Group Project Specification

300958 Social Web Analytics

Due Date: Friday of Week 13

1 Aim

The Group Project provides us with a chance to analyse data from Twitter as a Social Web Application using knowledge from this unit and a computer based statistical package. For this project, we will focus on identifying a chosen Public Figure's Twitter image.

2 Method

To complete this project:

1. Read through this specification.

2. Form a group and register the members in your group using the Project Groups section of vUWS.

3. Choose a famous person who is active on Twitter, check that the person is not already on the list of Group Project Twitter Handles . Then submit the Twitter handle of the person using the above link. It is It is your responsibility to ensure that your group chooses a person who is not already in the list. If you do so, the group with the later time stamp will be asked to find a new person.

4. Complete the data analysis required by the specification.

5. Write up your analysis using your favourite word processing/typesetting program, making sure that all of the working is shown and presented well. Include the necessary R code along with its output in your assignment.

6. Include the student declaration text on the front page of your report. Please make sure that the names and student numbers of each group member are clearly displayed on the front page. If a group member did not contribute to any part of the project, state their contribution as 0% (no contribution means 0 mark).

7. Submit the report as a PDF by the due date using the Submit Group Project link

8. All code and the outputs must be shown in the project, also include comments in the code to explain what you tried to do. Put all the code in the text (not to the Appendix). Any submissions other than a PDF file will not be marked.

3 Group Size and Organisation

Students in groups of size 3 or 4 are to work together to complete this project. One project report is to be submitted per group. Each group must be formed by signing-up to a group within the Project section of 300958 in vUWS. 0 marks will be awarded to lone submissions. Groups must be formed by week 7. Once the group is formed, one person should be nominated within the group to be responsible for submitting the report.

4 Due date and Submission

The project report is due by 11:59 p.m. on the Friday (January 29th) of Week 13. The report must be submitted as a PDF file using the assignment submission facilities in the Project section of 300958 in vUWS. Only one student from each group needs to submit the assignment.

5 Report Format

Once the required analysis is performed by the group, the members of the group are to write up the analysis as a report. Remember that the assessor will only see the groups' report. Therefore, the report should contain clear and concise description of the procedures carried out, comments on the code, explanations of what you tried to do, the analysis of results and any conclusions reached from the analysis.

The required analysis in this specification covers the material presented in lectures and labs. Students should use the computer software R to carry out the required analysis and then present the results from the analysis in the report.

6 Marks

This project is worth 30% of your final grade. The project consists of four investigations and will be marked using the following criteria:

Marks	**Criteria Satisfied**
8	First section completed correctly.
7	Second section completed correctly.
5	Third section completed correctly.

8	Fourth section completed correctly.
2	Marks allocated for presentation (based on the report formatting, style, grammar, clarity and mathematical notation).

If a report is submitted late, the maximum mark it can achieve will be reduced by 10% per day.

7 Declaration

The following declaration must be included in a clearly visible and readable place on the first page of the report.

"Names and Student IDs of all group members who contributed the project"

Student Name	Student Number	Contribution(%)

By including this statement, we the authors of this work, verify that:

- We hold a copy of this assignment that we can produce if the original is lost or damaged.
- We hereby certify that no part of this assignment/product has been copied from any other student's work or from any other source except where due acknowledgement is made in the assignment.
- No part of this assignment/product has been written/produced for us by another person except where such collaboration has been authorised by the subject lecturer/tutor concerned.
- We are aware that this work may be reproduced and submitted to plagiarism detection software programs for the purpose of detecting possible plagiarism (which may retain a copy on its database for future plagiarism checking).
- We hereby certify that we have read and understand what the School of Computing and Mathematics defines as minor and substantial breaches of misconduct as outlined in the learning guide for this unit.

Note: An examiner or lecturer/tutor has the right not to mark this project report if the above declaration has not been added to the cover of the report.

8 Project Description

A well-known public figure is investigating their public image and has approached your team to identify how the public associates with them. They want five pieces of analysis to be performed.

8.1 Analysis of Twitter language about the Public Figure

In this section, we want to examine the language used in tweets. Use the **rtweet** package to download tweets.

1. Use the **search_tweets** function from the rtweet library to search for 1000 tweets about the person you selected. Save these tweets as "*tweets*". The data in this set should be the same for all members in a group. Hence only one member should download the tweets and then save it in an RData file to be shared with your group members.
   For more information on how to save your objects see:
   https://stackoverflow.com/questions/19967478/how-to-save-data-file-into-rdata
   (https://stackoverflow.com/questions/19967478/how-to-save-data-file-into-rdata) .
2. Clean and pre-process the tweet text data in tweets.
3. Display the first two tweets before and after the cleaning/processing.
4. Use the text in the tweets as the source for the matrix to construct a document-term matrix referenced by the variable "*tweets.dtm*". Use TFIDF weights to find weighted document term matrix and weighted term document matrix. Store this in variable "*tweets.wdtm*".
5. Find how many documents were empty following the cleaning/processing.
6. Find all unique words in your document collection. Store them in a variable "words". Find the sum of TFIDF weights of these unique words. Store them in a variable "*wordsWeight*". Display the first 6 and last 6 unit words in your data set.
7. Find and display the top 100 words based on TFIDF weights.
8. Get a world cloud of the top 100 words.
9. Draw a bar plot of the top 100 words.
10. Find all terms in your weighted document term matrix with a minimum weight of 10 (or any other appropriate value).
11. Use cosine distance method to get a distance matrix between terms.
12. Create a dendrogram of the words identified by question 8.10. Try simple and complete linkage clustering.
13. What do these words tell us about the person? Comment on what people are saying about this person.

8.2 Clustering the Users Who Posted Tweets About the Public Figure

We want to categorize (cluster) the users of the tweets about the Public Figure based on the descriptions provided in their Twitter account. Descriptions in the users' Twitter profiles give a short piece of information about the Twitter handle. To cluster users, build a document term matrix by using the user descriptions of the tweets (note, not tweets themselves) you downloaded in section 8.1.

1. Use rtweet package's **users_data**(tweets) function to extract users' data from tweets data object you downloaded in Section 8.1. Store the unique author result in variable, *authors*.
2. Clean the data by pre-processing and then create a weighted Term Document Matrix using unique users' descriptions.
3. Compute the appropriate number of clusters using the elbow method. Use cosine distance.
4. Cluster the users and visualize the clusters in two dimensional vector space.
5. Display the count of users in each cluster
6. List a maximum of 10 screen names of users in each cluster.
7. List the top 10 words in each cluster
8. Display the description of the first five users in each cluster.
9. Comment on your findings.

8.3 Tweet Length Analysis of the User Clusters

We want to examine if the length of tweets is dependent on the user clusters you found in Section 8.2.

14. Find the tweet lengths of the tweets of the users in each cluster with respect to the data in *tweets*.
15. Find how many tweets are >= 100 characters in length and how many are below, in each cluster
16. Construct a 2×M table where M is the number of user clusters you found at Section 2. Each row (2 rows in total) should represent the total number of tweets with length >=100 and those below 100 in each cluster. For example, if M is 3, your data structure should be as follows:

	Cluster 1	Cluster 2	Cluster 3
Count of tweets with length >=100			
Count of tweets with length <100			

17. Is the length of tweets independent of user groups? Perform an appropriate test to answer this question.
18. Interpret your results in context.

8.4 Network of the Tweets

In this section, we want to examine the network of the tweets about the chosen person.

1. Document – Document similarity can be computed by multiplying Document Term Matrix (D) with its transpose as $S=DD^T$. Compute the tweet similarity matrix S from the tweets you downloaded at section 8.1. Use TFIDF weighting obtained from frequencies in your document term matrix. Note that the document similarity matrix will be symmetrical about the diagonal axis. An illustration of S is shown below:

	Doc1	**Doc2**	**Doc3**	**Doc4**
Doc1	Distance $_{11}$	Distance $_{12}$	Distance $_{13}$	Distance $_{14}$
Doc2	Distance $_{21}$	Distance $_{22}$	Distance $_{23}$	Distance $_{24}$
Doc3	Distance $_{31}$	Distance $_{32}$	Distance $_{33}$	Distance $_{34}$
Doc4	Distance $_{41}$	Distance $_{42}$	Distance $_{43}$	Distance $_{44}$

2. Construct a Data Frame to convert either the top triangle or the bottom triangle in S into an edge list as illustrated below. For example, distance$_{12}$ represents the distance between document 1 and 2.

Distance	**Row Number**	**Column Number**
Distance $_{12}$	1	2
Distance $_{13}$	1	3
Distance $_{14}$	1	4
Distance $_{23}$	2	3
Distance $_{24}$	2	4
Distance $_{34}$	3	4

Hint: https://stackoverflow.com/questions/31591546/sorting-the-output-of-dist

3. List the top 10 pairs of similar tweets based on the above distances
4. Find the corresponding usernames of the users involved in the top 10 pairs of similar tweets.
5. Use **graph.data.frame** function in igraph library to create a graph using the edge list data frame. Use the **set_edge_attr** function to set the weight of each edge to corresponding the distances between the tweets.
6. Find the sub-graph in which the degree of each node is greater than 2.
7. Plot the sub-graph. In case it consists of disconnected sub-graphs, use the **decompose.graph** function to plot each sub-graph.
8. Comment on your findings.

Note: The person wants the above analysis to be written up as a professional report. Ensure the report maintains the section numbers indicated in this document. Include only the relevant piece of code along with its output in the body of your assignment. Do not dump large chunks of output if it adds no value to your report.