In [1]:
```
import sklearn
import pandas as pd
import numpy as np
import seaborn as sns
from matplotlib import pyplot as plt
from mpl_toolkits.mplot3d import Axes3D
import scipy
import statistics
from sklearn import model_selection
from sklearn.neighbors import KNeighborsClassifier
from sklearn.metrics import accuracy_score
from sklearn.preprocessing import label_binarize
import os
```

In [2]:
```
path = os.getcwd()
iris_df = pd.read_csv(path+'\\Learn Dataset\\iris_dataset_missing.csv')
heart_df = pd.read_csv(path+'\\Learn Dataset\\heart_disease_missing.csv')
heart_df_nona = heart_df.dropna()
features = ["exang","thal","slope","cp"]
heart_df_sub = heart_df_nona.copy()
for i in heart_df_nona.columns:
    if i not in features and i not in ["target"]:
        heart_df_sub.drop(columns = [i], inplace=True)
```

CM4

Grouping variables by type and plotting a histogram

Heart Disease dataset

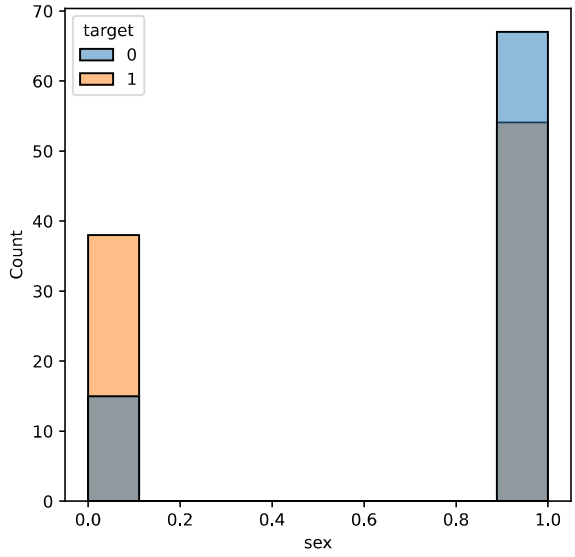
In [6]:
```
heart_df_bin = heart_df_nona.copy()
heart_df_cat = heart_df_nona.copy()
heart_df_ord = heart_df_nona.copy()
heart_df_num = heart_df_nona.copy()
cat = ["cp","restecg","slope","thal"]
ordinal = ["ca"]
num = ["age","oldpeak","trestbps","chol","thalach"]
binary = ["sex","fbs","exang"]

for i in heart_df_nona.columns:
    if i not in cat and i not in ["target"]:
        heart_df_cat.drop(columns=[i],inplace=True)
    if i not in ordinal and i not in ["target"]:
        heart_df_ord.drop(columns=[i],inplace=True)
    if i not in binary and i not in ["target"]:
        heart_df_bin.drop(columns=[i], inplace=True)
    if i not in num and i not in ["target"]:
        heart_df_num.drop(columns=[i], inplace=True)
```

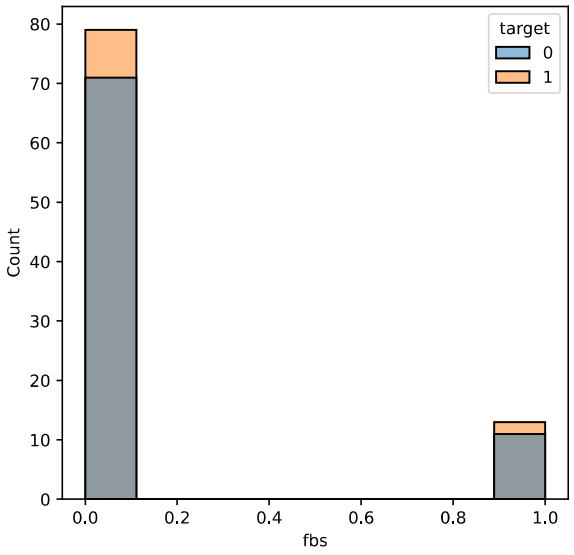
Binary Variables

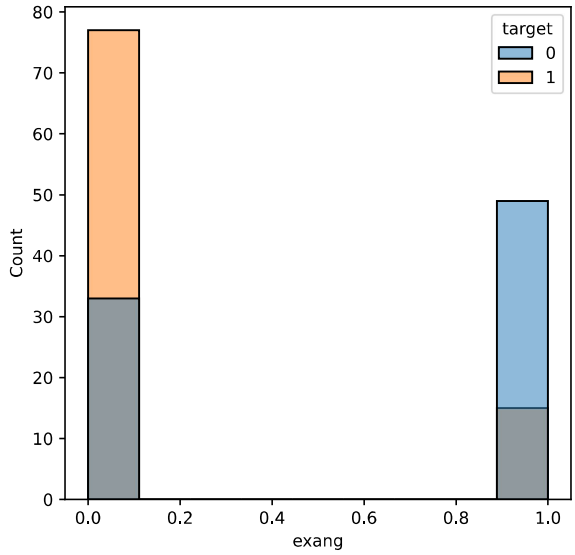
In [27]:
```
fig1 = plt.figure(figsize=(12,12))
plt.title("Binary Variables")
plt.subplot(2,2,1)
sns.histplot(data = heart_df_bin, x = "sex", hue="target")
plt.subplot(2,2,2)
sns.histplot(data = heart_df_bin, x = "fbs", hue="target")
```

```
plt.subplot(2,2,3)
sns.histplot(data = heart_df_bin, x = "exang", hue="target")
```

Out[27]: <matplotlib.axes._subplots.AxesSubplot at 0x27d1547d3d0>



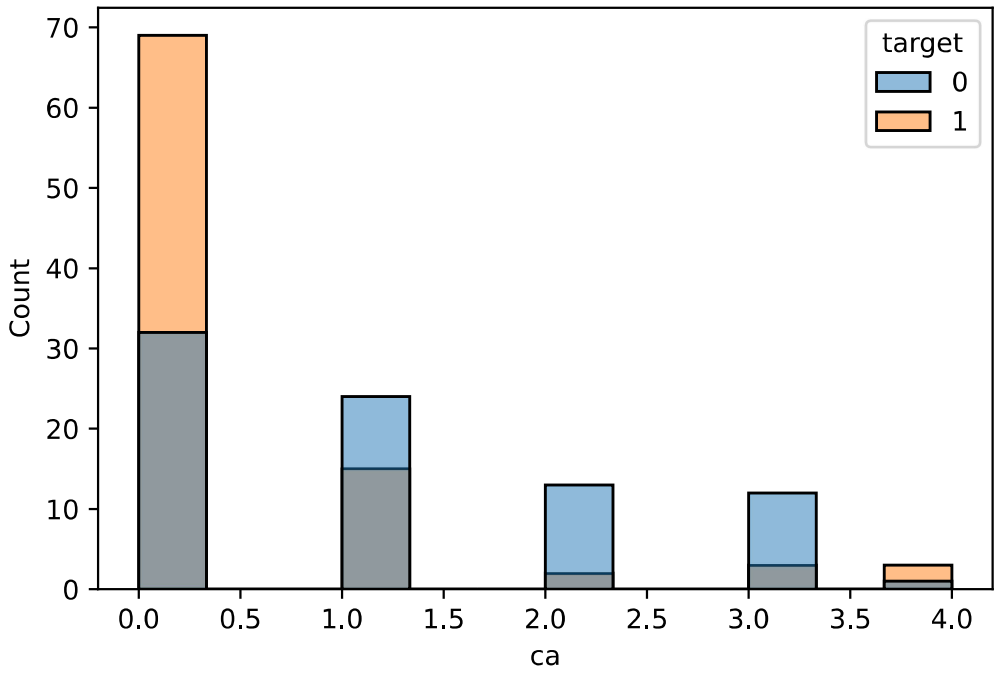




Ordinal

In [17]:
```
sns.histplot(data = heart_df_ord, x = "ca", hue="target")
```

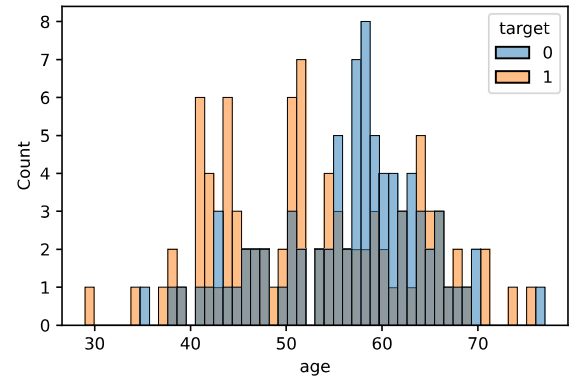
Out[17]: <matplotlib.axes._subplots.AxesSubplot at 0x27d12486a30>

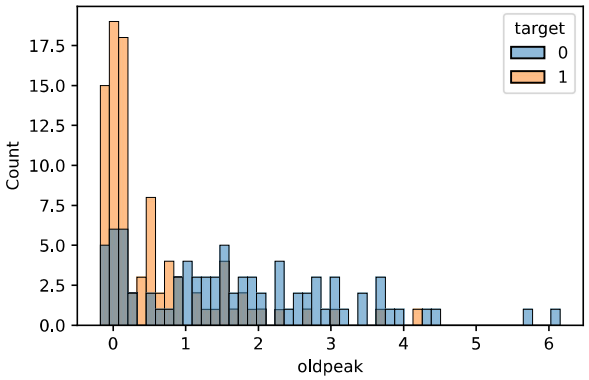


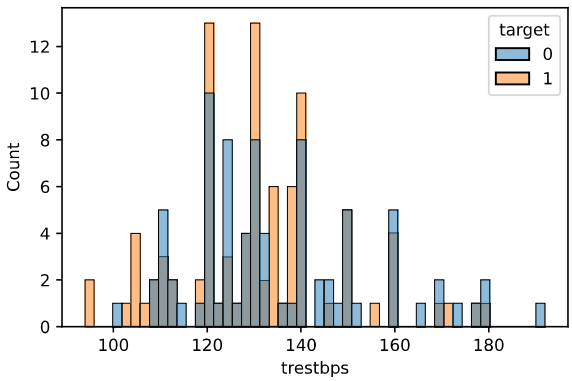
Numeric Type

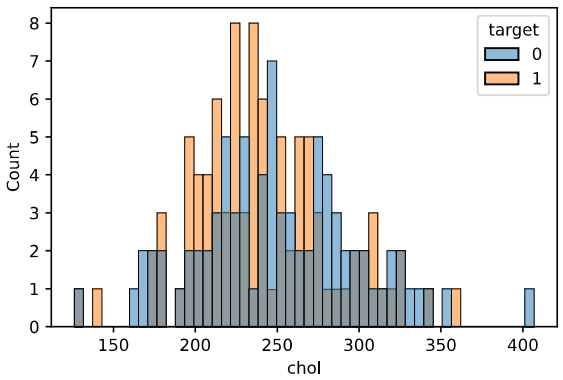
In [32]:
```
fig1 = plt.figure(figsize=(12,12))
plt.title("Numeric Variables")
plt.subplot(3,2,1)
sns.histplot(data = heart_df_num, x = "age", hue="target", bins = 50)
plt.subplot(3,2,2)
sns.histplot(data = heart_df_num, x = "oldpeak", hue="target",bins = 50)
plt.subplot(3,2,3)
sns.histplot(data = heart_df_num, x = "trestbps", hue="target",bins = 50)
plt.subplot(3,2,4)
sns.histplot(data = heart_df_num, x = "chol", hue="target",bins = 50)
plt.subplot(3,2,5)
sns.histplot(data = heart_df_num, x = "thalach", hue="target",bins = 50)
```

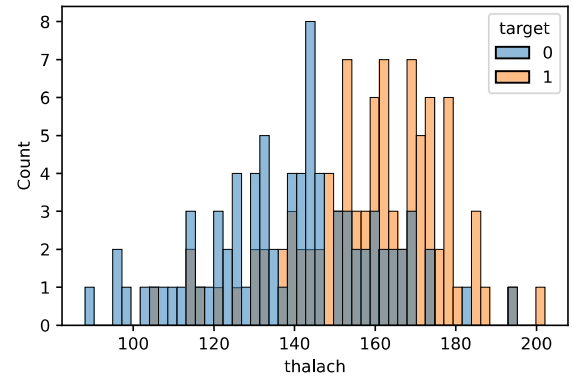
Out[32]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x27d170aed60>
```







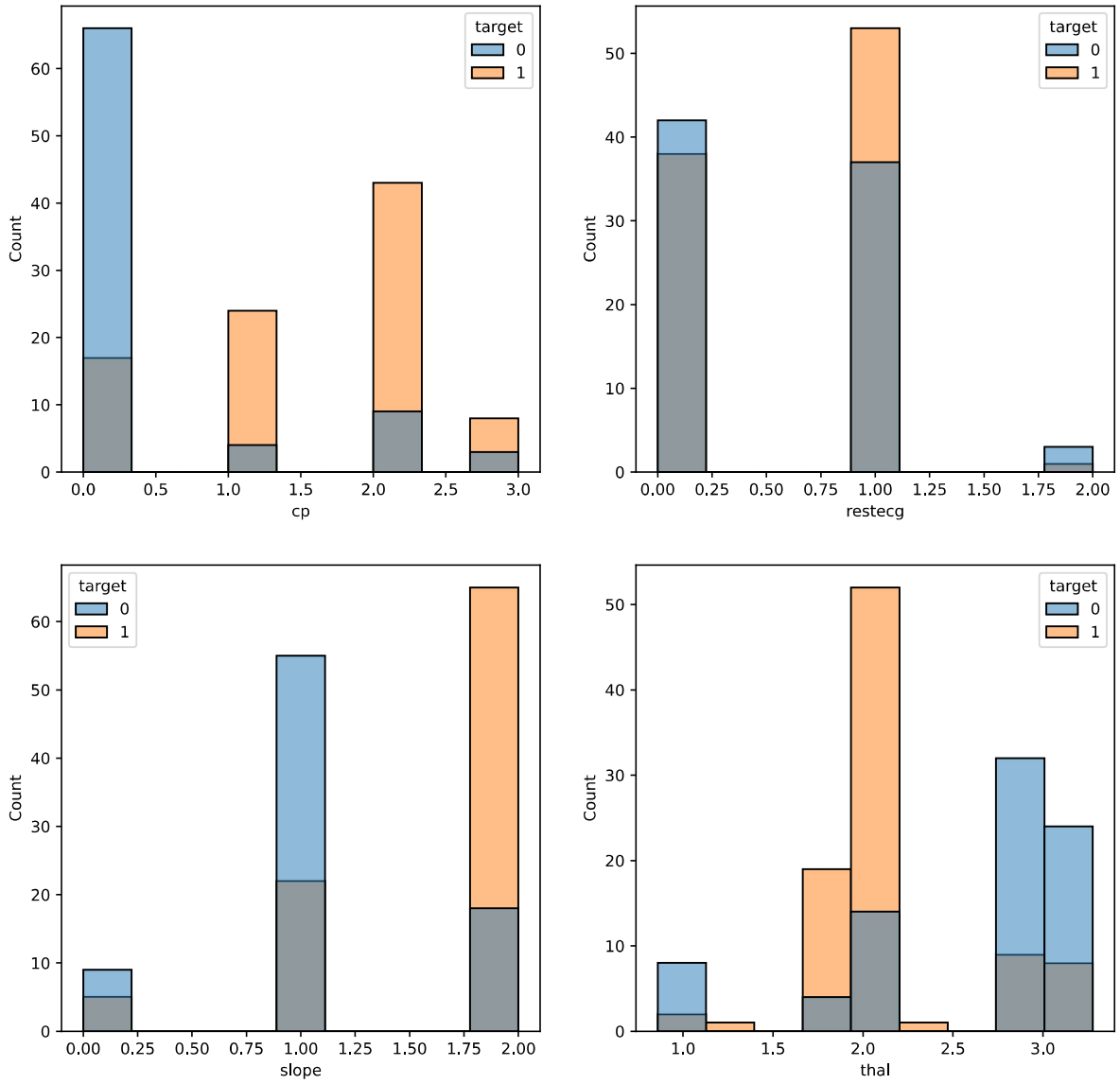




Categorical Type

In [33]:
```
fig1 = plt.figure(figsize=(12,12))
plt.title("categorical Variables")
plt.subplot(2,2,1)
sns.histplot(data = heart_df_cat, x = "cp", hue="target")
plt.subplot(2,2,2)
sns.histplot(data = heart_df_cat, x = "restecg", hue="target")
plt.subplot(2,2,3)
sns.histplot(data = heart_df_cat, x = "slope", hue="target")
plt.subplot(2,2,4)
sns.histplot(data = heart_df_cat, x = "thal", hue="target")
```

Out[33]: <matplotlib.axes._subplots.AxesSubplot at 0x27d1757a730>



We can see that the variables that we chose, namely {"exang","thal","slope","cp", "oldpeak"}, show less overlap of both target cases. If there is too much overlap, it is quite unwise to choose that feature.

In [ ]: