Continuous Feature	Number (%) missing	Min	Max	Mean	Median	Skewness
Assembled.Payload.Size	0	-1	77704	47068	46873	-0.4521
DYNRiskA.Score	0	0.2640622	0.8339315	0.6335027	0.5519468	-0.7072
Response.Size	0	233406	733978	504088	498696	-0.1534
Source.Ping.Time	0	110	453	267	266	0.0023
Connection.Rate	0	0.3118	1611.9322	427.3517	404.8002	0.6187
Server.Response.Packet.Time	0	65	397	226	218	0.1981
Packet.Size	0	1258	1369	1348	1328	0.0993
Packet.TTL	0	35	57	63.32	63	-0.0728
Source.IP.Concurrent.Connection	0	9	34	21.43	21	0.1037

Categorical Feature	Category	N(%)
IPV6.Traffic	-	25
		8.5
	FALSE	66.5
	TRUE	0
Operating.System	-	0
	Android	40.1666
	iOS	8
	Linux (Unknown)	0.5
	Windows (Unknown)	42.5
	Windows 10+	0.5
	Windows 7	8.3333
Connection State	ESTABLISHED	66.8333
	INVALID	30.6666
	NEW	1.5
	RELATED	1
Ingress.Router	mel-aus-01	63.16667
	syd-tls-04	36.83333
Class	1	50
	0	50

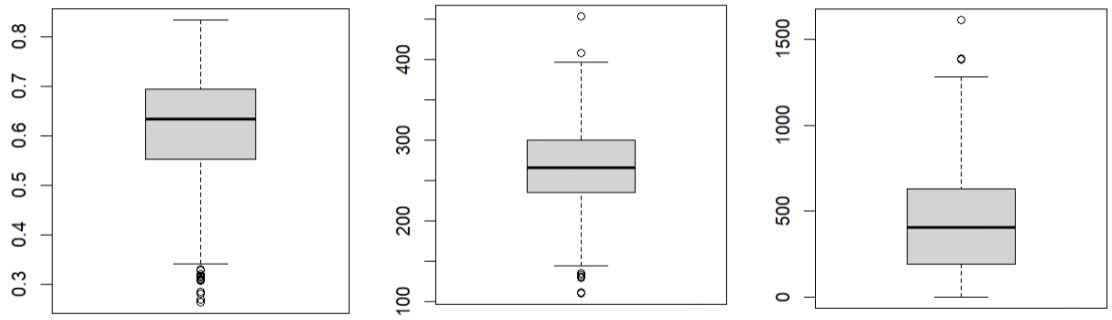
Part 1

(III)

There is not NA values can be found in the dataset. However, we could find may missing values in various variables. If we take the IP Traffick variable, 25% of data is '-'which can be considered as a missing value. Moreover, there are 8.5% of ' ' missing values which has not value at all. Thus, it is clear the data needs to be cleaned before we start model using an machine learning algorithm.

Other noticeable scenario in IPV6 Traffick column has not 'FLASE' values and there are no TRUE values. We assume that other missing values are actually Ipv6 address and changed it to 'TRUE'. However, still we do not have enough information to come to this conclusion and change them to 'TRUE'. Thus, the best option here is to remove the entire column and other wise it would leads the Machine Learning algorithms to have overfitting issue.

Likewise, I have changed Operating.System missing values into NA. Moreover, we can see few outliers in DYNRiskA.Score, Source.Ping.Time and Connection.Rate. But there is no enough information available for me to classify them exactly as outlier. However, in Source.Ping.Time and Connection.Rate time outliers in the boxplot has been changed with their mean values.



Part 2

(III)

3.1

Many supervised and unsupervised machine learning algorithms including Principal Component Analysis Anal predict well if the data is scaled. Scaling means that you are basically transforming your data so that it fits within a specific scale. Thus, these scaled values should have a mean value of 0.

Principal Component Analysis (PCA) is a technique which finds the major patterns in data for dimensionality reduction. Thus, variability of the data will play a huge role when it

is calculating the new dimensions. Scaled data will allow this unsupervised algorithm to predict accurate predictions. $z = \frac{value - mean}{standard\ deviation}$.

For Example: Source Ping Time, Server Response Packet Time are in microseconds. However, it is not even clear what scale Assembled Payload Size and Packet Size was measured.

The variables with higher standard deviation will have higher weights on the new axis. Thus, Standalization is the best method that could be used here. Once the data is standardized, all variables will be in the same weight in the new axis. After the PCA process has standardized the data, the coefficients can be directly interpreted without the need for any computation between each PC and each feature. Moreover, even though there is nothing to scale, doing so has no negative effects on PCA calculations.

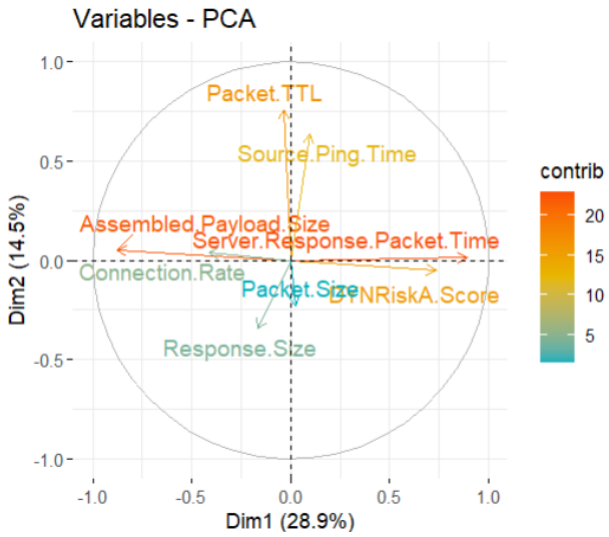
3.2

The individual proportion refers to the portion of the total variability in the data frame that is accounted for by each principal component separately. To calculate it, you divide the eigenvector of the component of interest by the sum of all eigenvectors. Specifically, PC1 explains 24% of the variance, while PC2, PC3, and PC4 explain 28.93%, 14.48%, and 12.87% of the variance, respectively. In the given dataset, PC1 accounts for 28.93% of the variance, and PC1 and PC2 together explain 43.41% of the variance. PC3 and PC4 explain 56.28% and 68.51% of the variance, respectively. We can rely on PC4 for further analysis since PC1, PC2, PC3, and PC4 together account for 68.51% of the cumulative proportion of variance. Once all eight principal components are considered, 100% of the variance will be explained.

3.3

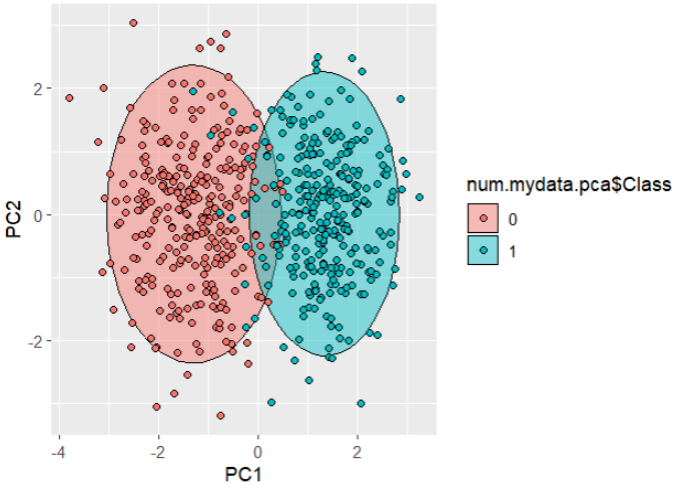
Using the first three principal components should be sufficient to explain the data, as their cumulative proportion accounts for more than 50% of the total variance. However, creating a 3D plot with three dimensions can be difficult, so it would be better to use only the first two principal components for data visualization purposes, as they collectively account for more than 43% of the variance in the dataset.

```
> summary(my.pca)
Importance of components:
                          PC1    PC2    PC3    PC4    PC5    PC6     PC7     PC8
Standard deviation     1.5214 1.0763 1.0147 0.9890 0.9441 0.9139 0.74468 0.48779
Proportion of Variance 0.2893 0.1448 0.1287 0.1223 0.1114 0.1044 0.06932 0.02974
Cumulative Proportion  0.2893 0.4341 0.5628 0.6851 0.7965 0.9009 0.97026 1.00000
```



Loading refers to the coefficient of a linear combination of the original variables that are projected onto the principal component. These coefficients represent the correlation between the original variables and the principal components. The threshold for identifying significant loadings may vary depending on the situation, but in this particular dataset, a threshold of 0.3 or higher is typically used. For instance, PC1 (Principal Component 1) is a linear combination of Assembled.Payload.Size (-0.576) and DYNRiskA.Score (0.483), among other variables, with the coefficients indicating strong contributions of these variables to PC1. In PC2, Server.Response.Packet.Time and Server Response Packet Time have similar correlation coefficients, both positively contributing to PC2. Additionally, Packet.TTL has the greatest influence on PC2 compared to other variables, with a positive contribution to PC2.

(IV)



PCA reveals a distinctive distinction between malevolent and non malicious network packets, which is not typically observed in other datasets. Packet TTL and Source Ping Time are significantly interrelated when their vectors are compared. Additionally, there exists a strong correlation between Response Size, Connection Rate, and Packet Size. Likewise, Assembly Payload Size and Server Response Packet Time are closely related to each other.

As per the Biplots analysis, the loading vectors for the first two principal components are represented by arrows. The highly correlated vectors mentioned earlier have longer arrows, indicating their greater significance compared to other

vectors.Furthermore, due to the nearly 90-degree difference between Assembled Payload Size and Source Ping Time, they are not correlated with each other. Because Packet TTL and Packet Size have an almost 180-degree difference, they are negatively correlated with each other.

(v)

As mentioned above, in the first dimension itself one can explain most of the variance. Which is around 28%. And the 2nd dimension can only explain 14% of variance, however, both dimensions can explain around 43% of variance. These numbers are better compared to most other real world data sets. However, with the dataset we can differentiate the packet classes well enough. And mainly PC1 or the 1st dimension can be used to differentiate packet maliciousness.