DSCI 6607 – Fall 2024

Assignment 4*

Question 1

Consider `sales_data.csv` with the following columns: [**20 points**]

a. Load the dataset in python and ensure that the `Date` column is converted to a datetime format.

b. Add a new column `Profit` where: If the `Region` is "North", the profit is `Revenue * 0.3`. If the `Region` is "South", the profit is `Revenue * 0.4`. For all other regions, the profit is `Revenue * 0.25`.

c. For each Product, calculate the total units sold and the average profit per unit sold. Store this information in a new DataFrame.

d. Filter the original dataset to include only rows where: The Date falls in the year 2023. The Revenue is above the median revenue for all rows.

e. For the filtered dataset: Group the data by Region and calculate the total Profit and Units Sold for each region. Sort the grouped data in descending order of total Profit.

f. Export the grouped data to a new CSV file called `region_summary.csv`.

```
sales_data = pd.read_csv('sales_data.csv')
sales_data.head()
```

Question 2

You are provided with a dataset in R that contains information about monthly sales, categorized by product and region, in a messy format. [**20 points**]

a. Separate the `Product.Region` column into two new columns: `Product` and `Region`.

b. Transform the dataset so that each `Product` becomes a column, and the `Sales` values are summarized by `Month` and `Region` (reshape the data from long to wide format).

c. Calculate an additional column for each row in the final dataset, representing the total sales across all products for the combination of `Month` and `Region`.

d. Filter the dataset to include only rows where the total sales exceed the average total sales across all rows.

e. Sort the resulting dataset by `Month` and `Region`, and save it to a CSV file called `tidy_sales_data.csv`.

```
messy_sales_data <- read_csv('messy_sales_data.csv')
head(messy_sales_data)
```

Question 3

Load the `movies.csv` from your directory in R. For data set, see the lecture notes. [**20 points**]

a. Plot the side-by-side histograms of the movie scores for the top three genres.

b. Plot the side-by-side boxplots of the movie scores for the top three genres.

```
movies_data <- read_csv('movies.csv')
head(movies_data)
```

Question 4

Download the following data:

```
download.file("https://raw.githubusercontent.com/biocorecrg/CRG_RIntroduction/master/ex12_normalized_in
```

1. Read file into object `intenseData`.
2. Using ggplot, create a simple scatter plot representing gene expression of `sampleB` on the x-axis and ‘`sampleH` on the y-axis.
3. Add a column to the data frame `intenseData` (call this column `expr_limits`), that will be filled the following way: if the expression of a gene is > 13 in both `sampleB` and `sampleH`, set to the value in `expr_limits` to `high` if the expression of a gene is < 6 in both `sampleB` and `sampleH`, set it to `low` if different, set it to `normal`.
4. Color the points of the scatter plot according to the newly created column `expr_limits`. Save that plot in the object `p`.
5. Add a layer to `p` in order to change the points colors to blue (for low), grey (for normal) and red (for high). Save this plot in the object `p2`.
6. Save p2 in a pdf file. [**20 points**]

Due on Tuesday, November 26, by 11 pm

Have fun!