Weekly assignment 4:

Collecting Data from APIs and Bayes

For this week's assignment, write your answers, code, explanation, and interpretation of the results in R Markdown. Upload the final document as a PDF file to *Canvas* on or before 9.00 a.m. in the morning of next Wednesday (April 26, 2023).

Applied - Collecting Data using APIs

In the following exercises you are asked to collect and plot data on the prices of the Ethereum blockchain.

[Q1] Use the API of CoinGecko[1] to download the historical USD prices of Ether (the cryptocurrency that powers the Ethereum network) for the past 365 days. You should have a data frame containing one column that indicates the date and a second column indicating the price for that date.[2] (1 point)

[Q2] Generate a line chart that shows how the price of Ether evolves over time. Take the following actions when creating the plot: (1) Only keep prices for the dates ranging from ~~April 8, 2022~~ May 8, 2022[3] up to February 8, 2023, (2) Make the line blue, (3) Insert a vertical black dotted line on the date September 15, 2022, (4) Name the x-axis "Prices (USD)" and the y-axis "Date". (1.5 point)

[Q3] The Ethereum merge was a major upgrade to the Ethereum network. This upgrade was intended to improve the network's scalability, energy efficiency, and security. The Ethereum merge took place on September 15, 2022. By looking at the chart can we reliably tell if the Ethereum merge

[1] You find documentation of the API at https://www.coingecko.com/en/api/documentation. It allows you to test the API directly on this webpage. We recommend finding the correct API call on this webpage, and then in a second step use the code from the tutorial to make the API call within R and download the data into R.

[2] The dates are returned in a strange format. Bring them into the correct one using the following command as.POSIXlt(xxxx/1e3, tz="GMT", origin="1970-01-01"). Replace "xxxx" with the column referring to the date.

[3] This was a mistake in the initial assignment.

affected the price of Ether? Explain your answer. What other analyses could you conduct to determine this? (1 point)

Applied - Bayesian Classification

The following exercises are based on the data set data_tripadvisor.csv that you can find on Canvas. It is a subset of a much larger dataset that contains TripAdvisor reviews of hotels. We will use a subset of the data consisting of 5000 reviews. The data includes the variables review_body, which includes the written review of a user and helpful_votes which displays the number of helpful votes that this review received.

[Q4] Load the data. Construct a new variable that indicates whether a review is helpful or not; a review is considered helpful if it has received at least one helpful vote. (0.5 point)

[Q5] Use string manipulation as explained in the tutorial to first convert the contents of column review_body to lower case and then remove any leading or trailing spaces from them. Afterwards, use the function get_nrc_sentiment from library "syuzhet" to extract the sentiment of every observation using the contents of column review_body.[4] This function, will return several columns. For every observation, find the difference of the columns "positive" and "negative" and insert that difference into a new column in the TripAdvisor data, named "sentiment". If the value of "sentiment" is greater than 5, then change the value to "positive", otherwise change it to "not positive". (1.5 points)

[Q6] Calculate the conditional probability $P(hotel \mid positive)$ and $P(beach \mid positive)$ where *hotel* indicates that a review contains the word "hotel", *beach* indicates that a review contains the word "beach", and *positive* indicates that a review has the value "positive" in column "sentiment". (0.5 point)

[Q7] Using Bayes' theorem, calculate the probability that a review containing the word "hotel", but not the word "beach", has positive sentiment (i.e., has value "positive" in column "sentiment"). Also calculate the probability that a review containing the word "beach", but not the word "hotel", has positive sentiment. Show all of your calculation steps. You can assume strict independence (page 62 in the session slides) in your calculations. (1 points)

[4] This may take a few minutes computing time. The tutorial gives some advice on how to speed this up using parallel processing. Still, you might want to not do this computation every time you compile your Markdown document. Instead, you can do the computation once and store its outcome in a file that you load into your Markdown document. In this case, leave the code to do the computation in your Markdown document but comment it out so it is not run every time you compile it but we can still verify that it is correct.

[Q8] Create nine separate variables that indicate whether the text of a review includes the strings: "hotel", "staff", "you", "breakfast", "room", "day", "clean", "noise", and "weather". Do a co-occurrence analysis using the *arules* and *arulesViz* packages in R for helpful reviews; these are reviews that have the value "helpful" in column helpful. Focus your analysis on rules containing only two separate words and that have a minimum support of 0.2 and a minimum confidence of 0.1. Output the ten learned rules with the highest lift value and plot the scatter plot of confidence and the graph plot containing all rules. (1.5 points)

(Hint: Before using the *apriori* command you have to convert your data frame into an object of class *transactions* using the command *as(. . . , transactions)*. Only the columns of your data frame containing the variables indicating which strings occur in a review should be passed to this function.).[5]

[Q9] Run a Naive Bayes model predicting whether a review is helpful or not. Use as attributes the nine variables created in the previous question plus the variable "sentiment". Use a random split of the data into a train-set of size 70% of the original data and a test-set containing the remaining 30% of the original data. Additionally, also estimate a Random Forest model using the same set-up. Compare the performance of both models on the test set using accuracy and a plot of the ROC-curves. Which of the two models performs better? (1.5 points)

[5] In case you experience errors when using the inspect() function, try using arules::inspect() instead.