

Introduction

Heart disease is taken to be the leading cause of death in the world because it is able to account for about 17.9 million deaths annually, according to the World Health Organization-2023 (WHO, 2023). In relation to heart diseases, much research has to do with blood pressure, cholesterol, as well as other clinical variables, since some of the major classifying causes that have been analyzed to produce, through comparison of results, a good method for early diagnosis and prevention of this disease. These indicators are critical not only in understanding cardiovascular health but also in guiding public health interventions and clinical decision-making (Benjamin et al., 2019). The project is to explore the relationship of relevant clinical variables such as resting blood pressure - trtbps, cholesterol level - chol, and maximum heart rate-thalachh, with the presence of heart disease. The dataset used for this analysis is taken from the UCI Machine Learning Repository and is based on the Heart Disease data prepared by The American Journal of Cardiology (Detrano et al., 1989). The dataset contains the medical measures of 303 patients and was analyzed with the aim of heart disease prediction in light of several physiological and lifestyle variables. Due to the dataset's credibility, considering its usage within multiple research situational analyses, it will be a reliable resource that should come useful when trying to answer critical questions related to cardiovascular health. The following analysis aims to test various hypotheses, such as that the greater the resting blood pressure, the lower the risk of heart disease, and identify thresholds for other clinical variables, such as cholesterol, which could be an early warning. These research questions also align with previous findings in cardiovascular research, such as the relationship between abnormal blood pressure and increased risk of heart disease (Whelton et al., 2018), and the value of cholesterol levels in predicting atherosclerotic cardiovascular disease (Stone et al.2013). The project will examine these relationships in an effort to provide actionable insights helpful in the prevention and early detection of heart disease. It also recognizes possible limitations because some datasets have relatively few features and explores how modern computational techniques can bring clarity to understanding such multifaceted medical conditions.

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.linear_model import LinearRegression, LogisticRegression
import statsmodels.api as sm
```

Start coding or generate with AI.

Research Questions:

1. Does the likelihood of heart disease decrease with increasing resting blood pressure (trtbps)?
2. Is there a significant difference in heart disease risk and o2 saturation between men and women?
3. How does high cholesterol (chol) affect the probability of heart disease? Can we identify a cholesterol threshold beyond which the risk significantly increases?

Start coding or generate with AI.


Start coding or generate with AI.

Double-click (or enter) to edit

Double-click (or enter) to edit

Double-click (or enter) to edit


```
heart_data = pd.read_csv("heart.csv")
heart_data.head()
```




	age	sex	cp	trtbps	chol	fbs	restecg	thalachh	exng	oldpeak	slp	caa	thall	output
0	63	1	3	145	233	1	0	150	0	2.3	0	0	1	1
1	37	1	2	130	250	0	1	187	0	3.5	0	0	2	1
2	41	0	1	130	204	0	0	172	0	1.4	2	0	2	1
3	56	1	1	120	236	0	1	178	0	0.8	2	0	2	1
4	57	0	0	120	354	0	1	163	1	0.6	2	0	2	1

Double-click (or enter) to edit

```
o2_saturation_df = pd.read_csv("o2Saturation.csv", header=None)
o2_saturation_df.columns = ['o2']
o2_saturation_df.head()
```



	o2
0	98.6
1	98.6
2	98.6
3	98.6
4	98.1



Start coding or [generate](#) with AI.

```
import nbimporter
from Data_Cleaning import clean_data
heart_data_cleaned = clean_data()
```

Summary stats and plots for before cleaning and after

```
summary_statistics_heart = heart_data.describe()
print(summary_statistics_heart)
```

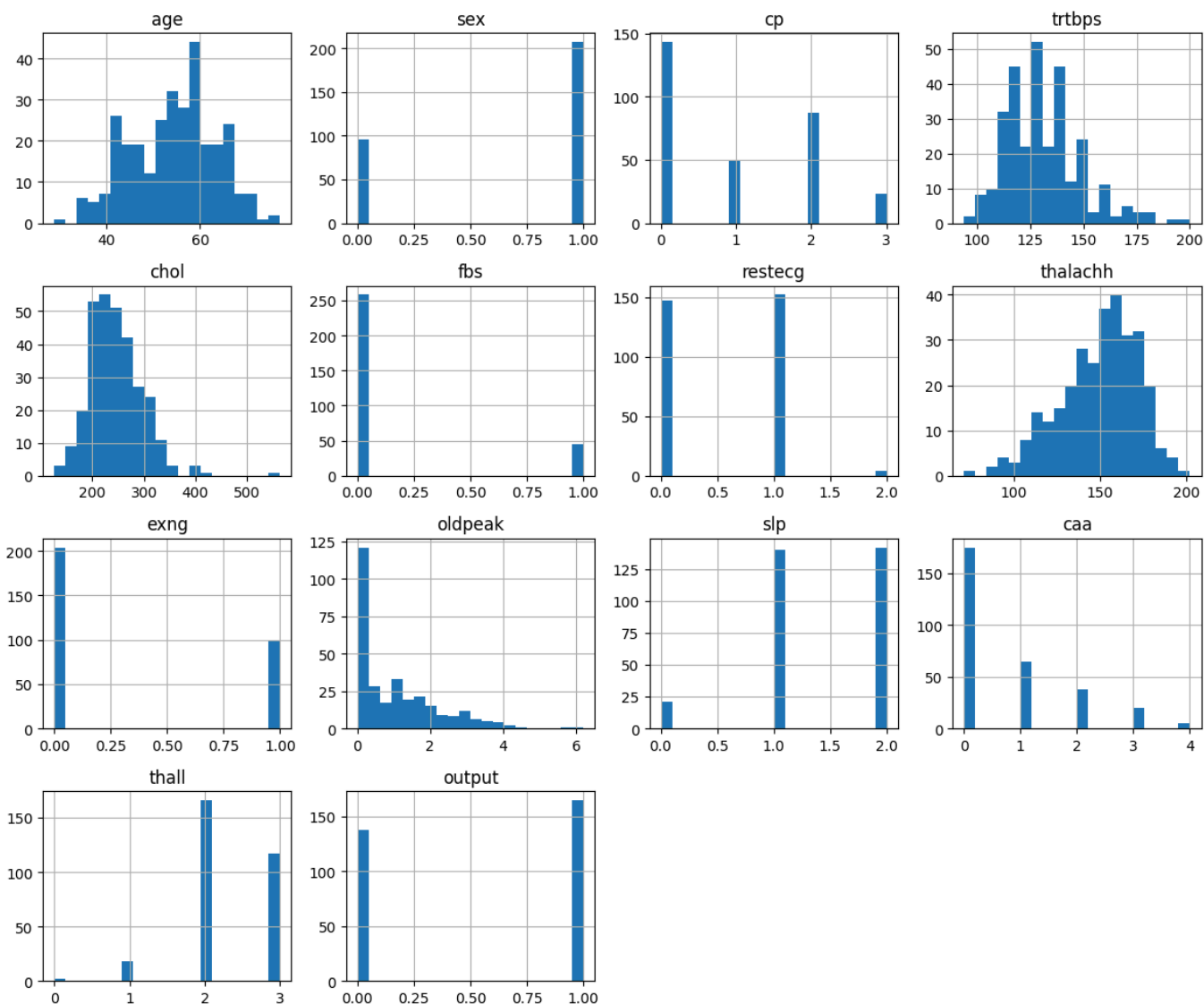
```
heart_data.hist(figsize=(12, 10), bins=20)
plt.tight_layout()
plt.show()
```



	age	sex	cp	trtbps	chol	fbf
count	303.000000	303.000000	303.000000	303.000000	303.000000	303.000000
mean	54.366337	0.683168	0.966997	131.623762	246.264026	0.148515
std	9.082101	0.466011	1.032052	17.538143	51.830751	0.356198
min	29.000000	0.000000	0.000000	94.000000	126.000000	0.000000
25%	47.500000	0.000000	0.000000	120.000000	211.000000	0.000000
50%	55.000000	1.000000	1.000000	130.000000	240.000000	0.000000
75%	61.000000	1.000000	2.000000	140.000000	274.500000	0.000000
max	77.000000	1.000000	3.000000	200.000000	564.000000	1.000000

	restecg	thalachh	exng	oldpeak	slp	caa
count	303.000000	303.000000	303.000000	303.000000	303.000000	303.000000
mean	0.528053	149.646865	0.326733	1.039604	1.399340	0.729373
std	0.525860	22.905161	0.469794	1.161075	0.616226	1.022606
min	0.000000	71.000000	0.000000	0.000000	0.000000	0.000000
25%	0.000000	133.500000	0.000000	0.000000	1.000000	0.000000
50%	1.000000	153.000000	0.000000	0.800000	1.000000	0.000000
75%	1.000000	166.000000	1.000000	1.600000	2.000000	1.000000
max	2.000000	202.000000	1.000000	6.200000	2.000000	4.000000

	thall	output
count	303.000000	303.000000
mean	2.313531	0.544554
std	0.612277	0.498835
min	0.000000	0.000000
25%	2.000000	0.000000
50%	2.000000	1.000000
75%	3.000000	1.000000
max	3.000000	1.000000



```
summary_statistics_o2 = o2_saturation_df.describe()
print(summary_statistics_o2)

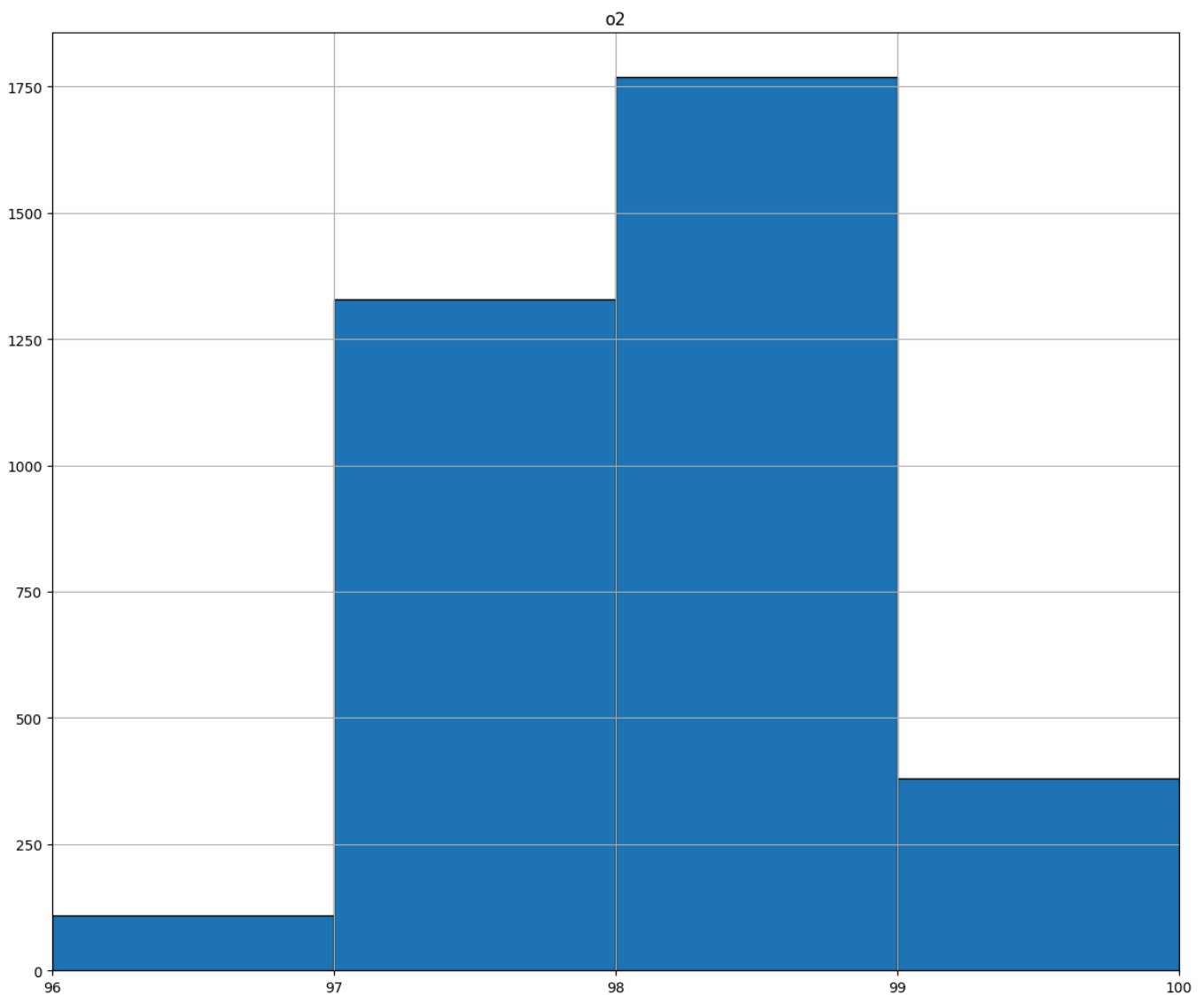
bin_edges = [96, 97, 98, 99, 100]

o2_saturation_df.hist(figsize=(12, 10), bins=bin_edges, edgecolor='black')

plt.xlim([96, 100])
plt.xticks(bin_edges)
plt.tight_layout()

# Show the plot
plt.show()
```

```
count  3586.000000
mean    98.239375
std      0.726260
min     96.500000
25%     97.600000
50%     98.600000
75%     98.600000
max     99.600000
```



```
summary_statistics_cleaned = heart_data_cleaned.describe()
print(summary_statistics_cleaned)
```

	age	sex	cp	trtbps	chol	fbs	\
count	298.000000	298.000000	298.000000	298.000000	298.000000	298.000000	
mean	54.228188	0.694631	0.969799	131.553691	243.043624	0.147651	
std	9.081836	0.461338	1.032678	17.612519	45.094679	0.355350	
min	29.000000	0.000000	0.000000	94.000000	126.000000	0.000000	
25%	47.000000	0.000000	0.000000	120.000000	211.000000	0.000000	
50%	55.000000	1.000000	1.000000	130.000000	240.000000	0.000000	
75%	60.750000	1.000000	2.000000	140.000000	273.000000	0.000000	
max	77.000000	1.000000	3.000000	200.000000	360.000000	1.000000	

	restecg	thalachh	exng	oldpeak	slp	caa	\
count	298.000000	298.000000	298.000000	298.000000	298.000000	298.000000	
mean	0.536913	149.546980	0.328859	1.025168	1.402685	0.721477	
std	0.525748	23.079853	0.470589	1.156392	0.618671	1.018156	
min	0.000000	71.000000	0.000000	0.000000	0.000000	0.000000	
25%	0.000000	133.000000	0.000000	0.000000	1.000000	0.000000	
50%	1.000000	152.000000	0.000000	0.650000	1.000000	0.000000	
75%	1.000000	166.750000	1.000000	1.600000	2.000000	1.000000	
max	2.000000	202.000000	1.000000	6.200000	2.000000	4.000000	

	thall	output	o2
count	298.000000	298.000000	298.000000
mean	2.308725	0.543624	97.484228
std	0.612983	0.498931	0.355599
min	0.000000	0.000000	96.500000
25%	2.000000	0.000000	97.500000
50%	2.000000	1.000000	97.500000
75%	3.000000	1.000000	97.500000
max	3.000000	1.000000	98.600000

Preregistration Statements

After reading through the data and considering our existing domain knowledge, we came up with the following hypotheses.

- Hypothesis 1: The likelihood of heart disease decreases with increasing resting blood pressure (trtbps). Reasoning: Anybody with abnormally low or high blood pressure is more likely to be at risk of heart disease compared to someone with a normal blood pressure as it is simply a sign that your heart is not working correctly but here we are trying to see if there is a difference between the likelihood of getting heart disease between having low bp and having high bp.

Analysis: Run a linear regression where we input resting blood pressure values and output the target values in the dataset that correspond to the values 0 and 1, where 0 is less chance of a heart attack and 1 is more chance of a heart attack. Since we are looking for a negative association, we will test whether $\beta_{trtbps} < 0$.

- Hypothesis 2: There is a significant difference in heart disease risk and o2 saturation between men and women. Reasoning: While it isn't clear what gender is more likely to get heart disease for a number of factors, a low o2 saturation is thought to be an indicator of a high chance of heart disease and we would like to establish whether there is a significant difference between men and women in such indicators.

Analysis: We will run a linear regression where we input the gender as a dummy variable, and output the different proportions in O2 saturation and heart disease risk. Because we are testing for the presence of significant difference overall, we will test whether $\beta_{o2sat} \neq 0$.

Exploratory data analysis

```
#Hypothesis One
visitor_model = LinearRegression().fit(heart_data_cleaned[['trtbps']], heart_data_cleaned[['output']])
slope = visitor_model.coef_[0][0].round(2)
intercept = visitor_model.intercept_[0].round(2)

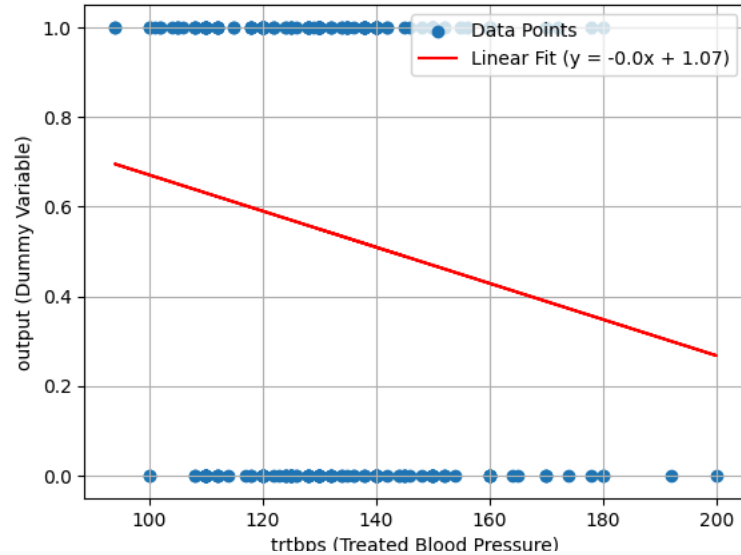
print(f"The model's slope is: {slope}")
print(f"The model's intercept is: {intercept}")

y_pred = visitor_model.predict(heart_data_cleaned[['trtbps']])

plt.scatter(heart_data_cleaned['trtbps'], heart_data_cleaned['output'], label='Data Points')
plt.plot(heart_data_cleaned[['trtbps']], y_pred, color='red', label=f'Linear Fit (y = {slope}x + {intercept})')
plt.title('Linear Regression of Blood Pressure (trtbps) vs Likelihood of Heart Attack ')
plt.xlabel('trtbps (Treated Blood Pressure)')
plt.ylabel('output (Dummy Variable)')
plt.legend()
plt.grid(True)
plt.show()
```

↻ The model's slope is: -0.0
The model's intercept is: 1.07

Linear Regression of Blood Pressure (trtbps) vs Likelihood of Heart Attack



```
#Hypothesis Two
# Linear regression for O2 saturation
X_o2 = sm.add_constant(heart_data_cleaned[['sex']])
y_o2 = heart_data_cleaned['o2']
model_o2 = sm.OLS(y_o2, X_o2).fit()

# Linear regression for heart disease risk
X_hd = sm.add_constant(heart_data_cleaned[['sex']])
y_hd = heart_data_cleaned['output']
model_hd = sm.OLS(y_hd, X_hd).fit()

# Display regression results
print("Linear Regression Results for O2 Saturation:")
print(model_o2.summary())
print("\nLinear Regression Results for Heart Disease Risk:")
print(model_hd.summary())

# Visualize the difference in O2 saturation by gender
plt.figure(figsize=(8, 6))
heart_data_cleaned.groupby('sex')['o2'].mean().plot(kind='bar', color=['skyblue', 'orange'])
plt.title('Average O2 Saturation by Gender')
plt.ylabel('O2 Saturation')
plt.xlabel('Gender')
plt.grid(axis='y')
plt.show()

# Visualize the difference in heart disease risk by gender
plt.figure(figsize=(8, 6))
heart_data_cleaned.groupby('sex')['output'].mean().plot(kind='bar', color=['skyblue', 'orange'])
plt.title('Average Heart Disease Risk by Gender')
plt.ylabel('Heart Disease Risk')
plt.xlabel('Gender')
plt.grid(axis='y')
plt.show()
```



Linear Regression Results for O2 Saturation:

OLS Regression Results

```

=====
Dep. Variable:          o2      R-squared:          0.011
Model:                  OLS      Adj. R-squared:       0.008
Method:                 Least Squares      F-statistic:       3.325
Date:                  Fri, 22 Nov 2024      Prob (F-statistic): 0.0692
Time:                  03:08:56      Log-Likelihood:    -112.56
No. Observations:      298      AIC:              229.1
Df Residuals:          296      BIC:              236.5
Df Model:               1
Covariance Type:       nonrobust
=====

```

	coef	std err	t	P> t	[0.025	0.975]
const	97.5407	0.037	2626.872	0.000	97.468	97.614
sex	-0.0812	0.045	-1.823	0.069	-0.169	0.006

```

=====
Omnibus:                 41.834      Durbin-Watson:       0.184
Prob(Omnibus):           0.000      Jarque-Bera (JB):    330.020
Skew:                    0.078      Prob(JB):            2.17e-72
Kurtosis:                8.153      Cond. No.            3.38
=====

```

Notes:

[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.

Linear Regression Results for Heart Disease Risk:

OLS Regression Results

```

=====
Dep. Variable:          output      R-squared:          0.082
Model:                  OLS      Adj. R-squared:       0.079
Method:                 Least Squares      F-statistic:       26.31
Date:                  Fri, 22 Nov 2024      Prob (F-statistic): 5.28e-07
Time:                  03:08:56      Log-Likelihood:    -202.46
No. Observations:      298      AIC:              408.9
Df Residuals:          296      BIC:              416.3
Df Model:               1
Covariance Type:       nonrobust
=====

```

	coef	std err	t	P> t	[0.025	0.975]
const	0.7582	0.050	15.102	0.000	0.659	0.857
sex	-0.3090	0.060	-5.129	0.000	-0.428	-0.190

```

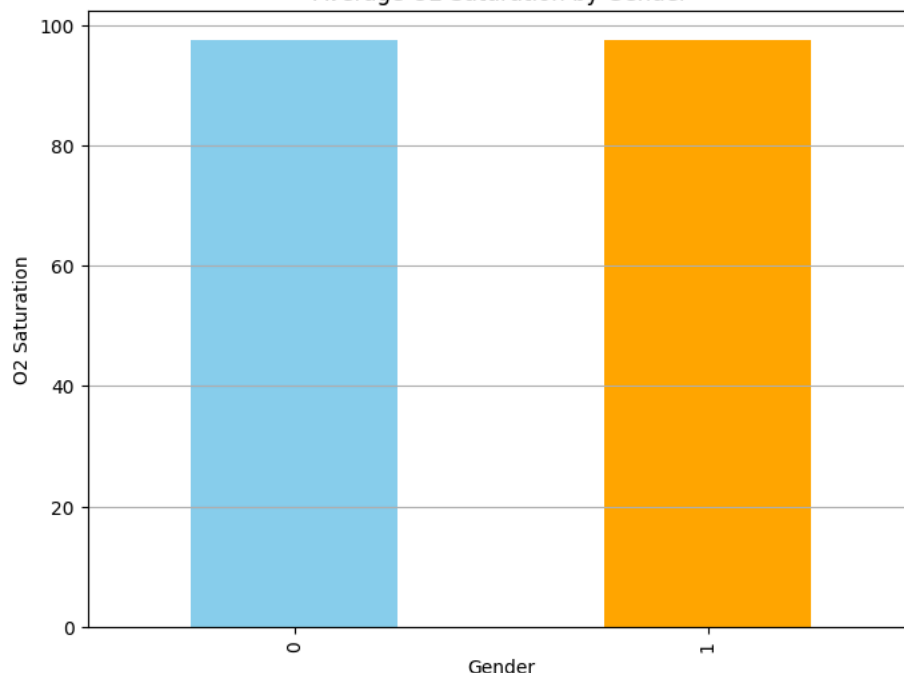
=====
Omnibus:                 2033.078      Durbin-Watson:       0.174
Prob(Omnibus):           0.000      Jarque-Bera (JB):    34.830
Skew:                    -0.106      Prob(JB):            2.73e-08
Kurtosis:                1.339      Cond. No.            3.38
=====

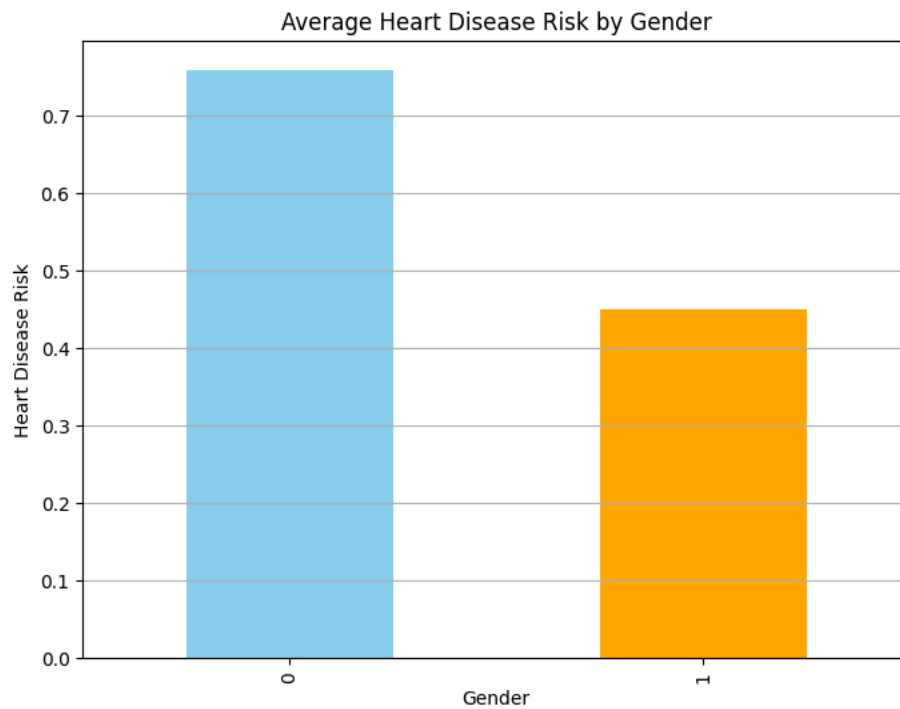
```

Notes:

[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.

Average O2 Saturation by Gender





Here is a further research question pertaining to whether high cholesterol significantly increases heart disease risk, and can a threshold be identified?

```
X = heart_data[['chol']] # Independent variable: cholesterol levels
X = sm.add_constant(X) # Add constant term for intercept
y = heart_data['output'] # Dependent variable: heart disease (1 = presence, 0 = absence)

# Build the logistic regression model
logit_model = sm.Logit(y, X)
result = logit_model.fit()

# Print the summary of the logistic regression
print(result.summary())

# Odds Ratios Calculation
odds_ratios = pd.DataFrame({
    "Variable": result.params.index,
    "Odds Ratio": result.params.apply(lambda x: round(np.exp(x), 2)), # Use np.exp for exponentiation
    "p-value": result.pvalues
})
print("\nOdds Ratios:")
print(odds_ratios)

# Visualizing Cholesterol Levels and Heart Disease Risk
plt.figure(figsize=(10, 6))
sns.histplot(data=heart_data, x='chol', hue='output', kde=True, bins=30, palette='coolwarm')
plt.title('Distribution of Cholesterol Levels by Heart Disease Presence')
plt.xlabel('Cholesterol Level')
plt.ylabel('Frequency')
plt.legend(title='Heart Disease', labels=['No Disease', 'Disease'])
plt.show()
```


Optimization terminated successfully.
 Current function value: 0.685527
 Iterations 4

Logit Regression Results

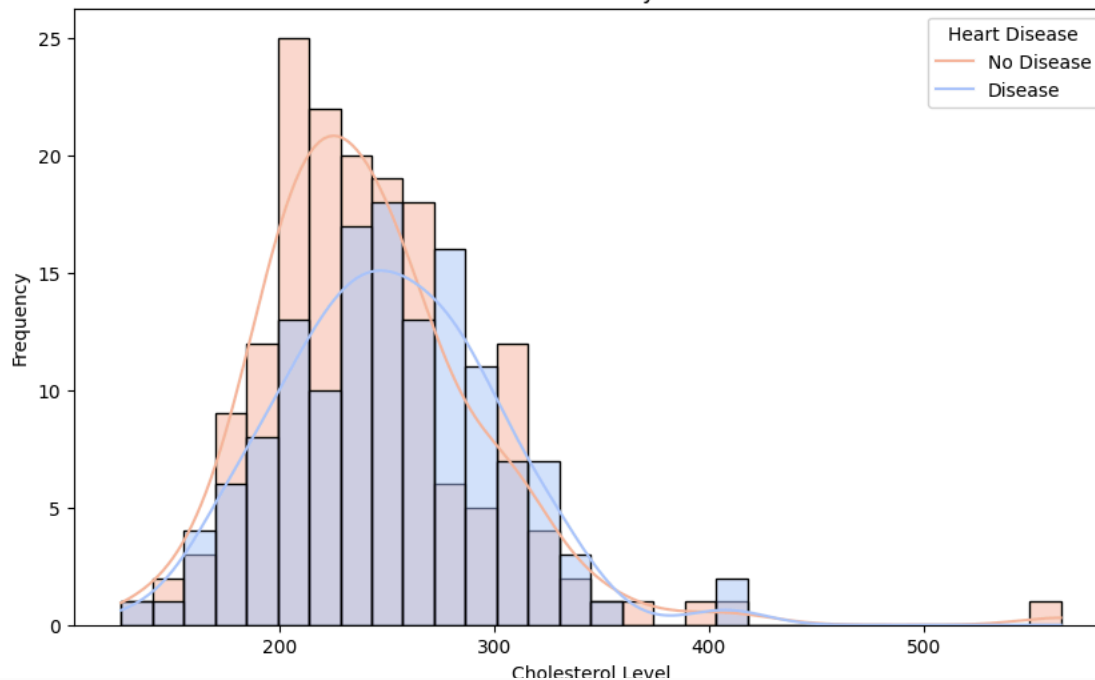
=====						
Dep. Variable:	output	No. Observations:	303			
Model:	Logit	Df Residuals:	301			
Method:	MLE	Df Model:	1			
Date:	Fri, 22 Nov 2024	Pseudo R-squ.:	0.005288			
Time:	03:30:44	Log-Likelihood:	-207.71			
converged:	True	LL-Null:	-208.82			
Covariance Type:	nonrobust	LLR p-value:	0.1373			
=====						
	coef	std err	z	P> z	[0.025	0.975]

const	1.0016	0.571	1.753	0.080	-0.118	2.122
chol	-0.0033	0.002	-1.471	0.141	-0.008	0.001
=====						

Odds Ratios:

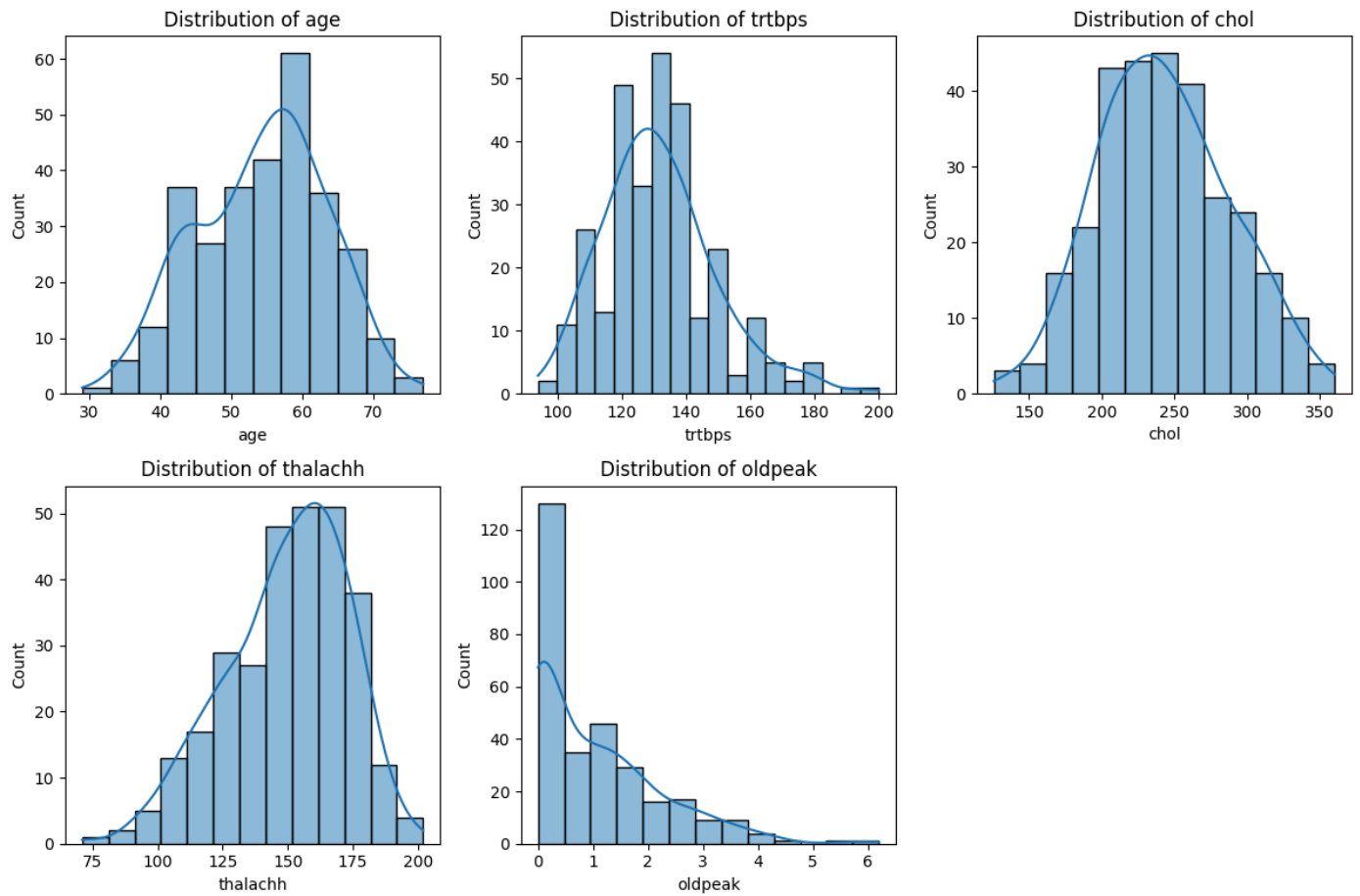
	Variable	Odds Ratio	p-value
const	const	2.72	0.079652
chol	chol	1.00	0.141226

Distribution of Cholesterol Levels by Heart Disease Presence



```
# List of continuous variables in heart indicator data
heart_continuous_vars = ['age', 'trtbps', 'chol', 'thalachh', 'oldpeak']
```

```
# Plot histograms for continuous variables
plt.figure(figsize=(12, 8))
for i, col in enumerate(heart_continuous_vars, 1):
    plt.subplot(2, 3, i)
    sns.histplot(heart_data_cleaned[col], kde=True)
    plt.title(f'Distribution of {col}')
plt.tight_layout()
plt.show()
```



```
# Boxplot to visualize the distribution of cholesterol levels based on heart disease outcome
plt.figure(figsize=(8, 6))
sns.boxplot(data=heart_data_cleaned, x='output', y='chol')
plt.title('Cholesterol Levels by Heart Disease Outcome (0 = No, 1 = Yes)')
plt.show()
```