Part 1

Continuous Feature	N (%)	Min	Max	Mean	Median	Skewness
Ping.Time.To.Server	88	30	1046	277.3	230.5	NA
File.Size..Bytes.	-	34035	38146187	8627123	7264877	1.3235
How.Many.Times.File.Seen	-	190	65535	715.6	595	22.2561
Calls.To.Low.level.System.Libraries	36	2	8927	1005.3	661	NA
Threads.Started	-	1	34	1.866	2	16.0033
Mean.Word.Length.of.Extracted.Strings	-	3.44	6.06	4.947	4.985	-0.2252
Similarity.Score	-	-3.00275	2.510398	-0.0009	0.011502	-0.0057
Characters.in.URL	-	21	162	70.19	70	0.2268

A simple for loop with an if statement has been utilized to retrieve the number of instances for each of categories and to retrieve summary of numeric values.

Categorical Features	Category	N (%)
Download.Source	Email	17.6
	Exposed Corporate Intranet Pages	0
	Exposed Hardware Manufacturers Site	0
	Git Repository	3.8
	Linked Directly to Google Search Result	35
	Linked to Web Search Advertisement	0.6
	Personal NAS Exposed to World	0
	University Web Pages	2
	Well Known Software Download Websites	28.6
	Wordpress Powered Web Site	12.4

And This will determine the number of instances for each of the categories in the mydata data frame.

Summary function has been used to summarize each of continuous or numeric values in the database:

TLD	.com	43.6893204

	.edu	7.2815534
	.gov	13.3495146
	.info	0
	.net	34.223301
	.net.au	0.7281553
	.org	0.7281553
	.tv	0
Download.Speed	-1	0
	100KB/s to 1MB/s	54.61165
	1MB/s to 10MB/s	26.941748
	Greater than 10MB/s	1.699029
	Less than 100 KB/s	16.747573
Executable.Code.Maybe.Present.in.Headers	YES	60.6
	NO	39.4
No.Executable.Code.Found.In.Headers	YES	100
Evidence.of.Code.Obfuscation	YES	60.2
	NO	39.8
Actually.Malicious	YES	47.2
	NO	52.8

1.1 Dealing with missing values:

According to the above-mentioned tables, clearly there are some “Not Applicable” values or missing values can be identified. Furthermore, the missing values in the data frame are also explained by the is.na(mydata) function. However, there are only five features include these NA values. If the data set is examined closely, a relation can be identified from the NA values in the data set. For example, all the NA values belong to ‘Download Source’ feature are emails. And it makes sense to have NA values for TLD, “Download Speed” and “Ping Time To Server” as well. Furthermore, “No Executable Code Found In Headers” contains some NA values. Because once the malware does not have executable or MZ header, Data set had been marked them as NA values.

Therefore, NA values cannot be removed from this data set since they represent something else. However, latter part of this report NA values replace with 0, to calculate PCA. NA values could also be omitted however, Omitting, or removing is not a good option. since it cuts out entire raw and data, if you remove all the NA value which means you will lose all the Email data.

1.2 Removing unwanted features from the data frame:

Nonetheless, Once the “Executable Code Maybe Present in Headers” and “No executable Code Found in Header" variables are compared together. A logical similarity can be identified. Basically, “Executable Code May Be Present in Header” variable is paradoxically similar to Executable Code Found in Header. 2.1 section of the R script if statement code proves it. And if two columns are identical the code will automatically remove “No Executable Code Found In Headers” feature from the data frame.

1.3 Removing outliers:

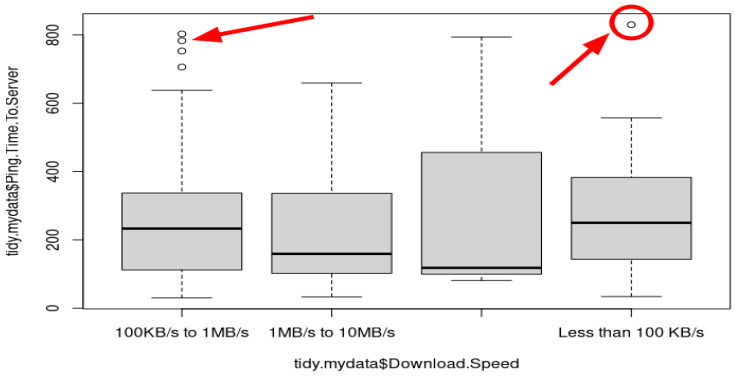
There are quite a few outliers that can be identified with a data frame.

Ping Time to Server

according to the boxplot number of outliers can be identified. However, with very limited information, it is hard to figure out if these outliers are invalid data. Nonetheless, the values which are more than 1000 can be removed. But without any further information about the data set, other outliers can be removed.

Download Speed

Furthermore, visualizing “Download Speed” and “Ping Time To Server” may also give a sense of outliers. Basically, Ping time



and Download or Upload speed related to each other in two ways. Therefore, an idea about Download Speed can be gained with ping time to the server also.

Finally, identified outliers can be filtered. then they can be removed.

How Many Times File Seen

above mentioned summary table and box plot illustrates that this feature has a great outlier. It has 65535 maximums of value when its mean is around 715. And this value either can be removed or edited with its mean value.

Threat Start

This also consists with few outliers. Once a program is executed at least one thread should be started. The maximum number of threads depends on the computer memory. But 34 does not make sense when its mean is around 1.8. Therefore, it is better change them with 2. Because its mean value is around 1.8

However, there is not enough evidence to remove other outliers from the data frame.

Part 2

2.3.1 why data should be scaled?

Not only PCA, but many machine learning algorithms also work perfectly if the data is on the same scale which means all the variables should be centralized and should have the mean value of 0. .(James, Witten, 2013) Basically, PCA calculates a new projection to the data frame. Therefore, the new axis will be based on the variability of the data frame. Thus, data should be scaled since the data frame has been defined in different units. For instance, this data set file size was measured in bytes, an unknown scoring system has been used for Similarity Score. Furthermore, it is not even clear to what scale Ping.Time.To.Server is measured (however, it could be milliseconds). Standardization is the best technique that could be used for scaling this data set. Once the data set is standardized, all variables will have the same weight on the new axis. Otherwise, variables with relatively high standard deviation may have a higher weight on the axis than relatively low standard deviation. Once the data is standardized in the PCA process, coefficient can be interpreted directly without any computation between each of the PCs with each of the features. Moreover, doing so does not do any harm to PCA calculations even though there is nothing to scale.

2.3.2 Explaining individual and cumulative proportions of variance by each first 4 components.

Individual proportion is the amount of variance each principal component accounts for in the data frame individually. This can be calculated, by dividing the component eigenvector which is in interest from the sum of eigenvectors. $\frac{\lambda_i}{\sum_{j=1}^{p} \lambda_j}$

Basically, PC1 holds 24% of the variance while PC2, PC3 and PC4 hold respectively 13%, 13% and 12% of the variance. However, Cumulative Proportion is the accumulated amount of explained variance. If the component is in interest is k, the k's cumulative proportion of variance can be calculated with this equation. $\frac{\lambda_1 + \lambda_2 + \cdots + \lambda_k}{\lambda_1 + \lambda_2 + \cdots + \lambda_p} \quad \text{for} \quad k \leq p$ Considering the given dataset, PC1 explains 24% of variance and both PC1 and PC2 explain 38% of variance. Moreover, when it comes to PC3 and PC4, all Principal components explain 51% and 65% of variances, respectively. We can rely on PC4 for further analysis, since all together (PC1,2,3 and 4) represent 64% of cumulative proportion of variance.

2.3.2 Outlining the coefficients for PC1 to PC4.

Loading is the coefficient of the linear combination of the original variables which are projected to the principal component. Basically, they represent the correlation between original variables and the principal components. The threshold depends on the circumstance However, with this data set it is normally at anything more than 0.3 can be identified as the threshold. For example, PC1 or Principal component 1 is a combination of **(Ping.Time.To.Server*-0.0006)+ (File.Size..Byte*0.6844)...\sum(EigenValues).** In the PC1 File Size Bytes and Calls to Low Level System Libraries, correlation coefficients are almost similar. Because they are both have plus values and have higher correlation. Likewise, PC2 is also have couple of similar values like that which are “Ping Time To Server” and “Similarity Score” is one (Loadings around 0.6) and other one is “Ping Time To Server” and “Characters in URL” (Loadings around 0.3).

As this report mentioned above, ‘File Size Bytes’ and ‘Calls To Low Level System Libraries’ contributed strongly to the PC1. Furthermore, ‘Ping Time To Server’, ‘Similarity Score’ and ‘Thread Started’ (Negatively) are strongly contributed to the PC2. ‘Thread Started’ variable also strong but positively contributed to PC3 as well.

2.3.3 How many principal components are adequate to explain at least 50% of the variability.

According to the answer that was given during the 2.3.2 section of this report, the first three components would be adequate to explain. Because first three principal component’s cumulative proportion is more than 50%. Since there are 8 variables, if each variable contributed equally same, one variable would have to contribute 12.5% from the total variables (as shown in 2.3.3 barplot). However, visualizing 3D draft is quite too hard with three dimensions, therefore, it is better to use first two principal components to visualize data since both PC represent almost 40% variance in the data set.

2.5 Biplot explanation.

There is no considerable separation can be seen between malicious and non-malicious files with the biplot. When the vectors are compared, especially their positions, Calls To Level System Libraries and File Size Bytes are highly corelated to one another. Moreover, Characters in URL, How Many Times File Seen and Ping Time to Server, Similarity Score also highly correlated to one another. According to 2.5.2 Biplots, the arrows represent loading vector for first two principal components, above mention corelated vectors also have higher length arrows. Therefore, they are more important than others. Nonetheless, these mentions correlated cluster vectors (Call Too Low level System Libraries, Files Size Bytes and Ping Time To server - Similarity Score) are not correlated to each other since they have almost 90 degree between their vectors. Thus, these vectors close to 0 correlation. And How Many Times File Seen vector and Threads Started are highly negatively corelated. Because these two vectors are almost directly opposite to each other.

Moreover, Non-Malicious files may tend to have greater ‘Calls To Level System Libraries’, ‘File Size Bytes’ as well as somewhat higher ‘Ping Time to Server and Similarity Score’. There is small possibility of having greater ‘File Size Bytes’ in Non-malicious files and greater ‘Word Length of extracted strings’ in Malicious Files. However, other vectors have no considerable separation between Malicious and Non-Malicious files.

2.6 Classification of malwares

First Dimension explains 24% of data variability while Dimension 2 explains 13% of variance. And both together explain almost 40% of variance. The PC1 can be used for the

classification of most of the points. Because it is PC that has fewer overlapping points once they are projected to the principal component and it is the PC that has the highest Eigenvalue. However, non-malicious file and malicious files cannot be differentiated well since most of the PCA value still overlap with each other on PC1. However, most right sided values have a very little tendency to be non-malicious files. Especially after PC1 -1.5. Thus, PC1 is better for the classification rather than any other principal component.

Some of the Non-malicious files tend to have a higher Calls To low Level systems and File Size Bytes sizers. Moreover, Non-Malicious files also tend to have a greater Ping Time To Server and Similarity Score. Furthermore, left most sample also tend to higher Calls to Low level Systems and File Size Byte. And sample towards the top of the 2.5.1 biplot tend to have almost 100% greater Ping Time To Server and Similarity Scores. (James, Witten, 2013)

Reference:

1. James, G., Witten, D., Hastie, T., & Tibshirani, R. (2013). *An introduction to statistical learning* (Vol. 112, p. 18). New York: springer.(James, Witten, 2013)
2. Holland, S. M. (2008). Principal components analysis (PCA). *Department of Geology, University of Georgia, Athens, GA*, 30602-2501.
3. Ding, C., Zhou, D., He, X., & Zha, H. (2006, June). R 1-pca: rotational invariant l 1-norm principal component analysis for robust subspace factorization. In *Proceedings of the 23rd international conference on Machine learning* (pp. 281-288).