Assignment 1 – Data Mining and Exploratory Data Analysis in R

Introduction:

In this assignment you will be using the "YouTube" dataset[1]. The purpose of this assignment is to introduce the basics of R while practising two key components of the CRISP-DM model you have learned: pre-processing and data exploration.

Guidelines:

Points will be taken off for not adhering to the following guidelines:

- Use the code file from moodle. **Do not remove comments from it!**
- For true/false or multiple choice questions – write only your selection (e.g.: "ANSWER: false")
- Questions will be answered in the forum ONLY.
- Points appear next to each question in brackets and provide an indication of the question's difficulty (so if you get stuck try to move on). Your **final grade**, however, will also be affected by the readability of your code, so make sure to make it as concise as possible and to use meaningful names for variables.
- Use Google – R is completely open-source so lots of info should be available online.

Submission:

Deadline for submission: 5.12.15 at midnight.
Teams: 2 people
Submission: R file uploaded to Moodle by **one** of the team members. Make sure to insert all IDs at the top of your code file.
Score: 1-100. A script that will fail execution will receive 0 points.
TA email: alonale1@mail.tau.ac.il

Useful Tips:

- You will likely need to install many packages along the way. In order to do so execute `install.packages` (look it up on google).
- If you want to know more about a certain function – execute the function with "?" before it, e.g.: `?nameOfFunction`
  At the bottom of the help page that will open you can usually find very good examples which you can copy and execute yourself and play around with. If the function's package is not installed – you won't see a help page about it, in which case you're better off searching google for it.

GOOD LUCK!!

[1] Taken from: http://netsg.cs.sfu.ca/youtubedata/

Loading the data:

1.a. Download and extract the data from moodle into a local folder designated for this assignment.

1.b. (2) Set your working directory to be your assignment folder for easy access.

1.c. (4) Import the text file 2.txt into R and save it by the name 'data'. This is slightly tricky so here's a hint – make sure you're using relevant input parameters.

1.d. (2) Use a command that shows you the first few rows of the data. Check that the function you used above read the file appropriately.

Pre-processing the data:

2.a. (3) As you can see the names of the variables are "V1", "V2", etc. and are not very informative. Rename the first 9 columns to the following (use this code):

```
c('Video_ID', 'uploader', 'age', 'category', 'length', 'views', 'rate', 'ratings', 'comments')
```

2.b. (4) Remove from the dataset columns 10 (inclusive) and above as we will not be using them (notice there is a long way and a short way of doing this – 2 points off for long).

2.c. (2) Write a function that counts the number of rows that contain missing values.

2.d. (2) According to the strategies you've learned - why does it make sense to remove these rows? (no code)

2.e. (2) Use a function to show the list of categories that exist - notice there's an empty category "" (we'll return to it later).

2.f (2) Remove the rows that contain missing values.

2.g (5) If you check category levels again you will see all categories including the empty "" category still exist (even though it's no longer in our data). Find a way to update this change so that the next time you run 'levels' on the category column you will see that "" disappeared.

Exploring the data

3.a. (5) Display a bar chart plotting the number of uploads per category. Display the category names **vertically**.

3.b. (5) Use a function to show the values above each bar.

3.c. Part of Google's revenue comes from advertisements on YouTube. Google wants to provide uploaders with guidelines of how to make a good video so that ratings are higher, possibly leading to more views and therefore more $$$ for Google. It is likely that different categories will have different guidelines, so as a pilot Google wants to choose only one category to start with.

> 3.c.i (4) To get a feel for which categories are doing well and which not – plot the density of the ratings by category colored by category.
>
> 3.c.ii (2) A more informative plot is the boxplot. Run a function that shows the boxplot of rating by category, colored by category.

3.d (6) Based solely on the boxplot above choose the right answer: if Google wants to provide potential uploaders with guidelines in the aim of increasing the number of videos that get a high rating they should:

A. Focus on Travel & Events category because its variance is very small.
B. Focus on the Science & Technology category as more than 22% of the videos have a rating below 1.
C. Focus on the Music category because there are too many outliers in the first quartile.

3.e (1) Use your code from 3.c.ii to show the boxplot of **views** by category, colored by category. As you will see there's not much we can understand due to the outliers.

3.f. (5) To make the plot clearer use the boxplot method to detect the samples that are outliers.

3.g. (4) Plot the boxplot again using the same code from 3.e. but this time the input data will be without the outlier rows.

3.h. (2) Comparing this graph to the barplot (of number of uploads per category from the beginning of the exercise) answer the following:
T/F - there is a mismatch between the 3 most uploaded categories and the 3 categories with highest median number of views (i.e. people are uploading a lot of videos to categories that are generally less viewed)

4.a. (6) If it's more views we're after - one of the recommendations for creating a highly viewed video can be related to the length of the video. Fill in the following code where appropriate. You are requested to fill in a vector of correlation values: each index in the vector represents one category and its value is the correlation coefficient between views and length for that category:

```
catEnumerated <- levels(data$category)

corVec <- 0

for (FILL IN) {

  catLength <- data[FILL IN]

  catViews <- data[FILL IN]

  corVec[i]<- FILL IN

  print(catEnumerated[i])

  print(corVec[i])

}
```

4.b (3) Find a way (using R code) to display the name of the category for which the correlation between views and length is maximal.

4.c. Transformations and correlation:

> # 4.c.i (3) (T/F (true or false) - The correlation coefficient of two variables is affected by normalizing them first to 0 mean and unit variance. Prove your answer with code showing the before and after (you can use catLength and catViews from 3.e).
>
> # 4.c.ii (3) T/F - The correlation coefficient of two variables is affected by log-scaling them first. Prove your answer with code by showing the before and after (you can use catLength and catViews from 3.e).

4.c.iii (3) T/F – correlation cannot be affected by outliers. 5 bonus points for proving with code on any data.

5.a. (3) Let's take a look at how much Google profits from YouTube videos versus how profitable it is for uploaders. In order to do so let's first run the following lines of code - explain in one short sentence what this **first** and **third** lines of code do:

```
subsetOfData = data[(data$category == "Music") | (data$category == "Gaming") | (data$category == "Sports"),]

levels(subsetOfData$category)

subsetOfData$category<-factor(subsetOfData$category)

levels(subsetOfData$category)
```

5.b. (6) Use a function to partition the length of the videos into 3 categories: short (up to 60 seconds), medium (between 60 seconds and 15 minutes) and long (above 15 minutes) and name them "short", "medium", "long".

5.c. (2) Use the function from above to partition the number of views into 2 categories: <=10000 and >10000 and name them “<=10000” and “>10000”.

5.d. (3) Run a mosaicplot displaying the relation between category, length category and view category (the two you created above) with data = subsetOfData provided in 5.a.

5.e. (2) Show that the mean number of videos per uploader is close to 1.

5.d. Answer the following based on the mosaicplot and the fact that people usually upload only one video as seen in 5.e.:

5.d.i (2) Assuming it takes a minimum of around 10,000 views per video to gain enough ad views before Google starts paying you, which category, length cut and view cut are the most profitable for Google? Your answer should be in the following format: (category, length cut, view cut)

5.d.ii (2) T/F - In the music category when comparing short and medium length videos uploaders who want to earn money form ads are generally better off creating short videos (if you think about it your answer should not only match the mosaicplot but it should also make some sense).

A final thought - following these results, does Google run a good business model with YouTube? Is it wise to think you can make a solid earning out of uploading? (answer these to yourself).