations of data

Computing the standard correlation coefficient between each pair of attributes can shows the correlations of data and the approximate contributions of a specific feature to predict the targets. We build a correlation heat map.

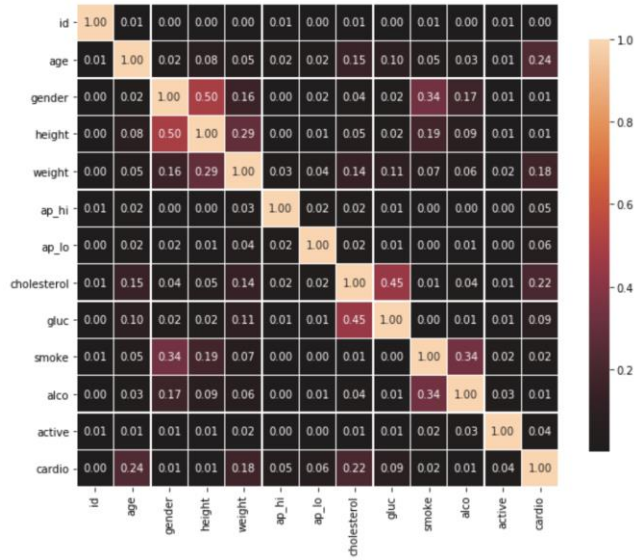

Figure 6: correlation heat map

In terms of the correlations of the data, we can find that the height, smoke and weight have a high correlation with gender, so we may need to consider to average the gender when splitting the training and testing dataset. The height also has high correlation with weight, so we can combine height and weight to a new attribute called BMI (Body Mass Index) which is weight / height^2 kg/(m^2). We can also combine the systolic blood pressure (ap_hi) and diastolic blood pressure (ap_lo) to a new attribute called MAP[2] (Mean Arterial Pressure) which is 1/3 ap_hi + 2/3 ap_lo. After we add these two attributes to the data, we get a new correlation heat map like the figure shown below.

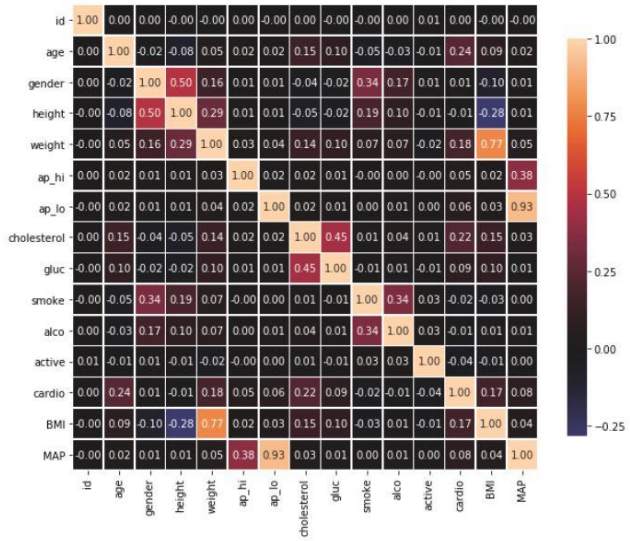

Figure 7: correlation heat map with BMI and MAP

From Figure 7, We can see that the correlations between BMI and cardio, MAP and cardio are not less than using height-weight and ap_hi-ap_lo combinations which means that using BMI and MAP

will have at least equivalent performance on predicting the cardiovascular disease.

4 Data analysis on the impact of gender

It's a reasonable inference from common sense that all the features in the datasets have something to do with the cardio disease except the gender remained unsure. In order to explore the impact of relationship between gender and other features on cardio disease. We implemented a simple multivariate analysis with sql(structured query language)[3] and pivot table[4]:

```
import pandasql as ps

# take the examples with cardio disease
q1 = """SELECT gender, id, smoke, alco FROM cardio where cardio=1"""
df1=ps.sqldf(q1, locals())

# the proportion of (non)smokers in different genders
df2=pd.pivot_table(df1, index=['gender'], values=['id'], columns=['smoke'], fill_value='',
                   aggfunc='count', margins=True, margins_name='Total')
print(df2.div( df2.iloc[:,-1], axis=0 ))
# the proportion of (non)drinkers in different genders
df3=pd.pivot_table(df1, index=['gender'], values=['id'], columns=['alco'], fill_value='',
                   aggfunc='count', margins=True, margins_name='Total')
print('\n')
print(df3.div( df3.iloc[:,-1], axis=0 ))
```

```
              id
smoke          0         1 Total
gender
1       0.983457  0.016543   1.0
2       0.795461  0.204539   1.0
Total   0.917051  0.082949   1.0


              id
alco           0         1 Total
gender
1       0.975711  0.024289   1.0
2       0.899504  0.100496   1.0
Total   0.948792  0.051208   1.0
```

Figure 8: multivariate analysis with sql and pivot table

The analysis above is based on the data with cardio disease(cardio=1) in the training set. The first pivot table above showed that men are more vulnerable to smoking than women, since about 20% of the men with cardio disease are smokers, while that of women is 1.65%. The similar phenomenon appeared on drinking and is demonstrated in the 2nd pivot table. About 10% of the men with cardio disease are alcohol takers, while that of women is 2.4%.

From this, we can conclude that gender is an important feature for predicting cardio disease.

5 Handling Categorical Attributes

Since the dataset doesn't have missing data and text data, so we just need to handle the categorical attribute. Luckily, we just have gender attributes to handle because male and female are originally represented by 1 and 2 which have magnitude difference while there is no magnitude difference between male and female. So, we use one-hot encoder to represent gender.

REFERENCES

[1] Géron A. Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems[M]. O'Reilly Media, 2019.

[2] https://en.wikipedia.org/wiki/Mean_arterial_pressure#Calculation

[3] https://pypi.org/project/pandasql/

[4] https://pandas.pydata.org/pandas-docs/stable/reference/api/pandas.pivot_table.html