Mestrado em Data Science and Advanced Analytics, com especialização em Data Science

ABCDEats Inc.

Project of Data Mining

Jan-Louis Schneider, 20240506
Marta Boavida, 20240519
Matilde Miguel, 20240549
Sofia Gomes, 20240848

Fall/Spring Semester 2024-2025

1 Introduction

In an increasingly competitive market, food delivery companies face the challenge of delivering unique and targeted experiences to diverse customer bases.

ABCDEats Inc., a fictional food delivery service, has tasked us with analyzing its customer data to uncover insights that can inform a data-driven marketing strategy. By focusing on data collected over three months from three different cities, this project aims to segment customers based on economic and behavioral perspectives.

This integrated segmentation strategy is expected to empower ABCDE to develop personalized marketing strategies that align with distinct customer segments and maximize engagement and profitability.

This project was divided into the following phases:

- Exploration of the dataset
- Relationship between features
- New features

Throughout the report there are graphs for easier and more intuitive viewing.

2 Exploration of the dataset

After importing the dataset ”DM2425_ABCDEats_DATASET.csv” and the necessary libraries, we used several functions already defined in the libraries in order to better understand the data.

To see all the columns:

```
df.columns.values
```

To see the first ten lines:

```
df.head(10)
```

To get information about the index dtype and columns, non-null values and memory usage:

```
df.info()
```

2.1 Incorrect data

After an analysis, we noticed that some variables are incorrect (the variable type is float and should be int), namely:

- **customer_age**
- **first_order**
- **HR_0**

So we converted the types using the function:

```
    df["customer_age"]= df["customer_age"].astype(’Int64’)
    df["first_order"] = df["first_order"].astype("Int64")
    df["HR_0"] = df["HR_0"].astype("Int64")
```

Another detail about incorrect data is between product_count and vendor_count. There is information about products that were not purchased but there is information about sellers who sold them.

2.2 Duplicates

We looked for duplicate values and found 0.041%, so we removed them using the function:

```
    df.drop_duplicates(inplace=True)
```

2.3 Missing values

We check for missing values and found some variables:

- Percentage of missing values in **customer_region** is: 1.387%
- Percentage of missing values in **customer_age** is: 2.281%
- Percentage of missing values in **first_order** is: 0.333%
- Percentage of missing values in **last_promo** is: 52.53%
- Percentage of missing values in **HR_0** is: 3.652%

2.4 Numerical variables

Numerical variables represent measurable quantities and can be analyzed mathematically. The numerical variables that exist in this dataset are:

- **customer_age**: 75% of customers are young, but there are a few older individuals
- **vendor_count**: Most customers ordered from few unique vendors, but there are customers with much higher count
- **product_count**: Most customers don't buy many products
- **is_chain**: relative small amount of orders made in chain restaurants
- **first_order**: On average customers place their first order 28 days after joining the app
- **last_order**: customers are buying more recently than they did earlier
- **CUI** (Overral analysis): 75% of customers spend nothing or a significant small amount but the maximum values of these cuisines are significantly high (specially for American (280) and Asian (896)) cuisines which indicate the presence of potential outliers who frequently order or spend heavily
- **DOW_0 to DOW_6** (Overral analysis): most customers only ordered 1 time in each day of the week
- **HR_0 to HR_23** (Overral analysis): No activity at midnight (HR_0). 75% of customers placed no order in HR_1 to HR_23

2.5 Categorical variables

Categorical variables represent characteristics or qualities that group data into distinct categories or labels. In this dataset, the categorical variables are:

- **customer_region**: Customers are from 8 different regions. Most customers are located in region 8670
- **last_promo**: There are promotions in 3 different categories. Most customers use promotions in the delivery category
- **payment_method**: There are 3 different payment methods used by customers. Most customers use Card as their preferred payment method

After analysing the categorical variables, we realise that isn´t strange data.

3 New features

Create new features can significantly enhance our analysis by providing additional insights and improving the performance of models.

In order to that, we create some:

1. **Customer Lifetime**: Interval of customer activity, so we have an idea of how many days the customer ordered

```
df['lifetime_days'] = df['last_order'] - df['first_order']
```

2. **Most frequent order day of the week:** Indicates the days of the week on which the customer placed the most orders: 6h-12h correspond to Morning (Breakfast), 12h-18h correspond to Afternoon (Lunch), 18h-00h correspond to Evening (Dinner) and 00h-6h correspond to Night

3. **Total monetary units spend**: Sum all total expenses

```
cuisine = df.filter(like='CUI_').columns.tolist()
df['total_expenses'] = df[cuisine].sum(axis=1)
```

4. **Average monetary units per product:** Show the average monetary of all products

```
df['avg_per_product'] =df['total_expenses'] / df['product_count'].
    replace(0, pd.NA)
```

5. **Average monetary units per order:** Show the average monetary per order

```
df['avg_per_order'] = df['total_expenses'] / df[dows].sum(axis=1).
    replace(0, pd.NA)
```

6. **Average order size:** Help identifing users who make larger orders

```
df['avg_order_size'] = df['product_count'] / df[dows].sum(axis=1).
    replace(0, pd.NA)
```

7. **Culinary profile:** A proportion of ordered cuisines. A higher number indicates more diversity of types of cuisine you ordered.

```
total_cuisine = len(cuisine)
df['culinary_variety'] = (df[cuisine].gt(0).sum(axis=1) / total_cuisine)
    * 100
```

8. **Loyalty to chain restaurants:** Proportion of orders from restaurant chains. A high value indicates that you prefer to try different restaurant chains. A lower value is only more faithful to certain chains.

```
df['chain_preference'] = df['is_chain'] / df[dows].sum(axis=1).replace
    (0, pd.NA)
```

9. **Loyalty to venders:** Proportion of orders from specific restaurants. A high value indicates that you prefer to try different restaurants. A lower tend to be more loyal to specific restaurants.

```
df['loyalty_to_venders'] = df[dows].sum(axis=1) / df['vendor_count'].
    replace(0, pd.NA)
```

4 Visualizations and relationship between features

In order to explore the relationships between the features, we looked for trends and patterns:

1. **Correlation of all numerical features:** We filter for all correlations that are greater than 0.7 and smaller than 1.0 to avoid correlations with itself, and we verify that the maximum correlation is between vendor_count and culinary_variety and the minimum correlation is between vendor_count and is_chain (see Fig 1).

2. **Visualization of total Orders placed per Hour:** The most orders placed is between 10h to 12h and 16h to 18h (see Fig 2).

3. **Visualization of total Orders placed per week day:** The most oders placed on Thursday and Saturday, and least on Sunday (see Fig 3).

4. **Percentage of each payment_method for each age_group:** DIGI is balanced, CASH is more used by older people above 50, while these use less CARD (see Fig 4).

5. **Proportions of each last_promo for each payment_method:** Delivery always has the highest proportion, but for CARD the last_promo are more balanced. For CASH, the proportion of freebie is lower than usual(see Fig 5).

6. **Means of each payment_method in vendor_count:** Card highest and cash lowest, maybe because people who try a lot different things and are more open to experimenting new restaurants also are more(see Fig 6)

7. **Means of each payment_method in lifetime_days:** Card highest and cash lowest maybe because who plans to use more frequently registers his card on website/app and first time users only use cash(see Fig 7)

8. **Means of each payment_method in total_expenses:** Total expenses highest with card, than digi and cash. Explanation could be that with cash you have better feeling for how much you spent(see Fig 8)

9. **Means of each payment_method in avg_per_product:** The difference between the average per product of CARD, CASH AND DIGI is not very big, with the one with the highest average being DIGI(see Fig 9)

10. **Means of each payment_method in avg_per_order:** The average monetary per order is greather if pay by DIGI(see Fig 10)

11. **Means of each payment_method in avg_order_size:** Who made majors order payed with DIGI, but the diferecence is minimal(see Fig 11)

12. **Means of each payment_method in culinary_variety:** Card highest, Cash lowest. Similar to vendor_count, people with cash more conservative people, less trying of new (”new” payment methods)(see Fig 12)

13. **Means of each payment_method in chain_preference:** In chain preference, the customers prefer pay in CASH, but the diference is minimal(see Fig 13)

14. **Total expenses per age group:** Low at younger age, after 23-28 more regular, probably because young people have less money(see Fig 14)

15. **Culinary variety per age group:** No big differences but peak at 23-28, maybe because people start to live on their own and try more different things(see Fig 15)

16. **Means of each last_promo in lifetime_days:** Freebie highest, delivery lowest(see Fig 16)

17. **Means of each last_promo in total_expenses:** Freebie highest, delivery lowest but more equal to discount. Freebie Discount leads to more expenses in total(see Fig 17)

18. **Means of each last_promo in avg_per_product:** In the last promo, the average per product is greather in delivery, but the diference is minimal(see Fig 18)

19. **Means of each last_promo in avg_per_order:** In the last promo, the average per order is greather in delivery, but the diference is minimal(see Fig 19)

20. **Means of each last_promo in chain_preferences:** Discount highest, people with discount promo tend to go to chains more(see Fig 20)

21. **Means of each last_promo in loyalty_to_venders:** Freebie leads to more loyality(see Fig 21)

22. **Relations between the costumer age and some types of cuisine:** We do with CUI_Asian, CUI_Desserts and CUI_Healthy per age, because are the most relevants. Asian increase by age, Dessert decrease with age and Healthy peak on 23-28(see Fig 22)

23. **Means of each total_expenses in chain_preferences:** Big spender tend to go to less different chains(see Fig 23)

24. **Proportions of each last_promo value for the two groups of people:** Big spenders use Freebie a lot more, while low spenders use delivery the most(see Fig 24)

25. **Cuisines and total_expenses:** The biggest spenders have their highest increase of spending compared to the other costumers in StreetFood/Snacks, Cafe, Asian Healthy and Desserts, while the lowest increase is in Chicken Dishes, Noodles and Indian(see Fig 25)

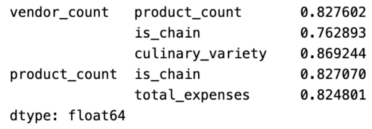
26. **Cuisines and loyalty_to_venders:** The most loyal costumers to restaurants are with Italian Cuisine, followed by Chinese and Cafe. The lowest are Desserts and Snacks where Costumers tend to choose any place without too much thought.(see Fig 26)

27. **Customer_regions:** The region 8550 speends more than the others(see Fig 27)

A Appendix

The most relevant graphics are in part 4 (Visualizations and relationships between features).

Figure 1: Correlation of all numerical features



```
vendor_count   product_count       0.827602
               is_chain            0.762893
               culinary_variety    0.869244
product_count  is_chain            0.827070
               total_expenses      0.824801
dtype: float64
```

Figure 2: Visualization of total Orders placed per Hour

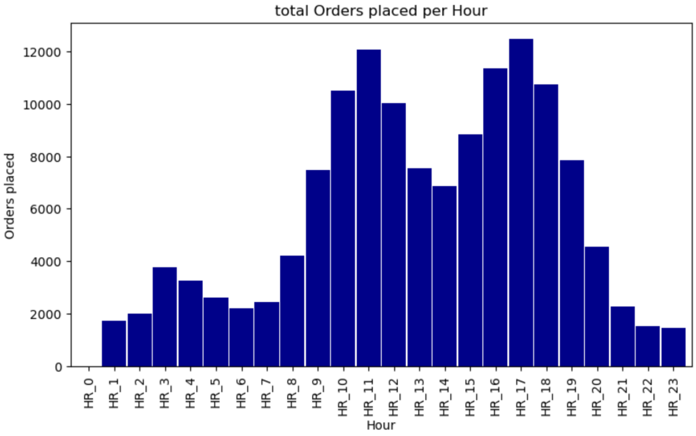


Figure 3: Visualization of total Orders placed per week day

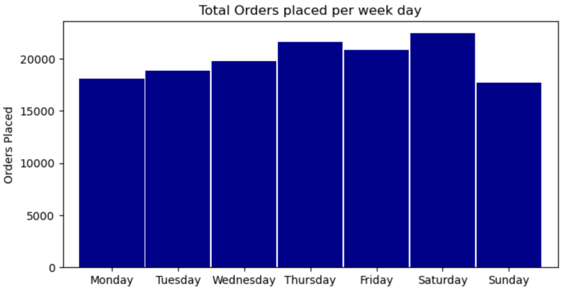


Figure 4: Percentage of each payment_method for each age_group

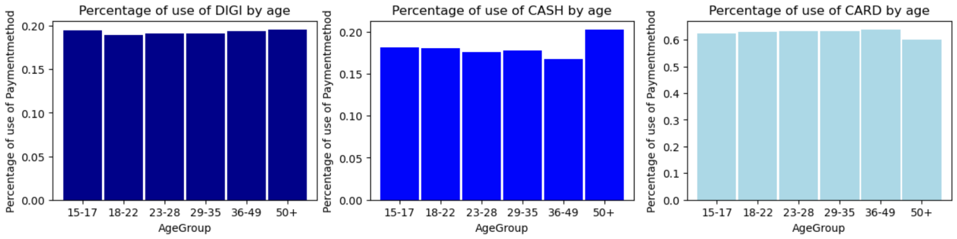


Figure 5: Proportions of each last_promo for each payment_method

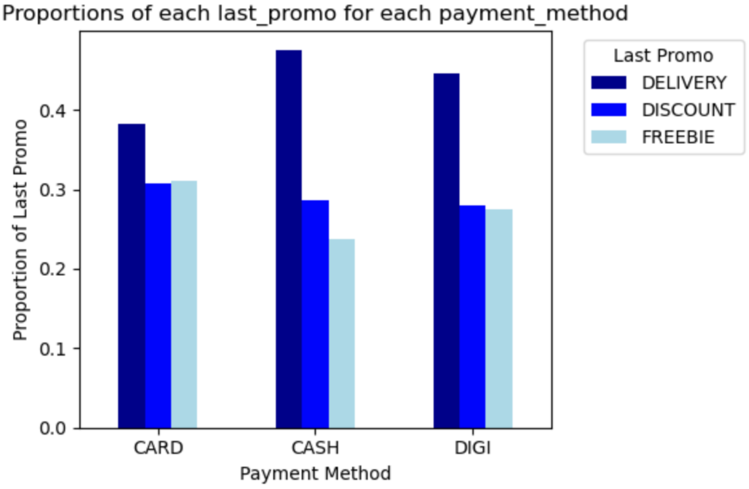
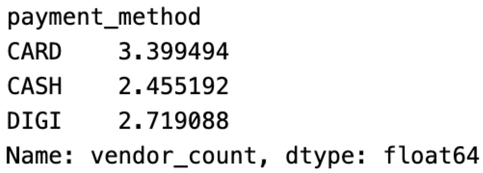
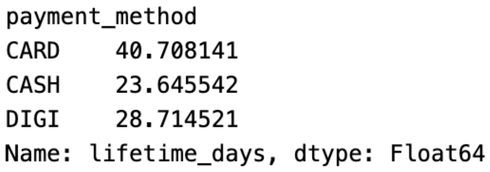


Figure 6: Means of each payment_method in vendor_count



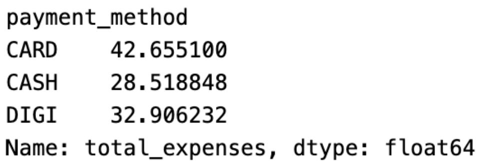
```
payment_method
CARD    3.399494
CASH    2.455192
DIGI    2.719088
Name: vendor_count, dtype: float64
```

Figure 7: Means of each payment_method in lifetime_days



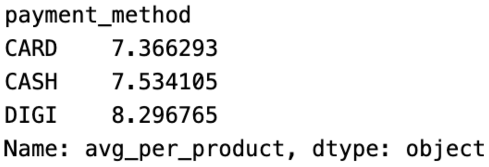
```
payment_method
CARD    40.708141
CASH    23.645542
DIGI    28.714521
Name: lifetime_days, dtype: Float64
```

Figure 8: Means of each payment_method in total_expenses



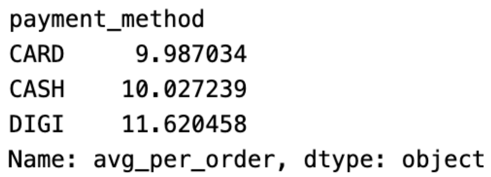
```
payment_method
CARD    42.655100
CASH    28.518848
DIGI    32.906232
Name: total_expenses, dtype: float64
```

Figure 9: Means of each payment_method in avg_per_product



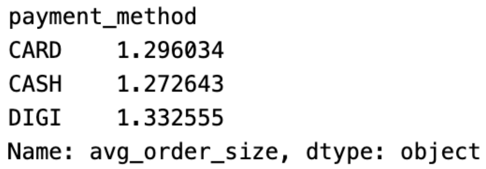
```
payment_method
CARD    7.366293
CASH    7.534105
DIGI    8.296765
Name: avg_per_product, dtype: object
```

Figure 10: Means of each payment_method in avg_per_order



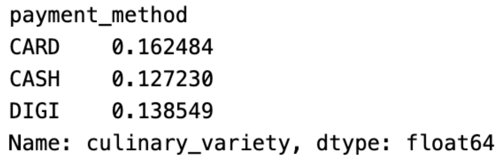
```
payment_method
CARD      9.987034
CASH     10.027239
DIGI     11.620458
Name: avg_per_order, dtype: object
```

Figure 11: Means of each payment_method in avg_order_size



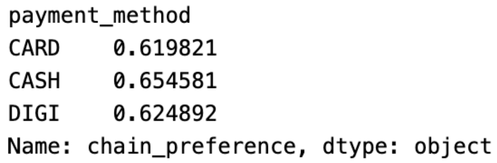
```
payment_method
CARD    1.296034
CASH    1.272643
DIGI    1.332555
Name: avg_order_size, dtype: object
```

Figure 12: Means of each payment_method in culinary_variety



```
payment_method
CARD    0.162484
CASH    0.127230
DIGI    0.138549
Name: culinary_variety, dtype: float64
```

Figure 13: Means of each payment_method in chain_preference



```
payment_method
CARD    0.619821
CASH    0.654581
DIGI    0.624892
Name: chain_preference, dtype: object
```

Figure 14: Total expenses per age group

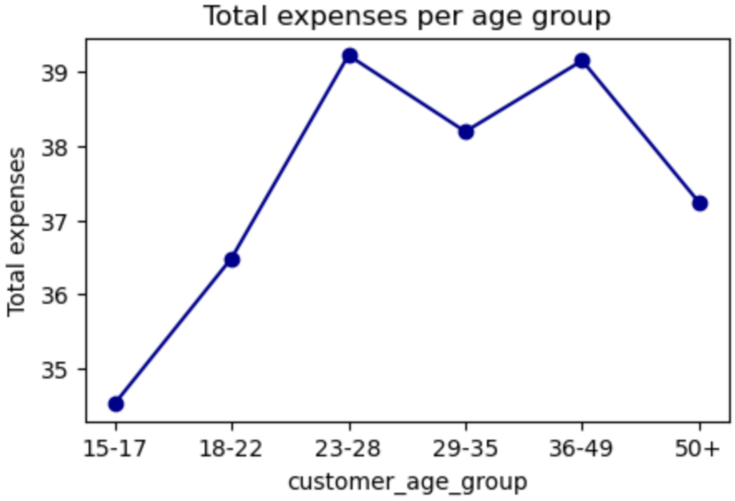


Figure 15: Culinary variety per age group

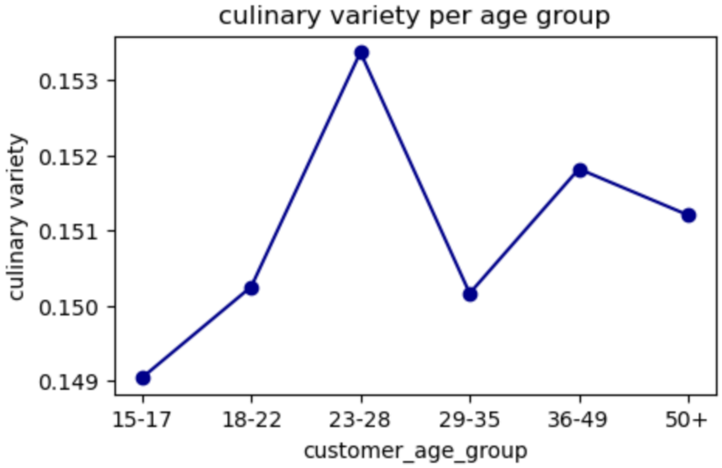
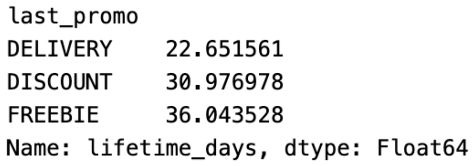
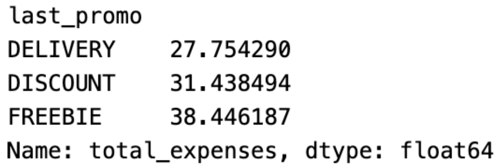


Figure 16: Means of each last_promo in lifetime_days



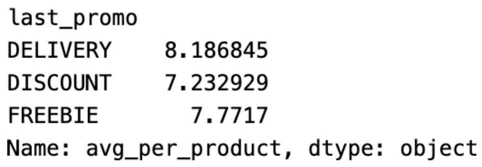
```
last_promo
DELIVERY    22.651561
DISCOUNT    30.976978
FREEBIE     36.043528
Name: lifetime_days, dtype: Float64
```

Figure 17: Means of each last_promo in total_expenses



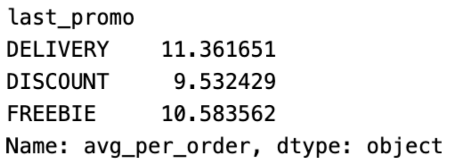
```
last_promo
DELIVERY    27.754290
DISCOUNT    31.438494
FREEBIE     38.446187
Name: total_expenses, dtype: float64
```

Figure 18: Means of each last_promo in avg_per_product



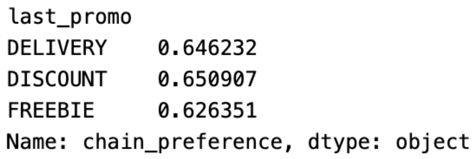
```
last_promo
DELIVERY    8.186845
DISCOUNT    7.232929
FREEBIE       7.7717
Name: avg_per_product, dtype: object
```

Figure 19: Means of each last_promo in avg_per_order



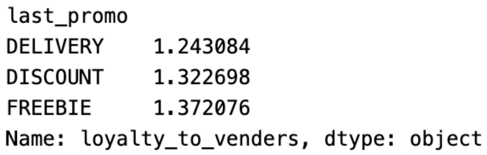
```
last_promo
DELIVERY    11.361651
DISCOUNT     9.532429
FREEBIE     10.583562
Name: avg_per_order, dtype: object
```

Figure 20: Means of each last_promo in chain_preferences



```
last_promo
DELIVERY    0.646232
DISCOUNT    0.650907
FREEBIE     0.626351
Name: chain_preference, dtype: object
```

Figure 21: Means of each last_promo in loyalty_to_venders



```
last_promo
DELIVERY    1.243084
DISCOUNT    1.322698
FREEBIE     1.372076
Name: loyalty_to_venders, dtype: object
```

Figure 22: Relations between the costumer age and some types of cuisine

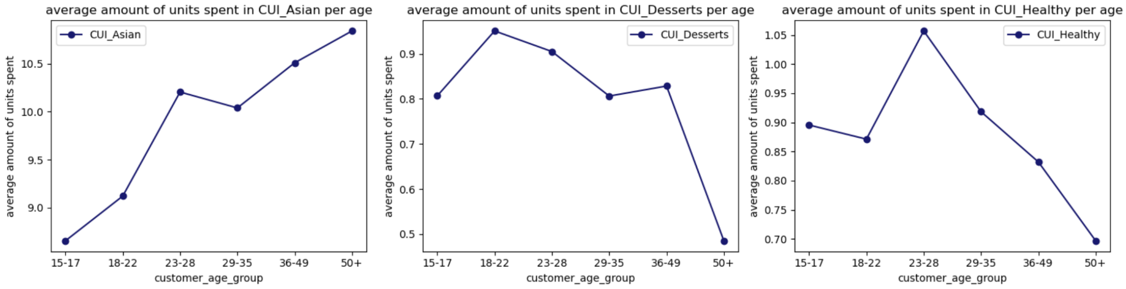
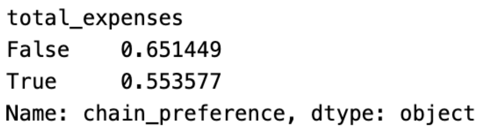
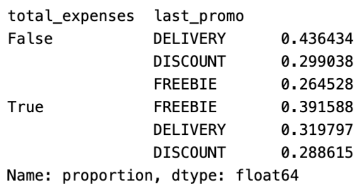


Figure 23: Means of each total_expenses in chain_preferences



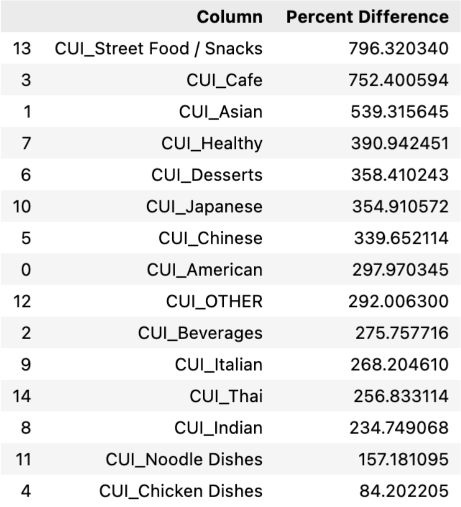
```
total_expenses
False    0.651449
True     0.553577
Name: chain_preference, dtype: object
```

Figure 24: Proportions of each last_promo value for the two groups of people



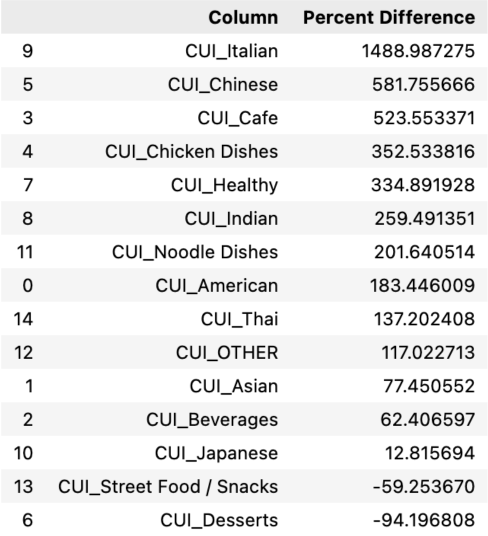
```
total_expenses  last_promo
False           DELIVERY      0.436434
                DISCOUNT      0.299038
                FREEBIE       0.264528
True            FREEBIE       0.391588
                DELIVERY      0.319797
                DISCOUNT      0.288615
Name: proportion, dtype: float64
```

Figure 25: Cuisines and total_expenses



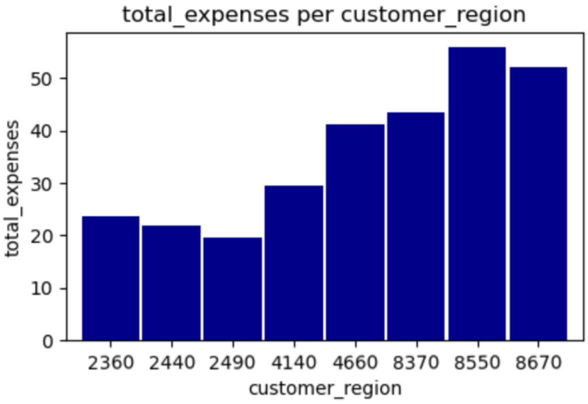
	Column	Percent Difference
13	CUI_Street Food / Snacks	796.320340
3	CUI_Cafe	752.400594
1	CUI_Asian	539.315645
7	CUI_Healthy	390.942451
6	CUI_Desserts	358.410243
10	CUI_Japanese	354.910572
5	CUI_Chinese	339.652114
0	CUI_American	297.970345
12	CUI_OTHER	292.006300
2	CUI_Beverages	275.757716
9	CUI_Italian	268.204610
14	CUI_Thai	256.833114
8	CUI_Indian	234.749068
11	CUI_Noodle Dishes	157.181095
4	CUI_Chicken Dishes	84.202205

Figure 26: Cuisines and loyalty_to_venders



	Column	Percent Difference
9	CUI_Italian	1488.987275
5	CUI_Chinese	581.755666
3	CUI_Cafe	523.553371
4	CUI_Chicken Dishes	352.533816
7	CUI_Healthy	334.891928
8	CUI_Indian	259.491351
11	CUI_Noodle Dishes	201.640514
0	CUI_American	183.446009
14	CUI_Thai	137.202408
12	CUI_OTHER	117.022713
1	CUI_Asian	77.450552
2	CUI_Beverages	62.406597
10	CUI_Japanese	12.815694
13	CUI_Street Food / Snacks	-59.253670
6	CUI_Desserts	-94.196808

Figure 27: Customer_regions


total_expenses per customer_region
total_expenses
customer_region