CS 249, Winter 2024

Final Project

William Zhong

Author	Version	Date
William Zhong	Milestone 1.0	Feb 27, 2024

M1 Goal:

This is the milestone one from the final project. During this phase I already collected two datasets for the first step of analysis. This milestone will focus on data exploration/feature analysis. I will go through each single feature that correlates to hypertension and other numerical/categorical features.

Data Description:

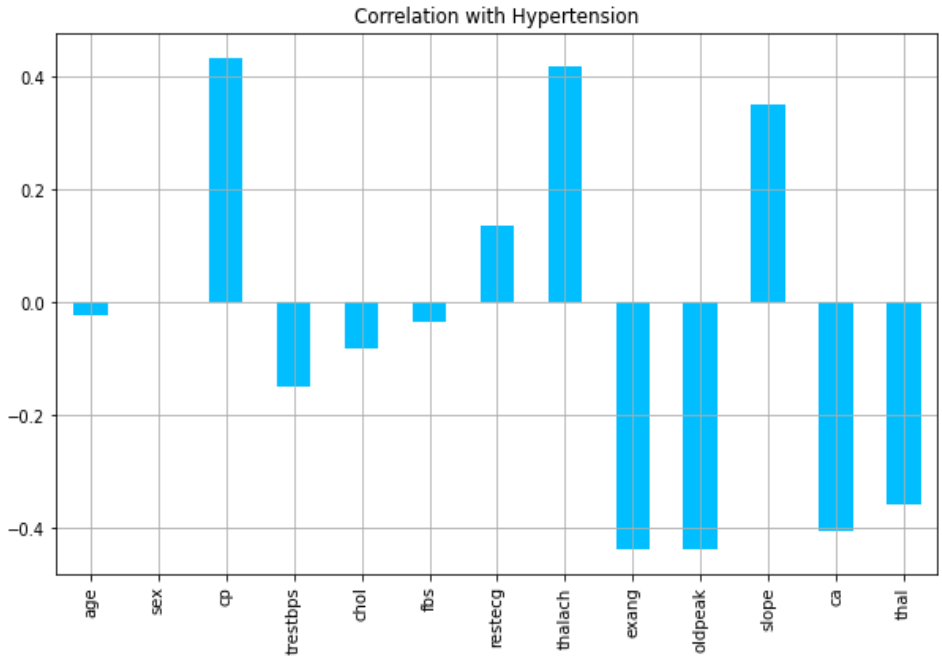
	age	sex	cp	trestbps	chol	fbs	restecg	thalach	exang	oldpeak	
count	26083.000000	26058.00000	26083.000000	26083.000000	26083.000000	26083.000000	26083.000000	26083.000000	26083.000000	26083.000000	26083.0
mean	55.661389	0.50000	0.958594	131.592992	246.246061	0.149753	0.526512	149.655024	0.326573	1.039512	1.4
std	15.189768	0.50001	1.023931	17.588809	51.643522	0.356836	0.525641	22.858109	0.468969	1.165138	0.6
min	11.000000	0.00000	0.000000	94.000000	126.000000	0.000000	0.000000	71.000000	0.000000	0.000000	0.0
25%	44.000000	0.00000	0.000000	120.000000	211.000000	0.000000	0.000000	133.000000	0.000000	0.000000	1.0
50%	56.000000	0.50000	1.000000	130.000000	240.000000	0.000000	1.000000	153.000000	0.000000	0.800000	1.0
75%	67.000000	1.00000	2.000000	140.000000	275.000000	0.000000	1.000000	166.000000	1.000000	1.600000	2.0
max	98.000000	1.00000	3.000000	200.000000	564.000000	1.000000	2.000000	202.000000	1.000000	6.200000	2.0

slope	ca	thal	target
26083.000000	26083.000000	26083.000000	26083.000000
1.400299	0.721849	2.318752	0.547253
0.616513	1.011608	0.604659	0.497772
0.000000	0.000000	0.000000	0.000000
1.000000	0.000000	2.000000	0.000000
1.000000	0.000000	2.000000	1.000000
2.000000	1.000000	3.000000	1.000000
2.000000	4.000000	3.000000	1.000000

Step-by-step Methodology:

1. Correlation between each feature and hypertension

Findings from step 1:



There are few features like ‘cp’, ‘thalach’ , ‘slope’, which are significantly correlated to hypertension. Only ‘restecg’ are weakly correlated.

Split the group 0 for non hypertension and group 1 for hypertension, I have the ATE result:

negative chest pain average outcome: 0.47023456685578796
positive chest pain average outcome: 1.3626173462239035
Estimated ATE is: 0.8923827793681156

Maximum heart rate achieved average outcome: 139.12439664662546
Maximum heart rate achieved average outcome: 158.36710102283874
Estimated ATE is: 19.242704376213283

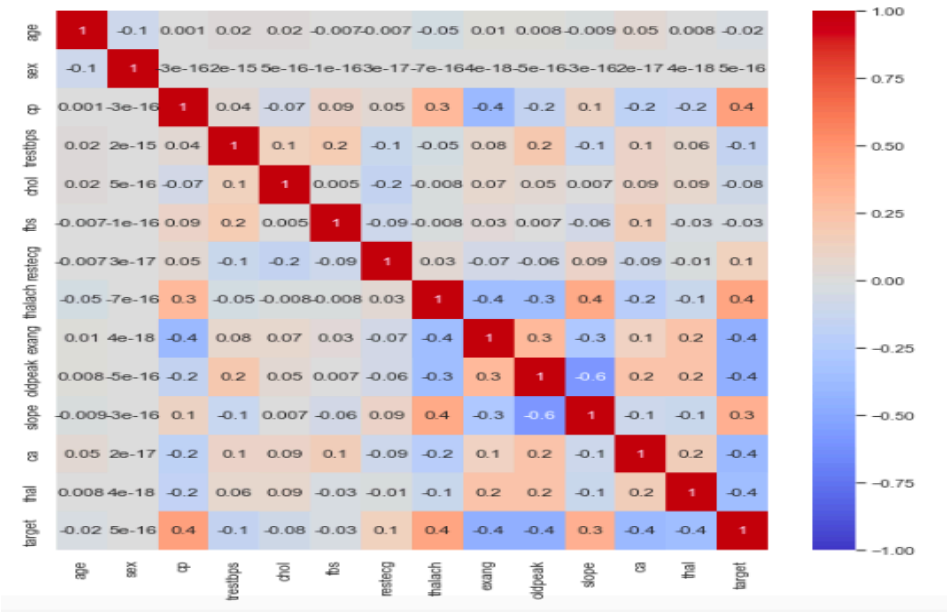
The slope of the peak exercise average outcome: 1.1631806249470742
The slope of the peak exercise average outcome: 1.596469104665826
Estimated ATE is: 0.43328847971875173

The Resting ECG average outcome: 0.4478787365568634
The Resting ECG average outcome: 0.5915650833683621
Estimated ATE is: 0.14368634681149872

It looks like the maximum heart rate has a significant effect across two groups split by the target.

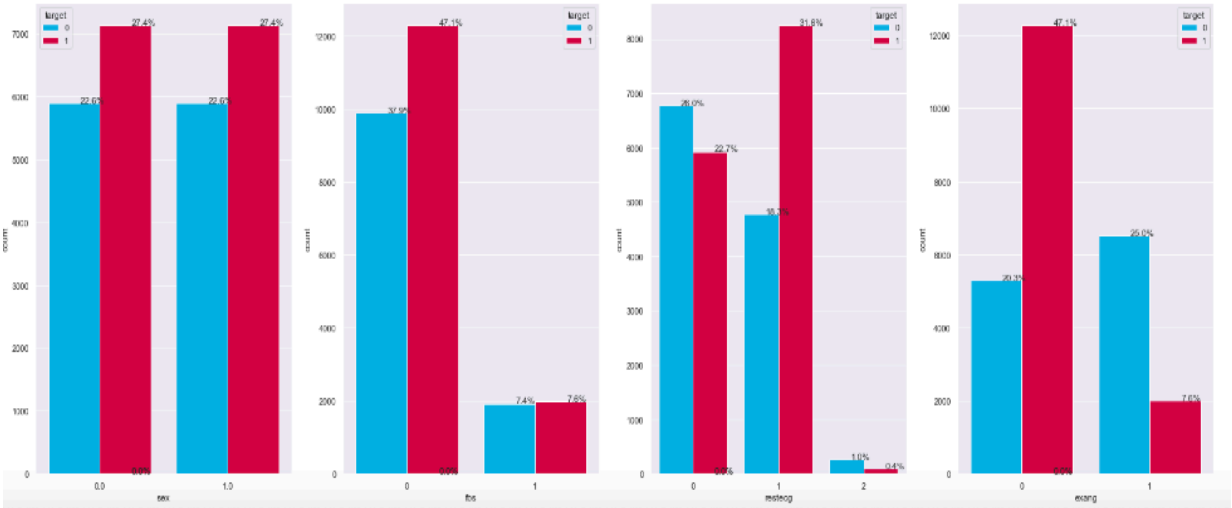
2. Correlation between any two features:

Findings from step 2:



Since ‘cp’, ‘thalach’,and ‘slope’ are not higher/smaller than +- 0.3. Hence I cannot drop any features at this moment.

3. Check if binary/categorical value correlated to hypertension



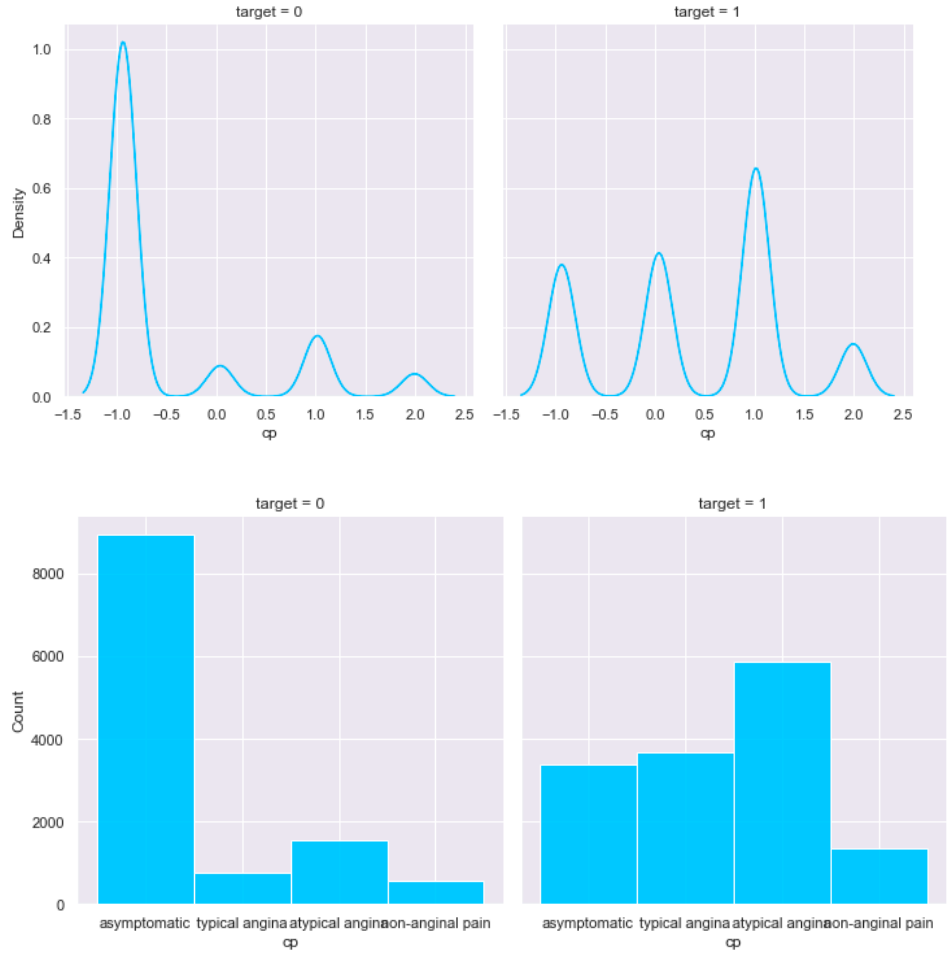
It seems sex has no effect on hypertension.
Patients who are not fasting blood sugar have a higher risk of hypertension.
Abnormal resting ECG has higher risk of hypertension
Looks like exercise induced angina has lower risk of hypertension

4. Distribution of correlated features to hypertension

Chest Pain:

cp target	asymptomatic	typical angina	atypical angina	non-anginal pain
0	8930.0	776.0	1532.0	571.0
1	3384.0	3680.0	5860.0	1350.0

```
<Figure size 864x360 with 0 Axes>
```

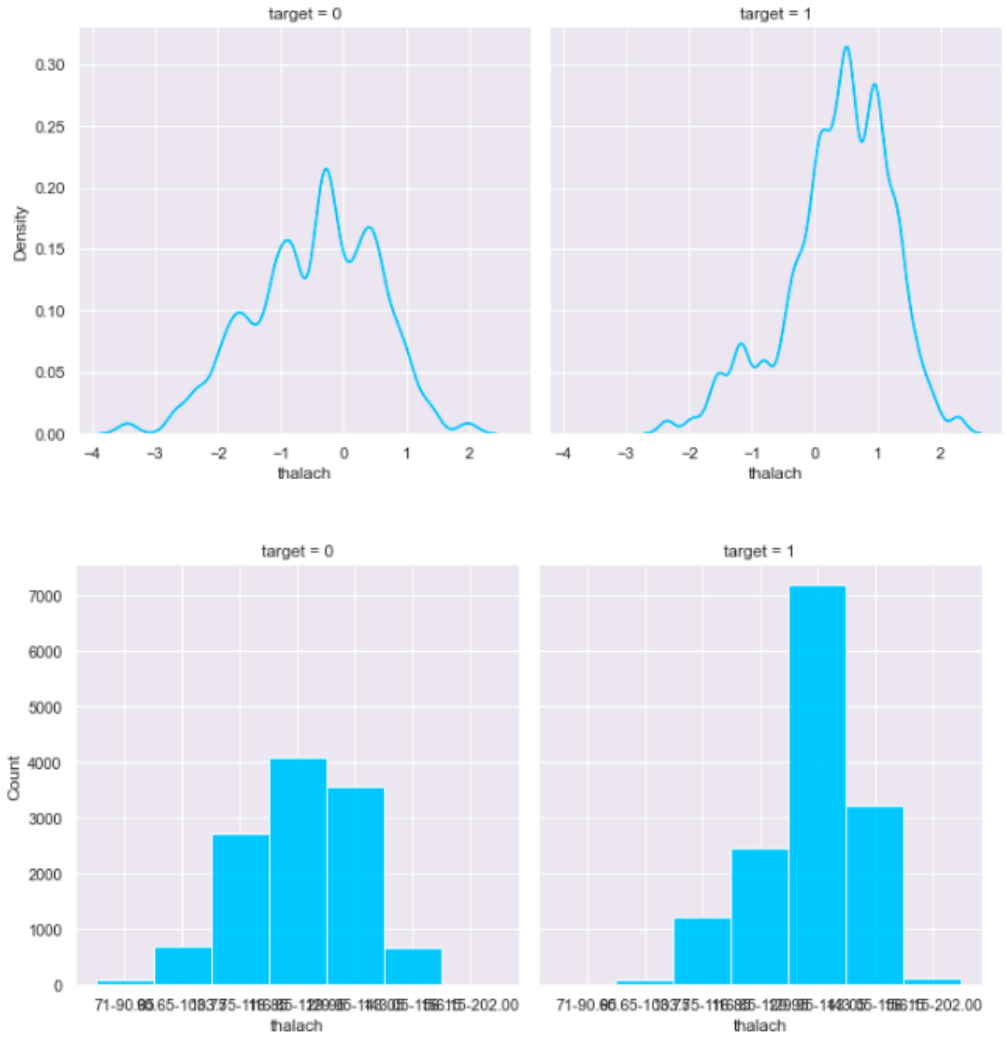


The distribution of the chest pain level of the target column values is different. This feature is an explanatory feature; People with high chest pain are more likely to have hypertension than those with normal values.

Maximum heart rate achieved:

thalach	71-90.65	90.65-103.75	103.75-116.85	116.85-129.95	129.95-143.05	143.05-156.15	156.15-202.00
target							
0	82.0	692.0	2730.0	4082.0	3569.0	654.0	0.0
1	0.0	84.0	1216.0	2448.0	7186.0	3232.0	108.0

```
<Figure size 864x360 with 0 Axes>
```

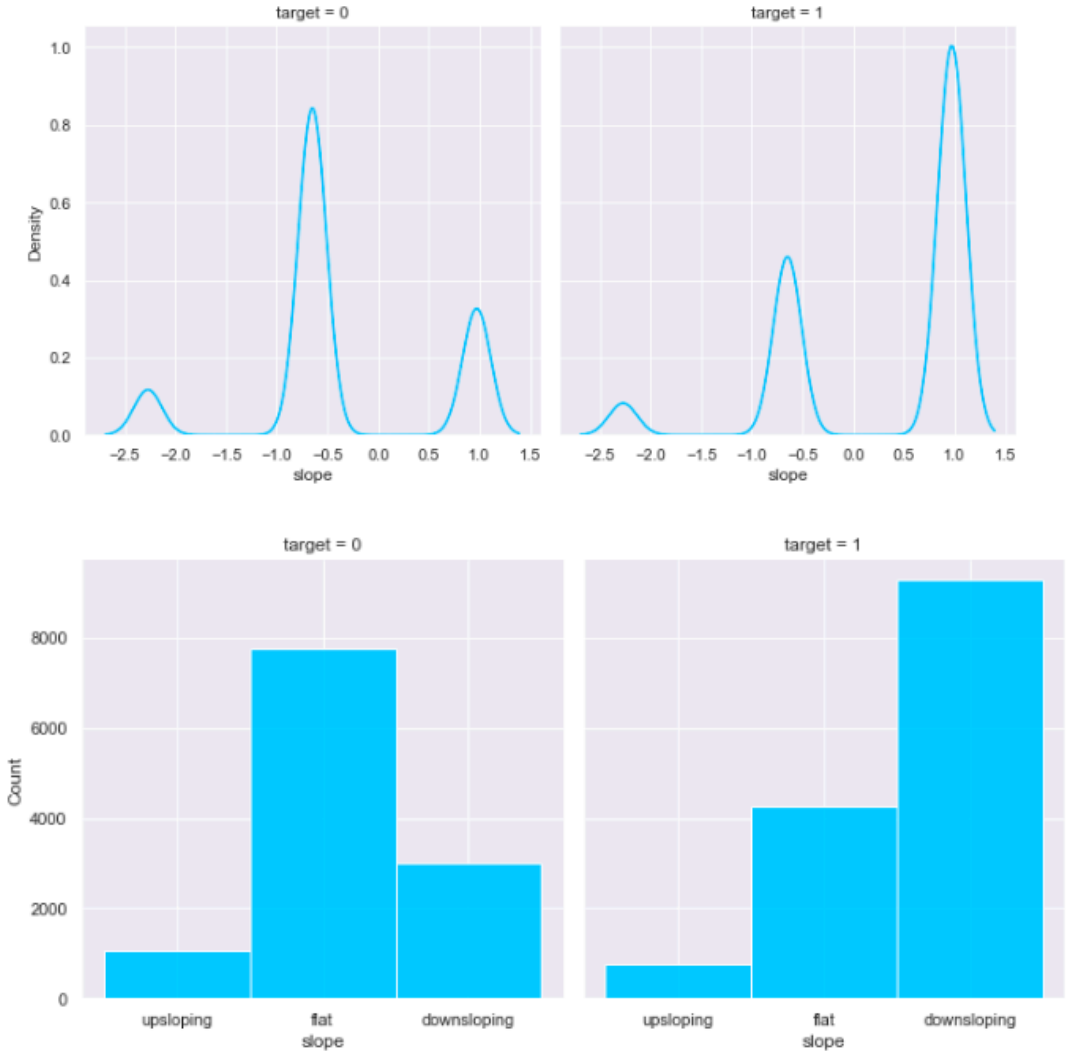


The distribution of the maximum heart rate of the target column values is different. This feature is an explanatory feature; People with high maximum heart rate are more likely to have hypertension than those with normal values.

The slope of the peak exercise ST segment:

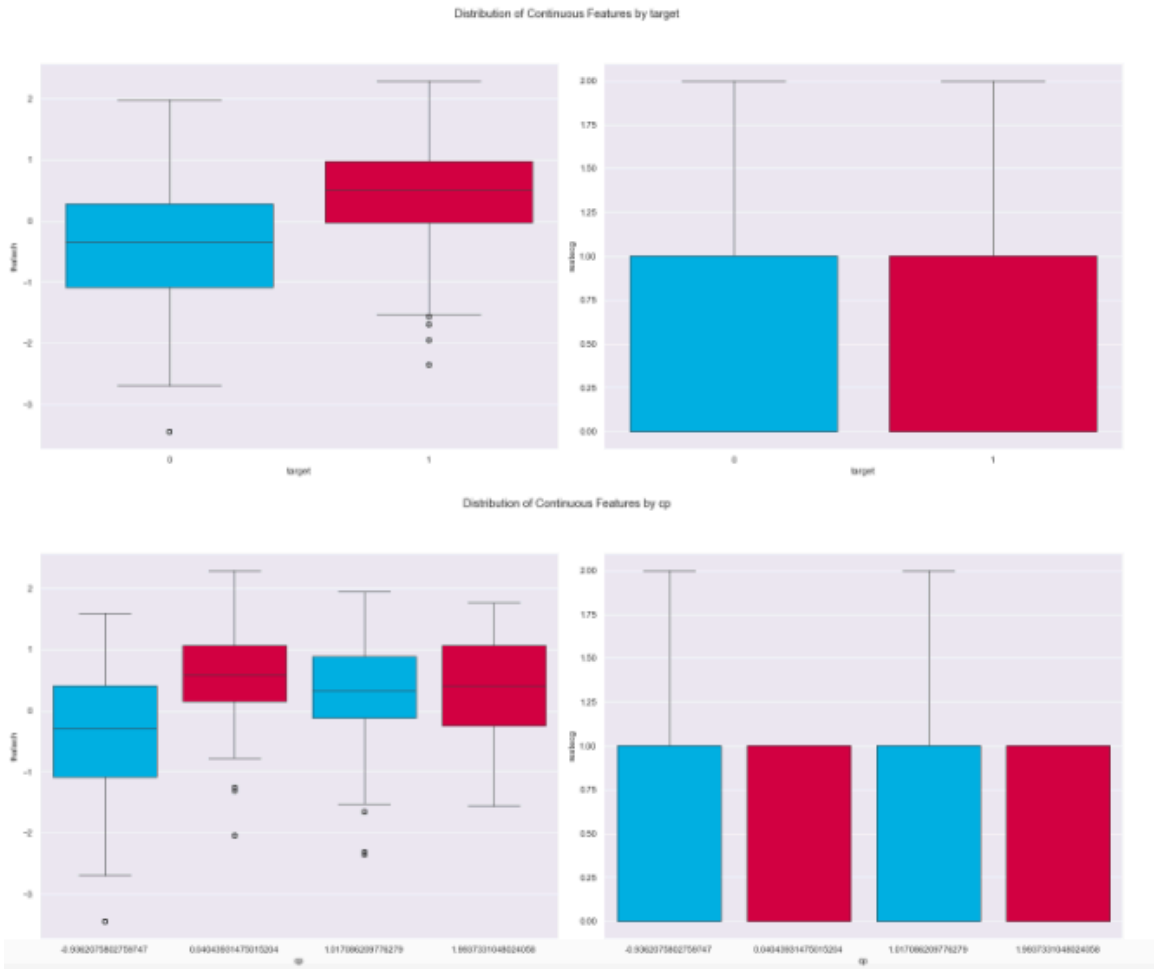
slope	upsloping	flat	downsloping
target			
0	1070.0	7742.0	2997.0
1	756.0	4248.0	9270.0

```
<Figure size 864x360 with 0 Axes>
```



The distribution of the slope of the peak exercise of the target column values is different. This feature is an explanatory feature; People with downsloping are more likely to have hypertension than those with flat slope.

5. Distribution of numerical features by hypertension and correlated features



The median, range and values in the "thalach" feature is higher when there is chest pain. The median and values in the same feature is higher when there is hypertension.

Code Appendix for M1

```
‘’’
import numpy as np
import pandas as pd
from scipy import stats
from sklearn.preprocessing import StandardScaler,MinMaxScaler
import matplotlib.pyplot as plt
import seaborn as sns

hypertesion_data = pd.read_csv('/Users/wiizh/CS249 Final Project/hypertension_data.csv')
stroke_data = pd.read_csv('/Users/wiizh/CS249 Final Project/stroke_data.csv')

print(hypertesion_data.head(3))
print(stroke_data.head(3))

negative_hypertension_data = hypertesion_data[hypertesion_data.target == 0]
positive_hypertension_data = hypertesion_data[hypertesion_data.target == 1]

negative_cp_mean_hypertension_data = np.mean(negative_hypertension_data.cp)
positive_cp_mean_hypertension_data = np.mean(positive_hypertension_data.cp)
ate = positive_cp_mean_hypertension_data - negative_cp_mean_hypertension_data

print("negative chest pain average outcome: ", negative_cp_mean_hypertension_data)
print("positive chest pain average outcome: ", positive_cp_mean_hypertension_data)
print("Estimated ATE is: ", ate)

negative_thalach_mean_hypertension_data = np.mean(negative_hypertension_data.thalach)
positive_thalach_mean_hypertension_data = np.mean(positive_hypertension_data.thalach)
ate = positive_thalach_mean_hypertension_data - negative_thalach_mean_hypertension_data

print("Maximum heart rate achieved average outcome: ", negative_thalach_mean_hypertension_data)
print("Maximum heart rate achieved average outcome: ", positive_thalach_mean_hypertension_data)
print("Estimated ATE is: ", ate)

negative_slope_mean_hypertension_data = np.mean(negative_hypertension_data.slope)
positive_slope_mean_hypertension_data = np.mean(positive_hypertension_data.slope)
ate = positive_slope_mean_hypertension_data - negative_slope_mean_hypertension_data

print("The slope of the peak exercise average outcome: ", negative_slope_mean_hypertension_data)
print("The slope of the peak exercise average outcome: ", positive_slope_mean_hypertension_data)
print("Estimated ATE is: ", ate)

negative_restecg_mean_hypertension_data = np.mean(negative_hypertension_data.restecg)
positive_restecg_mean_hypertension_data = np.mean(positive_hypertension_data.restecg)
ate = positive_restecg_mean_hypertension_data - negative_restecg_mean_hypertension_data

print("The Resting ECG average outcome: ", negative_restecg_mean_hypertension_data)
print("The Resting ECG average outcome: ", positive_restecg_mean_hypertension_data)
print("Estimated ATE is: ", ate)

hypertesion_data.drop('target', axis=1).corrwith(hypertesion_data.target).plot(kind='bar', grid=True, figsize=(10, 6), title="Correlation
with Hypertension",color="deepskyblue");

sns.set(rc = {'figure.figsize':(10,10)})
sns.heatmap(hypertesion_data.corr(),vmin=-1, vmax=1, annot = True, fmt='.1g',cmap= 'coolwarm')

features = [x for x in hypertesion_data.columns if x in ['sex', 'restecg', 'fbs', 'exang']]
plt.figure(figsize = (30,23))
plt.suptitle('Hypertension by categorical features')
```

```
#subplots
for i, feature in enumerate(features):
    plt.subplot(2,4, i+1)
    x = sns.countplot(x=feature ,hue='target', data=hypertesion_data, palette = ['deepskyblue','crimson'])
    for z in x.patches:
      x.annotate('{:.1f}'.format((z.get_height()/hypertesion_data.shape[0])*100)+'%',(z.get_x()+0.25, z.get_height()+0.01))

#scale the data before pairplot
data_pairplot = hypertesion_data
float_columns = [x for x in hypertesion_data.columns if x in ['slope','cp','thalach']]

sc = StandardScaler()
data_pairplot[float_columns] = sc.fit_transform(data_pairplot[float_columns])_
data_pairplot.head(4)

plt.figure(figsize=(12,5))
sns.displot(x='cp', col='target' , data = hypertesion_data, kind="kde" ,color = 'deepskyblue');

cp = pd.cut( hypertesion_data['cp'],bins=[-1.5,-0.5,0.5,1.5,2.5],labels=['asymptomatic','typical angina','atypical angina','non-anginal
pain'])
cp_temp = pd.crosstab(hypertesion_data['target'],cp,rownames=['target'])
cp_temp = cp_temp.astype(float)
cp_temp

cp_temp_sum_lst=list(cp_temp.transpose().sum().values)
for idx in range(cp_temp.values.shape[0]):
    cp_temp.values[idx]= cp_temp.values[idx]/cp_temp_sum_lst[idx]*100
cp_temp

plt.figure(figsize=(12,20))
sns.displot(data=hypertesion_data,col='target',x=cp,color='deepskyblue');

plt.figure(figsize=(12,5))
sns.displot(x='thalach', col='target' , data = hypertesion_data, kind="kde" ,color = 'deepskyblue');

thalach = pd.cut( hypertesion_data['thalach'],bins=[-4,-3,-2,-1,0,1,2,2.5],labels=['71-90.65','90.65-103.75','103.75-116.85',
'116.85-129.95','129.95-143.05','143.05-156.15','156.15-202.00'])
thalach_temp = pd.crosstab(hypertesion_data['target'],thalach,rownames=['target'])
thalach_temp = thalach_temp.astype(float)
thalach_temp
plt.figure(figsize=(12,20))
sns.displot(data=hypertesion_data,col='target',x=thalach,color='deepskyblue');

plt.figure(figsize=(12,5))
sns.displot(x='slope', col='target' , data = hypertesion_data, kind="kde" ,color = 'deepskyblue');

slope = pd.cut( hypertesion_data['slope'],bins=[-2.75,-1.5,0,1.5],labels=['upsloping','flat','downsloping'])
slope_temp = pd.crosstab(hypertesion_data['target'],slope,rownames=['target'])
slope_temp = slope_temp.astype(float)
slope_temp

plt.figure(figsize=(12,20))
sns.displot(data=hypertesion_data,col='target',x=slope,color='deepskyblue');

plt.figure(figsize=(12,5))
sns.displot(x='restecg', col='target' , data = hypertesion_data, kind="kde" ,color = 'deepskyblue');

restecg = pd.cut(hypertesion_data['restecg'],bins=[-0.75,0.5,1.5,2.6],labels=['normal','ST-T wave abnormality','probable'])
restecg_temp = pd.crosstab(hypertesion_data['target'],restecg,rownames=['target'])
restecg_temp = restecg_temp.astype(float)
restecg_temp
```

```
features = ['target','cp']
for i in enumerate(features):
  box_cols = ['thalach','restecg']
  fig, axes = plt.subplots(nrows=1, ncols=2, figsize=(20,8))
  fig.suptitle('Distribution of Continuous Features by '+i[1], y = 1.05);
  for col, ax in zip(box_cols, axes.ravel()):
      sns.boxplot(data=hypertesion_data, x=i[1], y=col ,palette = ['deepskyblue','crimson'], ax=ax)
  plt.tight_layout()
```