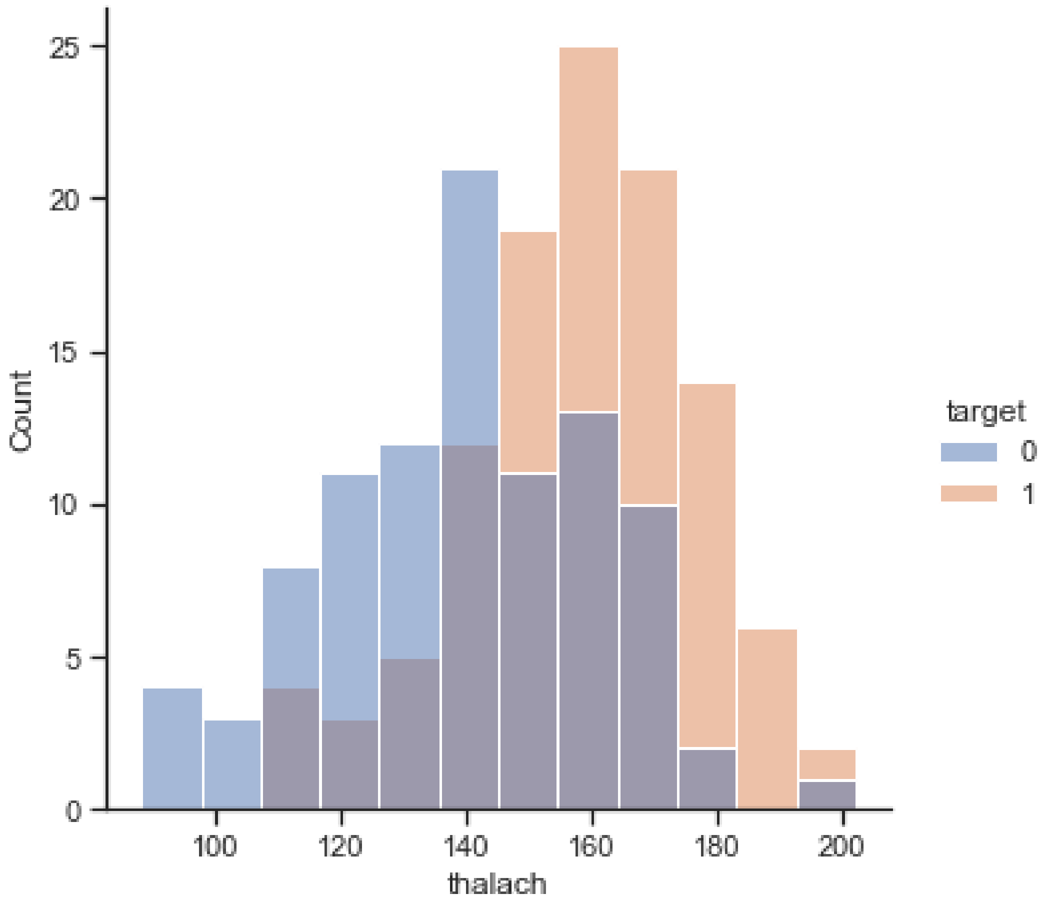

From the above histograms, we can see the number of present (1) and absent (0) heart disease cases in each features.

[CM5]

Data Cleaning

Checking for null / NaN values (missing data)

[16]:
```
# checking for any null / NaN values
df_heart.isnull().values.any()
```

[16]:
```
True
```

[17]:
```
# checking for any null / NaN values
df_heart.isna().sum()
```

[17]:
```
age          0
sex          0
cp           0
trestbps     7
chol        10
fbs          0
```

```
restecg      5
thalach      5
exang        0
oldpeak     19
slope        2
ca           0
thal         1
target       0
dtype: int64
```

We see NaN values in few features. These can be replaced with feature mean.

Checking for noise

We also observe that the column ‘thal’ which is a categorical variable, has float values. This can be categorised as ‘noise’. Rounding the values to get integer values.

[18]:

```
# rounding 'thal' values as we see noise in the column. 'thal' is expected to be␣
 ↪categorial
df_heart['thal'] = df_heart['thal'].round()
```

[19]:

```
# replacing NaN values with feature mean for nums and with median for other categories
for column in df_heart.columns[0:-1]:
    if column in nums:
        df_heart[column].fillna(value=df_heart[column].mean(), inplace=True)
    else:
        df_heart[column].fillna(value=df_heart[column].median(), inplace=True)
```

[20]:

```
# check if there are any null / NaN values
df_heart.isnull().values.any()
```

[20]:

```
False
```

[21]:

```
df_heart.isna().sum()
```

[21]:

```
age         0
sex         0
cp          0
trestbps    0
chol        0
fbs         0
restecg     0
thalach     0
exang       0
oldpeak     0
slope       0
ca          0
thal        0
target      0
dtype: int64
```

[22]:

```
df_heart.describe()
```

```
[22]:          age         sex          cp    trestbps        chol         fbs  \
count  212.000000  212.000000  212.000000  212.000000  212.000000  212.000000
mean    54.311321    0.688679    0.957547  131.784610  244.133256    0.132075
std      9.145339    0.464130    1.022537   17.755169   45.330324    0.339374
min     29.000000    0.000000    0.000000   93.944184  126.085811    0.000000
25%     47.000000    0.000000    0.000000  119.987220  212.793680    0.000000
50%     55.000000    1.000000    1.000000  130.021392  243.475116    0.000000
75%     61.000000    1.000000    2.000000  139.959811  269.275502    0.000000
max     77.000000    1.000000    3.000000  192.020200  406.932689    1.000000

          restecg     thalach       exang     oldpeak       slope          ca  \
count  212.000000  212.000000  212.000000  212.000000  212.000000  212.000000
mean     0.570755  149.863490    0.344340    1.010168    1.419811    0.731132
std      0.532982   21.648149    0.476277    1.071093    0.622016    1.038762
min      0.000000   88.032613    0.000000   -0.185668    0.000000    0.000000
25%      0.000000  137.712696    0.000000    0.083715    1.000000    0.000000
50%      1.000000  150.955534    0.000000    0.889500    1.000000    0.000000
75%      1.000000  164.991594    1.000000    1.569735    2.000000    1.000000
max      2.000000  202.138041    1.000000    4.404773    2.000000    4.000000

             thal      target
count  212.000000  212.000000
mean     2.353774    0.542453
std      0.586042    0.499374
min      1.000000    0.000000
25%      2.000000    0.000000
50%      2.000000    1.000000
75%      3.000000    1.000000
max      3.000000    1.000000
```

Data Cleaning :

- the NaN values (missing values) were replaced with feature mean for numeric and median for other type of features.
- the noise in 'thal' (non categorical values) were handled by rounding to integer.

If we attempt to drop the missing values, the performance of the classifier was observed to be low. Moreover, dropping the values reduces the size of the dataset affecting performance.

Question 2: KNN Classification

[CM6]

Basic Model