

Team ID: PNT2022TMID23576

PROJECT NAME: DemandEst - AI powered Food Demand Forecaster

Team Leader

The screenshot shows a Google Colab notebook titled "Data_preprocessing_and_Model_building (1).ipynb". The code cell contains Python imports for LabelEncoder from sklearn.preprocessing and assignments of three LabelEncoder objects (l1, l2, l3) to columns 'center_type', 'category', and 'cuisine' respectively. A call to trainfinal.head() displays the first five rows of the dataset, which includes columns like id, week, city_code, region_code, center_type, op_area, category, cuisine, checkout_price, base_price, emailer_for_promotion, homepage_featured, and num_orders. The output shows data for 177 rows.

```
[ ] from sklearn.preprocessing import LabelEncoder
l1 = LabelEncoder()
trainfinal['center_type'] = l1.fit_transform(trainfinal['center_type'])
l2 = LabelEncoder()
trainfinal['category'] = l2.fit_transform(trainfinal['category'])
l3 = LabelEncoder()
trainfinal['cuisine'] = l3.fit_transform(trainfinal['cuisine'])

[ ] trainfinal.head()

id week city_code region_code center_type op_area category cuisine checkout_price base_price emailer_for_promotion homepage_featured num_orders
0 1379560 1 647 56 2 2.0 0 3 136.83 152.29 0 0 177
1 1018704 2 647 56 2 2.0 0 3 135.83 152.29 0 0 323
2 1196273 3 647 56 2 2.0 0 3 132.92 133.92 0 0 96
3 1116527 4 647 56 2 2.0 0 3 135.86 134.86 0 0 163
4 1343872 5 647 56 2 2.0 0 3 146.50 147.50 0 0 215

[ ] trainfinal.shape
(456548, 13)
```

Team Member 1

The screenshot shows a Google Colab notebook titled "Data_preprocessing_and_Model_building (1).ipynb". The code cell contains Python imports for LabelEncoder from sklearn.preprocessing and assignments of three LabelEncoder objects (l1, l2, l3) to columns 'center_type', 'category', and 'cuisine' respectively. A call to trainfinal.head() displays the first five rows of the dataset, which includes columns like id, week, city_code, region_code, center_type, op_area, category, cuisine, checkout_price, base_price, emailer_for_promotion, homepage_featured, and num_orders. The output shows data for 177 rows.

```
[ ] from sklearn.preprocessing import LabelEncoder
l1 = LabelEncoder()
trainfinal['center_type'] = l1.fit_transform(trainfinal['center_type'])
l2 = LabelEncoder()
trainfinal['category'] = l2.fit_transform(trainfinal['category'])
l3 = LabelEncoder()
trainfinal['cuisine'] = l3.fit_transform(trainfinal['cuisine'])

[ ] trainfinal.head()

id week city_code region_code center_type op_area category cuisine checkout_price base_price emailer_for_promotion homepage_featured num_orders
0 1379560 1 647 56 2 2.0 0 3 136.83 152.29 0 0 177
1 1018704 2 647 56 2 2.0 0 3 135.83 152.29 0 0 323
2 1196273 3 647 56 2 2.0 0 3 132.92 133.92 0 0 96
3 1116527 4 647 56 2 2.0 0 3 135.86 134.86 0 0 163
4 1343872 5 647 56 2 2.0 0 3 146.50 147.50 0 0 215

[ ] trainfinal.shape
(456548, 13)
```

Team Member 2

The screenshot shows a Google Colab notebook titled "Data_preprocessing_and_Model_building (1).ipynb". The code cell contains Python code for importing LabelEncoder from sklearn.preprocessing and applying it to three categorical columns: center_type, category, and cuisine. The output cell displays the first five rows of the trainfinal DataFrame using the head() method, showing columns like id, week, city_code, region_code, center_type, op_area, category, cuisine, checkout_price, base_price, emailer_for_promotion, homepage_featured, and num_orders. The data is as follows:

	id	week	city_code	region_code	center_type	op_area	category	cuisine	checkout_price	base_price	emailer_for_promotion	homepage_featured	num_orders
0	1379560	1	647	56	2	2.0	0	3	136.83	152.29	0	0	177
1	1018704	2	647	56	2	2.0	0	3	135.83	152.29	0	0	323
2	1196273	3	647	56	2	2.0	0	3	132.92	133.92	0	0	96
3	1116527	4	647	56	2	2.0	0	3	135.86	134.86	0	0	163
4	1343872	5	647	56	2	2.0	0	3	146.50	147.50	0	0	215

[] trainfinal.shape
(456548, 13)

Team Member 3

The screenshot shows a Google Colab notebook titled "Data_preprocessing_and_Model_building (1).ipynb". The code cell contains Python code for importing LabelEncoder from sklearn.preprocessing and applying it to three categorical columns: center_type, category, and cuisine. The output cell displays the first five rows of the trainfinal DataFrame using the head() method, showing columns like id, week, city_code, region_code, center_type, op_area, category, cuisine, checkout_price, base_price, emailer_for_promotion, homepage_featured, and num_orders. The data is identical to Team Member 2's output.

	id	week	city_code	region_code	center_type	op_area	category	cuisine	checkout_price	base_price	emailer_for_promotion	homepage_featured	num_orders
0	1379560	1	647	56	2	2.0	0	3	136.83	152.29	0	0	177
1	1018704	2	647	56	2	2.0	0	3	135.83	152.29	0	0	323
2	1196273	3	647	56	2	2.0	0	3	132.92	133.92	0	0	96
3	1116527	4	647	56	2	2.0	0	3	135.86	134.86	0	0	163
4	1343872	5	647	56	2	2.0	0	3	146.50	147.50	0	0	215

[] trainfinal.shape
(456548, 13)