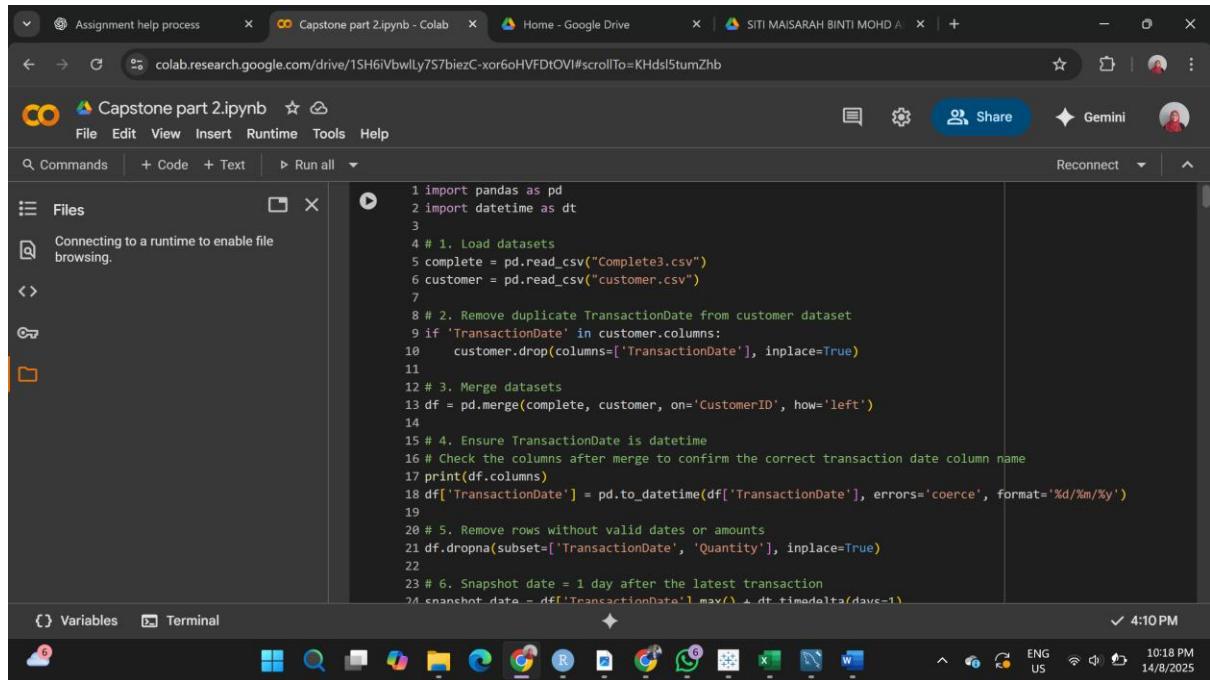


# Customer Retention and Sales Optimization in Retail

## Part 2 - BI Dashboard, Data Science, & R Programming

### 1. Data cleaning and transformation

#### a) Data cleaning

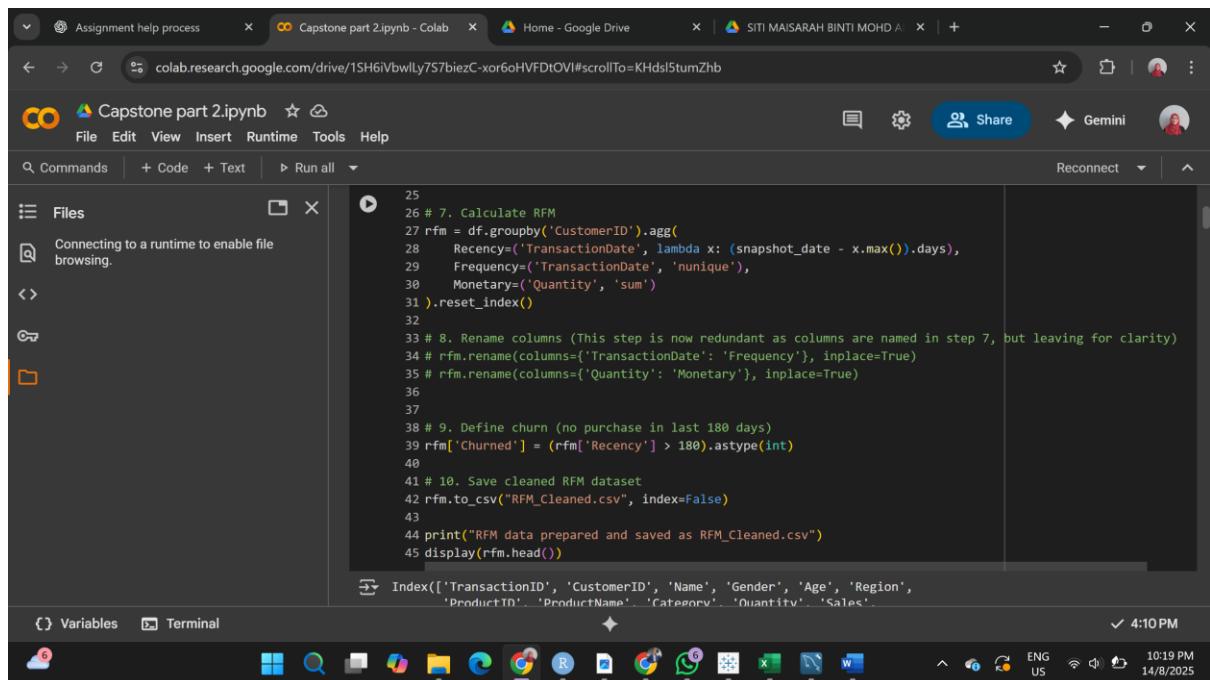


The screenshot shows a Google Colab notebook titled "Capstone part 2.ipynb". The code cell contains the following Python script:

```
1 import pandas as pd
2 import datetime as dt
3
4 # 1. Load datasets
5 complete = pd.read_csv("Complete3.csv")
6 customer = pd.read_csv("customer.csv")
7
8 # 2. Remove duplicate TransactionDate from customer dataset
9 if 'TransactionDate' in customer.columns:
10     customer.drop(columns=['TransactionDate'], inplace=True)
11
12 # 3. Merge datasets
13 df = pd.merge(complete, customer, on='CustomerID', how='left')
14
15 # 4. Ensure TransactionDate is datetime
16 # Check the columns after merge to confirm the correct transaction date column name
17 print(df.columns)
18 df['TransactionDate'] = pd.to_datetime(df['TransactionDate'], errors='coerce', format='%d/%m/%y')
19
20 # 5. Remove rows without valid dates or amounts
21 df.dropna(subset=['TransactionDate', 'Quantity'], inplace=True)
22
23 # 6. Snapshot date = 1 day after the latest transaction
24 snapshot_date = df['TransactionDate'].max() + dt.timedelta(days=1)
```

The code cell has been run successfully at 4:10 PM. The status bar at the bottom right shows ENG US, 10:18 PM, and 14/8/2025.

#### b) Calculate RFM; Recency, Frequency, and Monetary

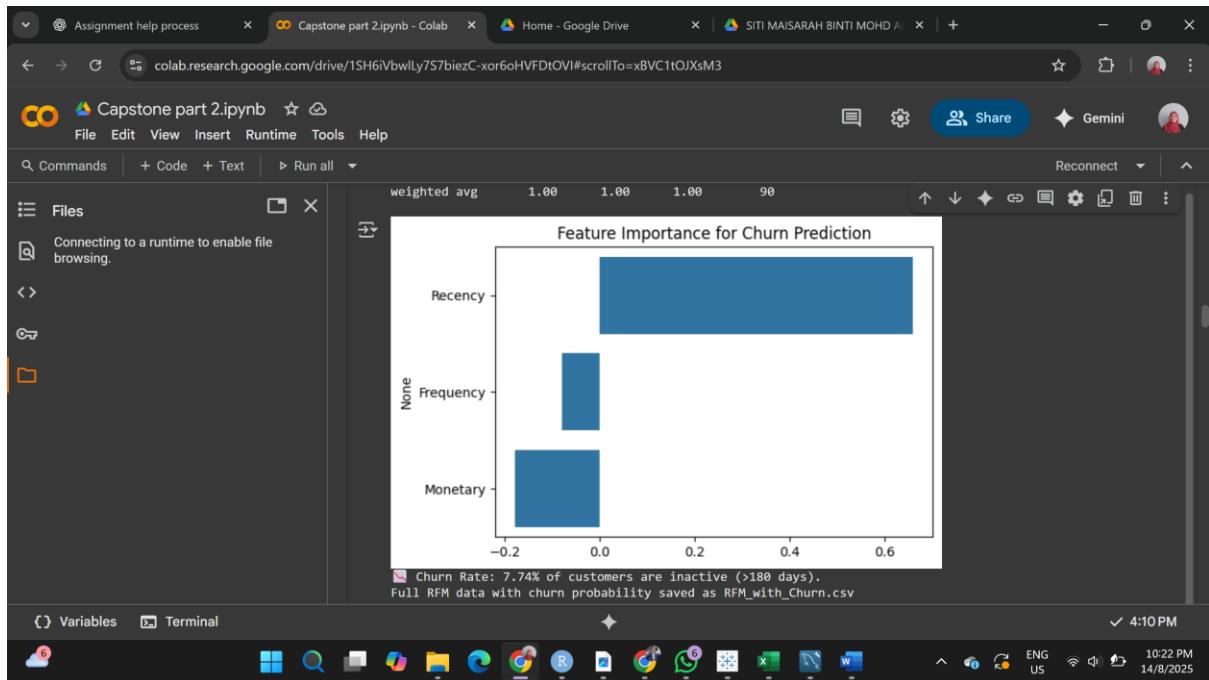


The screenshot shows a Google Colab notebook titled "Capstone part 2.ipynb". The code cell contains the following Python script:

```
25
26 # 7. Calculate RFM
27 rfm = df.groupby('CustomerID').agg(
28     Recency=('TransactionDate', lambda x: (snapshot_date - x.max()).days),
29     Frequency=('TransactionDate', 'nunique'),
30     Monetary=('Quantity', 'sum'))
31 ).reset_index()
32
33 # 8. Rename columns (This step is now redundant as columns are named in step 7, but leaving for clarity)
34 # rfm.rename(columns={'TransactionDate': 'Frequency'}, inplace=True)
35 # rfm.rename(columns={'Quantity': 'Monetary'}, inplace=True)
36
37
38 # 9. Define churn (no purchase in last 180 days)
39 rfm['Churned'] = (rfm['Recency'] > 180).astype(int)
40
41 # 10. Save cleaned RFM dataset
42 rfm.to_csv("RFM_Cleaned.csv", index=False)
43
44 print("RFM data prepared and saved as RFM_Cleaned.csv")
45 display(rfm.head())
46
```

The code cell has been run successfully at 4:10 PM. A tooltip at the bottom indicates the index of the columns: "Index(['TransactionID', 'CustomerID', 'Name', 'Gender', 'Age', 'Region', 'ProductID', 'ProductName', 'Category', 'Quantity', 'Sales'])". The status bar at the bottom right shows ENG US, 10:19 PM, and 14/8/2025.

c) Feature importance for churn prediction visualisation using barplot



2. Data science – R language

a) Chi-squared test result

### Pearson's Chi-squared test

```
data: table_data
X-squared = 0.59575, df = 5, p-value = 0.9882
```

The Chi-squared test result shows that there is no significant differences between male and female in preferred product category as the p-value (0.99) higher than 0.05.

b) ANOVA test result

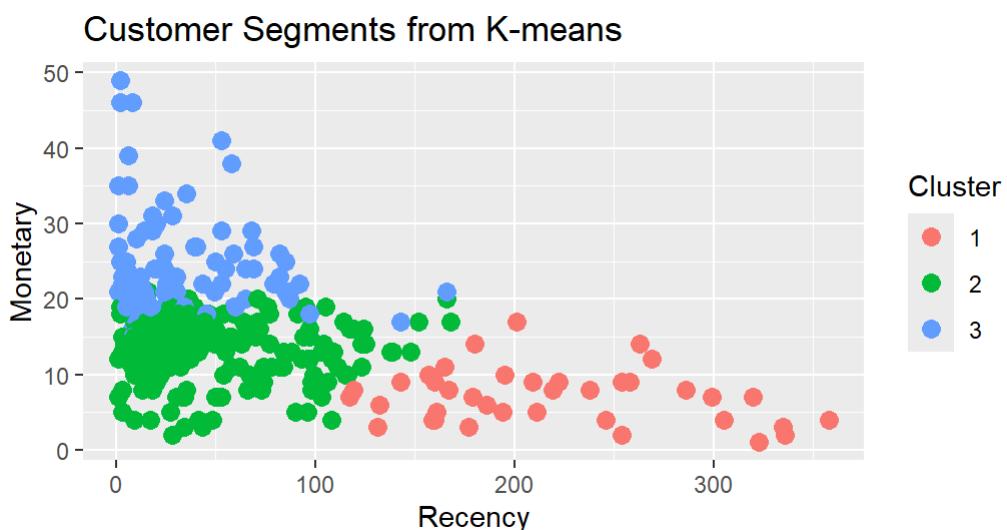
```
> anova_result <- aov(Monetary ~ Region, data = merged_data)
> summary(anova_result)
      Df Sum Sq Mean Sq F value    Pr(>F)
Region       3   1754   584.7   8.092 2.39e-05 ***
Residuals 1496 108082     72.2
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
>
> TukeyHSD(anova_result)
Tukey multiple comparisons of means
 95% family-wise confidence level

Fit: aov(formula = Monetary ~ Region, data = merged_data)

$Region
        diff      lwr      upr      p adj
North-East -0.2805521 -1.9117210 1.3506167 0.9710824
South-East -2.8835016 -4.5273787 -1.2396245 0.00000411
West-East   -1.4058425 -2.9026698 0.0909849 0.0745775
South-North -2.6029494 -4.3273432 -0.8785556 0.0006241
West-North  -1.1252903 -2.7101228 0.4595422 0.2613520
West-South   1.4776591 -0.1202502 3.0755685 0.0816908
```

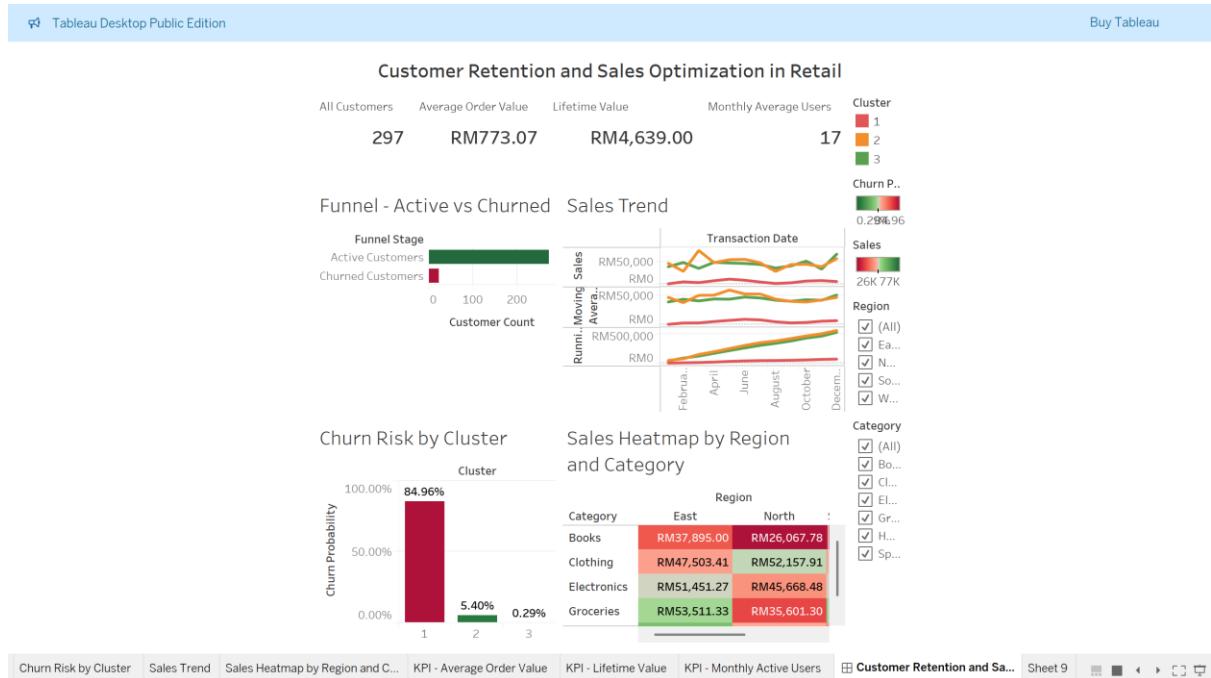
For the ANOVA result, it shows a very high significant differences for monetary value across the regions. TukeyHSD gives more in-depth result for the differences between two regions. It is found that South is significantly lower than East and North. Other than that, there is no significant difference.

c) Clustering using K-means



For cluster 1 with red color, it indicates high recency and low monetary. Most likely churned customers. For cluster 2 with green color shows moderately on both recency and monetary. For the blue one, cluster 3 is the loyal and high-value customers.

### 3. Visualisation using Tableau



The dashboard shows two charts focusing on churn and two charts on sales. It is also equipped by total customers, average order value, lifetime value, and monthly average users as its KPI. Region and product category are used as filters.