ASSIGNMENT 2

1. a) It is important for the trees to be correlated up to a certain extent while maintaining diversity because of the following reasons-
   Bias-Variance: Highly correlated trees tend to have low variance but high bias, while highly diverse trees have low bias but high variance. By striking the right balance, you can achieve a lower prediction error by combining both low-bias and low-variance models.
   Overfitting: Introducing diversity through bootstrapping and feature randomization helps reduce overfitting, while having some level of correlation among the trees helps in making accurate predictions based on the learning from the training data.

   b) The curse of dimensionality can become a problem in Naive Bayes when we have a large number of features(dimensions) in comparison to the amount of data available.
   Some strategies that can be employed to mitigate this problem in practice are as
   follows
   1. Feature Selection
   2. Dimensionality reduction using PCA, LDA T-Sne
   3. Using L1 regularization

   c) Yes it will affect the inference results. It can face the following problems
   1. If the new value is not present in the training dataset, its count will be zero. As a result, the conditional probability for that value will be zero, which can lead to issues in the classification process as that class will **never be classified**.
   2. The absence of the new value in the training data means that the model does not have any information about the relationship between that value and the class labels, hence resulting in **inefficient training**.

   To mitigate the problem of encountering new values in Naive Bayes, we can use the following approaches:
   1. Laplace Smoothing
   2. Discounted Smoothing
   3. Bayesian Estimation

   Now, let's illustrate this issue with an example:
   Let us consider a simple dataset with one feature(ice-cream) and one target label(taste) with values 0 and 1 now consider ice-cream cassata which is not present in the training data the conditional probability for both classes (0 and 1) will be 0, hence it won't be possible to classify the ice-cream cassata.

d) Yes, Information Gain can be biased if some attributes have more cardinality (distinct values) than others. This is because Information Gain measures the reduction in uncertainty or entropy, and attributes with higher cardinality tend to have a higher chance of reducing uncertainty by having more values. This can lead to biases towards attributes with more distinct values during the tree-building process.
To mitigate this problem we can use criterias that are less sensitive to cardinality of attributes such as

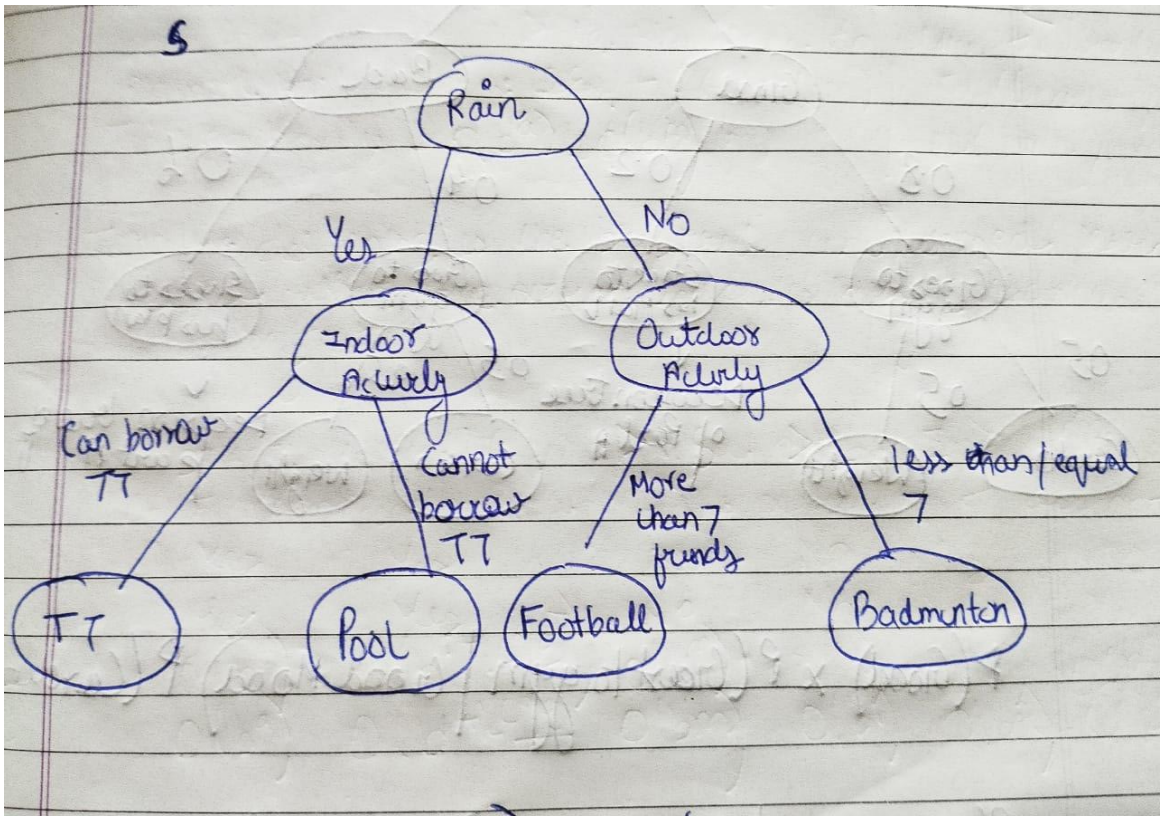
1. Gini-impurity
2. Chi-squared Statistic

To explain the issues with information gain let us consider an example of a dataset with two attributes “Age group” and “Pin Code” to predict whether someone will like a movie or not.

Age group: It has relatively low cardinality(4).
Pin Code: It has high cardinality. There are thousands of unique pin codes in a large geographic area.

Using Information Gain as the attribute selection criterion might lead the decision tree algorithm to prefer the “Pin Code” attribute due to its higher cardinality. However, the geographic location represented by the pin code may not be very informative in predicting whether someone will like a movie or not in contrast “Age group” could be a more relevant attribute for this prediction task, as people within the same age group may have similar movie preferences..

2.

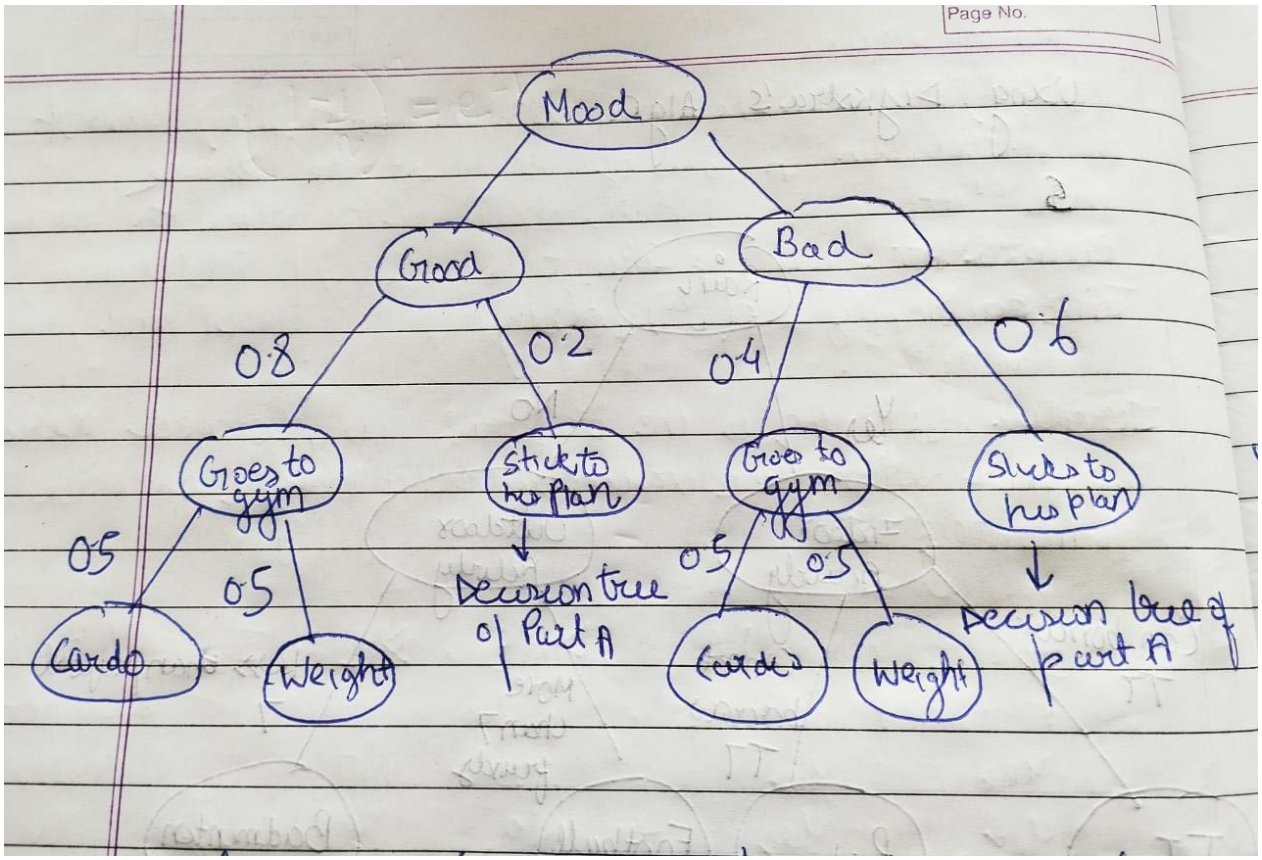


Outcome: Football, Conditional
Probability: P(Football | No Rain, >7 Friends) = P(No Rain)*P (>7 Friends | No Rain)
Outcome: Badminton
Probability: (P(Badminton | No Rain, <=7 Friends) = P(No Rain)*P (<=7 Friends | No Rain)
Outcome: Table Tennis
Probability: P(Table Tennis | Rain, Can BorrowRackets) = P(Rain)*P(Can BorrowRackets | Rain)
Outcome: Pool
Probability: P(Pool | Rain, Can't Borrow Rackets) = P(Rain)*P(Can't BorrowRackets | Rain)

b) P(Predicted Rain) =0.3
P(Predicted Clear) = 0.7
P(Predicted Rain | Rain) = 0.80
P(Predicted Clear | Clear) = 0.90
P(Predicted Clear| Rain) = 0.20
P(Predicted Rain| Clear) = 0.10

P(Predicted Rain) = P(Rain)*P(Predicted Rain | Rain) + P(Clear)*P(Predicted Rain| Clear)
0.3 = P(Rain)*0.8 + (1-P(Rain))*0.1
0.2 = 0.7*P(Rain)
2/7 = P(Rain)
P(Rain| Predicted Rain) = P(Predicted Rain | Rain)*P(Rain)/P(Predicted Rain)
= 0.8*P(Rain)/0.3
= 16/21
= 0.76

c)



Outcome : Cardio
Probability: P(Good Mood) * P(Go to Gym | Good Mood) * P(Cardio | Good Mood, Go to Gym) + P(Bad Mood) * P(Go to Gym | Bad Mood) * P(Cardio | Bad Mood, Go to Gym)
Outcome : Weight Training
Probability: P(Good Mood) * P(Go to Gym | Good Mood) * P(Weight Training | Good Mood, Go to Gym) + P(Bad Mood) * P(Go to Gym | Bad Mood) * P(Weight Training | Bad Mood, Go to Gym)

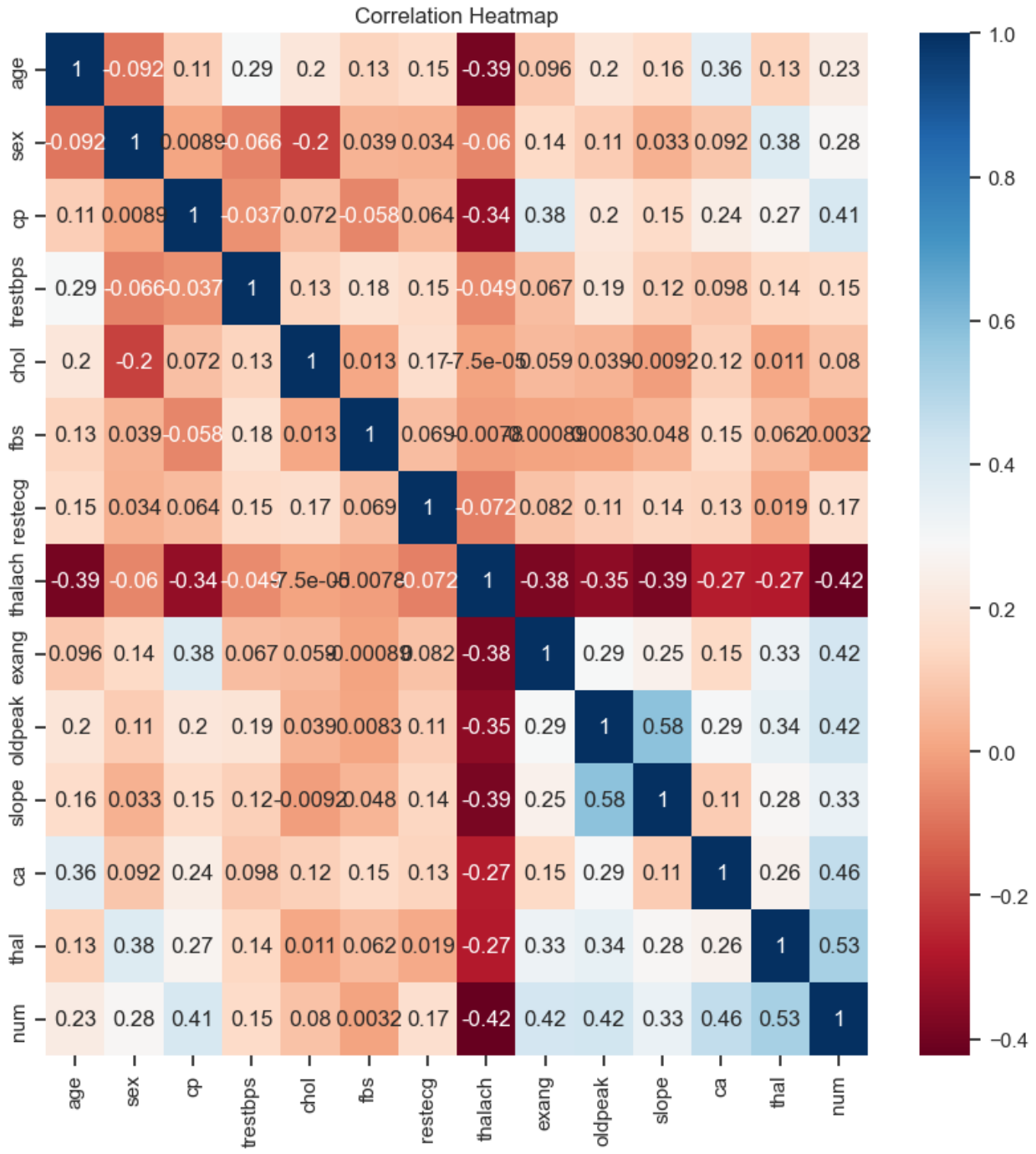
Outcome : Stick to Plan
Probabilities of Outcomes of previous decision tree will be multiplied by (P(Bad Mood) * P(Stick to Plan | Bad Mood) + P(Good Mood) * P(Stick to Plan | Good Mood)) as this represents P(Stick to Plan)

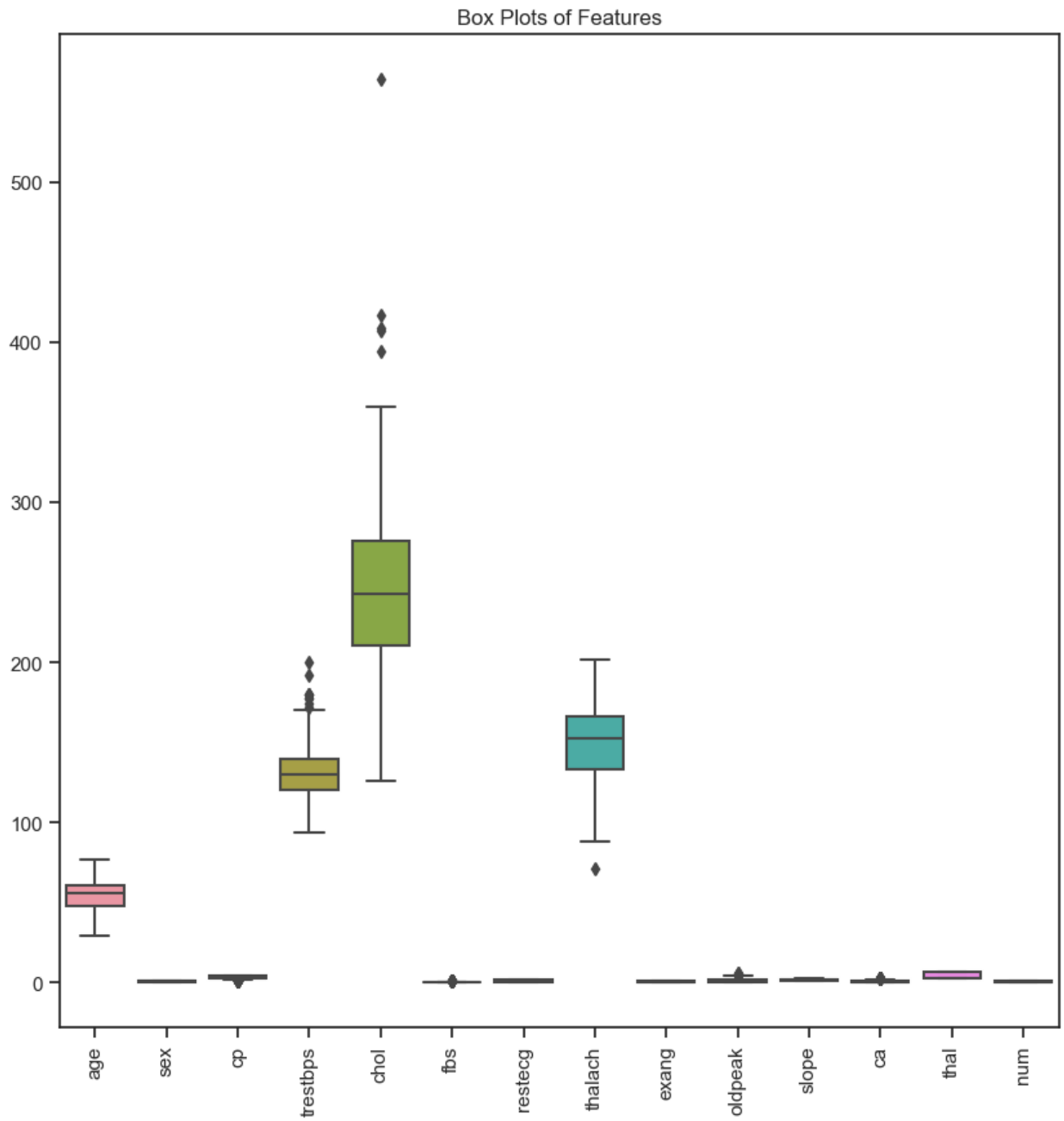
d) Rahul had 7 hours of sleep
P(Good Mood) = 0.6, P(Bad Mood) = 0.4
P(F = 7| Good Mood) = 0.7, P(F = 7| Bad Mood) = 0.45
P(F=7) = P(Good Mood)*P(F = 7| Good Mood) + P(Bad Mood)*P(F = 7| Bad Mood)
= 0.6*0.7 + 0.4*0.45 = 0.72
However his mood is also dependent on number of hours he slept
P(Good Mood | F = 7) = P(F = 7| Good Mood)*P(Good Mood)/P(F = 7) = 0.58
P(Bad Mood | F = 7) = P(F = 7| Bad Mood)*P(Bad Mood)/P(F = 7) = 0.25

In the expressions calculated in previous parts P(Good Mood) and P(Bad Mood) will be replaced by P(Good Mood | F = 7) and P(Bad Mood | F = 7) respectively
P(Cardio) = 0.58*0.8*0.5 + 0.25*0.4*0.5 = 0.282
P(Weight Training) = 0.58*0.8*0.5 + 0.25*0.4*0.5 = 0.282
P(Sticks to Plan) = 0.58*0.6+ 0.25*0.2 = 0.266

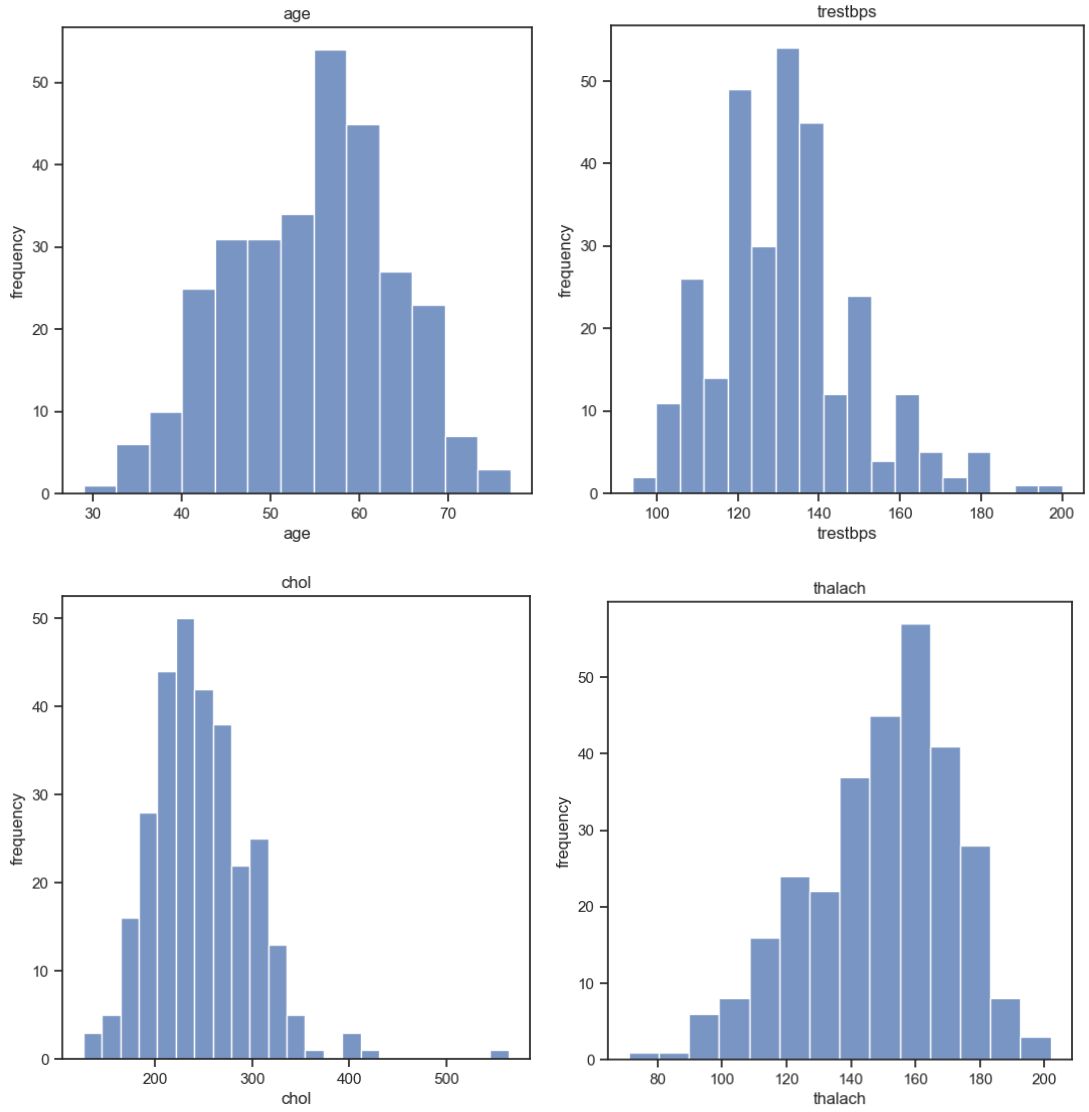
So he will either do cardio or weight training

3. **Exploratory data analysis**


Correlation Heatmap
age
sex
cp
trestbps
chol
fbs
restecg
thalach
exang
oldpeak
slope
ca
thal
num
1.0
0.8
0.6
0.4
0.2
0.0
−0.2
−0.4


Box Plots of Features
500
400
300
200
100
0
age
sex
cp
trestbps
chol
fbs
restecg
thalach
exang
oldpeak
slope
ca
thal
num

Histograms



c)Best criteria is entropy with training accuracy 1.0
Testing accuracy for the same is : 0.7333333333333333

d) Best parameters are {'max_features': 'log2', 'min_samples_split': 15}
Training score is : 0.9029535864978903
Testing score is : 0.75

e)Best hyperparameters for Random Forest: {'max_depth': 20, 'min_samples_split': 30, 'n_estimators': 250} with cross-validated accuracy 0.8524822695035462

Classification Report on Test Data:

	precision	recall	f1-score	support
0	0.76	0.93	0.84	30
1	0.91	0.70	0.79	30
accuracy			0.82	60
macro avg	0.83	0.82	0.81	60
weighted avg	0.83	0.82	0.81	60