Name: Rutvik Kolhe

Student ID: 200258232 Unity ID: rkolhe

Question 1:

Step 1: We find the entropy of training set- say T, using the following formula:

$$H[T] = -P_+ \log_2 P_+ - P_- \log_2 P_-$$

Here positive class : Buy
negative class : Not Buy.

$\therefore P_+ = \frac{5}{9} \quad P_- = \frac{4}{9}$

$$H[T] = -\frac{5}{9} \log_2 \frac{5}{9} - \frac{4}{9} \log_2 \frac{4}{9}$$

$$H[T] = 0.991.$$

Step 2: Now, for each attribute that- divides T into subsets Ti, we calculate the entropy for each subset-

→ Consider Age :- (n: no. of occurences in T)

Age	n	P+	P-	entropy
very young	2	1/2	1/2	$-1/2 \log_2 1/2 - 1/2 \log_2 1/2 = 1$
young	1	0	1	0
Middle	3	1/3	2/3	$-1/3 \log_2 1/3 - 2/3 \log_2 2/3 = 0.918$
old	1	1	0	0
very old	2	2/2	0	0

Now, calculate average entropy

Average entropy [H(T, age)] ⇒

Formula for calculating average entropy :-

$$H(T; a) = \sum P_i \times H(T_i)$$

$$\therefore H[T, age] = \frac{2}{9}(1) + 0 + \frac{3}{9} + 0 + 0$$

$$H[T, age] = \underline{0.528}$$

→ Consider Ticket type:

Ticket type	n	P+	P-	Entropy
Long	4	2/4	2/4	1
Local	1	1	0	0
Short	4	2/4	2/4	1

$$\therefore H[T, \text{Ticket type}] = \frac{4}{9}(1) + 0 + \frac{4}{9}(1)$$

$$H[T, \text{Ticket type}] = \underline{0.888}$$

Consider Language frequency

Language Frequency	n	p+	p-	Entropy
Fluent-	5	3/3	0	0
Not fluent-	3	1/3	2/3	0.918
Accent-	2	1/2	1/2	1
Foreign	1	1	0	0

$$\therefore H[T, \text{Language Frequency}] = 0 + \frac{3}{9}(0.918) + \frac{2}{9}(1) + 0$$

$$H[T, \text{Language Frequency}] = \underline{0.528}$$

→ Consider Type of call

Type of call	n	p+	p-	Entropy
Local	4	2/4	2/4	1
Long distance	2	2/2	0	0
Intern	3	1/3	2/3	0.918

$$\therefore H[T, \text{Type of call}] = \frac{4}{9}(1) + 0 + \frac{3}{9}(0.918)$$

$$H[T, \text{Type of call}] = 0.450$$

Step 3:- Now that we have calculated average entropy for each attribute, we calculate information gain using:-

$$I(T, a) = H[T] - H[T, a]$$

$\therefore I(T; age) = 0.991 - 0.528 = 0.463$

$I(T; Ticket\ type) = 0.991 - 0.888 = 0.103$

$I(T, Language\ frequency) = 0.991 - 0.528 = 0.463$

$I(T, type\ of\ call) = 0.991 - 0.750 = 0.241$

Ans:- The ranking (lowest to highest) of attributes based on information gain is:-

→ Ticket type
→ Type of call
→ Age, Language Freqllency

Question 2:

Step I: Center the data by subtracting mean of the attribute from each value

$$\bar{x}_1 = \frac{10+9+8+11+7}{5} = 9$$

$$\bar{x}_2 = \frac{-3-1-2-4}{5} = -2$$

∴ The new table created is:

x_1	x_2
1	-5
0	-3
-1	-4
2	-6
-2	-2

$n = 5$

Step II: We compute the covariance using:

$$C_{ij} = \frac{1}{n-1} \sum_{m=1}^{n} (x_{im} - \bar{x}_i)(x_{jm} - \bar{x}_j)$$

$$\therefore \quad cov(x_1, x_1) = \frac{1}{4}(1^2 + 0 + 1^2 + 2^2 + 2^2)$$

$$cov(x_1, x_1) = 2.5$$

$$cov(x_2, x_2) = \frac{1}{4}\left[(-1)^2 + (1)^2 + 0 + (-2)^2 + (2)^2\right]$$

$$cov(x_2, x_2) = 2.5$$

$$\therefore cov(x_1, x_2) = \frac{1}{4}\left[(1)(-1) + 0 + 0 + (2)(-2) + (-2)(2)\right]$$

$$cov(x_1, x_2) = -2.25$$

Hence, we get the covariance matrix C:

$$C = \begin{bmatrix} 2.5 & -2.25 \\ -2.25 & 2.5 \end{bmatrix}$$

step III :- Calculate eigen vectors :

Eigen values of C are solutions of λ to the equation $\Rightarrow$

$$|C - \lambda I| = 0$$

$$\left| \begin{bmatrix} 2.5 & -2.25 \\ -2.25 & 2.5 \end{bmatrix} - \lambda \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \right| = 0$$

$$\begin{vmatrix} 2.5-\lambda & -2.25 \\ -2.25 & 2.5-\lambda \end{vmatrix} = 0$$

using the determinent property

$$(2.5-\lambda)^2 - (2.25)^2 = 0$$

$$6.25 - 5\lambda + \lambda^2 - 5.0625 = 0$$

$$\lambda^2 - 5\lambda + 1.1875 = 0$$

∴ solving the quadratic eqn, we get-

$$\lambda_1 = 4.75 \qquad \lambda_2 = 0.25$$

Now, to find the eigen vectors, we solve the equation to get :-

$$Ce_i = \lambda_i e_i$$

where

i = no. of variables.

∴ We first solve for $\lambda_1 \rightarrow$

$$Ce_1 = \lambda_1 e_1$$

$$\begin{bmatrix} 2.5 & -2.25 \\ -2.25 & 2.5 \end{bmatrix} \begin{bmatrix} e_{11} \\ e_{12} \end{bmatrix} = 4.75 \begin{bmatrix} e_{11} \\ e_{12} \end{bmatrix}$$

$\therefore (2.5)(e_{11}) + (-2.25)(e_{12}) = 4.75(e_{11})$

$2.25(e_{11}) = -2.25(e_{12})$

$\therefore e_{11} = -e_{12}$

let $e_{12} = 1$

$\therefore e_{11} = -1$

$\therefore e_1 = \begin{bmatrix} -1 \\ 1 \end{bmatrix}$

$\therefore$ To avoid multiple solutions, we convert e_1 to a unit vector by dividing the Euclidian norm of the vector.
$(\sqrt{1^2 + -1^2} = \sqrt{2})$

$\therefore e_1 = \begin{bmatrix} -0.71 \\ 0.71 \end{bmatrix}$

Similarly, we find e_2 by substituting λ_2 in $Ce_2 = \lambda_2 e_2$

$$\begin{bmatrix} 2.5 & -2.25 \\ -2.25 & 2.5 \end{bmatrix} \begin{bmatrix} e_{21} \\ e_{22} \end{bmatrix} = 0.25 \begin{bmatrix} e_{21} \\ e_{22} \end{bmatrix}$$

$$\therefore (2.5)(e_{21}) - 2.25(e_{22}) = 0.25(e_{21})$$

$$\therefore \quad 2.25\, e_{21} = 2.25\, e_{22}$$

$$\therefore \quad e_{21} = e_{22}$$

$$\text{let } e_{22} = 1 \quad \therefore e_{21} = 1$$

$$\therefore \quad e_2 = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

By converting e_2 to a unit vector we divide values by $\sqrt{2}$

$$\therefore \quad e_2 = \begin{bmatrix} 0.71 \\ 0.71 \end{bmatrix}$$

Hence, the principal components for the given 2D data are :-

$$e_1 = \begin{bmatrix} -0.71 \\ 0.71 \end{bmatrix} \qquad e_2 = \begin{bmatrix} 0.71 \\ 0.71 \end{bmatrix}$$

Question 3 a):

Feature selection algorithms are used to obtain the best features from the given set, thereby decreasing the complexity of the dataset. There are different approaches to select these features of which one is the **Wrapper** based approach. This approach consists of three different algorithms:

- Best Subset Selection
- Forward Stepwise Selection
- Backward Stepwise Selection

As given in the question, we consider that the dataset consists of "p" covariates of which we want to select "k" predictors where k<<p. These algorithms are briefly described as follows:

- **Best Subset Selection:**
  - At first, we consider the value of k= 1,2,3...p
  - For k=1, we find the relation of each feature with the output by finding the value of RSS for that particular feature. In the end we will get "p" number of models for k=1.
  - Now, we find the model with the **least** value for RSS and name it as μ(1).
  - Similarly, we repeat the step till k=p. As a result, we will get "p" models- μ from (1 to p).
  - From these models, we choose the best model using different techniques such as C_p, AIC, BIC, adjusted R^2
- **Forward Stepwise Selection:**
  - Similar to the above algorithm, we find the bet model when k=1
  - Say the best model for k=1 is X_1. Now when k=2, we find the model with X_1 as one of the feature and find another feature from the remaining set of features in the dataset.
  - New features get added to the best model as the value of k increases, whereas the old features are not dropped.
  - From the different models that we obtain in μ(1 to p) we choose the best model using different techniques such as C_p, AIC, BIC, adjusted R^2
- **Backward Stepwise Selection:**
  - In this methods, we consider the value of k=p and then start the computation. At this step we get a single model with p features as the best model
  - Now at k=p-1, we choose the best "p-1" features to be part of the best model for k=1 (i.e. μ(p-1))
  - For k=p-2, we choose the best p-3 features from μ(p-1) which contains p-1 features only. The feature that was discarded in the previous step is not considered again.
  - This process continues till k=1 as we get "p" models (μ(p-1 to 1))
  - From these models, we choose the best model using different techniques such as C_p, AIC, BIC, adjusted R^2

From the above description, we see that the Best Subset Selection checks every possible combination to find the model with the least value of RSS. However, it is not the case with Forward/Backward stepwise selection techniques. In Forward Stepwise Selection method, the feature once not selected in the previous set of best models μ(k) (where value of k is smaller) cannot be a part of the model for consequent values of k. Similarly for Backward Stepwise Selection, the feature once discarded from being a part of the best model cannot occur in the consequent models. Hence, there may arise a case where these 2 methods ignore a feature that could potentially be a part of the best model. In contrast to this, the Best Subset Selection follows an exhaustive approach and checks the value of RSS for each

and every possible model from the given set of p covariates. Due to this the value of RSS differs, where **Best Subset Selection** method provides us with the smallest training RSS.

Question 3 b):

In the above question, we discussed about the Wrapper based approach to feature selection. Another approach is the Embedded approach, where the following two methods are widely used:

1. Ridge regression
2. LASSO

The **similarity** between these two is that they both are examples of Regularization method where an additional constraint is added to coefficient estimates so that they shift toward zero. This results in decreasing variance and hence improves the prediction

- **Ridge Regression:**
  - It is a regularization method which minimizes the value of RSS by adding a constraint to the coefficient as follows:

$$\Rightarrow \sum_{i=1}^{n}\left(y_i - \beta_0 - \sum_{j=1}^{p}\beta_j x_{ij}\right)^2 + \lambda \sum_{j=1}^{p}\beta_j^2$$

$$\Rightarrow \quad RSS + \lambda \sum_{j=1}^{p}\beta_j^2$$

  - We can see that ridge regression adds a penalty to the value of β. This, reduces the values of β towards zero. It's significance is that it minimizes the effect of certain features that are unimportant but have a high impact on the class variable/ target variable.
- **LASSO**
  - It is similar to the above method, but differs with respect to the penalty added to the coefficients. LASSO minimizes the RSS by the given formula:

$$\Rightarrow \sum_{i=1}^{n}\left(y_i - \beta_0 - \sum_{j=1}^{p}\beta_j x_{ij}\right)^2 + \lambda \sum_{j=1}^{p}|\beta_j|$$

- ✔ As mentioned above, the major **difference**, between the two methods is the constraint on β. In LASSO, the coefficient can be equal to zero in some cases, which is not possible in the the case of ridge regression
- ✔ The occurrence of $\beta = 0$, signifies that the feature with which it is associated need not be fitted in the model and hence is dropped. In this way, LASSO acts as a variable selection algorithm in some cases.
- ✔ This reduces the curse of dimensionality and provides us with better predicted values.
- ✔ Therefore, I would prefer **LASSO** because of the reasons stated above.