- thalach having a mild separation relation between disease and non-disease.
- similarly we see that cp , slope and exang have observable seperation between disease and non-disease.
- Other features don't form any clear separation and are mostly overlapping between disease and non-disease.

We are selecting a subset of the feature set as using all features run the risk of overfitting the training and validation sets. Using fewer features can speed up inference at the cost of predictive performance.

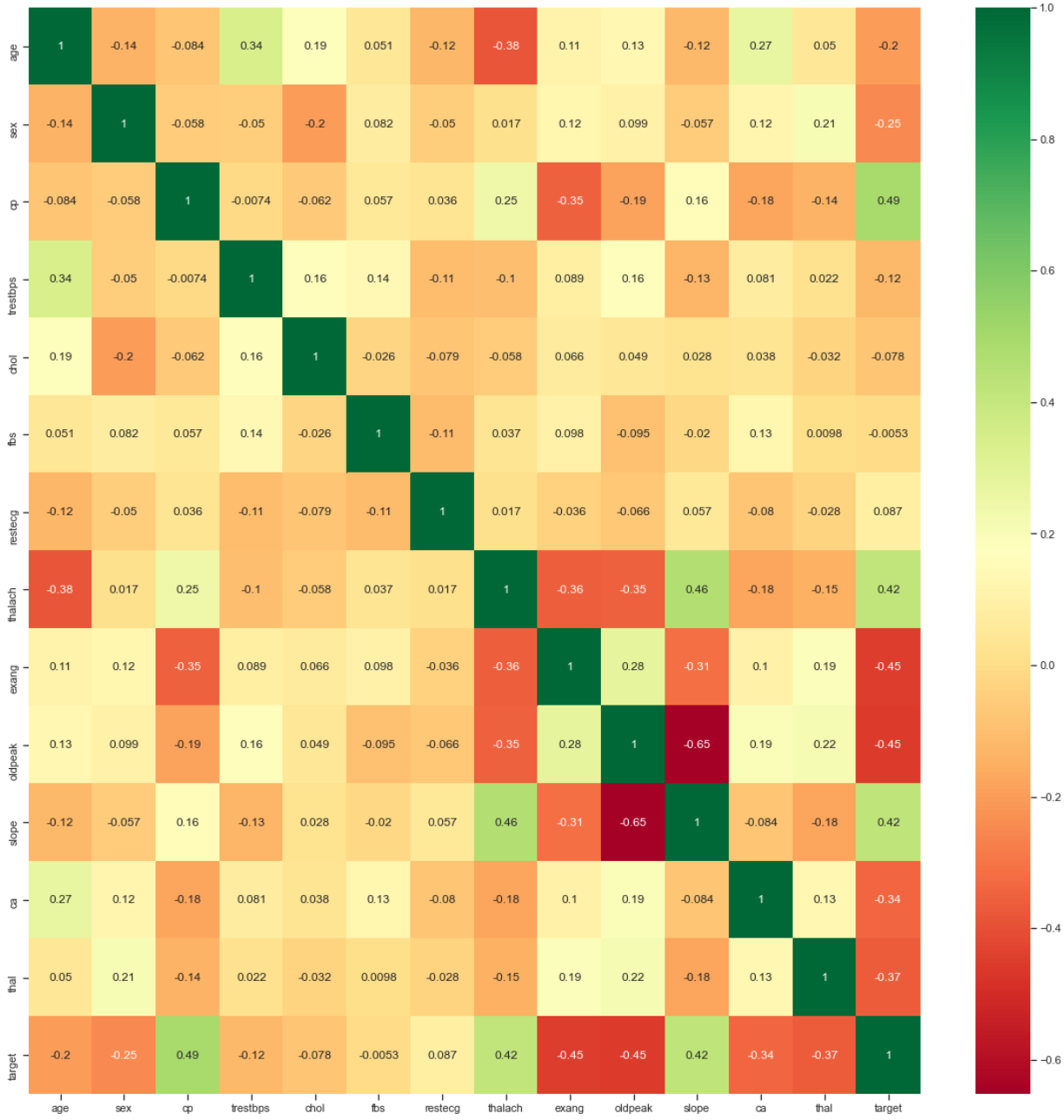
The features with less overlapping will lead to better model training and performance. Hence, choosing the features: 'cp', 'oldpeak', 'exang', 'slope', 'thalach'.

[CM2]

Correlation coefficient of each pair of features

Heat map is used to find out the correlation between different features in the dataset. High positive or negative value shows that the features have high correlation

[8]:
```
# plotting correlation coeeficients using heat map
#get correlations of each features in dataset
corrmat = df_heart.corr()
top_corr_features = corrmat.index
plt.figure(figsize=(20,20))
#plot heat map
heart_heat_map=sns.heatmap(df_heart[top_corr_features].corr(),annot=True,cmap="RdYlGn")
```



From the "Heat Map" visualization. We observe that:

- 'cp', 'thalach', 'slope' shows good positive correlation with target
- 'oldpeak', 'exang', 'ca', 'thal', 'sex', 'age' shows a good negative correlation with target
- 'fbs' 'chol', 'trestbps', 'restecg' has low correlation with the target

We observe that the chosen features ('cp', 'oldpeak', 'exang', 'slope', 'thalach') have good positive and negative correlation with the target.

Calculation of mean, variance, skew, kurtosis for the datasets

[9]:
```
# calculate mean
df_heart[choosen_features_nums].mean()
```

[9]:
```
oldpeak      1.113106
thalach    149.647978
dtype: float64
```

[10]:
```
# calculate variance
df_heart[choosen_features_nums].var()
```

[10]:
```
oldpeak      1.577304
thalach    487.358850
dtype: float64
```

[11]:
```
# calculate skew
df_heart[choosen_features_nums].skew()
```

[11]:
```
oldpeak    1.224053
thalach   -0.394100
dtype: float64
```

[12]:
```
# calculate kurtosis
df_heart[choosen_features_nums].kurtosis()
```

[12]:
```
oldpeak    1.363172
thalach   -0.214108
dtype: float64
```

From mean, var, skew, kurtosis, we observe that:

- oldpeak is left-skewed, whereas thalach is right skewed.
- the kurtosis values for thalach is negative indicating light tail distribution. Whereas, oldpeak has positive kurtosis value indicating heavy tail distribution.
- variance for oldpeak and thalach are high indicating that the values are highly spread out from the mean.

[CM3]

Checking for notable outliers using "Box Plots"

[13]:
```
# boxplot for outlier detection of numerical features
for column in df_heart[choosen_features_nums]:
    plt.figure()
    ax = sns.boxplot(x='target', y=column, data=df_heart)
    plt.show()
```