# Introduction

<!-- X Brief description of the dataset used; -->

In this report I will present a dataset obtained from the from [http://networkrepository.com/rt-barackobama.php](http://networkrepository.com/rt-barackobama.php) website; it is a retweet network of the US president Barack Obama. The network was downloaded in a form of an edge list.

I wanted to see if by performing a set of tests on the network, it would be possible to identify Barack Obama's vertex. I believe I succeded to identify the aformnetioned vertex with a considerable degree of confidence. I will present my reasoning and methods which led to me to such conclusion.

<!-- X what interested the student about the dataset; -->

<!-- X from where was the dataset obtained; -->

<!-- X what form was the dataset in (edge list or triangular matrix); -->

<!-- X what questions would they like to answer about the dataset; -->

# Methods

The network was analysed using R and Python programming languages, and Gephi software package. The data statistics were calculated using Python; the data preparation and manipulation was done using R; the data visualisation was done using R and Gephi.

<!-- X What data preparation did they have to carry out; -->

## Data Preparation

The data was prepared using R programming language and RStudio software; the data was downloaded, unziped and saved into a R table (within the code). \*Timestamp\* column was dropped (it would be useful if the dataset was an updatable one; then it could be used to update the set only from a certain timestamp but in this case the set was a complete one).

By default, R names the columns like this:

$$ x = \big\{V1, V2, ... , Vn \mid n \in \mathbb{N} \big\} $$

To make the data manipulation easier the columns were renamed such that:

| Previous column name | Changed column name |

| - | - |

| V1 | from |

| V2 | to |

Later, the data was saved in a `.csv` format so it could be easly loaded into memory using Python programming language.

<!-- X definitions of the network metrics used; any code should be provided in the Appendix of the report or as separate files; -->

<!-- Comment -->

<!-- methods section is not well presented - an explicit description of the nodes and edges is not given until near the end of this section. Some assumptions are stated as fact (Obama is the highest degree node for example). -->

<!-- X which metrics did they use to analyse the data; -->

## Metadata

The following metadata (which can be found on the [http://networkrepository.com/rt-barackobama.php](http://networkrepository.com/rt-barackobama.php) website) were used in the network analysis:

```{r, out.width="1.00\\linewidth", include=TRUE, fig.align="center", echo=FALSE}

knitr::include\_graphics("/Users/mateuszzaremba/dev/R/Networks/png/Metadata.png")

```

<!-- | | Metadata | | -->

<!-- | - | - | - | -->

<!-- | Category | Sparse Network | -->

<!-- | Collection | Retweet Network | -->

<!-- | Vertex type | User | -->

<!-- | Edge type | Retweet | -->

<!-- | Edge weights | Unweighted | -->

<!-- | Description | Nodes are twitter users and edges are retweets. These were collected from various social and political hashtags | -->

The dataset creators stated that the data was collected from "various social and political hashtags". It is not clear what methods they used but I suspect that once a user (which later will become a vertex in the network) which retweeted Barack Obama's tweet, a retweet link (the edge in the network) was created. Later, in the dataset visualisation, it is clearly visible that some people retweeted the retweets of someone who retweeted Barack Obama's tweet. This would create a path in the network to Barack Obama. The creators did not state if the network was simple or not but the fact the the network is sparse tells us that the creators took the steps to improve the network's efficiency in loading into memory and computation time. The sparcity statistic was not given so it is not certain to what degree the network's connection weights were reduced to 0 or very close to 0. The creators did not also specifically state if the network is simple or not but their efforts to make it as efficient as possible lead me to assumption that the network is a simple one, meaning it contains no loops or multiple edges; the visual analysis as well as the size metrics confirm this assumption which I elaborate in the next section.

## Network size metrics

The analysis conducted with Python relied on the correct data prepartion in R; it was crucial for further analysis for the statistics from the ‘Network Repository’ website, like \*number of nodes\*, \*number of edges\*, \*average degree\* etc. (Rossi and Ahmed, 2015) to match the ones I calculated. The statistics matched and the comparision of the reults is presented in the table below:

| | Calculated Statistics | Website Statistics |

| - | -------------------| --------------------- |

| Number of nodes | 9631 | 9.6K |

| Number of edges | 9775 | 9.8K |

| Average degree | 2.0299 | 2 |

## Network Structure Metrics

Our graph in undirected so it does not contain in- or out-degrees. Instead, we found the highest degree node - it had an id of 2506 and degree of 7655; this was likely Barack Obama’s twitter account since it is the most connected.

## Network Density

The network’s density is equal to 0.000210789557302036, which is close to zero, meaning - our graph is close to being ‘fully disconnected’.

## Shortest Path Between Two Nodes

We will find the shortest path between top two nodes with the highest degree and between the nodes with the highest and the lowest degree.

### Top Two

The top two nodes with the highest degree had ids: 2506 (Barack Obama) and 9302. Shortest path between these two nodes is: ['2506', '8474', '9302']; the length of this path is 2, which means they are not connected directly.

### Highest and Lowest Degree

As already mentioned, the id of the node with the highest degree is: 2506 and the id of the node with the lowest degree is: 2709; the shortest path between these two nodes is: ['2506', '2709'] and its length is 1, meaning the nodes are connected directly.

## Identifying Network Communities

We managed to identify 138 community groups:

`Counter({1: 7414, 6: 611, 7: 158, 12: 114, 4: 106, 2: 102, 50: 84, 5: 69, 88: 66, 61: 61, 127: 53, 35: 49, 19: 39, 25: 38, 17: 27, 111: 27, 77: 26, 23: 25, 112: 25, 0: 22, 49: 22, 26: 20, 54: 15, 37: 14, 102: 14, 22: 13, 41: 12, 48: 12, 55: 12, 9: 11, 15: 10, 32: 10, 75: 10, 97: 10, 34: 9, 38: 9, 82: 9, 85: 9, 13: 8, 3: 7, 29: 7, 84: 7, 105: 7, 108: 7, 68: 6, 78: 6, 83: 6, 117: 6, 86: 5, 110: 5, 8: 4, 27: 4, 31: 4, 62: 4, 67: 4, 79: 4, 93: 4, 100: 4, 129: 4, 10: 3, 20: 3, 36: 3, 40: 3, 47: 3, 66: 3, 70: 3, 80: 3, 81: 3, 87: 3, 90: 3, 95: 3, 104: 3, 107: 3, 114: 3, 125: 3, 133: 3, 134: 3, 139: 3, 11: 2, 14: 2, 16: 2, 18: 2, 21: 2, 24: 2, 28: 2, 30: 2, 33: 2, 39: 2, 42: 2, 43: 2, 44: 2, 45: 2, 46: 2, 51: 2, 52: 2, 53: 2, 56: 2, 57: 2, 58: 2, 59: 2, 60: 2, 63: 2, 64: 2, 65: 2, 69: 2, 71: 2, 72: 2, 73: 2, 74: 2, 76: 2, 89: 2, 91: 2, 92: 2, 94: 2, 96: 2, 98: 2, 99: 2, 101: 2, 103: 2, 106: 2, 109: 2, 113: 2, 115: 2, 116: 2, 118: 2, 119: 2, 120: 2, 121: 2, 122: 2, 123: 2, 124: 2, 126: 2, 128: 2, 130: 2, 131: 2, 132: 2, 135: 2, 136: 2, 137: 2, 138: 2})`

It was calculated that the sizes of the communities range from 2 to 7414. We can, again, assume that the biggest community is likely to be centred around Barack Obama; these would be the people who retweeted his tweets directly from him.

## Network Structure Connectivity

Investigation reveals that the Barack Obama network is fully connected, and it has no sub-components; this is not surprising because the edges are retweets, nodes are twitter users and the network consists only of users who retweeted Barack Obama's posts.

## Network Hubs/Brokers

Betweenness and closeness centrality was successfully calculated (Although, it took almost 40 minutes to compute) and sorted from the highest to the lowest score for top 20 results; but a `(PowerIterationFailedConvergence(...), 'power iteration failed to converge within 100 iterations')`

error kept occurring when calculating the nodes eigenvector centrality; replacing `nx.eigenvector\_centrality` with `nx.eigenvector\_centrality\_numpy` solved the issue (Stack Overflow, 2019) and after the fix, the computation was almost instantaneous.

Barack Obama’s node (id = 2506) had the highest score in closeness, betweenness and eigenvector centrality, which means:

- The highest closeness centrality score – it is the farthest away from all other nodes in the network or – it takes the most time to spread information sequentially from it to other nodes (Sciencedirect.com, 2019).

- The highest betweenness centrality score – it has the highest number of distinct paths that strictly contain it in-between (Sci.unich.it, 2019).

- The highest eigenvector centrality score – it is the most influential node in the network.

# Results

<!-- X Have they been able to answer the queries they wished to pose as described in the introduction; X visualisation(s) of the network structure; X results from the analyses described in methods; -->

The network is a retweet network of Barack Obama's tweets; it is not obvious that Barack Obama (as a user) feautres is in the network; the highest degree vertex could be someone who's retweeted Barack Obama a thousand of times. At the time, it could had been his officail \@POTUS twitter account which used to retweet every tweet from his private [\@BarackObama](https://twitter.com/barackobama) twitter account. This would mean that evey single person in the network has retweeted the POTUS account insted of Barack Obama's - \@POTUS account gets thousands of retweets so it is possible. But given the fact that it is "Barack Obama's retweet network" I think it is more plausible that the highest degree vertex is Barack Obama's private twitter account.

We could also think that the vertex with the highest degree could be [\@POTUS44](https://twitter.com/POTUS44) (archival twitter account of Barack Obama when he was the incumbent from 2009 to 2017). Although this is unlikely, given that the data reference request is dated to 2015. This would mean that the dataset was created was Barack Obama was incubent.

```{r, out.width="1.00\\linewidth", include=TRUE, fig.aligh="center", echo=FALSE}

knitr::include\_graphics("/Users/mateuszzaremba/dev/R/Networks/png/reference.png")

```

Which leaves us with a hypothesis that the data was collected from either \@POTUS (which in 2015 was Barack Obama) or \@BarackObama. The title of the network, and no apparent reason why it would be taken from any other twitter account retweeting the original tweets by Barack Obama, leads me to a conclusion that the data was gathered from Barack Obama's offical twitter account; but it is possible that, at the time, the data was gathered from the official POTUS twitter account.

<!-- Comments -->

<!-- Results section is mainly comprised of different visualisations of the network (metric calculations are in Methods section but are taken into account here). -->

## Data Visualisation

Becasue the data represents a network of users and was likely collected from Barack Obama's official twitter account, it was assumed that the biggest community of 7414 would be Barack Obama’s first-degree neighbours. This cannot be really well seen on the graph without any colouring or sizing:

```{r, out.width="0.55\\linewidth", keepaspectratio, include=TRUE, fig.align="center", echo=FALSE}

knitr::include\_graphics("/Users/mateuszzaremba/dev/R/Networks/Network/vanilla.pdf")

```

\pagebreak

But we can immediately see Barack Obama’s node (The big red one, obstructed by a bunch of smaller, blue nodes) when we distinguish the node’s degree using sizing and colouring:

```{r, out.width="0.95\\linewidth", include=TRUE, fig.align="center", echo=FALSE}

knitr::include\_graphics("/Users/mateuszzaremba/dev/R/Networks/Network/size\ colour\ node\ degree.pdf")

```

\pagebreak

The graph becomes even more readable when we use colouring to represent the node’s score of closeness and the node’s sizing to represent its measure of betweenness:

```{r, out.width="0.95\\linewidth", include=TRUE, fig.align="center", echo=FALSE}

knitr::include\_graphics("/Users/mateuszzaremba/dev/R/Networks/Network/colour\ closeness\ -\ size\ betweenness\ with\ NO\ lgl.pdf")

```

\pagebreak

The same graph was plotted using `Large Graph Layout` function which slightly improved the readability of the graph:

```{r, out.width="0.95\\linewidth", include=TRUE, fig.align="center", echo=FALSE}

knitr::include\_graphics("/Users/mateuszzaremba/dev/R/Networks/Network/colour\ closeness\ -\ size\ betweenness\ with\ lgl.pdf")

```

\pagebreak

Interesting results were achieved using `Gephi` (Gephi.org, 2019), were it is clearly visible that there is one central node in the network (Barack Obama); some of the 138 community groups (clusters) can be visible in the graph - there is one bigger cluster which was marked with an orange frame and many smaller ones which were marked with a green frame; these would be characterised by the size ranging from 2 to 611, i.e. the smallest community and the second largest (as mentioned before the largest community are the people who retweeted Barack Obama directly):

```{r, out.width="0.94\\linewidth", include=TRUE, fig.align="center", echo=FALSE}

knitr::include\_graphics("/Users/mateuszzaremba/dev/R/Networks/pdf/graph\ marked.pdf")

```

\pagebreak

The small cluster up close:

```{r, out.width="1.00\\linewidth", include=TRUE, fig.align="center", echo=FALSE}

knitr::include\_graphics("/Users/mateuszzaremba/dev/R/Networks/png/cluster.png")

```

\pagebreak

The same cluster showing the nodes id of 9302:

```{r, out.width="1.00\\linewidth", include=TRUE, fig.align="center", echo=FALSE}

knitr::include\_graphics("/Users/mateuszzaremba/dev/R/Networks/png/cluster\_label.png")

```

<!-- About the cluster 9302 -->

This, as well as the other smaller clusters, are likely to be some organisations, people actively supporting Barack Obama or his close family members and friends who would be actively retweeting his tweets.

\pagebreak

And for the final proof, here is a zoom of the same graph generated with Gephi and node labels on, where it is clearly visible that the most central node, has an id of 2506:

```{r, out.width="1.00\\linewidth", include=TRUE, fig.align="center", echo=FALSE}

knitr::include\_graphics("/Users/mateuszzaremba/dev/R/Networks/png/Screenshot\ 2019-12-17\ at\ 19.01.19.png")

```

# Conclusion

<!-- Comment-->

<!-- Student does state that they have "very likely" identified barack Obama's node (although I disagree - if edges are retweets then is he likely to have retweeted so much of his own posts? This is out of character for him in my opinion. This node is more likely a close family member, publicist or close friend). Conclusions are a bit scarce - further analysis on any communities within this network should have been suggested (or better carried out here). -->

<!-- What can be inferred from their analysis of the network; If they could answer their initial questions, what were the answers; was the dataset detailed enough - if not what other information could have been useful in the analysis; what problems did they encounter; how could they have overcame them; -->

To my best knowledge, the report presents complelling evidence for the conclusion that the node with an id of 2506 is Barack Obama's private twitter account; the vertices are the twitter users, the edges are the retweets, ergo the twitter users are connected by the retweets.

The only way the two users can get connected by a retweet is when one of them retweets a tweet of the other one; in other words, one of them must have retweeted the other because there is an edge (retweet) between the vertices (users).

The Vox study shows that the democrats and the republicans, in the US, live in their own social bubbles on the internet (Data based on twitter). This might further the assumptions that people who have interacted with Barack Obama's tweets (by retweeting them) are most likely living in one of these bubbles and are his devotees.

Inter alia, the report presented a way to find the shortest path between top two nodes with the highest degrees as well as the node with the highest and the lowest degree; identify 138 communities within the network and their range; calculate and analyse the network's closeness, betweenness and eigenvector centrality; conduct a comprehensive visual analysis.

Some of the challenges were rendered by the size of the network which made it quite difficult to work with; calculation of the node betweenness and closeness centrality took approx. 40 minutes. Plotting of the network in R took only a few moments but it was only due to the fact that the nodes' labels were not being displayed, otherwise, it was taking approx. 5-10 minutes to plot. Nonetheless, an attempt was made to speed up the process using `with\_lgl(…)` (Rdrr.io, 2019) function for `Large Graph Layout` (See Appendix B) but no improvement in rendering speed was noticed apart from a slightly better visual representation of the communities.

One of the things that could be done in the future, to improve calculation and render times, when analysing as big or even bigger networks, could be a usage of a GPU for network analysis. (Mathworks.com, 2019); e.g. Geforce GTS 250 has 450 cores working in parallel versus 4 on the CPU (Kajan and Slačka, 2019).

Overall, the data was well described and of a good quality; the original analysis provided a lot of good insight (table 1) and the size of the data was appropriate for a comprehensive analysis.

\pagebreak

# References

Gephi.org. (2019). Gephi - The Open Graph Viz Platform. [online] Available at: https://gephi.org/ [Accessed 16 Dec. 2019].

Kajan, S. and Slačka, J. (2019). COMPUTING OF NEURAL NETWORK ON GRAPHICS CARD. BSc. Slovak University of Technology in Bratislava.

Mathworks.com. (2019). Neural Networks with Parallel and GPU Computing- MATLAB & Simulink. [online] Available at: https://www.mathworks.com/help/deeplearning/ug/neural-networks-with-parallel-and-gpu-computing.html;jsessionid=fad53e284f66e53b4d2665c85a87 [Accessed 16 Dec. 2019].

Rdrr.io. (2019). layout\_with\_lgl: Large Graph Layout in igraph: Network Analysis and Visualization. [online] Available at: https://rdrr.io/cran/igraph/man/layout\_with\_lgl.html [Accessed 16 Dec. 2019].

Rossi, R. and Ahmed, N. (2015). The Network Data Repository with Interactive Graph Analytics and Visualization. [online] Network Repository. Available at: http://networkrepository.com/rt-barackobama.php [Accessed 16 Dec. 2019].

Sci.unich.it. (2019). Betweenness Centrality. [online] Available at: https://www.sci.unich.it/~francesc/teaching/network/betweeness.html [Accessed 16 Dec. 2019].

Sciencedirect.com. (2019). Closeness Centrality - an overview | ScienceDirect Topics. [online] Available at: https://www.sciencedirect.com/topics/computer-science/closeness-centrality [Accessed 16 Dec. 2019].

Stack Overflow. (2019). Using networkx to calculate eigenvector centrality. [online] Available at: https://stackoverflow.com/questions/43208737/using-networkx-to-calculate-eigenvector-centrality [Accessed 16 Dec. 2019].

Sadler, J. (2019). Introduction to Network Analysis with R. [online] Jesse Sadler. Available at: https://www.jessesadler.com/post/network-analysis-with-r/ [Accessed 16 Dec. 2019].