**1.INTRODUCTION**

A response to demand volatility is demand forecasting using Artificial Intelligence. Traditionally, demand forecasting is a form of predictive analytics, where the process of estimating customer Using demand is analysed using historical data .Typically, organisations use this form of AI to avoid inefficiencies caused by misalignment of demand and supply throughout the operational process. Honestly, this will never be 100% accurate . Yet it can offer companies the opportunity to significantly reduce supply chain costs and make improvements in financial planning, workforce planning, profit margins and risk assessment decisions.

**1.1 PROJECT OVERVIEW**

One of the biggest challenges for business executives today is demand volatility in relation to demand forecasting. Whereas data availability continues to increase, customer purchase patterns are becoming increasingly complex, and therefore harder to detect or predict There are too many factors influencing demand, ranging from weather fluctuations to posts by social media influencers, causing customers to frequently changing their minds. Traditional Forecasts are only as accurate as the data, models, resources and people that have to interpret them .

**1.2 PURPOSE**

Better forecasts will be made over time as machine learning algorithms learn from existing data. When products are ‘out of stock’, this will decrease customer satisfaction, whereas customer satisfaction will increase when products are always available. This improves customer loyalty and brand perceptionDemand forecasting can support the HR department in making efficient considerations between full-time or part-time staff mix, thus optimising HR costs and effectiveness.Cash-in-stock is a common situation for retail companies, where products remain unsold for a longer period than expected. This often causes higher expected inventory costs and the risk of products becoming obsolete and losing value. In this scenario, products are sold at lower selling prices. With demand forecasting, this scenario can be minimised

**2**. **LITERATURE** **SURVEY**

|  |  |  |
| --- | --- | --- |
| S.NO | AUTHOR | ABSTRACT |
| 1. | Claudimar Pereira Da  Veiga,cassia Rita Pereira Da veiga. | This paper concluded that the HWmodel  Performs better than ARIMA |
| 2. | Elcio Tarallo,getulio K.Akabane,Camilo I.  Shimabukuro. | This paper uses a Support vector machine to  Improve the sales management effectiveness  Of computing products that are highly replacable and subject to dramatic changes in demad. |
| 3. | N.de P.Barbosa,E.da  S.Christo,and K.A.Costa. | The forecasts are evaluated using the error measure MAPE and compared to the demand  Forecasting method. |
| 4. | Takashi Tanizaki,tomohiro Hoshinoa,Takeshi Takenaka. | In this paper ,demand forecasting in restaurants using machine learning is proposed. |
| 5. | Jakob Huber a,Alexander Gossman,  Heiner Stucknesschmidt. | It uses multivariate ARIMA model to forecast the daily demand to support operational decisions. |
| 6. | Ishtiyaq Otudi,  Abdulaziz Almaktoom | There are multiple models such as Naïve approach,moving average,weighted moving  And exponential smoothing. |
| 7. | Emily J.Filesa,Barry  W.Brooka,Linus Blomqvist. | Inded,time serious and simple income based  Model often make similar predictions to  Integrated assesments. |

**2.1 REFERENCE**

* C. Waterfield

### [Disaggregating food consumption parameters: designing targeted nutritional interventions](https://www.sciencedirect.com/science/article/pii/0306919285900259" \t "_self)

* H. Bouis et al.

### [Does it matter how we survey demand for food? Evidence from Kenya and the Philippines](https://www.sciencedirect.com/science/article/pii/0306919292900634" \t "_self)

* H. Alderman

### Estimates of consumer price response in Pakistan using market prices as data

### Pakistan Development Review

* J. Behrman

### Household behavior, preschool child health and nutrition, and the role of information

H. Bouis et al.

### [Are estimates of calorie-income elasticities too high?: A recalibration of the plausible range](https://www.sciencedirect.com/science/article/pii/0304387892900439" \t "_self)

### Journal of Development Economics

**2.2 EXISTING PROBLEM**

The World Bank Group and the G7 Presidency co-convened [Global Alliance for Food Security](https://www.worldbank.org/en/news/statement/2022/05/19/joint-statement-g7-presidency-wbg-establish-global-alliance-for-food-security), which aims to catalyze an immediate and concerted response to the unfolding global hunger crisis. The Alliance has developed the publicly accessible [Global Food and Nutrition Security Dashboard](https://www.gafs.info/), which provides timely information for global and local decision-makers to help improve coordination of the policy and financial response to the food crisis.

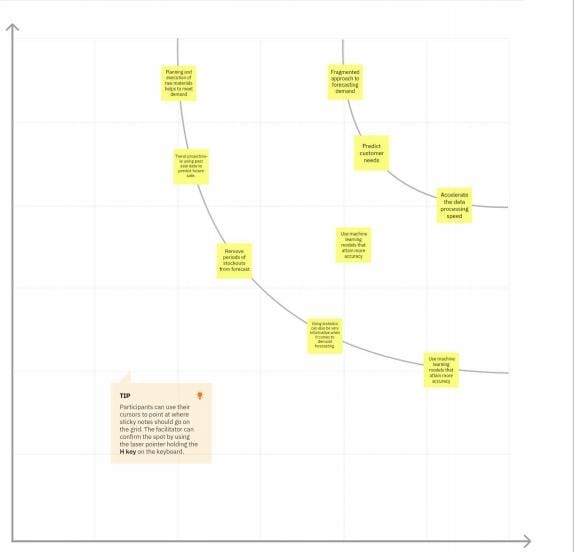
**2.3 PROBLEM STATEMENT DEFINITION**

problem of unreliable data on production and unrecorded trade is unavoidable, but may be serious for many of the most food insecure. The countries in sub-Saharan Africa. The current crisis in Southern Africa highlights this issue. Malawi appears to have been one of the twelve best-performing countries since the early 1990s in improving food security. However, there is currently much debate about the reliability of food production data, particularly for roots and tubers in this country. Trends for countries in which these are important staples, especially in subsistence, and comparisons between these and other countries are a source of ambiguity.

**3.IDEATION AND PROPOSED SOLUTION**

|  |  |  |
| --- | --- | --- |
| S.NO | PARAMETER | DESCRIPTION |
| * 1. | * Social impact | * Improvement in sales of the business * Analysing and understanding the demand for raw materials |
| **2.** | Business model | * Data analytics * Statistic * Futurepredication |
| **3.** | Scalability of the solution | The model is scalable from the architecture and data set training perspective. |

**3.1 Ideation and brainstroming**



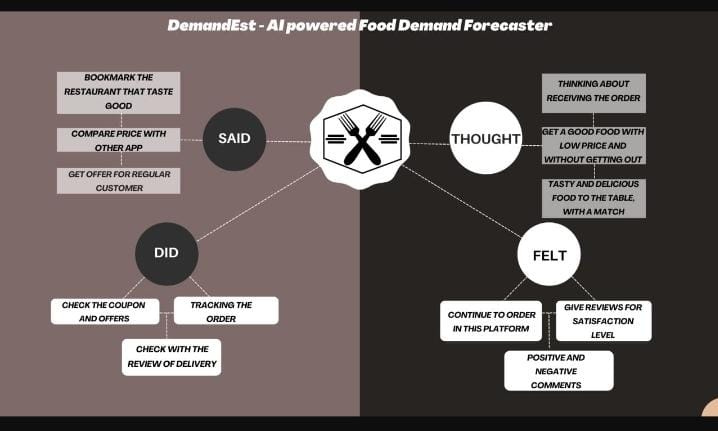
**3.2 PROBLEM SOLUTION FIT**



**3.3 PROPOSED SOLUTION**

|  |  |  |
| --- | --- | --- |
| S.No. | Parameter | Description |
| 1 | Problem Statement (Problem to be  solved) | Your client is a meal delivery company which  operates in multiple cities. They have various  fulfillment centres in these cities for  dispatching meal orders to their customers. The  client wants you to help these centres with  demand forecasting for upcoming weeks so  that these centres will plan the stock of raw  materials accordingly. The replenishment of  majority of raw materials is done on weekly  basis and since the raw material is perishable,  the procurement planning is of utmost  importance. Secondly, staffing of the centres is  also one area wherein accurate demand  forecasts are really helpful. |
| 2 | Idea / Solution description | The data set is related to a meal delivery  company which operates in multiple cities.  They have various fulfilment centres in these  cities for dispatching meal orders to their  customers.The dataset consists of historical data of  demand for a product-centre combination for  weeks 1 to 145.With the given data and information, the task isto predict the demand for the next 10 weeks(Weeks: 146-155) for the centre-mealcombinations, so that these fulfilment centres  stock the necessary raw materials accordingly. |
| 3 | Novelty / Uniqueness | As an alternative to the traditional demand  forecast format, there are opportunities to use  market and AI data to assist managers in the  S&amp;OP (Sales &amp; Operations Planning) process, as well as in the S&amp;OE (Sales and Operations Execution) process. During the S&amp;OP process, demand forecasting  supported by AI facilitates the work of the  marketing and sales areas, as well as reducing  uncertainty . |

**3.4 EMPATHY MAP CANVAS**



**4. REQUIRMENT ANALYSIS**

**4.1 FUNCTIONAL REQUIRMENT**

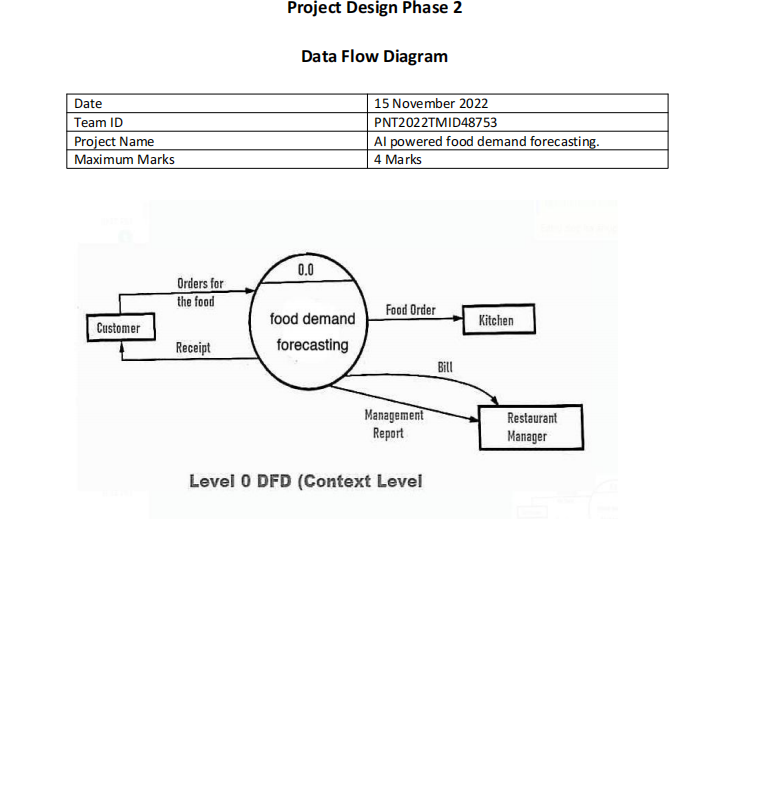
|  |  |  |
| --- | --- | --- |
| **FR No.** | **Functional Requirement (Epic)** | **Sub Requirement (Story / Sub-Task)** |
| FR-1 | User Registration | Registration through Form  Registration through Gmail  Registration through LinkedIN |
| FR-2 | User Confirmation | Confirmation via Email Confirmation via OTP |
| FR-3 | Executive administration | Regulation of monitoring the water environment status and regulatory compliance like pollution event emergency management, and it includes two different functions: early warning/forecast monitoring. |
| FR-4 | Data handling | File contains water quality metrics for different water bodies. |
| FR-5 | Quality analysis | Analyze with the acquired information of the water across various water quality indicator like (PH,  Turbidity, TDS, Temperature) using different models. |
| FR-6 | Model prediction | Confirming based on water quality index and shows the machine learning prediction (Good, Partially Good, Poor) with the percentage of presence of various parameter. |
| FR-7 | Remote Visualization | Visualization through charts based on present and past values of all the parameter for future forecast. |
| FR-8 | Notification services | Confirming through notification of water status prediction with parameter presence along with timestamp. |

**4.2 NON- FUNCTIONAL REQUIRMENT**

|  |  |  |
| --- | --- | --- |
| **FR No.** | **Non-Functional Requirement** | **Description** |
| NFR-1 | **Usability** | The system provides a natural interaction with the users. Accurate water quality prediction with short time analysis and provide prediction safe to drink or not using some parameters and provide a great significance for water environment protection. |
| NFR-2 | **Security** | The model enables with the high security system as the user’s data will not be shared to the other |

|  |  |  |
| --- | --- | --- |
|  |  | sources. The system is protected with the user name and password throughout the process. |
| NFR-3 | **Reliability** | The system is very reliable as it can last for long period of time when it is well maintained. The model can be extended in large scale by increasing the datasets. |
| NFR-4 | **Performance** | Our system should run on 32 bit (x86) or 64 bit (x64) Dual-core 2.66-GHZ or faster processor. It should not exceed 2 GB RAM. |
| NFR-5 | **Availability** | The system should be available for the duration of the user access the system until the user terminate the access. The system response to request of the user in less time and the recovery is done is less time. |
| NFR-6 | **Scalability** | It provides an efficient outcome and has the ability to increase or decrease the performance of the system based on the datasets. |

1. PROJECT DESIGN
   1. DATA FLOW DIAGRAMS

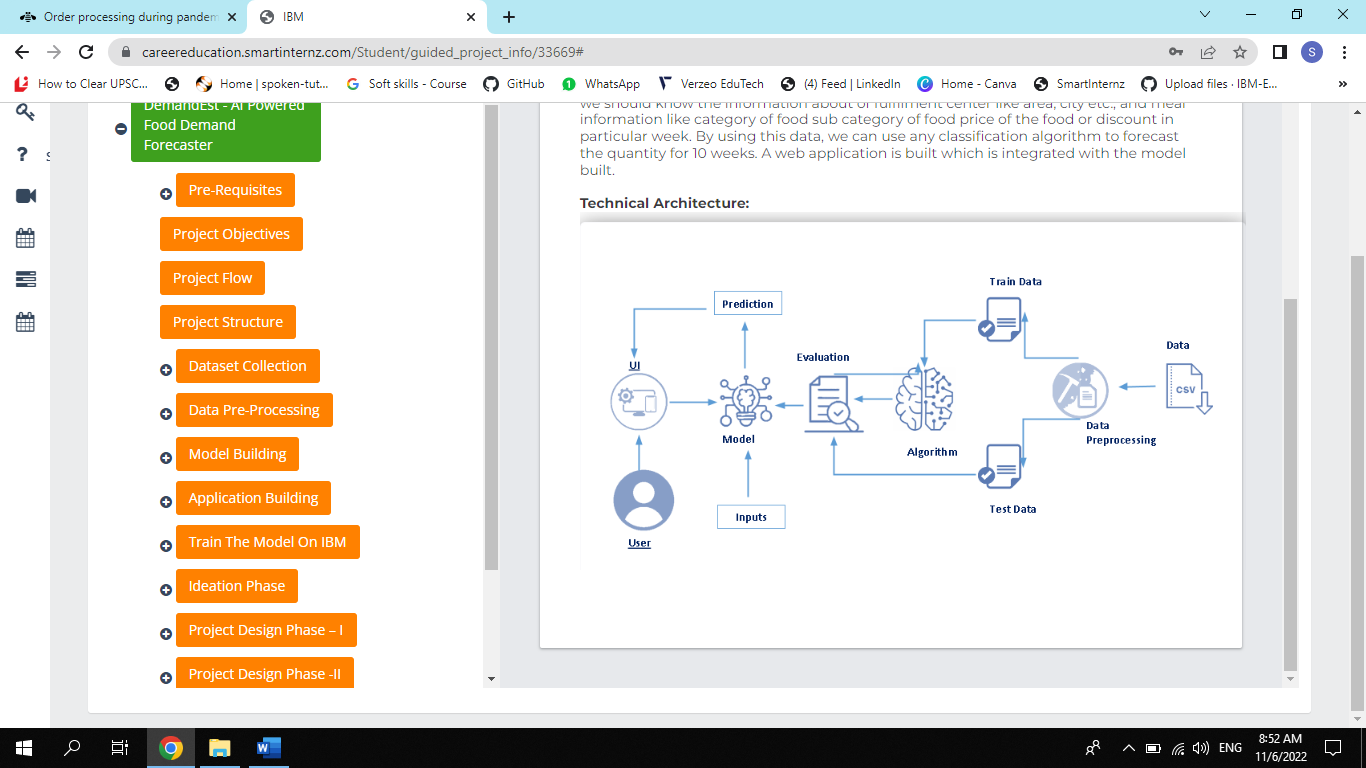


* 1. SOLUTION & TECHNICAL ARCHITECTURE

**Technology Stack (Architecture & Stack)**

|  |  |
| --- | --- |
| Date | 15 October 2022 |
| Team ID | PNT2022TMID48753 |
| Project Name | DemandEst-AI Powered Food Demand Forecaster |
| Maximum Marks | 4 Marks |

**Technical Architecture:**



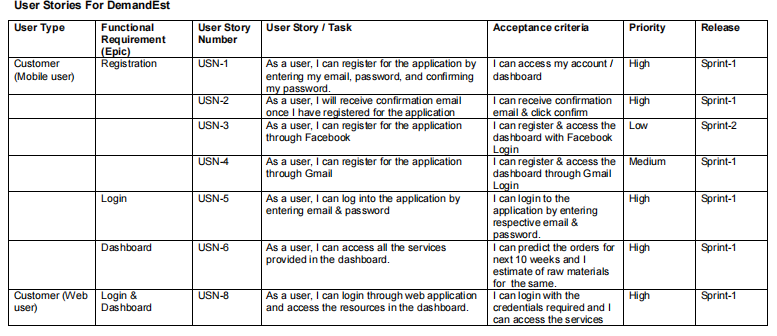
**Table-1 : Components & Technologies:**

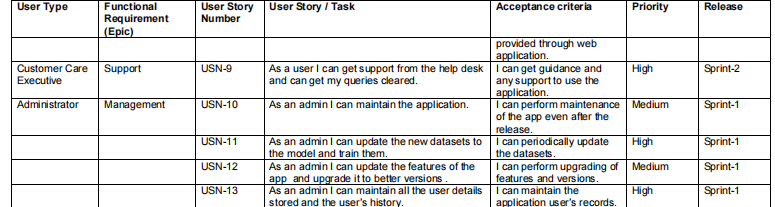
|  |  |  |  |
| --- | --- | --- | --- |
| S.No | Component | Description | Technology |
|  | User Interface | The user interface (UI) is the point of human-computer interaction and communication in a device | HTML, CSS, JavaScript, React JS |
|  | Application Logic-2 | To provide the train for the developed application using the dataset and to provide interaction between web app and the python codes. | IBM Watson STT service |
|  | Application Logic-1 | Google Collab is used for coding in the development of application. | Python |
|  | Database | The database is used for storing the history of the client requirement and the services provided b the web app | MySQL |
|  | Cloud Database | Database Service on Cloud for accessing the database whenever it is needed. | IBM DB2 |
|  | File Storage | File storage requirements | IBM Block Storage or Local Filesystem |
|  | Machine Learning Model | Fueled by data, machine learning (ML) models are the mathematical engines of artificial intelligence. | Cross-sectional forecasting, Statistical analysis. |
|  | Infrastructure (Server / Cloud) | Application Deployment on Local System | Local |

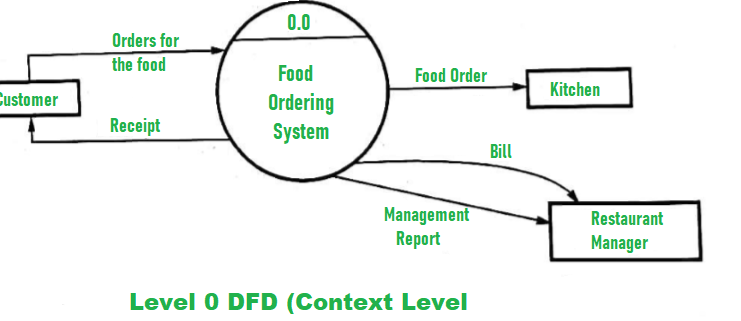
**Table-2: Application Characteristics:**

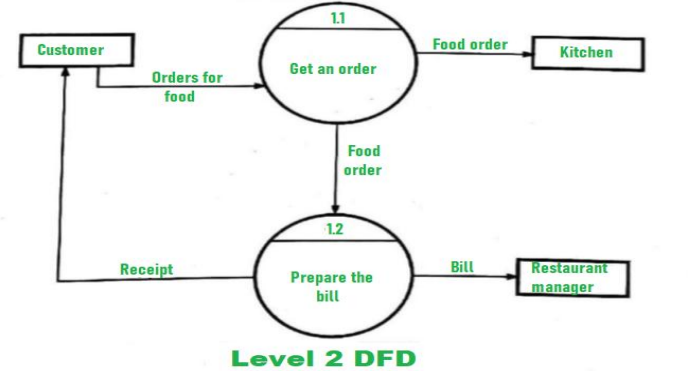
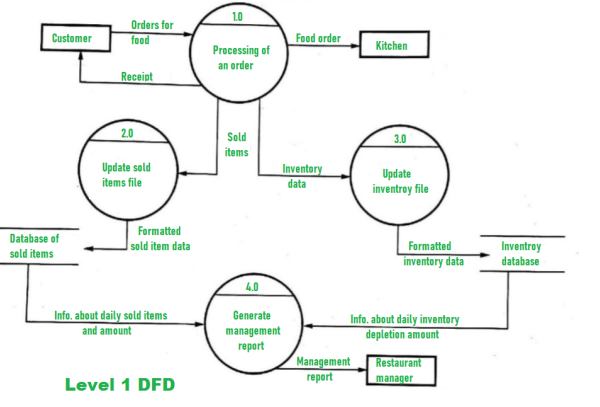
| S.No | Characteristics | Description | Technology |
| --- | --- | --- | --- |
|  | Open-Source Frameworks | Open source software is different. Its authors make its source code available to others who would like to view that code, copy it, learn from it, alter it, or share it. | Anaconda, IBM |
|  | Security Implementations | Software security is the concept of implementing mechanisms in the construction of security to help it remain functional (or resistant) to attacks. | OWASP, Firewall |
|  | Scalable Architecture | Scalability is the property of a system to handle a growing amount of work by adding resources to the system. | Cloud based product like IBM Cloud |
|  | Availability | Demand forecasting allows businesses to optimize inventory by predicting future sales. By analysing historical sales data, demand managers can make informed business decisions | Statistical Analysis |
|  | Performance | Application performance indicates how the app is functioning and how responsive the app is to the end-user. | React JS |

* 1. USER STORIES



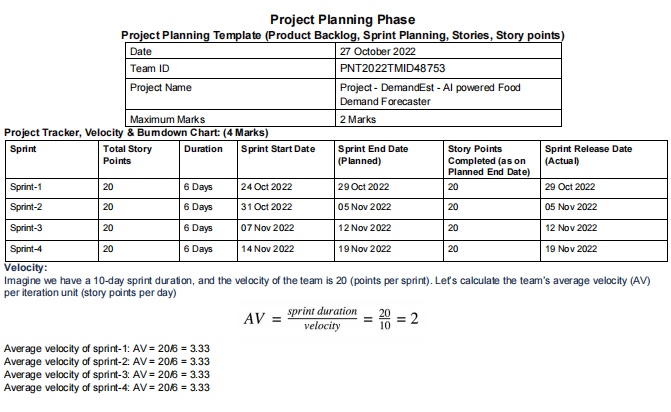






1. PROJECT PLANNING & SCHEDULING

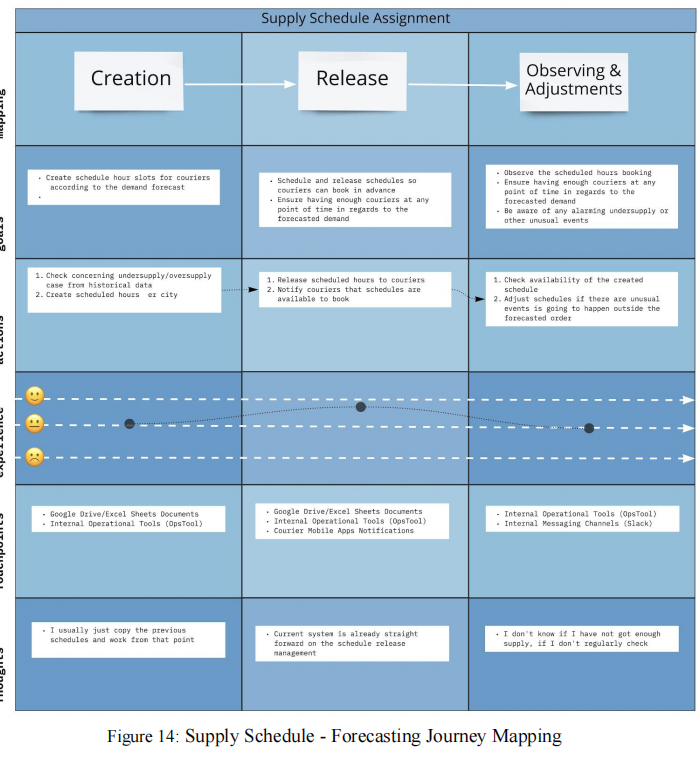
6.1 SPRINT PLANNING & ESTIMATION



6.2 SPRINT DELIVERY SCHEDULE

Creation, Release, Observing, and Adjustments

After the forecast is calculated, it is the schedule assignment stage where the OMs may need tocreate the Scheduled Hours slots. This process which is presented in Figure 14 is one of theapproaches to ensure the forecasted demands have sufficient courier supply and be planned outaccordingly. Some OMs may use less scheduled supply assignments to the couriers and use otherapproaches, e.g., surge pricing, bonus fee, where they see fit in their country. Then, the schedulesare released to the couriers to book the available slots via the Courier App. The last part is to observe the booked schedule and any alarming events that may impact the supply for the forecasted order number. The OMs need to ensure that the operation can run smoothly for all those order.



The OMs who are schedule-dependent put a significant amount of effort and time into demand forecasting to balance supply and demand. Tables 4, 5 and 6 define the manager’s journey which consists of using data collection, processing those data to demand predictions, and assigning the to match the delivery schedules. The focus of the flow is to plan couriers to operate efficiently in terms of delivery time and the cost of the couriers' guaranteed payment. These OMs do not have much flexibility in planning the schedules because they plan and create the Scheduled Hours slots usually one week in advance. Each of the stages of the specific forecasting journey mapping.

1. CODING & SOLUTIONING

7.1 FEATURES

There are several limitations to this research. The first limitation is the low number of sample interviewees, due to the OMs' availability for this research. As this research is a part of Wolt's ongoing initiative, the chosen OMs interviewed here were based voluntarily on their availability and experience. Out of the current 22 countries operating for Wolt, the research only interviewed OMs from 7 countries. The current settings were not covering all the nations that Wolt operates in so that the study could have some forecasting process biases from these countries. Moreover, the data was gathered through semi-structured interviews. Hence, the reliability and accuracy of the data depend highly on the OMs' expertise when the interview was conducted. It is challenging to research with different sets of characteristics and factors important in each country. However, because all the countries generally need the same forecasting information to operate, the research would address most of the uncovered countries' issues. Stakeholders taken as samples are also limited to OMs and do not include the Business Developer team in Wolt. As the forecasting service would be needed across the different teams in Wolt, it would be better to take more samples from different stakeholder needs in an ideal world. Thus, the improvements and problems would be balanced from different teams and different countries that are dependent or independent of Scheduled Hour. This approach would create even more comprehensive research that represents how the forecasting process is progressing.

Except for Finnish Operations, all the other OMs are located in different cities worldwide. The interviews and observations were mostly done via online video conferences. Hence, the interviews and observations were based on recorded interviews, mostly sound, and screens only. Most of the interviewed OMs explained the forecasting process in the online interview. Moreover, because the whole forecasting process would take weeks to observe, I observed the OMs only on the specific parts of the flow, such as the forecasting process and the supply assignments parts. If there were no time and location constraints, itwould be better to have the interviews and the observation face-to-face with these OMs.Then, I could observe better on their behavior when creating these forecasts. These would lead to a detailed observation, both from the problems they explain and reality when making the demand forecasts.

This study of demand forecasting service conveys that to comprehend the different internal and external factors that contribute to the demand forecasting service, further exploration of each country's local specificities is also required. The variations of local specificity and implications should be taken into account when thinking about how to forecast better and how the number of demand forecasts can help the local team expect and plan on their supply operations and meet their operational targets in the long term. Hence, creating a demand forecasting service that caters to the forecasting service's local factors, such as weather or eating time behaviors in different countries, could significantly enhance the service. The current service approach would not fit all markets, and there needs to be a way to customize the learnings of these factors based

on localities in future research.

The demand forecasting service could also be expanded to the supply forecasting service. Based on demand forecasting, the OMs need to forecast how much couriers supply that they should have to be online and deliver the food demand in a specific time. If both parts of the demand and supply forecasting calculation are automated by machine learning, then OMs would monitor ad hoc or exceptional cases more frequently. The service would also benefit and enhance their working time to other parts of the food delivery operations .

1. ADVANTAGES & DISADVANTAGES

ADVANTAGES

Demand forcasting helps reduce risk and make efficient financial decisions that impact profit margins,cash flow,allocation of resources,opportunities for expension,inventory acounting,operating costs,staffing,and overall spends. All strategic and operational plans are formulated around forcasting demand.

Forcasting projections is one of the toughest things to get right. Gradual sales or is in high-growth mode.Demand forcasting is the process of using predictive analysis of historical data t estimate and predict customers future demand for a product or service.

DISADVANTAGES

Forcasting are never 100% accurate.Lets face it; its hard to predict the future.It can be time -consuming and resource -intensive .Forcasting involves a lot of data gathering ,data organizing,and coordination.It can also be costly .

CONCLUSION

This study aims to understand the current demand forecasting service in Wolt and integrate the user-centric design approach to establish a service that supports the forecasting process efficiently. Based on the primary users, the focus group of the research is the OMs in Wolt. The service design methods helped reveal the pain points, wishes, and gaps in redefining how the

demand forecast process should move forward in the future. The service design approach helps shape suggestions to match user expectations and needs for the demand forecasting service. The approach explains the OMs' underlying experience when forecasting demand and uncovering the hidden necessities and insights, which are useful to improve the future forecasting service

development. With the wisdom of the OMs, the company could suggest better and more robust design solutions to the service. Service design methods also revealed the connection between the service provider and the primary users by clarifying the connection and touchpoints to develop the enhancement ideas.

The Affinity Diagram maps that consisted of the OMs insight were useful for uncovering insights and patterns between the OMs on the forecasting process. Using the diagram, I can cluster the common patterns to understand the current process better. Furthermore, the Persona method displayed the OM's common approach when creating a demand forecasting and supply schedules. The two profiles are Schedule Dependent OMs and Non-Schedule Dependent OMs,

each with different approaches and need to gather and process the demand forecasting.

These two profiles also differ in how they are using the result of the demand forecasting. The User journey map explained the OM’s journey in the demand forecasting cycle. Using the map, I can identify and explore the challenges and opportunities of the current forecasting process with OMs. Moreover, the service blueprint combined all the insights from the previous methods and

conveyed all the services' interaction, both the visible and the background process.

In the gig-economy industry, planning the right supply to the forecasted demand is one of the most critical factors to reach the efficiency of the *giggers*

and the company cost, as discussed . The extent of this research indicates that the area of order demand forecasting can be effectively improved by automating the process and fine-tuning the results based on external and local factors. In the scope of forecasting service, automated machine learning model-generated

results could offer the accuracy and time-efficiency that many companies encounter in this digital era. With the capability to automate the forecasting process and maintenance, the service can be scalable for more country expansion while still bridging the local factors and necessities

during the forecasting process. An automated forecasting service would help the OMs become aware of the expected growth changes, which improves the flexibility and precision to prepare

the courier supply. By allowing these local OMs to set and tune their local forecasting factors, the forecast results will be more reliable and beneficial for future learnings of the demand forecasting model.

SOURCE CODE

窗体顶端

{

"cells": [

{

"cell\_type": "code",

"execution\_count": 1,

"metadata": {},

"outputs": [],

"source": [

"import numpy as np\n",

"import pandas as pd\n",

"pd.set\_option('display.max\_columns', None) \n",

"import matplotlib.pyplot as plt\n",

"import matplotlib.patches as mpatches\n",

"%matplotlib inline \n",

"import seaborn as sns\n",

"import plotly.io as pio\n",

"import plotly.graph\_objects as go\n",

"from plotly.offline import init\_notebook\_mode, iplot\n",

"from sklearn.preprocessing import StandardScaler\n",

"import warnings\n",

"warnings.filterwarnings('ignore')"

]

},

{

"cell\_type": "code",

"execution\_count": 2,

"metadata": {},

"outputs": [],

"source": [

"data=pd.read\_csv('train.csv')\n",

"center=pd.read\_csv('fulfilment\_center\_info.csv')\n",

"meal=pd.read\_csv('meal\_info.csv')\n",

"test=pd.read\_csv('test\_QoiMO9B.csv')"

]

},

{

"cell\_type": "markdown",

"metadata": {},

"source": [

"##### Data Pre-Processing"

]

},

{

"cell\_type": "code",

"execution\_count": 3,

"metadata": {},

"outputs": [

{

"name": "stdout",

"output\_type": "stream",

"text": [

"The Shape of Demand dataset : (456548, 9)\n",

"The Shape of Fulmilment Center Information dataset : (77, 5)\n",

"The Shape of Meal information dataset : (51, 3)\n",

"The Shape of Test dataset : (32573, 8)\n"

]

}

],

"source": [

"print(\"The Shape of Demand dataset :\",data.shape)\n",

"print(\"The Shape of Fulmilment Center Information dataset :\",center.shape)\n",

"print(\"The Shape of Meal information dataset :\",meal.shape)\n",

"print(\"The Shape of Test dataset :\",test.shape)"

]

},

{

"cell\_type": "code",

"execution\_count": 4,

"metadata": {},

"outputs": [

{

"data": {

"text/html": [

"<div>\n",

"<style scoped>\n",

" .dataframe tbody tr th:only-of-type {\n",

" vertical-align: middle;\n",

" }\n",

"\n",

" .dataframe tbody tr th {\n",

" vertical-align: top;\n",

" }\n",

"\n",

" .dataframe thead th {\n",

" text-align: right;\n",

" }\n",

"</style>\n",

"<table border=\"1\" class=\"dataframe\">\n",

" <thead>\n",

" <tr style=\"text-align: right;\">\n",

" <th></th>\n",

" <th>id</th>\n",

" <th>week</th>\n",

" <th>center\_id</th>\n",

" <th>meal\_id</th>\n",

" <th>checkout\_price</th>\n",

" <th>base\_price</th>\n",

" <th>emailer\_for\_promotion</th>\n",

" <th>homepage\_featured</th>\n",

" <th>num\_orders</th>\n",

" </tr>\n",

" </thead>\n",

" <tbody>\n",

" <tr>\n",

" <th>0</th>\n",

" <td>1379560</td>\n",

" <td>1</td>\n",

" <td>55</td>\n",

" <td>1885</td>\n",

" <td>136.83</td>\n",

" <td>152.29</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>177</td>\n",

" </tr>\n",

" <tr>\n",

" <th>1</th>\n",

" <td>1466964</td>\n",

" <td>1</td>\n",

" <td>55</td>\n",

" <td>1993</td>\n",

" <td>136.83</td>\n",

" <td>135.83</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>270</td>\n",

" </tr>\n",

" <tr>\n",

" <th>2</th>\n",

" <td>1346989</td>\n",

" <td>1</td>\n",

" <td>55</td>\n",

" <td>2539</td>\n",

" <td>134.86</td>\n",

" <td>135.86</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>189</td>\n",

" </tr>\n",

" <tr>\n",

" <th>3</th>\n",

" <td>1338232</td>\n",

" <td>1</td>\n",

" <td>55</td>\n",

" <td>2139</td>\n",

" <td>339.50</td>\n",

" <td>437.53</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>54</td>\n",

" </tr>\n",

" <tr>\n",

" <th>4</th>\n",

" <td>1448490</td>\n",

" <td>1</td>\n",

" <td>55</td>\n",

" <td>2631</td>\n",

" <td>243.50</td>\n",

" <td>242.50</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>40</td>\n",

" </tr>\n",

" </tbody>\n",

"</table>\n",

"</div>"

],

"text/plain": [

" id week center\_id meal\_id checkout\_price base\_price \\\n",

"0 1379560 1 55 1885 136.83 152.29 \n",

"1 1466964 1 55 1993 136.83 135.83 \n",

"2 1346989 1 55 2539 134.86 135.86 \n",

"3 1338232 1 55 2139 339.50 437.53 \n",

"4 1448490 1 55 2631 243.50 242.50 \n",

"\n",

" emailer\_for\_promotion homepage\_featured num\_orders \n",

"0 0 0 177 \n",

"1 0 0 270 \n",

"2 0 0 189 \n",

"3 0 0 54 \n",

"4 0 0 40 "

]

},

"execution\_count": 4,

"metadata": {},

"output\_type": "execute\_result"

}

],

"source": [

"data.head()"

]

},

{

"cell\_type": "code",

"execution\_count": 5,

"metadata": {},

"outputs": [],

"source": [

"test['num\_orders']=123456 ### Assigning random number for Target Variable of Test Data."

]

},

{

"cell\_type": "code",

"execution\_count": 6,

"metadata": {},

"outputs": [

{

"data": {

"text/html": [

"<div>\n",

"<style scoped>\n",

" .dataframe tbody tr th:only-of-type {\n",

" vertical-align: middle;\n",

" }\n",

"\n",

" .dataframe tbody tr th {\n",

" vertical-align: top;\n",

" }\n",

"\n",

" .dataframe thead th {\n",

" text-align: right;\n",

" }\n",

"</style>\n",

"<table border=\"1\" class=\"dataframe\">\n",

" <thead>\n",

" <tr style=\"text-align: right;\">\n",

" <th></th>\n",

" <th>id</th>\n",

" <th>week</th>\n",

" <th>center\_id</th>\n",

" <th>meal\_id</th>\n",

" <th>checkout\_price</th>\n",

" <th>base\_price</th>\n",

" <th>emailer\_for\_promotion</th>\n",

" <th>homepage\_featured</th>\n",

" <th>num\_orders</th>\n",

" </tr>\n",

" </thead>\n",

" <tbody>\n",

" <tr>\n",

" <th>0</th>\n",

" <td>1028232</td>\n",

" <td>146</td>\n",

" <td>55</td>\n",

" <td>1885</td>\n",

" <td>158.11</td>\n",

" <td>159.11</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>123456</td>\n",

" </tr>\n",

" <tr>\n",

" <th>1</th>\n",

" <td>1127204</td>\n",

" <td>146</td>\n",

" <td>55</td>\n",

" <td>1993</td>\n",

" <td>160.11</td>\n",

" <td>159.11</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>123456</td>\n",

" </tr>\n",

" <tr>\n",

" <th>2</th>\n",

" <td>1212707</td>\n",

" <td>146</td>\n",

" <td>55</td>\n",

" <td>2539</td>\n",

" <td>157.14</td>\n",

" <td>159.14</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>123456</td>\n",

" </tr>\n",

" <tr>\n",

" <th>3</th>\n",

" <td>1082698</td>\n",

" <td>146</td>\n",

" <td>55</td>\n",

" <td>2631</td>\n",

" <td>162.02</td>\n",

" <td>162.02</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>123456</td>\n",

" </tr>\n",

" <tr>\n",

" <th>4</th>\n",

" <td>1400926</td>\n",

" <td>146</td>\n",

" <td>55</td>\n",

" <td>1248</td>\n",

" <td>163.93</td>\n",

" <td>163.93</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>123456</td>\n",

" </tr>\n",

" </tbody>\n",

"</table>\n",

"</div>"

],

"text/plain": [

" id week center\_id meal\_id checkout\_price base\_price \\\n",

"0 1028232 146 55 1885 158.11 159.11 \n",

"1 1127204 146 55 1993 160.11 159.11 \n",

"2 1212707 146 55 2539 157.14 159.14 \n",

"3 1082698 146 55 2631 162.02 162.02 \n",

"4 1400926 146 55 1248 163.93 163.93 \n",

"\n",

" emailer\_for\_promotion homepage\_featured num\_orders \n",

"0 0 0 123456 \n",

"1 0 0 123456 \n",

"2 0 0 123456 \n",

"3 0 0 123456 \n",

"4 0 0 123456 "

]

},

"execution\_count": 6,

"metadata": {},

"output\_type": "execute\_result"

}

],

"source": [

"test.head()"

]

},

{

"cell\_type": "code",

"execution\_count": 7,

"metadata": {},

"outputs": [

{

"data": {

"text/html": [

"<div>\n",

"<style scoped>\n",

" .dataframe tbody tr th:only-of-type {\n",

" vertical-align: middle;\n",

" }\n",

"\n",

" .dataframe tbody tr th {\n",

" vertical-align: top;\n",

" }\n",

"\n",

" .dataframe thead th {\n",

" text-align: right;\n",

" }\n",

"</style>\n",

"<table border=\"1\" class=\"dataframe\">\n",

" <thead>\n",

" <tr style=\"text-align: right;\">\n",

" <th></th>\n",

" <th>center\_id</th>\n",

" <th>city\_code</th>\n",

" <th>region\_code</th>\n",

" <th>center\_type</th>\n",

" <th>op\_area</th>\n",

" </tr>\n",

" </thead>\n",

" <tbody>\n",

" <tr>\n",

" <th>0</th>\n",

" <td>11</td>\n",

" <td>679</td>\n",

" <td>56</td>\n",

" <td>TYPE\_A</td>\n",

" <td>3.7</td>\n",

" </tr>\n",

" <tr>\n",

" <th>1</th>\n",

" <td>13</td>\n",

" <td>590</td>\n",

" <td>56</td>\n",

" <td>TYPE\_B</td>\n",

" <td>6.7</td>\n",

" </tr>\n",

" <tr>\n",

" <th>2</th>\n",

" <td>124</td>\n",

" <td>590</td>\n",

" <td>56</td>\n",

" <td>TYPE\_C</td>\n",

" <td>4.0</td>\n",

" </tr>\n",

" <tr>\n",

" <th>3</th>\n",

" <td>66</td>\n",

" <td>648</td>\n",

" <td>34</td>\n",

" <td>TYPE\_A</td>\n",

" <td>4.1</td>\n",

" </tr>\n",

" <tr>\n",

" <th>4</th>\n",

" <td>94</td>\n",

" <td>632</td>\n",

" <td>34</td>\n",

" <td>TYPE\_C</td>\n",

" <td>3.6</td>\n",

" </tr>\n",

" </tbody>\n",

"</table>\n",

"</div>"

],

"text/plain": [

" center\_id city\_code region\_code center\_type op\_area\n",

"0 11 679 56 TYPE\_A 3.7\n",

"1 13 590 56 TYPE\_B 6.7\n",

"2 124 590 56 TYPE\_C 4.0\n",

"3 66 648 34 TYPE\_A 4.1\n",

"4 94 632 34 TYPE\_C 3.6"

]

},

"execution\_count": 7,

"metadata": {},

"output\_type": "execute\_result"

}

],

"source": [

"center.head()"

]

},

{

"cell\_type": "code",

"execution\_count": 8,

"metadata": {},

"outputs": [

{

"data": {

"text/html": [

"<div>\n",

"<style scoped>\n",

" .dataframe tbody tr th:only-of-type {\n",

" vertical-align: middle;\n",

" }\n",

"\n",

" .dataframe tbody tr th {\n",

" vertical-align: top;\n",

" }\n",

"\n",

" .dataframe thead th {\n",

" text-align: right;\n",

" }\n",

"</style>\n",

"<table border=\"1\" class=\"dataframe\">\n",

" <thead>\n",

" <tr style=\"text-align: right;\">\n",

" <th></th>\n",

" <th>meal\_id</th>\n",

" <th>category</th>\n",

" <th>cuisine</th>\n",

" </tr>\n",

" </thead>\n",

" <tbody>\n",

" <tr>\n",

" <th>0</th>\n",

" <td>1885</td>\n",

" <td>Beverages</td>\n",

" <td>Thai</td>\n",

" </tr>\n",

" <tr>\n",

" <th>1</th>\n",

" <td>1993</td>\n",

" <td>Beverages</td>\n",

" <td>Thai</td>\n",

" </tr>\n",

" <tr>\n",

" <th>2</th>\n",

" <td>2539</td>\n",

" <td>Beverages</td>\n",

" <td>Thai</td>\n",

" </tr>\n",

" <tr>\n",

" <th>3</th>\n",

" <td>1248</td>\n",

" <td>Beverages</td>\n",

" <td>Indian</td>\n",

" </tr>\n",

" <tr>\n",

" <th>4</th>\n",

" <td>2631</td>\n",

" <td>Beverages</td>\n",

" <td>Indian</td>\n",

" </tr>\n",

" </tbody>\n",

"</table>\n",

"</div>"

],

"text/plain": [

" meal\_id category cuisine\n",

"0 1885 Beverages Thai\n",

"1 1993 Beverages Thai\n",

"2 2539 Beverages Thai\n",

"3 1248 Beverages Indian\n",

"4 2631 Beverages Indian"

]

},

"execution\_count": 8,

"metadata": {},

"output\_type": "execute\_result"

}

],

"source": [

"meal.head()"

]

},

{

"cell\_type": "code",

"execution\_count": 9,

"metadata": {},

"outputs": [],

"source": [

"data=pd.concat([data,test],axis=0)"

]

},

{

"cell\_type": "code",

"execution\_count": 10,

"metadata": {},

"outputs": [],

"source": [

"data=data.merge(center,on='center\_id',how='left')"

]

},

{

"cell\_type": "code",

"execution\_count": 11,

"metadata": {},

"outputs": [],

"source": [

"data=data.merge(meal,on='meal\_id',how='left')"

]

},

{

"cell\_type": "code",

"execution\_count": 12,

"metadata": {

"scrolled": false

},

"outputs": [

{

"data": {

"text/html": [

"<div>\n",

"<style scoped>\n",

" .dataframe tbody tr th:only-of-type {\n",

" vertical-align: middle;\n",

" }\n",

"\n",

" .dataframe tbody tr th {\n",

" vertical-align: top;\n",

" }\n",

"\n",

" .dataframe thead th {\n",

" text-align: right;\n",

" }\n",

"</style>\n",

"<table border=\"1\" class=\"dataframe\">\n",

" <thead>\n",

" <tr style=\"text-align: right;\">\n",

" <th></th>\n",

" <th>id</th>\n",

" <th>week</th>\n",

" <th>center\_id</th>\n",

" <th>meal\_id</th>\n",

" <th>checkout\_price</th>\n",

" <th>base\_price</th>\n",

" <th>emailer\_for\_promotion</th>\n",

" <th>homepage\_featured</th>\n",

" <th>num\_orders</th>\n",

" <th>city\_code</th>\n",

" <th>region\_code</th>\n",

" <th>center\_type</th>\n",

" <th>op\_area</th>\n",

" <th>category</th>\n",

" <th>cuisine</th>\n",

" </tr>\n",

" </thead>\n",

" <tbody>\n",

" <tr>\n",

" <th>0</th>\n",

" <td>1379560</td>\n",

" <td>1</td>\n",

" <td>55</td>\n",

" <td>1885</td>\n",

" <td>136.83</td>\n",

" <td>152.29</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>177</td>\n",

" <td>647</td>\n",

" <td>56</td>\n",

" <td>TYPE\_C</td>\n",

" <td>2.0</td>\n",

" <td>Beverages</td>\n",

" <td>Thai</td>\n",

" </tr>\n",

" <tr>\n",

" <th>1</th>\n",

" <td>1466964</td>\n",

" <td>1</td>\n",

" <td>55</td>\n",

" <td>1993</td>\n",

" <td>136.83</td>\n",

" <td>135.83</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>270</td>\n",

" <td>647</td>\n",

" <td>56</td>\n",

" <td>TYPE\_C</td>\n",

" <td>2.0</td>\n",

" <td>Beverages</td>\n",

" <td>Thai</td>\n",

" </tr>\n",

" <tr>\n",

" <th>2</th>\n",

" <td>1346989</td>\n",

" <td>1</td>\n",

" <td>55</td>\n",

" <td>2539</td>\n",

" <td>134.86</td>\n",

" <td>135.86</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>189</td>\n",

" <td>647</td>\n",

" <td>56</td>\n",

" <td>TYPE\_C</td>\n",

" <td>2.0</td>\n",

" <td>Beverages</td>\n",

" <td>Thai</td>\n",

" </tr>\n",

" <tr>\n",

" <th>3</th>\n",

" <td>1338232</td>\n",

" <td>1</td>\n",

" <td>55</td>\n",

" <td>2139</td>\n",

" <td>339.50</td>\n",

" <td>437.53</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>54</td>\n",

" <td>647</td>\n",

" <td>56</td>\n",

" <td>TYPE\_C</td>\n",

" <td>2.0</td>\n",

" <td>Beverages</td>\n",

" <td>Indian</td>\n",

" </tr>\n",

" <tr>\n",

" <th>4</th>\n",

" <td>1448490</td>\n",

" <td>1</td>\n",

" <td>55</td>\n",

" <td>2631</td>\n",

" <td>243.50</td>\n",

" <td>242.50</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>40</td>\n",

" <td>647</td>\n",

" <td>56</td>\n",

" <td>TYPE\_C</td>\n",

" <td>2.0</td>\n",

" <td>Beverages</td>\n",

" <td>Indian</td>\n",

" </tr>\n",

" </tbody>\n",

"</table>\n",

"</div>"

],

"text/plain": [

" id week center\_id meal\_id checkout\_price base\_price \\\n",

"0 1379560 1 55 1885 136.83 152.29 \n",

"1 1466964 1 55 1993 136.83 135.83 \n",

"2 1346989 1 55 2539 134.86 135.86 \n",

"3 1338232 1 55 2139 339.50 437.53 \n",

"4 1448490 1 55 2631 243.50 242.50 \n",

"\n",

" emailer\_for\_promotion homepage\_featured num\_orders city\_code \\\n",

"0 0 0 177 647 \n",

"1 0 0 270 647 \n",

"2 0 0 189 647 \n",

"3 0 0 54 647 \n",

"4 0 0 40 647 \n",

"\n",

" region\_code center\_type op\_area category cuisine \n",

"0 56 TYPE\_C 2.0 Beverages Thai \n",

"1 56 TYPE\_C 2.0 Beverages Thai \n",

"2 56 TYPE\_C 2.0 Beverages Thai \n",

"3 56 TYPE\_C 2.0 Beverages Indian \n",

"4 56 TYPE\_C 2.0 Beverages Indian "

]

},

"execution\_count": 12,

"metadata": {},

"output\_type": "execute\_result"

}

],

"source": [

"data.head()"

]

},

{

"cell\_type": "code",

"execution\_count": 13,

"metadata": {},

"outputs": [

{

"data": {

"text/plain": [

"id 0\n",

"week 0\n",

"center\_id 0\n",

"meal\_id 0\n",

"checkout\_price 0\n",

"base\_price 0\n",

"emailer\_for\_promotion 0\n",

"homepage\_featured 0\n",

"num\_orders 0\n",

"city\_code 0\n",

"region\_code 0\n",

"center\_type 0\n",

"op\_area 0\n",

"category 0\n",

"cuisine 0\n",

"dtype: int64"

]

},

"execution\_count": 13,

"metadata": {},

"output\_type": "execute\_result"

}

],

"source": [

"data.isnull().sum()"

]

},

{

"cell\_type": "markdown",

"metadata": {},

"source": [

"##### Deriving New Features"

]

},

{

"cell\_type": "code",

"execution\_count": 14,

"metadata": {},

"outputs": [],

"source": [

"#Discount Amount\n",

"data['discount amount']=data['base\_price']-data['checkout\_price']"

]

},

{

"cell\_type": "code",

"execution\_count": 15,

"metadata": {},

"outputs": [],

"source": [

"#Discount Percent\n",

"data['discount percent'] = ((data['base\_price']-data['checkout\_price'])/data['base\_price'])\*100"

]

},

{

"cell\_type": "code",

"execution\_count": 16,

"metadata": {},

"outputs": [],

"source": [

"#Discount Y/N\n",

"data['discount y/n'] = [1 if x>0 else 0 for x in (data['base\_price']-data['checkout\_price'])]"

]

},

{

"cell\_type": "code",

"execution\_count": 17,

"metadata": {},

"outputs": [],

"source": [

"data=data.sort\_values(['center\_id', 'meal\_id', 'week']).reset\_index()"

]

},

{

"cell\_type": "code",

"execution\_count": 18,

"metadata": {},

"outputs": [],

"source": [

"#Compare Week Price\n",

"data['compare\_week\_price'] = data['checkout\_price'] - data['checkout\_price'].shift(1) "

]

},

{

"cell\_type": "code",

"execution\_count": 19,

"metadata": {

"scrolled": false

},

"outputs": [],

"source": [

"data['compare\_week\_price'][data['week']==1]=0"

]

},

{

"cell\_type": "code",

"execution\_count": 20,

"metadata": {},

"outputs": [],

"source": [

"data=data.sort\_values(by='index').reset\_index().drop(['level\_0','index'],axis=1)"

]

},

{

"cell\_type": "code",

"execution\_count": 21,

"metadata": {},

"outputs": [],

"source": [

"# Compare Week Price Y/N\n",

"data['compare\_week\_price y/n'] = [1 if x>0 else 0 for x in data['compare\_week\_price']]"

]

},

{

"cell\_type": "code",

"execution\_count": 22,

"metadata": {},

"outputs": [

{

"data": {

"text/html": [

"<div>\n",

"<style scoped>\n",

" .dataframe tbody tr th:only-of-type {\n",

" vertical-align: middle;\n",

" }\n",

"\n",

" .dataframe tbody tr th {\n",

" vertical-align: top;\n",

" }\n",

"\n",

" .dataframe thead th {\n",

" text-align: right;\n",

" }\n",

"</style>\n",

"<table border=\"1\" class=\"dataframe\">\n",

" <thead>\n",

" <tr style=\"text-align: right;\">\n",

" <th></th>\n",

" <th>id</th>\n",

" <th>week</th>\n",

" <th>center\_id</th>\n",

" <th>meal\_id</th>\n",

" <th>checkout\_price</th>\n",

" <th>base\_price</th>\n",

" <th>emailer\_for\_promotion</th>\n",

" <th>homepage\_featured</th>\n",

" <th>num\_orders</th>\n",

" <th>city\_code</th>\n",

" <th>region\_code</th>\n",

" <th>center\_type</th>\n",

" <th>op\_area</th>\n",

" <th>category</th>\n",

" <th>cuisine</th>\n",

" <th>discount amount</th>\n",

" <th>discount percent</th>\n",

" <th>discount y/n</th>\n",

" <th>compare\_week\_price</th>\n",

" <th>compare\_week\_price y/n</th>\n",

" </tr>\n",

" </thead>\n",

" <tbody>\n",

" <tr>\n",

" <th>0</th>\n",

" <td>1379560</td>\n",

" <td>1</td>\n",

" <td>55</td>\n",

" <td>1885</td>\n",

" <td>136.83</td>\n",

" <td>152.29</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>177</td>\n",

" <td>647</td>\n",

" <td>56</td>\n",

" <td>TYPE\_C</td>\n",

" <td>2.0</td>\n",

" <td>Beverages</td>\n",

" <td>Thai</td>\n",

" <td>15.46</td>\n",

" <td>10.151684</td>\n",

" <td>1</td>\n",

" <td>0.0</td>\n",

" <td>0</td>\n",

" </tr>\n",

" <tr>\n",

" <th>1</th>\n",

" <td>1466964</td>\n",

" <td>1</td>\n",

" <td>55</td>\n",

" <td>1993</td>\n",

" <td>136.83</td>\n",

" <td>135.83</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>270</td>\n",

" <td>647</td>\n",

" <td>56</td>\n",

" <td>TYPE\_C</td>\n",

" <td>2.0</td>\n",

" <td>Beverages</td>\n",

" <td>Thai</td>\n",

" <td>-1.00</td>\n",

" <td>-0.736214</td>\n",

" <td>0</td>\n",

" <td>0.0</td>\n",

" <td>0</td>\n",

" </tr>\n",

" <tr>\n",

" <th>2</th>\n",

" <td>1346989</td>\n",

" <td>1</td>\n",

" <td>55</td>\n",

" <td>2539</td>\n",

" <td>134.86</td>\n",

" <td>135.86</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>189</td>\n",

" <td>647</td>\n",

" <td>56</td>\n",

" <td>TYPE\_C</td>\n",

" <td>2.0</td>\n",

" <td>Beverages</td>\n",

" <td>Thai</td>\n",

" <td>1.00</td>\n",

" <td>0.736052</td>\n",

" <td>1</td>\n",

" <td>0.0</td>\n",

" <td>0</td>\n",

" </tr>\n",

" <tr>\n",

" <th>3</th>\n",

" <td>1338232</td>\n",

" <td>1</td>\n",

" <td>55</td>\n",

" <td>2139</td>\n",

" <td>339.50</td>\n",

" <td>437.53</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>54</td>\n",

" <td>647</td>\n",

" <td>56</td>\n",

" <td>TYPE\_C</td>\n",

" <td>2.0</td>\n",

" <td>Beverages</td>\n",

" <td>Indian</td>\n",

" <td>98.03</td>\n",

" <td>22.405321</td>\n",

" <td>1</td>\n",

" <td>0.0</td>\n",

" <td>0</td>\n",

" </tr>\n",

" <tr>\n",

" <th>4</th>\n",

" <td>1448490</td>\n",

" <td>1</td>\n",

" <td>55</td>\n",

" <td>2631</td>\n",

" <td>243.50</td>\n",

" <td>242.50</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>40</td>\n",

" <td>647</td>\n",

" <td>56</td>\n",

" <td>TYPE\_C</td>\n",

" <td>2.0</td>\n",

" <td>Beverages</td>\n",

" <td>Indian</td>\n",

" <td>-1.00</td>\n",

" <td>-0.412371</td>\n",

" <td>0</td>\n",

" <td>0.0</td>\n",

" <td>0</td>\n",

" </tr>\n",

" </tbody>\n",

"</table>\n",

"</div>"

],

"text/plain": [

" id week center\_id meal\_id checkout\_price base\_price \\\n",

"0 1379560 1 55 1885 136.83 152.29 \n",

"1 1466964 1 55 1993 136.83 135.83 \n",

"2 1346989 1 55 2539 134.86 135.86 \n",

"3 1338232 1 55 2139 339.50 437.53 \n",

"4 1448490 1 55 2631 243.50 242.50 \n",

"\n",

" emailer\_for\_promotion homepage\_featured num\_orders city\_code \\\n",

"0 0 0 177 647 \n",

"1 0 0 270 647 \n",

"2 0 0 189 647 \n",

"3 0 0 54 647 \n",

"4 0 0 40 647 \n",

"\n",

" region\_code center\_type op\_area category cuisine discount amount \\\n",

"0 56 TYPE\_C 2.0 Beverages Thai 15.46 \n",

"1 56 TYPE\_C 2.0 Beverages Thai -1.00 \n",

"2 56 TYPE\_C 2.0 Beverages Thai 1.00 \n",

"3 56 TYPE\_C 2.0 Beverages Indian 98.03 \n",

"4 56 TYPE\_C 2.0 Beverages Indian -1.00 \n",

"\n",

" discount percent discount y/n compare\_week\_price compare\_week\_price y/n \n",

"0 10.151684 1 0.0 0 \n",

"1 -0.736214 0 0.0 0 \n",

"2 0.736052 1 0.0 0 \n",

"3 22.405321 1 0.0 0 \n",

"4 -0.412371 0 0.0 0 "

]

},

"execution\_count": 22,

"metadata": {},

"output\_type": "execute\_result"

}

],

"source": [

"data.head()"

]

},

{

"cell\_type": "code",

"execution\_count": 23,

"metadata": {},

"outputs": [

{

"data": {

"text/plain": [

"id 0\n",

"week 0\n",

"center\_id 0\n",

"meal\_id 0\n",

"checkout\_price 0\n",

"base\_price 0\n",

"emailer\_for\_promotion 0\n",

"homepage\_featured 0\n",

"num\_orders 0\n",

"city\_code 0\n",

"region\_code 0\n",

"center\_type 0\n",

"op\_area 0\n",

"category 0\n",

"cuisine 0\n",

"discount amount 0\n",

"discount percent 0\n",

"discount y/n 0\n",

"compare\_week\_price 0\n",

"compare\_week\_price y/n 0\n",

"dtype: int64"

]

},

"execution\_count": 23,

"metadata": {},

"output\_type": "execute\_result"

}

],

"source": [

"data.isnull().sum()"

]

},

{

"cell\_type": "markdown",

"metadata": {},

"source": [

"##### Train Test Split"

]

},

{

"cell\_type": "code",

"execution\_count": 24,

"metadata": {},

"outputs": [],

"source": [

"train=data[data['week'].isin(range(1,146))]\n",

"test=data[data['week'].isin(range(146,156))]"

]

},

{

"cell\_type": "code",

"execution\_count": 25,

"metadata": {},

"outputs": [

{

"name": "stdout",

"output\_type": "stream",

"text": [

"The Shape of Train dataset : (456548, 20)\n",

"The Shape of Test dataset : (32573, 20)\n"

]

}

],

"source": [

"print(\"The Shape of Train dataset :\",train.shape)\n",

"print(\"The Shape of Test dataset :\",test.shape)"

]

},

{

"cell\_type": "code",

"execution\_count": 26,

"metadata": {},

"outputs": [

{

"data": {

"text/plain": [

"<matplotlib.axes.\_subplots.AxesSubplot at 0x1ce5fe6f828>"

]

},

"execution\_count": 26,

"metadata": {},

"output\_type": "execute\_result"

},

{

"data": {

"image/png": "\n",

"text/plain": [

"<Figure size 1152x1008 with 2 Axes>"

]

},

"metadata": {

"needs\_background": "light"

},

"output\_type": "display\_data"

}

],

"source": [

"plt.figure(figsize=(16,14))\n",

"sns.heatmap(train.corr(),annot=True, square=True, cmap='Reds')"

]

},

{

"cell\_type": "code",

"execution\_count": 27,

"metadata": {},

"outputs": [

{

"data": {

"image/png": "\n",

"text/plain": [

"<Figure size 288x504 with 1 Axes>"

]

},

"metadata": {

"needs\_background": "light"

},

"output\_type": "display\_data"

}

],

"source": [

"fig=plt.figure(figsize=(4,7))\n",

"plt.title('Total No. of Orders for Each Center type',fontdict={'fontsize':13})\n",

"sns.barplot(y='num\_orders', x='center\_type', data=train.groupby('center\_type').sum()['num\_orders'].reset\_index(),palette='autumn');\n",

"plt.ylabel('No. of Orders',fontdict={'fontsize':12})\n",

"plt.xlabel('Center Type',fontdict={'fontsize':12})\n",

"sns.despine(bottom = True, left = True);\n"

]

},

{

"cell\_type": "markdown",

"metadata": {},

"source": [

"Type\_A Centers have the highest number of Orders placed and Type\_C has the least."

]

},

{

"cell\_type": "code",

"execution\_count": 28,

"metadata": {},

"outputs": [

{

"data": {

"text/plain": [

"77"

]

},

"execution\_count": 28,

"metadata": {},

"output\_type": "execute\_result"

}

],

"source": [

"train['center\_id'].nunique()"

]

},

{

"cell\_type": "markdown",

"metadata": {},

"source": [

"The are are 77 Fullfilment Centers in total."

]

},

{

"cell\_type": "code",

"execution\_count": 29,

"metadata": {},

"outputs": [

{

"data": {

"image/png": "

\n",

"text/plain": [

"<Figure size 720x576 with 1 Axes>"

]

},

"metadata": {

"needs\_background": "light"

},

"output\_type": "display\_data"

}

],

"source": [

"fig=plt.figure(figsize=(10,8))\n",

"plt.title('Top 20 Centers with Highest No. of Orders',fontdict={'fontsize':14})\n",

"sns.barplot(y='num\_orders', x='center\_id', data=train.groupby(['center\_id','center\_type']).num\_orders.sum().sort\_values(ascending=False).reset\_index().head(20),palette='YlOrRd\_r',order=list(train.groupby(['center\_id','center\_type']).num\_orders.sum().sort\_values(ascending=False).reset\_index().head(20)['center\_id']));\n",

"plt.ylabel('No. of Orders',fontdict={'fontsize':12})\n",

"plt.xlabel('Center ID',fontdict={'fontsize':12})\n",

"sns.despine(bottom = True, left = True);\n"

]

},

{

"cell\_type": "markdown",

"metadata": {},

"source": [

"Initially, when we checked, which Center Type has the highest number of Orders, We found that Center Type\_A has the highest number of orders, but now when we check individually, we could see that Center 13 of Type\_B has the highest number of Orders. Let’s analyze the reason behind that."

]

},

{

"cell\_type": "code",

"execution\_count": 30,

"metadata": {},

"outputs": [

{

"data": {

"image/png": "\n",

"text/plain": [

"<Figure size 288x504 with 1 Axes>"

]

},

"metadata": {

"needs\_background": "light"

},

"output\_type": "display\_data"

}

],

"source": [

"fig=plt.figure(figsize=(4,7))\n",

"plt.title('Total No. of Centers under Each Center type',fontdict={'fontsize':13})\n",

"sns.barplot(y=train.groupby(['center\_id','center\_type']).num\_orders.sum().reset\_index()['center\_type'].value\_counts(), x=train.groupby(['center\_id','center\_type']).num\_orders.sum().reset\_index()['center\_type'].value\_counts().index,palette='autumn');\n",

"plt.ylabel('No. of Centers',fontdict={'fontsize':12})\n",

"plt.xlabel('Center Type',fontdict={'fontsize':12})\n",

"sns.despine(bottom = True, left = True);"

]

},

{

"cell\_type": "markdown",

"metadata": {},

"source": [

"Type\_A has the most number of orders because, Type\_A has the most number of Centers - 43 Centers."

]

},

{

"cell\_type": "code",

"execution\_count": 31,

"metadata": {},

"outputs": [

{

"data": {

"image/png": "\n",

"text/plain": [

"<Figure size 576x576 with 1 Axes>"

]

},

"metadata": {},

"output\_type": "display\_data"

}

],

"source": [

"sns.set\_style(\"white\")\n",

"plt.figure(figsize=(8,8))\n",

"sns.scatterplot(y=train['base\_price']-train['checkout\_price'],x=train['num\_orders'],color='coral')\n",

"plt.ylabel('Discount (Base price - Checkout Price)')\n",

"sns.despine(bottom = True, left = True)"

]

},

{

"cell\_type": "markdown",

"metadata": {},

"source": [

"We created a new feature: Discount which is the difference of base price and checkout price and tried to find out if there is any relationship between the discount and the number of orders. But surprisingly there are no good correlation between the discount and the number of orders."

]

},

{

"cell\_type": "code",

"execution\_count": 32,

"metadata": {},

"outputs": [

{

"data": {

"image/png": "\n",

"text/plain": [

"<Figure size 1152x504 with 1 Axes>"

]

},

"metadata": {},

"output\_type": "display\_data"

}

],

"source": [

"fig=plt.figure(figsize=(16,7))\n",

"sns.set\_style(\"whitegrid\")\n",

"plt.title('Pattern of Orders',fontdict={'fontsize':14})\n",

"\n",

"sns.pointplot(x=train.groupby('week').sum().reset\_index()['week'],y=train.groupby('week').sum().reset\_index()['num\_orders'],color='coral')\n",

"plt.xticks(range(0, 150, 10), list(np.arange(0,160,10)))\n",

"plt.ylabel('No. of Orders',fontdict={'fontsize':12})\n",

"plt.xlabel('Week',fontdict={'fontsize':12})\n",

"sns.despine(bottom = True, left = True);"

]

},

{

"cell\_type": "markdown",

"metadata": {},

"source": [

"When we analysed the trend of order placed over the weeks, we could see that the highest number of orders were received in week 48 and the lowest in week 62."

]

},

{

"cell\_type": "code",

"execution\_count": 33,

"metadata": {},

"outputs": [

{

"data": {

"image/png": "\n",

"text/plain": [

"<Figure size 432x432 with 1 Axes>"

]

},

"metadata": {},

"output\_type": "display\_data"

}

],

"source": [

"plt.figure(figsize=(6,6))\n",

"colors = ['coral','#FFDAB9','yellowgreen','#6495ED']\n",

"plt.pie(train.groupby(['cuisine']).num\_orders.sum(),\n",

" labels=train.groupby(['cuisine']).num\_orders.sum().index,\n",

" shadow=False,\n",

" colors=colors,\n",

" explode=(0.05, 0.05, 0.03,0.05),\n",

" startangle=90, \n",

" autopct='%1.1f%%',pctdistance=0.6,\n",

" textprops={'fontsize': 12})\n",

"plt.title('Total Number of Orders for Each Category')\n",

"plt.tight\_layout()\n",

"plt.show()"

]

},

{

"cell\_type": "markdown",

"metadata": {},

"source": [

"Italian Cuisine has the highest number of orders with Continental cuisine being the least."

]

},

{

"cell\_type": "code",

"execution\_count": 34,

"metadata": {},

"outputs": [

{

"data": {

"image/png": "\n",

"text/plain": [

"<Figure size 792x576 with 1 Axes>"

]

},

"metadata": {},

"output\_type": "display\_data"

}

],

"source": [

"fig=plt.figure(figsize=(11,8))\n",

"sns.set\_style(\"white\")\n",

"\n",

"plt.xticks(rotation=90,fontsize=12)\n",

"plt.title('Total Number of Orders for Each Category',fontdict={'fontsize':14})\n",

"sns.barplot(y='num\_orders', x='category', data=train.groupby('category').num\_orders.sum().sort\_values(ascending=False).reset\_index(),palette='YlOrRd\_r');\n",

"plt.ylabel('No. of Orders',fontdict={'fontsize':12})\n",

"plt.xlabel('Category',fontdict={'fontsize':12})\n",

"sns.despine(bottom = True, left = True);\n"

]

},

{

"cell\_type": "markdown",

"metadata": {},

"source": [

"We could see that Beverages are the food category which has the higest number of orders and Biriyani is the food category with least number of orders."

]

},

{

"cell\_type": "code",

"execution\_count": 35,

"metadata": {},

"outputs": [

{

"data": {

"image/png": "

QkG2bev+/l7H41HJZFLj8VgXF58/vymVSnIcR5ZlnXorAADAb/BXDgAAcBYGg4EajYZM09Rms1GtVtNisVA8Hv/SOsvlUo7jaDabKRAI/KNpAQA4HzQmAADAWUin07JtW4Zh6OPjQ71e78uhxPPzszzP08vLC6EEAAB/CY0JAAAAAADgGz5+CQAAAAAAfEMwAQAAAAAAfEMwAQAAAAAAfEMwAQAAAAAAfEMwAQAAAAAAfEMwAQAAAAAAfPMNW1+lOVV4gEkAAAAASUVORK5CYII=\n",

"text/plain": [

"<Figure size 1296x576 with 1 Axes>"

]

},

"metadata": {},

"output\_type": "display\_data"

}

],

"source": [

"fig=plt.figure(figsize=(18,8))\n",

"sns.set\_style(\"white\")\n",

"plt.xticks(rotation=90,fontsize=12)\n",

"plt.title('Total Number of Orders for Each Cuisine-Category',fontdict={'fontsize':14})\n",

"\n",

"sns.barplot(x='category',y='num\_orders',data=train.groupby(['cuisine','category']).sum().sort\_values(by='num\_orders', ascending=False).reset\_index(),hue='cuisine',palette='YlOrRd\_r')\n",

"\n",

"plt.ylabel('No. of Orders',fontdict={'fontsize':12})\n",

"plt.xlabel('Cuisine-Category',fontdict={'fontsize':12})\n",

"sns.despine(bottom = True, left = True);"

]

},

{

"cell\_type": "markdown",

"metadata": {},

"source": [

"Similary when we checked which specific cuisne-food category has the highest number of orders, we could see that Indian-Rice Bowl has the highest number of orders and Indian-Biriyani has the least."

]

},

{

"cell\_type": "code",

"execution\_count": 36,

"metadata": {},

"outputs": [

{

"data": {

"text/plain": [

"[56, 34, 77, 85, 71, 23, 93, 35]"

]

},

"execution\_count": 36,

"metadata": {},

"output\_type": "execute\_result"

}

],

"source": [

"list(data.groupby('region\_code').num\_orders.sum().sort\_values(ascending=False).reset\_index().values[:,0])"

]

},

{

"cell\_type": "code",

"execution\_count": 37,

"metadata": {},

"outputs": [

{

"data": {

"image/png": "\n",

"text/plain": [

"<Figure size 576x576 with 1 Axes>"

]

},

"metadata": {},

"output\_type": "display\_data"

}

],

"source": [

"fig=plt.figure(figsize=(8,8))\n",

"sns.set\_style(\"white\")\n",

"plt.xticks(fontsize=13)\n",

"plt.title('Total Number of Orders for Each Region',fontdict={'fontsize':14})\n",

"sns.barplot(y='num\_orders', x='region\_code', data=data.groupby('region\_code').num\_orders.sum().sort\_values(ascending=False).reset\_index(),palette='YlOrRd\_r',order=list(data.groupby('region\_code').num\_orders.sum().sort\_values(ascending=False).reset\_index().values[:,0]));\n",

"plt.ylabel('No. of Orders',fontdict={'fontsize':12})\n",

"plt.xlabel('Region',fontdict={'fontsize':12})\n",

"plt.xticks()\n",

"sns.despine(bottom = True, left = True);\n"

]

},

{

"cell\_type": "markdown",

"metadata": {},

"source": [

"Also when we checked the number of orders with respect to Region, we could see that Region - 56 has the highest number of orders - 60.5M orders which is almost 35M orders higher than the Region with second highest number of orders - Region 34 - 24M orders. "

]

},

{

"cell\_type": "code",

"execution\_count": 38,

"metadata": {},

"outputs": [

{

"data": {

"image/png": "\n",

"text/plain": [

"<Figure size 1296x576 with 1 Axes>"

]

},

"metadata": {},

"output\_type": "display\_data"

}

],

"source": [

"fig=plt.figure(figsize=(18,8))\n",

"sns.set\_style(\"white\")\n",

"plt.xticks(rotation=90,fontsize=13)\n",

"plt.title('Total Number of Orders for each Meal ID',fontdict={'fontsize':14})\n",

"sns.barplot(y='num\_orders', x='meal\_id', data=data.groupby('meal\_id').num\_orders.sum().sort\_values(ascending=False).reset\_index(),palette='YlOrRd\_r',order=list(data.groupby('meal\_id').num\_orders.sum().sort\_values(ascending=False).reset\_index()['meal\_id'].values));\n",

"plt.ylabel('No. of Orders',fontdict={'fontsize':12})\n",

"plt.xlabel('Meal',fontdict={'fontsize':12})\n",

"sns.despine(bottom = True, left = True);\n"

]

},

{

"cell\_type": "markdown",

"metadata": {},

"source": [

"Meal ID 2290 has the higest number of Orders. There is not much significant differences between number of orders for different Meal IDs."

]

},

{

"cell\_type": "code",

"execution\_count": 39,

"metadata": {},

"outputs": [

{

"data": {

"image/png": "\n",

"text/plain": [

"<Figure size 1296x576 with 1 Axes>"

]

},

"metadata": {},

"output\_type": "display\_data"

}

],

"source": [

"fig=plt.figure(figsize=(18,8))\n",

"sns.set\_style(\"white\")\n",

"plt.xticks(rotation=90,fontsize=13)\n",

"plt.title('Total Number of Orders for each City',fontdict={'fontsize':14})\n",

"sns.barplot(y='num\_orders', x='city\_code', data=train.groupby('city\_code').num\_orders.sum().sort\_values(ascending=False).reset\_index(),palette='YlOrRd\_r',order=list(train.groupby('city\_code').num\_orders.sum().sort\_values(ascending=False).reset\_index()['city\_code'].values));\n",

"plt.ylabel('No. of Orders',fontdict={'fontsize':12})\n",

"plt.xlabel('City',fontdict={'fontsize':12})\n",

"sns.despine(bottom = True, left = True);\n"

]

},

{

"cell\_type": "markdown",

"metadata": {},

"source": [

"Also when we checked the number of orders with respect to City, we could see that City - 590 has the highest number of orders - 18.5M orders which is almost 10M orders higher than the City with second highest number of orders - City 526 - 8.6M orders."

]

},

{

"cell\_type": "markdown",

"metadata": {},

"source": [

"##### Encoding City "

]

},

{

"cell\_type": "markdown",

"metadata": {},

"source": [

"As per our observation from our barchart of the City against the number of orders. There the high significant difference between the Top 3 cities which has the highest number of orders. Therefore, in our first approach we will encode the City with Highest No. of Orders as CH1, City with 2nd Highest No. of Orders as CH2 and City with 3rd Highest No. of Orders as CH3 and rest all of the cities which does not have much significant differences between the number of orders as CH4."

]

},

{

"cell\_type": "code",

"execution\_count": 40,

"metadata": {},

"outputs": [],

"source": [

"city4={590:'CH1', 526:'CH2', 638:'CH3'}\n",

"data['city\_enc\_4']=data['city\_code'].map(city4)\n",

"data['city\_enc\_4']=data['city\_enc\_4'].fillna('CH4')"

]

},

{

"cell\_type": "code",

"execution\_count": 41,

"metadata": {},

"outputs": [

{

"data": {

"text/plain": [

"CH4 362341\n",

"CH1 58708\n",

"CH2 46589\n",

"CH3 21483\n",

"Name: city\_enc\_4, dtype: int64"

]

},

"execution\_count": 41,

"metadata": {},

"output\_type": "execute\_result"

}

],

"source": [

"data['city\_enc\_4'].value\_counts()"

]

},

{

"cell\_type": "code",

"execution\_count": 42,

"metadata": {},

"outputs": [

{

"data": {

"text/html": [

"<div>\n",

"<style scoped>\n",

" .dataframe tbody tr th:only-of-type {\n",

" vertical-align: middle;\n",

" }\n",

"\n",

" .dataframe tbody tr th {\n",

" vertical-align: top;\n",

" }\n",

"\n",

" .dataframe thead th {\n",

" text-align: right;\n",

" }\n",

"</style>\n",

"<table border=\"1\" class=\"dataframe\">\n",

" <thead>\n",

" <tr style=\"text-align: right;\">\n",

" <th></th>\n",

" <th>id</th>\n",

" <th>week</th>\n",

" <th>center\_id</th>\n",

" <th>meal\_id</th>\n",

" <th>checkout\_price</th>\n",

" <th>base\_price</th>\n",

" <th>emailer\_for\_promotion</th>\n",

" <th>homepage\_featured</th>\n",

" <th>num\_orders</th>\n",

" <th>city\_code</th>\n",

" <th>region\_code</th>\n",

" <th>center\_type</th>\n",

" <th>op\_area</th>\n",

" <th>category</th>\n",

" <th>cuisine</th>\n",

" <th>discount amount</th>\n",

" <th>discount percent</th>\n",

" <th>discount y/n</th>\n",

" <th>compare\_week\_price</th>\n",

" <th>compare\_week\_price y/n</th>\n",

" <th>city\_enc\_4</th>\n",

" </tr>\n",

" </thead>\n",

" <tbody>\n",

" <tr>\n",

" <th>0</th>\n",

" <td>1379560</td>\n",

" <td>1</td>\n",

" <td>55</td>\n",

" <td>1885</td>\n",

" <td>136.83</td>\n",

" <td>152.29</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>177</td>\n",

" <td>647</td>\n",

" <td>56</td>\n",

" <td>TYPE\_C</td>\n",

" <td>2.0</td>\n",

" <td>Beverages</td>\n",

" <td>Thai</td>\n",

" <td>15.46</td>\n",

" <td>10.151684</td>\n",

" <td>1</td>\n",

" <td>0.0</td>\n",

" <td>0</td>\n",

" <td>CH4</td>\n",

" </tr>\n",

" <tr>\n",

" <th>1</th>\n",

" <td>1466964</td>\n",

" <td>1</td>\n",

" <td>55</td>\n",

" <td>1993</td>\n",

" <td>136.83</td>\n",

" <td>135.83</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>270</td>\n",

" <td>647</td>\n",

" <td>56</td>\n",

" <td>TYPE\_C</td>\n",

" <td>2.0</td>\n",

" <td>Beverages</td>\n",

" <td>Thai</td>\n",

" <td>-1.00</td>\n",

" <td>-0.736214</td>\n",

" <td>0</td>\n",

" <td>0.0</td>\n",

" <td>0</td>\n",

" <td>CH4</td>\n",

" </tr>\n",

" <tr>\n",

" <th>2</th>\n",

" <td>1346989</td>\n",

" <td>1</td>\n",

" <td>55</td>\n",

" <td>2539</td>\n",

" <td>134.86</td>\n",

" <td>135.86</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>189</td>\n",

" <td>647</td>\n",

" <td>56</td>\n",

" <td>TYPE\_C</td>\n",

" <td>2.0</td>\n",

" <td>Beverages</td>\n",

" <td>Thai</td>\n",

" <td>1.00</td>\n",

" <td>0.736052</td>\n",

" <td>1</td>\n",

" <td>0.0</td>\n",

" <td>0</td>\n",

" <td>CH4</td>\n",

" </tr>\n",

" <tr>\n",

" <th>3</th>\n",

" <td>1338232</td>\n",

" <td>1</td>\n",

" <td>55</td>\n",

" <td>2139</td>\n",

" <td>339.50</td>\n",

" <td>437.53</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>54</td>\n",

" <td>647</td>\n",

" <td>56</td>\n",

" <td>TYPE\_C</td>\n",

" <td>2.0</td>\n",

" <td>Beverages</td>\n",

" <td>Indian</td>\n",

" <td>98.03</td>\n",

" <td>22.405321</td>\n",

" <td>1</td>\n",

" <td>0.0</td>\n",

" <td>0</td>\n",

" <td>CH4</td>\n",

" </tr>\n",

" <tr>\n",

" <th>4</th>\n",

" <td>1448490</td>\n",

" <td>1</td>\n",

" <td>55</td>\n",

" <td>2631</td>\n",

" <td>243.50</td>\n",

" <td>242.50</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>40</td>\n",

" <td>647</td>\n",

" <td>56</td>\n",

" <td>TYPE\_C</td>\n",

" <td>2.0</td>\n",

" <td>Beverages</td>\n",

" <td>Indian</td>\n",

" <td>-1.00</td>\n",

" <td>-0.412371</td>\n",

" <td>0</td>\n",

" <td>0.0</td>\n",

" <td>0</td>\n",

" <td>CH4</td>\n",

" </tr>\n",

" </tbody>\n",

"</table>\n",

"</div>"

],

"text/plain": [

" id week center\_id meal\_id checkout\_price base\_price \\\n",

"0 1379560 1 55 1885 136.83 152.29 \n",

"1 1466964 1 55 1993 136.83 135.83 \n",

"2 1346989 1 55 2539 134.86 135.86 \n",

"3 1338232 1 55 2139 339.50 437.53 \n",

"4 1448490 1 55 2631 243.50 242.50 \n",

"\n",

" emailer\_for\_promotion homepage\_featured num\_orders city\_code \\\n",

"0 0 0 177 647 \n",

"1 0 0 270 647 \n",

"2 0 0 189 647 \n",

"3 0 0 54 647 \n",

"4 0 0 40 647 \n",

"\n",

" region\_code center\_type op\_area category cuisine discount amount \\\n",

"0 56 TYPE\_C 2.0 Beverages Thai 15.46 \n",

"1 56 TYPE\_C 2.0 Beverages Thai -1.00 \n",

"2 56 TYPE\_C 2.0 Beverages Thai 1.00 \n",

"3 56 TYPE\_C 2.0 Beverages Indian 98.03 \n",

"4 56 TYPE\_C 2.0 Beverages Indian -1.00 \n",

"\n",

" discount percent discount y/n compare\_week\_price compare\_week\_price y/n \\\n",

"0 10.151684 1 0.0 0 \n",

"1 -0.736214 0 0.0 0 \n",

"2 0.736052 1 0.0 0 \n",

"3 22.405321 1 0.0 0 \n",

"4 -0.412371 0 0.0 0 \n",

"\n",

" city\_enc\_4 \n",

"0 CH4 \n",

"1 CH4 \n",

"2 CH4 \n",

"3 CH4 \n",

"4 CH4 "

]

},

"execution\_count": 42,

"metadata": {},

"output\_type": "execute\_result"

}

],

"source": [

"data.head()"

]

},

{

"cell\_type": "code",

"execution\_count": 43,

"metadata": {},

"outputs": [

{

"data": {

"text/plain": [

"id 0\n",

"week 0\n",

"center\_id 0\n",

"meal\_id 0\n",

"checkout\_price 0\n",

"base\_price 0\n",

"emailer\_for\_promotion 0\n",

"homepage\_featured 0\n",

"num\_orders 0\n",

"city\_code 0\n",

"region\_code 0\n",

"center\_type 0\n",

"op\_area 0\n",

"category 0\n",

"cuisine 0\n",

"discount amount 0\n",

"discount percent 0\n",

"discount y/n 0\n",

"compare\_week\_price 0\n",

"compare\_week\_price y/n 0\n",

"city\_enc\_4 0\n",

"dtype: int64"

]

},

"execution\_count": 43,

"metadata": {},

"output\_type": "execute\_result"

}

],

"source": [

"data.isnull().sum()"

]

},

{

"cell\_type": "markdown",

"metadata": {},

"source": [

"##### Copying to New DataFrame"

]

},

{

"cell\_type": "code",

"execution\_count": 44,

"metadata": {},

"outputs": [

{

"data": {

"text/html": [

"<div>\n",

"<style scoped>\n",

" .dataframe tbody tr th:only-of-type {\n",

" vertical-align: middle;\n",

" }\n",

"\n",

" .dataframe tbody tr th {\n",

" vertical-align: top;\n",

" }\n",

"\n",

" .dataframe thead th {\n",

" text-align: right;\n",

" }\n",

"</style>\n",

"<table border=\"1\" class=\"dataframe\">\n",

" <thead>\n",

" <tr style=\"text-align: right;\">\n",

" <th></th>\n",

" <th>id</th>\n",

" <th>week</th>\n",

" <th>center\_id</th>\n",

" <th>meal\_id</th>\n",

" <th>checkout\_price</th>\n",

" <th>base\_price</th>\n",

" <th>emailer\_for\_promotion</th>\n",

" <th>homepage\_featured</th>\n",

" <th>num\_orders</th>\n",

" <th>city\_code</th>\n",

" <th>region\_code</th>\n",

" <th>center\_type</th>\n",

" <th>op\_area</th>\n",

" <th>category</th>\n",

" <th>cuisine</th>\n",

" <th>discount amount</th>\n",

" <th>discount percent</th>\n",

" <th>discount y/n</th>\n",

" <th>compare\_week\_price</th>\n",

" <th>compare\_week\_price y/n</th>\n",

" <th>city\_enc\_4</th>\n",

" </tr>\n",

" </thead>\n",

" <tbody>\n",

" <tr>\n",

" <th>0</th>\n",

" <td>1379560</td>\n",

" <td>1</td>\n",

" <td>55</td>\n",

" <td>1885</td>\n",

" <td>136.83</td>\n",

" <td>152.29</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>177</td>\n",

" <td>647</td>\n",

" <td>56</td>\n",

" <td>TYPE\_C</td>\n",

" <td>2.0</td>\n",

" <td>Beverages</td>\n",

" <td>Thai</td>\n",

" <td>15.46</td>\n",

" <td>10.151684</td>\n",

" <td>1</td>\n",

" <td>0.0</td>\n",

" <td>0</td>\n",

" <td>CH4</td>\n",

" </tr>\n",

" <tr>\n",

" <th>1</th>\n",

" <td>1466964</td>\n",

" <td>1</td>\n",

" <td>55</td>\n",

" <td>1993</td>\n",

" <td>136.83</td>\n",

" <td>135.83</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>270</td>\n",

" <td>647</td>\n",

" <td>56</td>\n",

" <td>TYPE\_C</td>\n",

" <td>2.0</td>\n",

" <td>Beverages</td>\n",

" <td>Thai</td>\n",

" <td>-1.00</td>\n",

" <td>-0.736214</td>\n",

" <td>0</td>\n",

" <td>0.0</td>\n",

" <td>0</td>\n",

" <td>CH4</td>\n",

" </tr>\n",

" <tr>\n",

" <th>2</th>\n",

" <td>1346989</td>\n",

" <td>1</td>\n",

" <td>55</td>\n",

" <td>2539</td>\n",

" <td>134.86</td>\n",

" <td>135.86</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>189</td>\n",

" <td>647</td>\n",

" <td>56</td>\n",

" <td>TYPE\_C</td>\n",

" <td>2.0</td>\n",

" <td>Beverages</td>\n",

" <td>Thai</td>\n",

" <td>1.00</td>\n",

" <td>0.736052</td>\n",

" <td>1</td>\n",

" <td>0.0</td>\n",

" <td>0</td>\n",

" <td>CH4</td>\n",

" </tr>\n",

" <tr>\n",

" <th>3</th>\n",

" <td>1338232</td>\n",

" <td>1</td>\n",

" <td>55</td>\n",

" <td>2139</td>\n",

" <td>339.50</td>\n",

" <td>437.53</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>54</td>\n",

" <td>647</td>\n",

" <td>56</td>\n",

" <td>TYPE\_C</td>\n",

" <td>2.0</td>\n",

" <td>Beverages</td>\n",

" <td>Indian</td>\n",

" <td>98.03</td>\n",

" <td>22.405321</td>\n",

" <td>1</td>\n",

" <td>0.0</td>\n",

" <td>0</td>\n",

" <td>CH4</td>\n",

" </tr>\n",

" <tr>\n",

" <th>4</th>\n",

" <td>1448490</td>\n",

" <td>1</td>\n",

" <td>55</td>\n",

" <td>2631</td>\n",

" <td>243.50</td>\n",

" <td>242.50</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>40</td>\n",

" <td>647</td>\n",

" <td>56</td>\n",

" <td>TYPE\_C</td>\n",

" <td>2.0</td>\n",

" <td>Beverages</td>\n",

" <td>Indian</td>\n",

" <td>-1.00</td>\n",

" <td>-0.412371</td>\n",

" <td>0</td>\n",

" <td>0.0</td>\n",

" <td>0</td>\n",

" <td>CH4</td>\n",

" </tr>\n",

" </tbody>\n",

"</table>\n",

"</div>"

],

"text/plain": [

" id week center\_id meal\_id checkout\_price base\_price \\\n",

"0 1379560 1 55 1885 136.83 152.29 \n",

"1 1466964 1 55 1993 136.83 135.83 \n",

"2 1346989 1 55 2539 134.86 135.86 \n",

"3 1338232 1 55 2139 339.50 437.53 \n",

"4 1448490 1 55 2631 243.50 242.50 \n",

"\n",

" emailer\_for\_promotion homepage\_featured num\_orders city\_code \\\n",

"0 0 0 177 647 \n",

"1 0 0 270 647 \n",

"2 0 0 189 647 \n",

"3 0 0 54 647 \n",

"4 0 0 40 647 \n",

"\n",

" region\_code center\_type op\_area category cuisine discount amount \\\n",

"0 56 TYPE\_C 2.0 Beverages Thai 15.46 \n",

"1 56 TYPE\_C 2.0 Beverages Thai -1.00 \n",

"2 56 TYPE\_C 2.0 Beverages Thai 1.00 \n",

"3 56 TYPE\_C 2.0 Beverages Indian 98.03 \n",

"4 56 TYPE\_C 2.0 Beverages Indian -1.00 \n",

"\n",

" discount percent discount y/n compare\_week\_price compare\_week\_price y/n \\\n",

"0 10.151684 1 0.0 0 \n",

"1 -0.736214 0 0.0 0 \n",

"2 0.736052 1 0.0 0 \n",

"3 22.405321 1 0.0 0 \n",

"4 -0.412371 0 0.0 0 \n",

"\n",

" city\_enc\_4 \n",

"0 CH4 \n",

"1 CH4 \n",

"2 CH4 \n",

"3 CH4 \n",

"4 CH4 "

]

},

"execution\_count": 44,

"metadata": {},

"output\_type": "execute\_result"

}

],

"source": [

"datax=data.copy()\n",

"datax.head()"

]

},

{

"cell\_type": "markdown",

"metadata": {},

"source": [

"##### Enoding all the Categorical Features"

]

},

{

"cell\_type": "code",

"execution\_count": 45,

"metadata": {},

"outputs": [],

"source": [

"datax['center\_id']=datax['center\_id'].astype('object')\n",

"datax['meal\_id']=datax['meal\_id'].astype('object')\n",

"datax['region\_code']=datax['region\_code'].astype('object')"

]

},

{

"cell\_type": "code",

"execution\_count": 46,

"metadata": {},

"outputs": [],

"source": [

"obj=datax[['center\_id','meal\_id','region\_code','center\_type','category','cuisine','city\_enc\_4']]\n",

"num=datax.drop(['center\_id','meal\_id','region\_code','center\_type','category','cuisine','city\_enc\_4'],axis=1)"

]

},

{

"cell\_type": "code",

"execution\_count": 47,

"metadata": {},

"outputs": [],

"source": [

"encode1=pd.get\_dummies(obj,drop\_first = True)"

]

},

{

"cell\_type": "code",

"execution\_count": 48,

"metadata": {},

"outputs": [],

"source": [

"datax=pd.concat([num,encode1],axis=1)"

]

},

{

"cell\_type": "code",

"execution\_count": 49,

"metadata": {},

"outputs": [

{

"data": {

"text/html": [

"<div>\n",

"<style scoped>\n",

" .dataframe tbody tr th:only-of-type {\n",

" vertical-align: middle;\n",

" }\n",

"\n",

" .dataframe tbody tr th {\n",

" vertical-align: top;\n",

" }\n",

"\n",

" .dataframe thead th {\n",

" text-align: right;\n",

" }\n",

"</style>\n",

"<table border=\"1\" class=\"dataframe\">\n",

" <thead>\n",

" <tr style=\"text-align: right;\">\n",

" <th></th>\n",

" <th>id</th>\n",

" <th>week</th>\n",

" <th>checkout\_price</th>\n",

" <th>base\_price</th>\n",

" <th>emailer\_for\_promotion</th>\n",

" <th>homepage\_featured</th>\n",

" <th>num\_orders</th>\n",

" <th>city\_code</th>\n",

" <th>op\_area</th>\n",

" <th>discount amount</th>\n",

" <th>discount percent</th>\n",

" <th>discount y/n</th>\n",

" <th>compare\_week\_price</th>\n",

" <th>compare\_week\_price y/n</th>\n",

" <th>center\_id\_11</th>\n",

" <th>center\_id\_13</th>\n",

" <th>center\_id\_14</th>\n",

" <th>center\_id\_17</th>\n",

" <th>center\_id\_20</th>\n",

" <th>center\_id\_23</th>\n",

" <th>center\_id\_24</th>\n",

" <th>center\_id\_26</th>\n",

" <th>center\_id\_27</th>\n",

" <th>center\_id\_29</th>\n",

" <th>center\_id\_30</th>\n",

" <th>center\_id\_32</th>\n",

" <th>center\_id\_34</th>\n",

" <th>center\_id\_36</th>\n",

" <th>center\_id\_39</th>\n",

" <th>center\_id\_41</th>\n",

" <th>center\_id\_42</th>\n",

" <th>center\_id\_43</th>\n",

" <th>center\_id\_50</th>\n",

" <th>center\_id\_51</th>\n",

" <th>center\_id\_52</th>\n",

" <th>center\_id\_53</th>\n",

" <th>center\_id\_55</th>\n",

" <th>center\_id\_57</th>\n",

" <th>center\_id\_58</th>\n",

" <th>center\_id\_59</th>\n",

" <th>center\_id\_61</th>\n",

" <th>center\_id\_64</th>\n",

" <th>center\_id\_65</th>\n",

" <th>center\_id\_66</th>\n",

" <th>center\_id\_67</th>\n",

" <th>center\_id\_68</th>\n",

" <th>center\_id\_72</th>\n",

" <th>center\_id\_73</th>\n",

" <th>center\_id\_74</th>\n",

" <th>center\_id\_75</th>\n",

" <th>center\_id\_76</th>\n",

" <th>center\_id\_77</th>\n",

" <th>center\_id\_80</th>\n",

" <th>center\_id\_81</th>\n",

" <th>center\_id\_83</th>\n",

" <th>center\_id\_86</th>\n",

" <th>center\_id\_88</th>\n",

" <th>center\_id\_89</th>\n",

" <th>center\_id\_91</th>\n",

" <th>center\_id\_92</th>\n",

" <th>center\_id\_93</th>\n",

" <th>center\_id\_94</th>\n",

" <th>center\_id\_97</th>\n",

" <th>center\_id\_99</th>\n",

" <th>center\_id\_101</th>\n",

" <th>center\_id\_102</th>\n",

" <th>center\_id\_104</th>\n",

" <th>center\_id\_106</th>\n",

" <th>center\_id\_108</th>\n",

" <th>center\_id\_109</th>\n",

" <th>center\_id\_110</th>\n",

" <th>center\_id\_113</th>\n",

" <th>center\_id\_124</th>\n",

" <th>center\_id\_126</th>\n",

" <th>center\_id\_129</th>\n",

" <th>center\_id\_132</th>\n",

" <th>center\_id\_137</th>\n",

" <th>center\_id\_139</th>\n",

" <th>center\_id\_143</th>\n",

" <th>center\_id\_145</th>\n",

" <th>center\_id\_146</th>\n",

" <th>center\_id\_149</th>\n",

" <th>center\_id\_152</th>\n",

" <th>center\_id\_153</th>\n",

" <th>center\_id\_157</th>\n",

" <th>center\_id\_161</th>\n",

" <th>center\_id\_162</th>\n",

" <th>center\_id\_174</th>\n",

" <th>center\_id\_177</th>\n",

" <th>center\_id\_186</th>\n",

" <th>meal\_id\_1109</th>\n",

" <th>meal\_id\_1198</th>\n",

" <th>meal\_id\_1207</th>\n",

" <th>meal\_id\_1216</th>\n",

" <th>meal\_id\_1230</th>\n",

" <th>meal\_id\_1247</th>\n",

" <th>meal\_id\_1248</th>\n",

" <th>meal\_id\_1311</th>\n",

" <th>meal\_id\_1438</th>\n",

" <th>meal\_id\_1445</th>\n",

" <th>meal\_id\_1525</th>\n",

" <th>meal\_id\_1543</th>\n",

" <th>meal\_id\_1558</th>\n",

" <th>meal\_id\_1571</th>\n",

" <th>meal\_id\_1727</th>\n",

" <th>meal\_id\_1754</th>\n",

" <th>meal\_id\_1770</th>\n",

" <th>meal\_id\_1778</th>\n",

" <th>meal\_id\_1803</th>\n",

" <th>meal\_id\_1847</th>\n",

" <th>meal\_id\_1878</th>\n",

" <th>meal\_id\_1885</th>\n",

" <th>meal\_id\_1902</th>\n",

" <th>meal\_id\_1962</th>\n",

" <th>meal\_id\_1971</th>\n",

" <th>meal\_id\_1993</th>\n",

" <th>meal\_id\_2104</th>\n",

" <th>meal\_id\_2126</th>\n",

" <th>meal\_id\_2139</th>\n",

" <th>meal\_id\_2290</th>\n",

" <th>meal\_id\_2304</th>\n",

" <th>meal\_id\_2306</th>\n",

" <th>meal\_id\_2322</th>\n",

" <th>meal\_id\_2444</th>\n",

" <th>meal\_id\_2490</th>\n",

" <th>meal\_id\_2492</th>\n",

" <th>meal\_id\_2494</th>\n",

" <th>meal\_id\_2539</th>\n",

" <th>meal\_id\_2569</th>\n",

" <th>meal\_id\_2577</th>\n",

" <th>meal\_id\_2581</th>\n",

" <th>meal\_id\_2631</th>\n",

" <th>meal\_id\_2640</th>\n",

" <th>meal\_id\_2664</th>\n",

" <th>meal\_id\_2704</th>\n",

" <th>meal\_id\_2707</th>\n",

" <th>meal\_id\_2760</th>\n",

" <th>meal\_id\_2826</th>\n",

" <th>meal\_id\_2867</th>\n",

" <th>meal\_id\_2956</th>\n",

" <th>region\_code\_34</th>\n",

" <th>region\_code\_35</th>\n",

" <th>region\_code\_56</th>\n",

" <th>region\_code\_71</th>\n",

" <th>region\_code\_77</th>\n",

" <th>region\_code\_85</th>\n",

" <th>region\_code\_93</th>\n",

" <th>center\_type\_TYPE\_B</th>\n",

" <th>center\_type\_TYPE\_C</th>\n",

" <th>category\_Biryani</th>\n",

" <th>category\_Desert</th>\n",

" <th>category\_Extras</th>\n",

" <th>category\_Fish</th>\n",

" <th>category\_Other Snacks</th>\n",

" <th>category\_Pasta</th>\n",

" <th>category\_Pizza</th>\n",

" <th>category\_Rice Bowl</th>\n",

" <th>category\_Salad</th>\n",

" <th>category\_Sandwich</th>\n",

" <th>category\_Seafood</th>\n",

" <th>category\_Soup</th>\n",

" <th>category\_Starters</th>\n",

" <th>cuisine\_Indian</th>\n",

" <th>cuisine\_Italian</th>\n",

" <th>cuisine\_Thai</th>\n",

" <th>city\_enc\_4\_CH2</th>\n",

" <th>city\_enc\_4\_CH3</th>\n",

" <th>city\_enc\_4\_CH4</th>\n",

" </tr>\n",

" </thead>\n",

" <tbody>\n",

" <tr>\n",

" <th>0</th>\n",

" <td>1379560</td>\n",

" <td>1</td>\n",

" <td>136.83</td>\n",

" <td>152.29</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>177</td>\n",

" <td>647</td>\n",

" <td>2.0</td>\n",

" <td>15.46</td>\n",

" <td>10.151684</td>\n",

" <td>1</td>\n",

" <td>0.0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" </tr>\n",

" <tr>\n",

" <th>1</th>\n",

" <td>1466964</td>\n",

" <td>1</td>\n",

" <td>136.83</td>\n",

" <td>135.83</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>270</td>\n",

" <td>647</td>\n",

" <td>2.0</td>\n",

" <td>-1.00</td>\n",

" <td>-0.736214</td>\n",

" <td>0</td>\n",

" <td>0.0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" </tr>\n",

" <tr>\n",

" <th>2</th>\n",

" <td>1346989</td>\n",

" <td>1</td>\n",

" <td>134.86</td>\n",

" <td>135.86</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>189</td>\n",

" <td>647</td>\n",

" <td>2.0</td>\n",

" <td>1.00</td>\n",

" <td>0.736052</td>\n",

" <td>1</td>\n",

" <td>0.0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" </tr>\n",

" <tr>\n",

" <th>3</th>\n",

" <td>1338232</td>\n",

" <td>1</td>\n",

" <td>339.50</td>\n",

" <td>437.53</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>54</td>\n",

" <td>647</td>\n",

" <td>2.0</td>\n",

" <td>98.03</td>\n",

" <td>22.405321</td>\n",

" <td>1</td>\n",

" <td>0.0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" </tr>\n",

" <tr>\n",

" <th>4</th>\n",

" <td>1448490</td>\n",

" <td>1</td>\n",

" <td>243.50</td>\n",

" <td>242.50</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>40</td>\n",

" <td>647</td>\n",

" <td>2.0</td>\n",

" <td>-1.00</td>\n",

" <td>-0.412371</td>\n",

" <td>0</td>\n",

" <td>0.0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" </tr>\n",

" </tbody>\n",

"</table>\n",

"</div>"

],

"text/plain": [

" id week checkout\_price base\_price emailer\_for\_promotion \\\n",

"0 1379560 1 136.83 152.29 0 \n",

"1 1466964 1 136.83 135.83 0 \n",

"2 1346989 1 134.86 135.86 0 \n",

"3 1338232 1 339.50 437.53 0 \n",

"4 1448490 1 243.50 242.50 0 \n",

"\n",

" homepage\_featured num\_orders city\_code op\_area discount amount \\\n",

"0 0 177 647 2.0 15.46 \n",

"1 0 270 647 2.0 -1.00 \n",

"2 0 189 647 2.0 1.00 \n",

"3 0 54 647 2.0 98.03 \n",

"4 0 40 647 2.0 -1.00 \n",

"\n",

" discount percent discount y/n compare\_week\_price compare\_week\_price y/n \\\n",

"0 10.151684 1 0.0 0 \n",

"1 -0.736214 0 0.0 0 \n",

"2 0.736052 1 0.0 0 \n",

"3 22.405321 1 0.0 0 \n",

"4 -0.412371 0 0.0 0 \n",

"\n",

" center\_id\_11 center\_id\_13 center\_id\_14 center\_id\_17 center\_id\_20 \\\n",

"0 0 0 0 0 0 \n",

"1 0 0 0 0 0 \n",

"2 0 0 0 0 0 \n",

"3 0 0 0 0 0 \n",

"4 0 0 0 0 0 \n",

"\n",

" center\_id\_23 center\_id\_24 center\_id\_26 center\_id\_27 center\_id\_29 \\\n",

"0 0 0 0 0 0 \n",

"1 0 0 0 0 0 \n",

"2 0 0 0 0 0 \n",

"3 0 0 0 0 0 \n",

"4 0 0 0 0 0 \n",

"\n",

" center\_id\_30 center\_id\_32 center\_id\_34 center\_id\_36 center\_id\_39 \\\n",

"0 0 0 0 0 0 \n",

"1 0 0 0 0 0 \n",

"2 0 0 0 0 0 \n",

"3 0 0 0 0 0 \n",

"4 0 0 0 0 0 \n",

"\n",

" center\_id\_41 center\_id\_42 center\_id\_43 center\_id\_50 center\_id\_51 \\\n",

"0 0 0 0 0 0 \n",

"1 0 0 0 0 0 \n",

"2 0 0 0 0 0 \n",

"3 0 0 0 0 0 \n",

"4 0 0 0 0 0 \n",

"\n",

" center\_id\_52 center\_id\_53 center\_id\_55 center\_id\_57 center\_id\_58 \\\n",

"0 0 0 1 0 0 \n",

"1 0 0 1 0 0 \n",

"2 0 0 1 0 0 \n",

"3 0 0 1 0 0 \n",

"4 0 0 1 0 0 \n",

"\n",

" center\_id\_59 center\_id\_61 center\_id\_64 center\_id\_65 center\_id\_66 \\\n",

"0 0 0 0 0 0 \n",

"1 0 0 0 0 0 \n",

"2 0 0 0 0 0 \n",

"3 0 0 0 0 0 \n",

"4 0 0 0 0 0 \n",

"\n",

" center\_id\_67 center\_id\_68 center\_id\_72 center\_id\_73 center\_id\_74 \\\n",

"0 0 0 0 0 0 \n",

"1 0 0 0 0 0 \n",

"2 0 0 0 0 0 \n",

"3 0 0 0 0 0 \n",

"4 0 0 0 0 0 \n",

"\n",

" center\_id\_75 center\_id\_76 center\_id\_77 center\_id\_80 center\_id\_81 \\\n",

"0 0 0 0 0 0 \n",

"1 0 0 0 0 0 \n",

"2 0 0 0 0 0 \n",

"3 0 0 0 0 0 \n",

"4 0 0 0 0 0 \n",

"\n",

" center\_id\_83 center\_id\_86 center\_id\_88 center\_id\_89 center\_id\_91 \\\n",

"0 0 0 0 0 0 \n",

"1 0 0 0 0 0 \n",

"2 0 0 0 0 0 \n",

"3 0 0 0 0 0 \n",

"4 0 0 0 0 0 \n",

"\n",

" center\_id\_92 center\_id\_93 center\_id\_94 center\_id\_97 center\_id\_99 \\\n",

"0 0 0 0 0 0 \n",

"1 0 0 0 0 0 \n",

"2 0 0 0 0 0 \n",

"3 0 0 0 0 0 \n",

"4 0 0 0 0 0 \n",

"\n",

" center\_id\_101 center\_id\_102 center\_id\_104 center\_id\_106 center\_id\_108 \\\n",

"0 0 0 0 0 0 \n",

"1 0 0 0 0 0 \n",

"2 0 0 0 0 0 \n",

"3 0 0 0 0 0 \n",

"4 0 0 0 0 0 \n",

"\n",

" center\_id\_109 center\_id\_110 center\_id\_113 center\_id\_124 center\_id\_126 \\\n",

"0 0 0 0 0 0 \n",

"1 0 0 0 0 0 \n",

"2 0 0 0 0 0 \n",

"3 0 0 0 0 0 \n",

"4 0 0 0 0 0 \n",

"\n",

" center\_id\_129 center\_id\_132 center\_id\_137 center\_id\_139 center\_id\_143 \\\n",

"0 0 0 0 0 0 \n",

"1 0 0 0 0 0 \n",

"2 0 0 0 0 0 \n",

"3 0 0 0 0 0 \n",

"4 0 0 0 0 0 \n",

"\n",

" center\_id\_145 center\_id\_146 center\_id\_149 center\_id\_152 center\_id\_153 \\\n",

"0 0 0 0 0 0 \n",

"1 0 0 0 0 0 \n",

"2 0 0 0 0 0 \n",

"3 0 0 0 0 0 \n",

"4 0 0 0 0 0 \n",

"\n",

" center\_id\_157 center\_id\_161 center\_id\_162 center\_id\_174 center\_id\_177 \\\n",

"0 0 0 0 0 0 \n",

"1 0 0 0 0 0 \n",

"2 0 0 0 0 0 \n",

"3 0 0 0 0 0 \n",

"4 0 0 0 0 0 \n",

"\n",

" center\_id\_186 meal\_id\_1109 meal\_id\_1198 meal\_id\_1207 meal\_id\_1216 \\\n",

"0 0 0 0 0 0 \n",

"1 0 0 0 0 0 \n",

"2 0 0 0 0 0 \n",

"3 0 0 0 0 0 \n",

"4 0 0 0 0 0 \n",

"\n",

" meal\_id\_1230 meal\_id\_1247 meal\_id\_1248 meal\_id\_1311 meal\_id\_1438 \\\n",

"0 0 0 0 0 0 \n",

"1 0 0 0 0 0 \n",

"2 0 0 0 0 0 \n",

"3 0 0 0 0 0 \n",

"4 0 0 0 0 0 \n",

"\n",

" meal\_id\_1445 meal\_id\_1525 meal\_id\_1543 meal\_id\_1558 meal\_id\_1571 \\\n",

"0 0 0 0 0 0 \n",

"1 0 0 0 0 0 \n",

"2 0 0 0 0 0 \n",

"3 0 0 0 0 0 \n",

"4 0 0 0 0 0 \n",

"\n",

" meal\_id\_1727 meal\_id\_1754 meal\_id\_1770 meal\_id\_1778 meal\_id\_1803 \\\n",

"0 0 0 0 0 0 \n",

"1 0 0 0 0 0 \n",

"2 0 0 0 0 0 \n",

"3 0 0 0 0 0 \n",

"4 0 0 0 0 0 \n",

"\n",

" meal\_id\_1847 meal\_id\_1878 meal\_id\_1885 meal\_id\_1902 meal\_id\_1962 \\\n",

"0 0 0 1 0 0 \n",

"1 0 0 0 0 0 \n",

"2 0 0 0 0 0 \n",

"3 0 0 0 0 0 \n",

"4 0 0 0 0 0 \n",

"\n",

" meal\_id\_1971 meal\_id\_1993 meal\_id\_2104 meal\_id\_2126 meal\_id\_2139 \\\n",

"0 0 0 0 0 0 \n",

"1 0 1 0 0 0 \n",

"2 0 0 0 0 0 \n",

"3 0 0 0 0 1 \n",

"4 0 0 0 0 0 \n",

"\n",

" meal\_id\_2290 meal\_id\_2304 meal\_id\_2306 meal\_id\_2322 meal\_id\_2444 \\\n",

"0 0 0 0 0 0 \n",

"1 0 0 0 0 0 \n",

"2 0 0 0 0 0 \n",

"3 0 0 0 0 0 \n",

"4 0 0 0 0 0 \n",

"\n",

" meal\_id\_2490 meal\_id\_2492 meal\_id\_2494 meal\_id\_2539 meal\_id\_2569 \\\n",

"0 0 0 0 0 0 \n",

"1 0 0 0 0 0 \n",

"2 0 0 0 1 0 \n",

"3 0 0 0 0 0 \n",

"4 0 0 0 0 0 \n",

"\n",

" meal\_id\_2577 meal\_id\_2581 meal\_id\_2631 meal\_id\_2640 meal\_id\_2664 \\\n",

"0 0 0 0 0 0 \n",

"1 0 0 0 0 0 \n",

"2 0 0 0 0 0 \n",

"3 0 0 0 0 0 \n",

"4 0 0 1 0 0 \n",

"\n",

" meal\_id\_2704 meal\_id\_2707 meal\_id\_2760 meal\_id\_2826 meal\_id\_2867 \\\n",

"0 0 0 0 0 0 \n",

"1 0 0 0 0 0 \n",

"2 0 0 0 0 0 \n",

"3 0 0 0 0 0 \n",

"4 0 0 0 0 0 \n",

"\n",

" meal\_id\_2956 region\_code\_34 region\_code\_35 region\_code\_56 \\\n",

"0 0 0 0 1 \n",

"1 0 0 0 1 \n",

"2 0 0 0 1 \n",

"3 0 0 0 1 \n",

"4 0 0 0 1 \n",

"\n",

" region\_code\_71 region\_code\_77 region\_code\_85 region\_code\_93 \\\n",

"0 0 0 0 0 \n",

"1 0 0 0 0 \n",

"2 0 0 0 0 \n",

"3 0 0 0 0 \n",

"4 0 0 0 0 \n",

"\n",

" center\_type\_TYPE\_B center\_type\_TYPE\_C category\_Biryani category\_Desert \\\n",

"0 0 1 0 0 \n",

"1 0 1 0 0 \n",

"2 0 1 0 0 \n",

"3 0 1 0 0 \n",

"4 0 1 0 0 \n",

"\n",

" category\_Extras category\_Fish category\_Other Snacks category\_Pasta \\\n",

"0 0 0 0 0 \n",

"1 0 0 0 0 \n",

"2 0 0 0 0 \n",

"3 0 0 0 0 \n",

"4 0 0 0 0 \n",

"\n",

" category\_Pizza category\_Rice Bowl category\_Salad category\_Sandwich \\\n",

"0 0 0 0 0 \n",

"1 0 0 0 0 \n",

"2 0 0 0 0 \n",

"3 0 0 0 0 \n",

"4 0 0 0 0 \n",

"\n",

" category\_Seafood category\_Soup category\_Starters cuisine\_Indian \\\n",

"0 0 0 0 0 \n",

"1 0 0 0 0 \n",

"2 0 0 0 0 \n",

"3 0 0 0 1 \n",

"4 0 0 0 1 \n",

"\n",

" cuisine\_Italian cuisine\_Thai city\_enc\_4\_CH2 city\_enc\_4\_CH3 \\\n",

"0 0 1 0 0 \n",

"1 0 1 0 0 \n",

"2 0 1 0 0 \n",

"3 0 0 0 0 \n",

"4 0 0 0 0 \n",

"\n",

" city\_enc\_4\_CH4 \n",

"0 1 \n",

"1 1 \n",

"2 1 \n",

"3 1 \n",

"4 1 "

]

},

"execution\_count": 49,

"metadata": {},

"output\_type": "execute\_result"

}

],

"source": [

"datax.head()"

]

},

{

"cell\_type": "code",

"execution\_count": 50,

"metadata": {},

"outputs": [

{

"data": {

"text/plain": [

"num\_orders 1.000000\n",

"homepage\_featured 0.294490\n",

"checkout\_price 0.282108\n",

"emailer\_for\_promotion 0.277147\n",

"base\_price 0.222306\n",

"discount percent 0.205188\n",

"op\_area 0.176976\n",

"discount amount 0.152950\n",

"compare\_week\_price 0.137849\n",

"discount y/n 0.111766\n",

"center\_id 0.053035\n",

"city\_code 0.041596\n",

"region\_code 0.029744\n",

"compare\_week\_price y/n 0.022898\n",

"week 0.017210\n",

"meal\_id 0.010597\n",

"id 0.000702\n",

"Name: num\_orders, dtype: float64"

]

},

"execution\_count": 50,

"metadata": {},

"output\_type": "execute\_result"

}

],

"source": [

"abs(train.corr()['num\_orders']).sort\_values(ascending=False)"

]

},

{

"cell\_type": "markdown",

"metadata": {},

"source": [

"##### Base Model"

]

},

{

"cell\_type": "markdown",

"metadata": {},

"source": [

"Building base model by splitting the last 10 week of the train dataset as test."

]

},

{

"cell\_type": "code",

"execution\_count": 51,

"metadata": {},

"outputs": [],

"source": [

"from sklearn.linear\_model import LinearRegression\n",

"from sklearn.metrics import mean\_absolute\_error\n",

"from sklearn.metrics import mean\_squared\_error\n",

"from sklearn.metrics import r2\_score"

]

},

{

"cell\_type": "code",

"execution\_count": 52,

"metadata": {},

"outputs": [

{

"name": "stdout",

"output\_type": "stream",

"text": [

"Train Score : 0.4928484355256698\n",

"Test Score : 0.5060097204451955\n",

"R squared : 0.5060097204451955\n",

"RMSE : 194.4023205177741\n"

]

}

],

"source": [

"train=datax[datax['week'].isin(range(1,136))]\n",

"test=datax[datax['week'].isin(range(136,146))]\n",

"\n",

"X\_train=train.drop(['id','num\_orders','week'],axis=1)\n",

"y\_train=train['num\_orders']\n",

"\n",

"X\_test=test.drop(['id','num\_orders','week'],axis=1)\n",

"y\_test=test['num\_orders']\n",

"\n",

"reg = LinearRegression()\n",

"reg.fit(X\_train,y\_train)\n",

"print('Train Score :',reg.score(X\_train,y\_train))\n",

"print('Test Score :',reg.score(X\_test,y\_test))\n",

"\n",

"y\_pred = reg.predict(X\_test)\n",

"print('R squared :',(r2\_score(y\_test,y\_pred)))\n",

"print('RMSE :',np.sqrt(mean\_squared\_error(y\_test,y\_pred)))"

]

},

{

"cell\_type": "markdown",

"metadata": {},

"source": [

"##### Linear Model 2 : Applying Standard Scaling & Log Transformation"

]

},

{

"cell\_type": "code",

"execution\_count": 53,

"metadata": {},

"outputs": [

{

"name": "stdout",

"output\_type": "stream",

"text": [

"Train Score : 0.727389363420345\n",

"Test Score : 0.703197653881066\n",

"R squared : 0.703197653881066\n",

"RMSLE : 0.6348071399794168\n"

]

}

],

"source": [

"sc=StandardScaler()\n",

"\n",

"cat=datax.drop(['checkout\_price','base\_price','discount amount','discount percent','compare\_week\_price'],axis=1)\n",

"num=datax[['checkout\_price','base\_price','discount amount','discount percent','compare\_week\_price']]\n",

"scal= pd.DataFrame(sc.fit\_transform(num),columns=num.columns)\n",

"datas=pd.concat([scal,cat],axis=1)\n",

"\n",

"train=datas[datas['week'].isin(range(1,136))]\n",

"test=datas[datas['week'].isin(range(136,146))]\n",

"\n",

"\n",

"X\_train=train.drop(['id','num\_orders','week'],axis=1)\n",

"y\_train=np.log(train['num\_orders']) #Applying Log Transformation on the Target Feature\n",

"\n",

"X\_test=test.drop(['id','num\_orders','week'],axis=1)\n",

"y\_test=np.log(test['num\_orders']) #Applying Log Transformation on the Target Feature\n",

"\n",

"reg = LinearRegression()\n",

"reg.fit(X\_train,y\_train)\n",

"print('Train Score :',reg.score(X\_train,y\_train))\n",

"print('Test Score :',reg.score(X\_test,y\_test))\n",

"\n",

"y\_pred = reg.predict(X\_test)\n",

"print('R squared :',(r2\_score(y\_test,y\_pred)))\n",

"print('RMSLE :',np.sqrt(mean\_squared\_error(y\_test,y\_pred)))"

]

},

{

"cell\_type": "markdown",

"metadata": {},

"source": [

"##### Copying to New DataFrame"

]

},

{

"cell\_type": "code",

"execution\_count": 54,

"metadata": {},

"outputs": [],

"source": [

"datay=datas.copy()"

]

},

{

"cell\_type": "code",

"execution\_count": 55,

"metadata": {},

"outputs": [],

"source": [

"datay['Quarter']=(datas['week']/13).astype('int64')"

]

},

{

"cell\_type": "code",

"execution\_count": 56,

"metadata": {},

"outputs": [],

"source": [

"datay['Quarter'] = datay['Quarter'].map({0:'Q1',\n",

" 1:'Q2',\n",

" 2:'Q3',\n",

" 3:'Q4',\n",

" 4:'Q1',\n",

" 5:'Q2',\n",

" 6:'Q3',\n",

" 7:'Q4',\n",

" 8:'Q1',\n",

" 9:'Q2',\n",

" 10:'Q3',\n",

" 11:'Q4'})"

]

},

{

"cell\_type": "code",

"execution\_count": 57,

"metadata": {},

"outputs": [

{

"data": {

"text/plain": [

"Q4 124839\n",

"Q3 124225\n",

"Q2 122386\n",

"Q1 117671\n",

"Name: Quarter, dtype: int64"

]

},

"execution\_count": 57,

"metadata": {},

"output\_type": "execute\_result"

}

],

"source": [

"datay['Quarter'].value\_counts()"

]

},

{

"cell\_type": "code",

"execution\_count": 58,

"metadata": {},

"outputs": [],

"source": [

"datay['Year']=(datas['week']/52).astype('int64')"

]

},

{

"cell\_type": "code",

"execution\_count": 59,

"metadata": {},

"outputs": [],

"source": [

"datay['Year'] = datay['Year'].map({0:'Y1',\n",

" 1:'Y2',\n",

" 2:'Y3'})"

]

},

{

"cell\_type": "code",

"execution\_count": 60,

"metadata": {},

"outputs": [],

"source": [

"objy=datay[['Quarter', 'Year']]\n",

"numy=datay.drop(['Quarter', 'Year'],axis=1)\n",

"\n",

"encode1y=pd.get\_dummies(objy,drop\_first = True)\n",

"encode1y.head()\n",

"\n",

"datay=pd.concat([numy,encode1y],axis=1)"

]

},

{

"cell\_type": "code",

"execution\_count": 61,

"metadata": {},

"outputs": [

{

"data": {

"text/html": [

"<div>\n",

"<style scoped>\n",

" .dataframe tbody tr th:only-of-type {\n",

" vertical-align: middle;\n",

" }\n",

"\n",

" .dataframe tbody tr th {\n",

" vertical-align: top;\n",

" }\n",

"\n",

" .dataframe thead th {\n",

" text-align: right;\n",

" }\n",

"</style>\n",

"<table border=\"1\" class=\"dataframe\">\n",

" <thead>\n",

" <tr style=\"text-align: right;\">\n",

" <th></th>\n",

" <th>checkout\_price</th>\n",

" <th>base\_price</th>\n",

" <th>discount amount</th>\n",

" <th>discount percent</th>\n",

" <th>compare\_week\_price</th>\n",

" <th>id</th>\n",

" <th>week</th>\n",

" <th>emailer\_for\_promotion</th>\n",

" <th>homepage\_featured</th>\n",

" <th>num\_orders</th>\n",

" <th>city\_code</th>\n",

" <th>op\_area</th>\n",

" <th>discount y/n</th>\n",

" <th>compare\_week\_price y/n</th>\n",

" <th>center\_id\_11</th>\n",

" <th>center\_id\_13</th>\n",

" <th>center\_id\_14</th>\n",

" <th>center\_id\_17</th>\n",

" <th>center\_id\_20</th>\n",

" <th>center\_id\_23</th>\n",

" <th>center\_id\_24</th>\n",

" <th>center\_id\_26</th>\n",

" <th>center\_id\_27</th>\n",

" <th>center\_id\_29</th>\n",

" <th>center\_id\_30</th>\n",

" <th>center\_id\_32</th>\n",

" <th>center\_id\_34</th>\n",

" <th>center\_id\_36</th>\n",

" <th>center\_id\_39</th>\n",

" <th>center\_id\_41</th>\n",

" <th>center\_id\_42</th>\n",

" <th>center\_id\_43</th>\n",

" <th>center\_id\_50</th>\n",

" <th>center\_id\_51</th>\n",

" <th>center\_id\_52</th>\n",

" <th>center\_id\_53</th>\n",

" <th>center\_id\_55</th>\n",

" <th>center\_id\_57</th>\n",

" <th>center\_id\_58</th>\n",

" <th>center\_id\_59</th>\n",

" <th>center\_id\_61</th>\n",

" <th>center\_id\_64</th>\n",

" <th>center\_id\_65</th>\n",

" <th>center\_id\_66</th>\n",

" <th>center\_id\_67</th>\n",

" <th>center\_id\_68</th>\n",

" <th>center\_id\_72</th>\n",

" <th>center\_id\_73</th>\n",

" <th>center\_id\_74</th>\n",

" <th>center\_id\_75</th>\n",

" <th>center\_id\_76</th>\n",

" <th>center\_id\_77</th>\n",

" <th>center\_id\_80</th>\n",

" <th>center\_id\_81</th>\n",

" <th>center\_id\_83</th>\n",

" <th>center\_id\_86</th>\n",

" <th>center\_id\_88</th>\n",

" <th>center\_id\_89</th>\n",

" <th>center\_id\_91</th>\n",

" <th>center\_id\_92</th>\n",

" <th>center\_id\_93</th>\n",

" <th>center\_id\_94</th>\n",

" <th>center\_id\_97</th>\n",

" <th>center\_id\_99</th>\n",

" <th>center\_id\_101</th>\n",

" <th>center\_id\_102</th>\n",

" <th>center\_id\_104</th>\n",

" <th>center\_id\_106</th>\n",

" <th>center\_id\_108</th>\n",

" <th>center\_id\_109</th>\n",

" <th>center\_id\_110</th>\n",

" <th>center\_id\_113</th>\n",

" <th>center\_id\_124</th>\n",

" <th>center\_id\_126</th>\n",

" <th>center\_id\_129</th>\n",

" <th>center\_id\_132</th>\n",

" <th>center\_id\_137</th>\n",

" <th>center\_id\_139</th>\n",

" <th>center\_id\_143</th>\n",

" <th>center\_id\_145</th>\n",

" <th>center\_id\_146</th>\n",

" <th>center\_id\_149</th>\n",

" <th>center\_id\_152</th>\n",

" <th>center\_id\_153</th>\n",

" <th>center\_id\_157</th>\n",

" <th>center\_id\_161</th>\n",

" <th>center\_id\_162</th>\n",

" <th>center\_id\_174</th>\n",

" <th>center\_id\_177</th>\n",

" <th>center\_id\_186</th>\n",

" <th>meal\_id\_1109</th>\n",

" <th>meal\_id\_1198</th>\n",

" <th>meal\_id\_1207</th>\n",

" <th>meal\_id\_1216</th>\n",

" <th>meal\_id\_1230</th>\n",

" <th>meal\_id\_1247</th>\n",

" <th>meal\_id\_1248</th>\n",

" <th>meal\_id\_1311</th>\n",

" <th>meal\_id\_1438</th>\n",

" <th>meal\_id\_1445</th>\n",

" <th>meal\_id\_1525</th>\n",

" <th>meal\_id\_1543</th>\n",

" <th>meal\_id\_1558</th>\n",

" <th>meal\_id\_1571</th>\n",

" <th>meal\_id\_1727</th>\n",

" <th>meal\_id\_1754</th>\n",

" <th>meal\_id\_1770</th>\n",

" <th>meal\_id\_1778</th>\n",

" <th>meal\_id\_1803</th>\n",

" <th>meal\_id\_1847</th>\n",

" <th>meal\_id\_1878</th>\n",

" <th>meal\_id\_1885</th>\n",

" <th>meal\_id\_1902</th>\n",

" <th>meal\_id\_1962</th>\n",

" <th>meal\_id\_1971</th>\n",

" <th>meal\_id\_1993</th>\n",

" <th>meal\_id\_2104</th>\n",

" <th>meal\_id\_2126</th>\n",

" <th>meal\_id\_2139</th>\n",

" <th>meal\_id\_2290</th>\n",

" <th>meal\_id\_2304</th>\n",

" <th>meal\_id\_2306</th>\n",

" <th>meal\_id\_2322</th>\n",

" <th>meal\_id\_2444</th>\n",

" <th>meal\_id\_2490</th>\n",

" <th>meal\_id\_2492</th>\n",

" <th>meal\_id\_2494</th>\n",

" <th>meal\_id\_2539</th>\n",

" <th>meal\_id\_2569</th>\n",

" <th>meal\_id\_2577</th>\n",

" <th>meal\_id\_2581</th>\n",

" <th>meal\_id\_2631</th>\n",

" <th>meal\_id\_2640</th>\n",

" <th>meal\_id\_2664</th>\n",

" <th>meal\_id\_2704</th>\n",

" <th>meal\_id\_2707</th>\n",

" <th>meal\_id\_2760</th>\n",

" <th>meal\_id\_2826</th>\n",

" <th>meal\_id\_2867</th>\n",

" <th>meal\_id\_2956</th>\n",

" <th>region\_code\_34</th>\n",

" <th>region\_code\_35</th>\n",

" <th>region\_code\_56</th>\n",

" <th>region\_code\_71</th>\n",

" <th>region\_code\_77</th>\n",

" <th>region\_code\_85</th>\n",

" <th>region\_code\_93</th>\n",

" <th>center\_type\_TYPE\_B</th>\n",

" <th>center\_type\_TYPE\_C</th>\n",

" <th>category\_Biryani</th>\n",

" <th>category\_Desert</th>\n",

" <th>category\_Extras</th>\n",

" <th>category\_Fish</th>\n",

" <th>category\_Other Snacks</th>\n",

" <th>category\_Pasta</th>\n",

" <th>category\_Pizza</th>\n",

" <th>category\_Rice Bowl</th>\n",

" <th>category\_Salad</th>\n",

" <th>category\_Sandwich</th>\n",

" <th>category\_Seafood</th>\n",

" <th>category\_Soup</th>\n",

" <th>category\_Starters</th>\n",

" <th>cuisine\_Indian</th>\n",

" <th>cuisine\_Italian</th>\n",

" <th>cuisine\_Thai</th>\n",

" <th>city\_enc\_4\_CH2</th>\n",

" <th>city\_enc\_4\_CH3</th>\n",

" <th>city\_enc\_4\_CH4</th>\n",

" <th>Quarter\_Q2</th>\n",

" <th>Quarter\_Q3</th>\n",

" <th>Quarter\_Q4</th>\n",

" <th>Year\_Y2</th>\n",

" <th>Year\_Y3</th>\n",

" </tr>\n",

" </thead>\n",

" <tbody>\n",

" <tr>\n",

" <th>0</th>\n",

" <td>-1.281184</td>\n",

" <td>-1.259867</td>\n",

" <td>-0.124498</td>\n",

" <td>0.399910</td>\n",

" <td>-0.003326</td>\n",

" <td>1379560</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>177</td>\n",

" <td>647</td>\n",

" <td>2.0</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" </tr>\n",

" <tr>\n",

" <th>1</th>\n",

" <td>-1.281184</td>\n",

" <td>-1.362517</td>\n",

" <td>-0.467581</td>\n",

" <td>-0.562824</td>\n",

" <td>-0.003326</td>\n",

" <td>1466964</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>270</td>\n",

" <td>647</td>\n",

" <td>2.0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" </tr>\n",

" <tr>\n",

" <th>2</th>\n",

" <td>-1.294058</td>\n",

" <td>-1.362329</td>\n",

" <td>-0.425894</td>\n",

" <td>-0.432643</td>\n",

" <td>-0.003326</td>\n",

" <td>1346989</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>189</td>\n",

" <td>647</td>\n",

" <td>2.0</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" </tr>\n",

" <tr>\n",

" <th>3</th>\n",

" <td>0.043267</td>\n",

" <td>0.518969</td>\n",

" <td>1.596548</td>\n",

" <td>1.483406</td>\n",

" <td>-0.003326</td>\n",

" <td>1338232</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>54</td>\n",

" <td>647</td>\n",

" <td>2.0</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" </tr>\n",

" <tr>\n",

" <th>4</th>\n",

" <td>-0.584095</td>\n",

" <td>-0.697293</td>\n",

" <td>-0.467581</td>\n",

" <td>-0.534189</td>\n",

" <td>-0.003326</td>\n",

" <td>1448490</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>40</td>\n",

" <td>647</td>\n",

" <td>2.0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" </tr>\n",

" </tbody>\n",

"</table>\n",

"</div>"

],

"text/plain": [

" checkout\_price base\_price discount amount discount percent \\\n",

"0 -1.281184 -1.259867 -0.124498 0.399910 \n",

"1 -1.281184 -1.362517 -0.467581 -0.562824 \n",

"2 -1.294058 -1.362329 -0.425894 -0.432643 \n",

"3 0.043267 0.518969 1.596548 1.483406 \n",

"4 -0.584095 -0.697293 -0.467581 -0.534189 \n",

"\n",

" compare\_week\_price id week emailer\_for\_promotion \\\n",

"0 -0.003326 1379560 1 0 \n",

"1 -0.003326 1466964 1 0 \n",

"2 -0.003326 1346989 1 0 \n",

"3 -0.003326 1338232 1 0 \n",

"4 -0.003326 1448490 1 0 \n",

"\n",

" homepage\_featured num\_orders city\_code op\_area discount y/n \\\n",

"0 0 177 647 2.0 1 \n",

"1 0 270 647 2.0 0 \n",

"2 0 189 647 2.0 1 \n",

"3 0 54 647 2.0 1 \n",

"4 0 40 647 2.0 0 \n",

"\n",

" compare\_week\_price y/n center\_id\_11 center\_id\_13 center\_id\_14 \\\n",

"0 0 0 0 0 \n",

"1 0 0 0 0 \n",

"2 0 0 0 0 \n",

"3 0 0 0 0 \n",

"4 0 0 0 0 \n",

"\n",

" center\_id\_17 center\_id\_20 center\_id\_23 center\_id\_24 center\_id\_26 \\\n",

"0 0 0 0 0 0 \n",

"1 0 0 0 0 0 \n",

"2 0 0 0 0 0 \n",

"3 0 0 0 0 0 \n",

"4 0 0 0 0 0 \n",

"\n",

" center\_id\_27 center\_id\_29 center\_id\_30 center\_id\_32 center\_id\_34 \\\n",

"0 0 0 0 0 0 \n",

"1 0 0 0 0 0 \n",

"2 0 0 0 0 0 \n",

"3 0 0 0 0 0 \n",

"4 0 0 0 0 0 \n",

"\n",

" center\_id\_36 center\_id\_39 center\_id\_41 center\_id\_42 center\_id\_43 \\\n",

"0 0 0 0 0 0 \n",

"1 0 0 0 0 0 \n",

"2 0 0 0 0 0 \n",

"3 0 0 0 0 0 \n",

"4 0 0 0 0 0 \n",

"\n",

" center\_id\_50 center\_id\_51 center\_id\_52 center\_id\_53 center\_id\_55 \\\n",

"0 0 0 0 0 1 \n",

"1 0 0 0 0 1 \n",

"2 0 0 0 0 1 \n",

"3 0 0 0 0 1 \n",

"4 0 0 0 0 1 \n",

"\n",

" center\_id\_57 center\_id\_58 center\_id\_59 center\_id\_61 center\_id\_64 \\\n",

"0 0 0 0 0 0 \n",

"1 0 0 0 0 0 \n",

"2 0 0 0 0 0 \n",

"3 0 0 0 0 0 \n",

"4 0 0 0 0 0 \n",

"\n",

" center\_id\_65 center\_id\_66 center\_id\_67 center\_id\_68 center\_id\_72 \\\n",

"0 0 0 0 0 0 \n",

"1 0 0 0 0 0 \n",

"2 0 0 0 0 0 \n",

"3 0 0 0 0 0 \n",

"4 0 0 0 0 0 \n",

"\n",

" center\_id\_73 center\_id\_74 center\_id\_75 center\_id\_76 center\_id\_77 \\\n",

"0 0 0 0 0 0 \n",

"1 0 0 0 0 0 \n",

"2 0 0 0 0 0 \n",

"3 0 0 0 0 0 \n",

"4 0 0 0 0 0 \n",

"\n",

" center\_id\_80 center\_id\_81 center\_id\_83 center\_id\_86 center\_id\_88 \\\n",

"0 0 0 0 0 0 \n",

"1 0 0 0 0 0 \n",

"2 0 0 0 0 0 \n",

"3 0 0 0 0 0 \n",

"4 0 0 0 0 0 \n",

"\n",

" center\_id\_89 center\_id\_91 center\_id\_92 center\_id\_93 center\_id\_94 \\\n",

"0 0 0 0 0 0 \n",

"1 0 0 0 0 0 \n",

"2 0 0 0 0 0 \n",

"3 0 0 0 0 0 \n",

"4 0 0 0 0 0 \n",

"\n",

" center\_id\_97 center\_id\_99 center\_id\_101 center\_id\_102 center\_id\_104 \\\n",

"0 0 0 0 0 0 \n",

"1 0 0 0 0 0 \n",

"2 0 0 0 0 0 \n",

"3 0 0 0 0 0 \n",

"4 0 0 0 0 0 \n",

"\n",

" center\_id\_106 center\_id\_108 center\_id\_109 center\_id\_110 center\_id\_113 \\\n",

"0 0 0 0 0 0 \n",

"1 0 0 0 0 0 \n",

"2 0 0 0 0 0 \n",

"3 0 0 0 0 0 \n",

"4 0 0 0 0 0 \n",

"\n",

" center\_id\_124 center\_id\_126 center\_id\_129 center\_id\_132 center\_id\_137 \\\n",

"0 0 0 0 0 0 \n",

"1 0 0 0 0 0 \n",

"2 0 0 0 0 0 \n",

"3 0 0 0 0 0 \n",

"4 0 0 0 0 0 \n",

"\n",

" center\_id\_139 center\_id\_143 center\_id\_145 center\_id\_146 center\_id\_149 \\\n",

"0 0 0 0 0 0 \n",

"1 0 0 0 0 0 \n",

"2 0 0 0 0 0 \n",

"3 0 0 0 0 0 \n",

"4 0 0 0 0 0 \n",

"\n",

" center\_id\_152 center\_id\_153 center\_id\_157 center\_id\_161 center\_id\_162 \\\n",

"0 0 0 0 0 0 \n",

"1 0 0 0 0 0 \n",

"2 0 0 0 0 0 \n",

"3 0 0 0 0 0 \n",

"4 0 0 0 0 0 \n",

"\n",

" center\_id\_174 center\_id\_177 center\_id\_186 meal\_id\_1109 meal\_id\_1198 \\\n",

"0 0 0 0 0 0 \n",

"1 0 0 0 0 0 \n",

"2 0 0 0 0 0 \n",

"3 0 0 0 0 0 \n",

"4 0 0 0 0 0 \n",

"\n",

" meal\_id\_1207 meal\_id\_1216 meal\_id\_1230 meal\_id\_1247 meal\_id\_1248 \\\n",

"0 0 0 0 0 0 \n",

"1 0 0 0 0 0 \n",

"2 0 0 0 0 0 \n",

"3 0 0 0 0 0 \n",

"4 0 0 0 0 0 \n",

"\n",

" meal\_id\_1311 meal\_id\_1438 meal\_id\_1445 meal\_id\_1525 meal\_id\_1543 \\\n",

"0 0 0 0 0 0 \n",

"1 0 0 0 0 0 \n",

"2 0 0 0 0 0 \n",

"3 0 0 0 0 0 \n",

"4 0 0 0 0 0 \n",

"\n",

" meal\_id\_1558 meal\_id\_1571 meal\_id\_1727 meal\_id\_1754 meal\_id\_1770 \\\n",

"0 0 0 0 0 0 \n",

"1 0 0 0 0 0 \n",

"2 0 0 0 0 0 \n",

"3 0 0 0 0 0 \n",

"4 0 0 0 0 0 \n",

"\n",

" meal\_id\_1778 meal\_id\_1803 meal\_id\_1847 meal\_id\_1878 meal\_id\_1885 \\\n",

"0 0 0 0 0 1 \n",

"1 0 0 0 0 0 \n",

"2 0 0 0 0 0 \n",

"3 0 0 0 0 0 \n",

"4 0 0 0 0 0 \n",

"\n",

" meal\_id\_1902 meal\_id\_1962 meal\_id\_1971 meal\_id\_1993 meal\_id\_2104 \\\n",

"0 0 0 0 0 0 \n",

"1 0 0 0 1 0 \n",

"2 0 0 0 0 0 \n",

"3 0 0 0 0 0 \n",

"4 0 0 0 0 0 \n",

"\n",

" meal\_id\_2126 meal\_id\_2139 meal\_id\_2290 meal\_id\_2304 meal\_id\_2306 \\\n",

"0 0 0 0 0 0 \n",

"1 0 0 0 0 0 \n",

"2 0 0 0 0 0 \n",

"3 0 1 0 0 0 \n",

"4 0 0 0 0 0 \n",

"\n",

" meal\_id\_2322 meal\_id\_2444 meal\_id\_2490 meal\_id\_2492 meal\_id\_2494 \\\n",

"0 0 0 0 0 0 \n",

"1 0 0 0 0 0 \n",

"2 0 0 0 0 0 \n",

"3 0 0 0 0 0 \n",

"4 0 0 0 0 0 \n",

"\n",

" meal\_id\_2539 meal\_id\_2569 meal\_id\_2577 meal\_id\_2581 meal\_id\_2631 \\\n",

"0 0 0 0 0 0 \n",

"1 0 0 0 0 0 \n",

"2 1 0 0 0 0 \n",

"3 0 0 0 0 0 \n",

"4 0 0 0 0 1 \n",

"\n",

" meal\_id\_2640 meal\_id\_2664 meal\_id\_2704 meal\_id\_2707 meal\_id\_2760 \\\n",

"0 0 0 0 0 0 \n",

"1 0 0 0 0 0 \n",

"2 0 0 0 0 0 \n",

"3 0 0 0 0 0 \n",

"4 0 0 0 0 0 \n",

"\n",

" meal\_id\_2826 meal\_id\_2867 meal\_id\_2956 region\_code\_34 region\_code\_35 \\\n",

"0 0 0 0 0 0 \n",

"1 0 0 0 0 0 \n",

"2 0 0 0 0 0 \n",

"3 0 0 0 0 0 \n",

"4 0 0 0 0 0 \n",

"\n",

" region\_code\_56 region\_code\_71 region\_code\_77 region\_code\_85 \\\n",

"0 1 0 0 0 \n",

"1 1 0 0 0 \n",

"2 1 0 0 0 \n",

"3 1 0 0 0 \n",

"4 1 0 0 0 \n",

"\n",

" region\_code\_93 center\_type\_TYPE\_B center\_type\_TYPE\_C category\_Biryani \\\n",

"0 0 0 1 0 \n",

"1 0 0 1 0 \n",

"2 0 0 1 0 \n",

"3 0 0 1 0 \n",

"4 0 0 1 0 \n",

"\n",

" category\_Desert category\_Extras category\_Fish category\_Other Snacks \\\n",

"0 0 0 0 0 \n",

"1 0 0 0 0 \n",

"2 0 0 0 0 \n",

"3 0 0 0 0 \n",

"4 0 0 0 0 \n",

"\n",

" category\_Pasta category\_Pizza category\_Rice Bowl category\_Salad \\\n",

"0 0 0 0 0 \n",

"1 0 0 0 0 \n",

"2 0 0 0 0 \n",

"3 0 0 0 0 \n",

"4 0 0 0 0 \n",

"\n",

" category\_Sandwich category\_Seafood category\_Soup category\_Starters \\\n",

"0 0 0 0 0 \n",

"1 0 0 0 0 \n",

"2 0 0 0 0 \n",

"3 0 0 0 0 \n",

"4 0 0 0 0 \n",

"\n",

" cuisine\_Indian cuisine\_Italian cuisine\_Thai city\_enc\_4\_CH2 \\\n",

"0 0 0 1 0 \n",

"1 0 0 1 0 \n",

"2 0 0 1 0 \n",

"3 1 0 0 0 \n",

"4 1 0 0 0 \n",

"\n",

" city\_enc\_4\_CH3 city\_enc\_4\_CH4 Quarter\_Q2 Quarter\_Q3 Quarter\_Q4 \\\n",

"0 0 1 0 0 0 \n",

"1 0 1 0 0 0 \n",

"2 0 1 0 0 0 \n",

"3 0 1 0 0 0 \n",

"4 0 1 0 0 0 \n",

"\n",

" Year\_Y2 Year\_Y3 \n",

"0 0 0 \n",

"1 0 0 \n",

"2 0 0 \n",

"3 0 0 \n",

"4 0 0 "

]

},

"execution\_count": 61,

"metadata": {},

"output\_type": "execute\_result"

}

],

"source": [

"datay.head()"

]

},

{

"cell\_type": "markdown",

"metadata": {},

"source": [

"##### Applying Log Transformation on the Target Feature"

]

},

{

"cell\_type": "code",

"execution\_count": 62,

"metadata": {},

"outputs": [],

"source": [

"datay['num\_orders']=np.log1p(datay['num\_orders'])"

]

},

{

"cell\_type": "code",

"execution\_count": 63,

"metadata": {},

"outputs": [],

"source": [

"train=datay[datay['week'].isin(range(1,146))]\n",

"def outliers\_3(col):\n",

" q3=round(train[col].quantile(0.75),6)\n",

" q1=round(train[col].quantile(0.25),6)\n",

" iqr=q3-q1\n",

" lw = q1 - (3\*iqr)\n",

" hw = q3 + (3\*iqr) \n",

" uo=train[train[col]>hw].shape[0]\n",

" lo=train[train[col]<lw].shape[0]\n",

" print('Number of Upper Outliers :',uo)\n",

" print('Number of Lower Outliers :',lo)\n",

" print('Percentage of Outliers :',((uo+lo)/train.shape[0])\*100)"

]

},

{

"cell\_type": "code",

"execution\_count": 64,

"metadata": {},

"outputs": [

{

"name": "stdout",

"output\_type": "stream",

"text": [

"Number of Upper Outliers : 0\n",

"Number of Lower Outliers : 0\n",

"Percentage of Outliers : 0.0\n"

]

}

],

"source": [

"outliers\_3('num\_orders')"

]

},

{

"cell\_type": "code",

"execution\_count": 65,

"metadata": {},

"outputs": [

{

"data": {

"text/html": [

"<div>\n",

"<style scoped>\n",

" .dataframe tbody tr th:only-of-type {\n",

" vertical-align: middle;\n",

" }\n",

"\n",

" .dataframe tbody tr th {\n",

" vertical-align: top;\n",

" }\n",

"\n",

" .dataframe thead th {\n",

" text-align: right;\n",

" }\n",

"</style>\n",

"<table border=\"1\" class=\"dataframe\">\n",

" <thead>\n",

" <tr style=\"text-align: right;\">\n",

" <th></th>\n",

" <th>checkout\_price</th>\n",

" <th>base\_price</th>\n",

" <th>discount amount</th>\n",

" <th>discount percent</th>\n",

" <th>compare\_week\_price</th>\n",

" <th>id</th>\n",

" <th>week</th>\n",

" <th>emailer\_for\_promotion</th>\n",

" <th>homepage\_featured</th>\n",

" <th>num\_orders</th>\n",

" <th>city\_code</th>\n",

" <th>op\_area</th>\n",

" <th>discount y/n</th>\n",

" <th>compare\_week\_price y/n</th>\n",

" <th>center\_id\_11</th>\n",

" <th>center\_id\_13</th>\n",

" <th>center\_id\_14</th>\n",

" <th>center\_id\_17</th>\n",

" <th>center\_id\_20</th>\n",

" <th>center\_id\_23</th>\n",

" <th>center\_id\_24</th>\n",

" <th>center\_id\_26</th>\n",

" <th>center\_id\_27</th>\n",

" <th>center\_id\_29</th>\n",

" <th>center\_id\_30</th>\n",

" <th>center\_id\_32</th>\n",

" <th>center\_id\_34</th>\n",

" <th>center\_id\_36</th>\n",

" <th>center\_id\_39</th>\n",

" <th>center\_id\_41</th>\n",

" <th>center\_id\_42</th>\n",

" <th>center\_id\_43</th>\n",

" <th>center\_id\_50</th>\n",

" <th>center\_id\_51</th>\n",

" <th>center\_id\_52</th>\n",

" <th>center\_id\_53</th>\n",

" <th>center\_id\_55</th>\n",

" <th>center\_id\_57</th>\n",

" <th>center\_id\_58</th>\n",

" <th>center\_id\_59</th>\n",

" <th>center\_id\_61</th>\n",

" <th>center\_id\_64</th>\n",

" <th>center\_id\_65</th>\n",

" <th>center\_id\_66</th>\n",

" <th>center\_id\_67</th>\n",

" <th>center\_id\_68</th>\n",

" <th>center\_id\_72</th>\n",

" <th>center\_id\_73</th>\n",

" <th>center\_id\_74</th>\n",

" <th>center\_id\_75</th>\n",

" <th>center\_id\_76</th>\n",

" <th>center\_id\_77</th>\n",

" <th>center\_id\_80</th>\n",

" <th>center\_id\_81</th>\n",

" <th>center\_id\_83</th>\n",

" <th>center\_id\_86</th>\n",

" <th>center\_id\_88</th>\n",

" <th>center\_id\_89</th>\n",

" <th>center\_id\_91</th>\n",

" <th>center\_id\_92</th>\n",

" <th>center\_id\_93</th>\n",

" <th>center\_id\_94</th>\n",

" <th>center\_id\_97</th>\n",

" <th>center\_id\_99</th>\n",

" <th>center\_id\_101</th>\n",

" <th>center\_id\_102</th>\n",

" <th>center\_id\_104</th>\n",

" <th>center\_id\_106</th>\n",

" <th>center\_id\_108</th>\n",

" <th>center\_id\_109</th>\n",

" <th>center\_id\_110</th>\n",

" <th>center\_id\_113</th>\n",

" <th>center\_id\_124</th>\n",

" <th>center\_id\_126</th>\n",

" <th>center\_id\_129</th>\n",

" <th>center\_id\_132</th>\n",

" <th>center\_id\_137</th>\n",

" <th>center\_id\_139</th>\n",

" <th>center\_id\_143</th>\n",

" <th>center\_id\_145</th>\n",

" <th>center\_id\_146</th>\n",

" <th>center\_id\_149</th>\n",

" <th>center\_id\_152</th>\n",

" <th>center\_id\_153</th>\n",

" <th>center\_id\_157</th>\n",

" <th>center\_id\_161</th>\n",

" <th>center\_id\_162</th>\n",

" <th>center\_id\_174</th>\n",

" <th>center\_id\_177</th>\n",

" <th>center\_id\_186</th>\n",

" <th>meal\_id\_1109</th>\n",

" <th>meal\_id\_1198</th>\n",

" <th>meal\_id\_1207</th>\n",

" <th>meal\_id\_1216</th>\n",

" <th>meal\_id\_1230</th>\n",

" <th>meal\_id\_1247</th>\n",

" <th>meal\_id\_1248</th>\n",

" <th>meal\_id\_1311</th>\n",

" <th>meal\_id\_1438</th>\n",

" <th>meal\_id\_1445</th>\n",

" <th>meal\_id\_1525</th>\n",

" <th>meal\_id\_1543</th>\n",

" <th>meal\_id\_1558</th>\n",

" <th>meal\_id\_1571</th>\n",

" <th>meal\_id\_1727</th>\n",

" <th>meal\_id\_1754</th>\n",

" <th>meal\_id\_1770</th>\n",

" <th>meal\_id\_1778</th>\n",

" <th>meal\_id\_1803</th>\n",

" <th>meal\_id\_1847</th>\n",

" <th>meal\_id\_1878</th>\n",

" <th>meal\_id\_1885</th>\n",

" <th>meal\_id\_1902</th>\n",

" <th>meal\_id\_1962</th>\n",

" <th>meal\_id\_1971</th>\n",

" <th>meal\_id\_1993</th>\n",

" <th>meal\_id\_2104</th>\n",

" <th>meal\_id\_2126</th>\n",

" <th>meal\_id\_2139</th>\n",

" <th>meal\_id\_2290</th>\n",

" <th>meal\_id\_2304</th>\n",

" <th>meal\_id\_2306</th>\n",

" <th>meal\_id\_2322</th>\n",

" <th>meal\_id\_2444</th>\n",

" <th>meal\_id\_2490</th>\n",

" <th>meal\_id\_2492</th>\n",

" <th>meal\_id\_2494</th>\n",

" <th>meal\_id\_2539</th>\n",

" <th>meal\_id\_2569</th>\n",

" <th>meal\_id\_2577</th>\n",

" <th>meal\_id\_2581</th>\n",

" <th>meal\_id\_2631</th>\n",

" <th>meal\_id\_2640</th>\n",

" <th>meal\_id\_2664</th>\n",

" <th>meal\_id\_2704</th>\n",

" <th>meal\_id\_2707</th>\n",

" <th>meal\_id\_2760</th>\n",

" <th>meal\_id\_2826</th>\n",

" <th>meal\_id\_2867</th>\n",

" <th>meal\_id\_2956</th>\n",

" <th>region\_code\_34</th>\n",

" <th>region\_code\_35</th>\n",

" <th>region\_code\_56</th>\n",

" <th>region\_code\_71</th>\n",

" <th>region\_code\_77</th>\n",

" <th>region\_code\_85</th>\n",

" <th>region\_code\_93</th>\n",

" <th>center\_type\_TYPE\_B</th>\n",

" <th>center\_type\_TYPE\_C</th>\n",

" <th>category\_Biryani</th>\n",

" <th>category\_Desert</th>\n",

" <th>category\_Extras</th>\n",

" <th>category\_Fish</th>\n",

" <th>category\_Other Snacks</th>\n",

" <th>category\_Pasta</th>\n",

" <th>category\_Pizza</th>\n",

" <th>category\_Rice Bowl</th>\n",

" <th>category\_Salad</th>\n",

" <th>category\_Sandwich</th>\n",

" <th>category\_Seafood</th>\n",

" <th>category\_Soup</th>\n",

" <th>category\_Starters</th>\n",

" <th>cuisine\_Indian</th>\n",

" <th>cuisine\_Italian</th>\n",

" <th>cuisine\_Thai</th>\n",

" <th>city\_enc\_4\_CH2</th>\n",

" <th>city\_enc\_4\_CH3</th>\n",

" <th>city\_enc\_4\_CH4</th>\n",

" <th>Quarter\_Q2</th>\n",

" <th>Quarter\_Q3</th>\n",

" <th>Quarter\_Q4</th>\n",

" <th>Year\_Y2</th>\n",

" <th>Year\_Y3</th>\n",

" </tr>\n",

" </thead>\n",

" <tbody>\n",

" <tr>\n",

" <th>0</th>\n",

" <td>-1.281184</td>\n",

" <td>-1.259867</td>\n",

" <td>-0.124498</td>\n",

" <td>0.399910</td>\n",

" <td>-0.003326</td>\n",

" <td>1379560</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>5.181784</td>\n",

" <td>647</td>\n",

" <td>2.0</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" </tr>\n",

" <tr>\n",

" <th>1</th>\n",

" <td>-1.281184</td>\n",

" <td>-1.362517</td>\n",

" <td>-0.467581</td>\n",

" <td>-0.562824</td>\n",

" <td>-0.003326</td>\n",

" <td>1466964</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>5.602119</td>\n",

" <td>647</td>\n",

" <td>2.0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" </tr>\n",

" <tr>\n",

" <th>2</th>\n",

" <td>-1.294058</td>\n",

" <td>-1.362329</td>\n",

" <td>-0.425894</td>\n",

" <td>-0.432643</td>\n",

" <td>-0.003326</td>\n",

" <td>1346989</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>5.247024</td>\n",

" <td>647</td>\n",

" <td>2.0</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" </tr>\n",

" <tr>\n",

" <th>3</th>\n",

" <td>0.043267</td>\n",

" <td>0.518969</td>\n",

" <td>1.596548</td>\n",

" <td>1.483406</td>\n",

" <td>-0.003326</td>\n",

" <td>1338232</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>4.007333</td>\n",

" <td>647</td>\n",

" <td>2.0</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" </tr>\n",

" <tr>\n",

" <th>4</th>\n",

" <td>-0.584095</td>\n",

" <td>-0.697293</td>\n",

" <td>-0.467581</td>\n",

" <td>-0.534189</td>\n",

" <td>-0.003326</td>\n",

" <td>1448490</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>3.713572</td>\n",

" <td>647</td>\n",

" <td>2.0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>1</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

" <td>0</td>\n",

GITHUB & PROJECT DEMO LINK