Section A)

Q1)

(a)

A1)(a) More diversity means the ensemble trees would give different results & the a strong classifier could be made by combining all of them. otherwise we ~~wouldn't~~ would have ~~the~~ a single tree instead of multiple trees that produce the same results. However more diversity means lesser correlation b/w the trees which would make the model suffer from high variance. This is the tradeoff b/w correlation & diversity in RFs. The trees need to be correlated upto a certain extent which is achieved by selecting a subset of features for each model to train on & by creating bootstrap samples.

(b)

(b) When the no of features is relatively large compared to the no of data points the curse of dimensionality becomes a problem for Naive Bayes. We can reduce their dimensionality by doing feature selection or feature extraction through PCA or SVD. We can use TSNE for the same. Also, we can use K-fold cross-validation ~~to tune the hyperparameters~~ ~~to~~ to reduce overfitting of the sparse data present in higher dimensions. or we can use ensemble methods too.

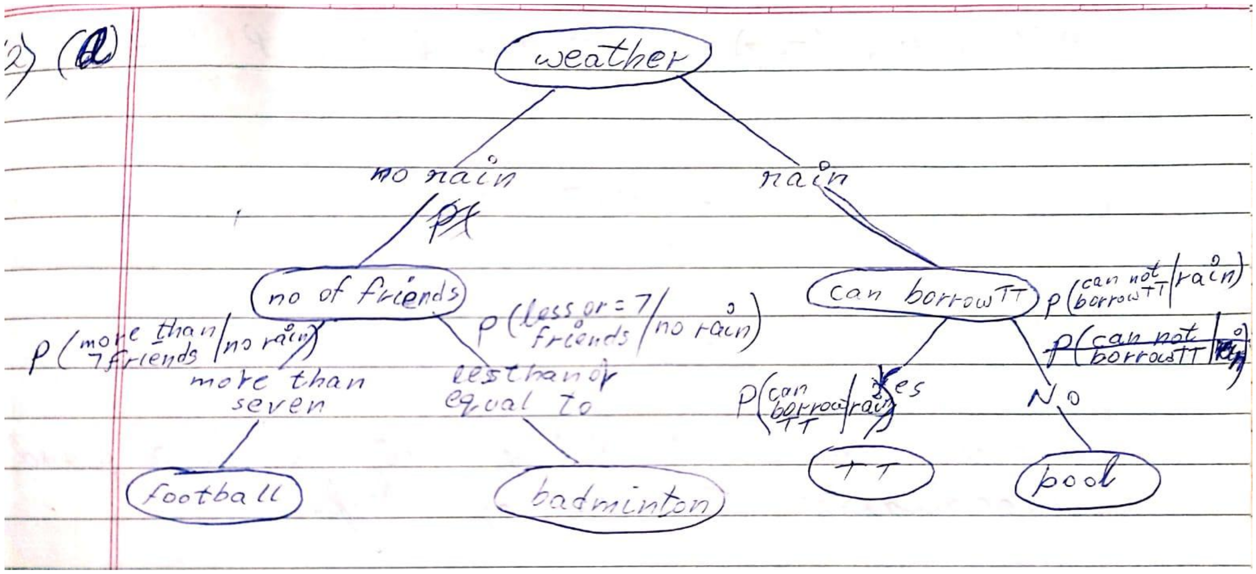
(c)

(c) If some attributes are missing then it maybe that $P(Y=y|X=x_i)$ may be zero for x_i attribute. This will lead to its whole probability being zero (which shouldn't be the case). To mitigate this we can use Laplace Smoothing (adding a constant value to the ~~numerator & deno~~ count of each attribute in training data). Example - Let's suppose we have the data of a football team that hasn't lost a match for the past 5 years. So, it doesn't mean that for the next year its probability of losing would be zero. However without Laplace smoothing the NB model would predict a probability zero.

(d)

(d) Yes. If an attribute has higher cardinality then the information gain associated with that attribute would be higher as $P(X=x)$ would be higher. An alternative way could be to use the Gini Index (~~or Gain~~) (or Gain Ratio). ~~Gain~~ Example - Predicting whether a customer will ~~predict~~ tip the waiter based on 'meal type' & 'customer satisfaction' with 'meal type' being (Breakfast, Lunch, Dinner) & the other being ratings from 1-5. If we use IG then there may be bias towards 'customer satisfaction' since the cardinality is higher over here. However if we use Gain Ratio then it'll penalize the 'customer satisfaction' attribute thus providing a more balanced assessment.

A2)
(a) TT here refers to Table Tennis.
Assumption -
P(no rain) = 0.5 and P(rain) = 0.5



(b) Assumption: Rainy and Clear Weather are mutually exclusive. Thus ‘not rainy’ can be modeled as ‘clear’.
So, P(~Prediction of rainy) = P(Prediction of Clear) and P(~Rainy) = P(Clear)

(b) Let P = prediction of the app that its rainy
R = It is Rainy

Thus, $P(P|R) = 0.8$ — ①
$P(\sim P|\sim R) = 0.9$ — ②
$P(P) = 0.3$
$P(\sim P) = 0.7$
given

Hence, $P(R|P) = P(P|R) \times \frac{P(R)}{P(P)}$

~~$P(R) = P(R|R) \times P($~~

~~$P(R) = P(R, P) + P(R, \sim P)$~~
~~$= P(R|P) \times P(P) + P(R|\sim P) P(\sim P)$~~

Using the first two eq's -

$P(P|R) = 0.8$
$\Rightarrow P(P, R)/P(R) = 0.8$
$\Rightarrow P(P, R) = 0.8\, P(R)$ — ③

And, $P(\sim P|\sim R) = 0.9$
$\therefore P(P|\sim R) = 1 - 0.9 = 0.1$
$\Rightarrow P(P, \sim R)/P(\sim R) = 0.1$
$\Rightarrow P(P, \sim R) = 0.1\, P(\sim R) = 0.1 - 0.1 P(R)$ — ④

Adding ③ & ④,
$P(P, R) + P(P, \sim R) = 0.7 P(R) + 0.1$
We know that $P(P) = P(P, R) + P(P, \sim R)$
Thus, $P(P) = 0.7 P(R) + 0.1$
$\Rightarrow P(R) = \frac{2}{7}$

Thus, $P(R|P) = P(P|R)\ \dfrac{P(R)}{P(P)}$

$\Rightarrow P(R|P) = 0.8 \times \dfrac{2}{7 \times 0.3}$

$\Rightarrow \boxed{P(R|P) = \dfrac{16}{21}}$

Alternate Solution considering that the probability of rainy weather and clear weather were given instead of probability of prediction of rainy and clear weather:

(b) Let P_r = Prediction of Rain through app
& R = Rainy Weather

Thus, $P(P_r|R) = 0.8$
$P(\sim P_r|\sim R) = 0.9$
$P(R) = 0.3$
$P(\sim R) = 0.7$

Hence, a/q –

$$P(R|P_r) = P(P_r|R) \times \frac{P(R)}{P(P_r)}$$

$$\begin{aligned} P(P_r) &= P(P_r, R) + P(P_r, \sim R) \\ &= P(P_r|R)P(R) + P(P_r|\sim R)\,P(\sim R) \\ &= 0.8 \times 0.3 + (1 - 0.9)\,0.7 \\ &= 0.24 + 0.07 \\ &= 0.31 \end{aligned}$$

$$P(R|P_r) = 0.8 \times \frac{0.3}{0.31}$$

$$\boxed{P(R|P_r) = \frac{24}{31} = 0.774}$$

(c)
Assumption: P(Good Mood) and P(Bad Mood) are same
Therefore, P(Good Mood) = 0.5 and P(Bad Mood) = 0.5

(c) $P(\text{Gym} \mid \text{Good Mood}) = 0.8$ — (1)

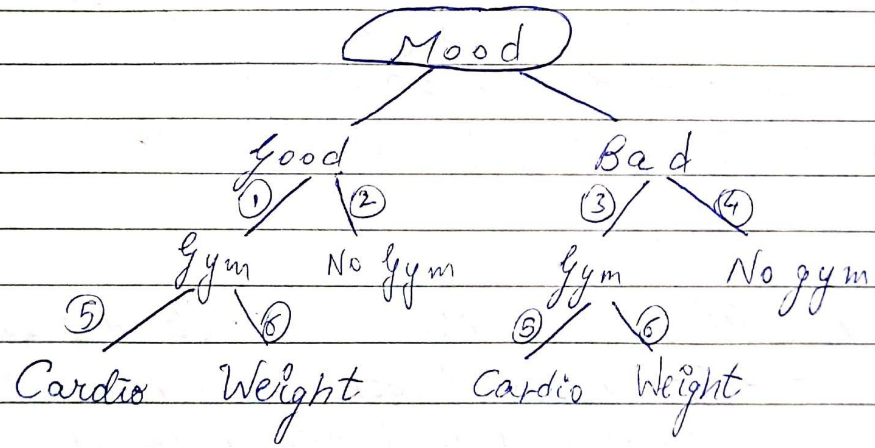
$P(\text{No gym} \mid \text{Good Mood}) = 0.2$ — (2)

$P(\text{Gym} \mid \text{Bad Mood}) = 0.4$ — (3)

$P(\text{No gym} \mid \text{Bad Mood}) = 0.6$ — (4)

$P(\text{Cardio} \mid \text{Gym}) = 0.5$ — (5)

$P(\text{Weight} \mid \text{Gym}) = 0.5$ — (6)



$$P(\text{Cardio} \mid \text{Good Mood}) = P(\text{Cardio} \mid \text{Gym}) \times P(\text{Gym} \mid \text{Good Mood})$$

$$= 0.5 \times 0.8 = 0.4$$

$$P(\text{Cardio} \mid \text{Bad Mood}) = P(\text{Cardio} \mid \text{Gym}) \times P(\text{Gym} \mid \text{Bad Mood})$$

$$= 0.5 \times 0.4 = 0.2$$

$$P(\text{Cardio}) = P(\text{Cardio} \mid \text{Good Mood})\, P(\text{Good Mood}) + P(\text{Cardio} \mid \text{Bad Mood})\, P(\text{Bad Mood})$$

$$= 0.4 \times 0.5 + 0.2 \times 0.5 = 0.3$$

$$P(\text{Weights} \mid \text{Good Mood}) = P(\text{Weights} \mid \text{Gym}) \times P(\text{Gym} \mid \text{Good})$$

$$= 0.5 \times 0.8 = 0.4$$

$$P(\text{Weights} \mid \text{Bad Mood}) = P(\text{Weights} \mid \text{Gym}) \times P(\text{Gym} \mid \text{Bad Mood})$$

$$= 0.5 \times 0.4 = 0.2$$

$$P(\text{Weights}) = P(\text{Weights} \mid \text{Good Mood})\, P(\text{Good Mood}) + P(\text{Weights} \mid \text{Bad Mood})\, P(\text{Bad Mood})$$

$$= 0.4 \times 0.5 + 0.2 \times 0.5 = 0.3$$

$$P(\text{Gym}) = P(\text{Gym} \mid \text{Good Mood})\, P(\text{Good Mood}) + P(\text{Gym} \mid \text{Bad Mood})\, P(\text{Bad Mood})$$

$$= 0.8 \times 0.5 + 0.4 \times 0.5 = 0.6$$

$$P(\text{No Gym}) = 1 - 0.6 = 0.4$$

(d)

(d) $P(\text{Good mood}) = 0.6$
$P(\text{not Good mood}) = 0.4$
$P(F = 7 \text{ hours} | \text{Good mood}) = 0.7$
$P(F = 7 \text{ hours} | \text{Bad mood}) = 0.45$
Given

$P(\text{Bad mood} | F = 7 \text{ hours}) = \frac{P(F = 7 \text{ hours} | \text{Bad mood}) \times P(\text{Bad mood})}{P(F = 7)}$

$$
\begin{aligned}
P(F=7) &= P(F=7, \text{Good Mood}) + P(F=7, \text{Bad Mood}) \\
&= P(F=7|\text{Good Mood})\,P(\text{Good Mood}) + P(F=7|\text{Bad Mood})\,P(\text{Bad Mood}) \\
&= 0.7 \times 0.6 + 0.45 \times 0.4 \\
&= 0.42 + 0.18 \\
&= 0.6
\end{aligned}
$$

Hence, $P(\text{Bad Mood} | F=7) = 0.45 \times \frac{0.4}{0.6} = 0.3$

$$
\begin{aligned}
P(\text{Good Mood}/F=7) &= P(F=7|\text{Good Mood}) \frac{P(\text{Good Mood})}{P(F=7)} \\
&= 0.7 \times \frac{0.6}{0.6} \\
&= 0.7
\end{aligned}
$$

Hence, the most likely outcome is a good mood after 7 hours of sleep.

Elaborating further on his decisions-

$P(\text{cardio}|\cancel{F=7}) = P(\text{cardio}|\text{good mood})P(\text{good}|F=7)$
$+ P(\text{cardio}|\text{bad mood})P(\text{bad mood}|F=7)$
$= 0.5 \times 0.8 \times 0.7 + 0.5 \times 0.4 \times 0.3$

$P(\text{Cardio}|\cancel{F=7}) = 0.34$

$P(\text{Cardio}|\text{Good}) = 0.5 \times 0.8 \times 0.7 = 0.28$
$P(\text{Cardio}|\text{Bad}) = 0.5 \times 0.4 \times 0.3 = 0.06$

$P(\text{no gym}) = P(\text{no gym}|\text{good}) \times P(\text{good}) + P(\text{no gym}|\text{bad})P(\text{bad})$
$= 0.2 \times 0.7 + 0.6 \times 0.3$
$= 0.32$

$P(\text{weights}|\text{good mood}) = 0.5 \times 0.8 \times 0.7 = 0.28$
$P(\text{weights}|\text{bad mood}) = 0.06$
$P(\text{weights}) = 0.34$

Hence, Rahul will go to the gym for cardio or weights if he has 7 hours of sleep.

Section B

Q3)

Decision Trees:-

The best results obtained was with the Entropy criterion

```
Average accuracy of Decision Tree Classifier with Entropy as the splitting criterion:  0.7336666666666666
Average accuracy of Decision Tree Classifier with Gini as the splitting criterion:  0.7105000000000001
```

Through GridSearchCV -

```
Best accuracy:  0.8568840579710144
Best parameters:  {'max_depth': 3, 'max_features': 7, 'min_samples_split': 2}
```

Random Forests:-

```
Average accuracy of Random Forest Classifier with Entropy as the splitting criterion:  0.8003333333333333
Average accuracy of Random Forest Classifier with Gini as the splitting criterion:  0.7939999999999999
```

Through GridSearchCV -

```
Best parameters:  {'max_depth': 2, 'min_samples_split': 3, 'n_estimators': 250}
```

Random Forest Classifier with best parameters obtained using GridSearchCV -

```
Random Forest Classifier with best parameters-
Test Accuracy:  0.8333333333333334
Classification Report:
              precision    recall  f1-score   support

           0       0.84      0.84      0.84        32
           1       0.82      0.82      0.82        28

    accuracy                           0.83        60
   macro avg       0.83      0.83      0.83        60
weighted avg       0.83      0.83      0.83        60
```

Section C

Q4)
Preprocessing steps that were implemented for the given dataset -

1) Checking for missing values
2) Merging the classes of output label column to make it into a binary classification problem
3) Encoding the categorical features.

The MyDecisionTree was implemented and evaluated. The following accuracy was obtained on 70:30 train-test split -

```
Testing Accuracy of MyDecisionTree: 0.9785714285714285
```

On comparing this with the sklearn’s Decision Tree Classifer, the accuracy of it was -

```
Testing accuracy of sklearn DT: 0.9809523809523809
```

We can see that our implemented Decision Tree Class performs really good on the given dataset.