Feature Extraction

April 8, 2024

1 Feature Extraction

[ ]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import gc

import category_encoders as ce

root = './data/'
```

[ ]:
```
aisles = pd.read_csv(root + 'aisles.csv')

departments = pd.read_csv(root + 'departments.csv')

orders = pd.read_csv(root + 'orders.csv',
                  dtype={
                         'order_id': np.int32,
                         'user_id': np.int64,
                         'eval_set': 'category',
                         'order_number': np.int16,
                         'order_dow': np.int8,
                         'order_hour_of_day': np.int8,
                         'days_since_prior_order': np.float32})

order_products_prior = pd.read_csv(root + 'order_products__prior.csv',
                                   dtype={
                                          'order_id': np.int32,
                                          'product_id': np.uint16,
                                          'add_to_cart_order': np.int16,
                                          'reordered': np.int8})

order_products_train = pd.read_csv(root + 'order_products__train.csv',
                                   dtype={
                                          'order_id': np.int32,
                                          'product_id': np.uint16,
```

```
                                        'add_to_cart_order': np.int16,
                                        'reordered': np.int8})
products = pd.read_csv(root + 'products.csv')
```

1.0.1 Preparing Data

[ ]:
```
prior_df = order_products_prior.merge(orders, on ='order_id', how='inner')
prior_df = prior_df.merge(products, on = 'product_id', how = 'left')
prior_df.head()
```

[ ]:
```
   order_id  product_id  add_to_cart_order  reordered  user_id eval_set  \
0         2       33120                  1          1   202279    prior
1         2       28985                  2          1   202279    prior
2         2        9327                  3          0   202279    prior
3         2       45918                  4          1   202279    prior
4         2       30035                  5          0   202279    prior

   order_number  order_dow  order_hour_of_day  days_since_prior_order  \
0             3          5                  9                     8.0
1             3          5                  9                     8.0
2             3          5                  9                     8.0
3             3          5                  9                     8.0
4             3          5                  9                     8.0

            product_name  aisle_id  department_id
0     Organic Egg Whites        86             16
1  Michigan Organic Kale        83              4
2          Garlic Powder       104             13
3         Coconut Butter        19             13
4      Natural Sweetener        17             13
```

1.1 Features creation

Calculating how many times a user buy the product

[ ]:
```
prior_df['user_buy_product_times'] = prior_df.groupby(['user_id',␣
 ↪'product_id']).cumcount() + 1
prior_df.head()
```

[ ]:
```
   order_id  product_id  add_to_cart_order  reordered  user_id eval_set  \
0         2       33120                  1          1   202279    prior
1         2       28985                  2          1   202279    prior
2         2        9327                  3          0   202279    prior
3         2       45918                  4          1   202279    prior
4         2       30035                  5          0   202279    prior

   order_number  order_dow  order_hour_of_day  days_since_prior_order  \
```

```
0              3          5                  9                      8.0
1              3          5                  9                      8.0
2              3          5                  9                      8.0
3              3          5                  9                      8.0
4              3          5                  9                      8.0
```

```
             product_name  aisle_id  department_id  user_buy_product_times
0     Organic Egg Whites        86             16                       1
1  Michigan Organic Kale        83              4                       1
2          Garlic Powder       104             13                       1
3         Coconut Butter        19             13                       1
4      Natural Sweetener        17             13                       1
```

1.1.1 Product level features

(1) Product's average add-to-cart-order

(2) Total times the product was ordered

(3) Total times the product was reordered

(4) Reorder percentage of a product

(5) Total unique users of a product

(6) Is the product Organic?

(7) Percentage of users that buy the product second time

[ ]:
```
agg_dict1 = {'add_to_cart_order' : {'mean_add_to_cart_order':'mean'},
             'reordered' : {'total_orders':'count', 'total_reorders':'sum',␣
 ↪'reorder_percentage':'mean'},
             'user_id': {'unique_users' :lambda x: x.nunique()},
             'user_buy_product_times': {'order_first_time_total_cnt' : lambda x:␣
 ↪sum(x==1),
                                        'order_second_time_total_cnt' :lambda x:␣
 ↪sum(x==2)},
             'product_name':{'is_organic': lambda x: 1 if 'Organic' in x else 0}}
```

[ ]:
```
prod_feats1 = prior_df.groupby('product_id').agg(
    mean_add_to_cart_order=('add_to_cart_order', 'mean'),
    total_orders=('reordered', 'count'),
    total_reorders=('reordered', 'sum'),
    reorder_percentage=('reordered', 'mean'),
    unique_users=('user_id', lambda x: x.nunique()),
    order_first_time_total_cnt=('user_buy_product_times', lambda x: sum(x ==␣
 ↪1)),
    order_second_time_total_cnt=('user_buy_product_times', lambda x: sum(x ==␣
 ↪2)),
    is_organic=('product_name', lambda x: 1 if 'Organic' in x else 0)
```

```
)

prod_feats1.reset_index(inplace = True)
prod_feats1.head()
```

[ ]:

	product_id	mean_add_to_cart_order	total_orders	total_reorders	\
0	1	5.801836	1852	1136	
1	2	9.888889	90	12	
2	3	6.415162	277	203	
3	4	9.507599	329	147	
4	5	6.466667	15	9	

	reorder_percentage	unique_users	order_first_time_total_cnt	\
0	0.613391	716	716	
1	0.133333	78	78	
2	0.732852	74	74	
3	0.446809	182	182	
4	0.600000	6	6	

	order_second_time_total_cnt	is_organic
0	276	0
1	8	0
2	36	0
3	64	0
4	4	0

[ ]:
```
prod_feats1['second_time_percent'] = prod_feats1.order_second_time_total_cnt/
↪prod_feats1.order_first_time_total_cnt
```

1.1.2 Aisle and department features

(8) Reorder percentage, Total orders and reorders of a product aisle

(9) Mean and std of aisle add-to-cart-order

(10) Aisle unique users

[ ]:
```
agg_dict2 = {
    'add_to_cart_order': [('aisle_mean_add_to_cart_order', 'mean'),␣
↪('aisle_std_add_to_cart_order', 'std')],
    'reordered': [('aisle_total_orders', 'count'), ('aisle_total_reorders',␣
↪'sum'), ('aisle_reorder_percentage', 'mean')],
    'user_id': [('aisle_unique_users', lambda x: x.nunique())]
}
```

[ ]:
```
aisle_feats = prior_df.groupby('aisle_id').agg(agg_dict2)
aisle_feats.columns = aisle_feats.columns.droplevel(0)
aisle_feats.reset_index(inplace = True)
```

```
aisle_feats.head()
```

[ ]:
```
   aisle_id  aisle_mean_add_to_cart_order  aisle_std_add_to_cart_order  \
0         1                      8.167640                     7.104166
1         2                      9.275497                     7.473802
2         3                      9.571935                     7.899672
3         4                     10.161450                     7.745705
4         5                     10.297600                     8.187047

   aisle_total_orders  aisle_total_reorders  aisle_reorder_percentage  \
0               71928                 42912                  0.596597
1               82491                 40365                  0.489326
2              456386                272922                  0.598007
3              200687                 98243                  0.489533
4               62510                 17542                  0.280627

   aisle_unique_users
0               20711
1               31222
2               63592
3               53892
4               32312
```

features

(10) Reorder percentage, Total orders and reorders of a product department

(11) Mean and std of department add-to-cart-order

(12) Department unique users

[ ]:
```
agg_dict3 = {
    'add_to_cart_order': [('department_mean_add_to_cart_order', 'mean'),␣
 ↪('department_std_add_to_cart_order', 'std')],
    'reordered': [('department_total_orders', 'count'),␣
 ↪('department_total_reorders', 'sum'), ('department_reorder_percentage',␣
 ↪'mean')],
    'user_id': [('department_unique_users', lambda x: x.nunique())]
}
```

[ ]:
```
dpt_feats = prior_df.groupby('department_id').agg(agg_dict3)
dpt_feats.columns = dpt_feats.columns.droplevel(0)
dpt_feats.reset_index(inplace = True)
dpt_feats.head()
```

[ ]:
```
   department_id  department_mean_add_to_cart_order  \
0              1                           8.996414
1              2                           8.277645
2              3                           8.084397
```

```
3              4                          8.022875
4              5                          5.428346

   department_std_add_to_cart_order  department_total_orders  \
0                          7.393502                 2236432
1                          7.526272                   36291
2                          6.904849                 1176787
3                          6.658899                 9479291
4                          5.778253                  153696

   department_total_reorders  department_reorder_percentage  \
0                    1211890                       0.541885
1                      14806                       0.407980
2                     739188                       0.628141
3                    6160710                       0.649913
4                      87595                       0.569924

   department_unique_users
0                   163233
1                    17875
2                   140612
3                   193237
4                    15798
```

features

(13) Binary encoding of aisle feature

(14) Binary encoding of department feature

[ ]:
```
prod_feats1 = prod_feats1.merge(products, on = 'product_id', how = 'left')
prod_feats1 = prod_feats1.merge(aisle_feats, on = 'aisle_id', how = 'left')
prod_feats1 = prod_feats1.merge(aisles, on = 'aisle_id', how = 'left')
prod_feats1 = prod_feats1.merge(dpt_feats, on = 'department_id', how = 'left')
prod_feats1 = prod_feats1.merge(departments, on = 'department_id', how = 'left')
prod_feats1.head()
```

[ ]:
```
   product_id  mean_add_to_cart_order  total_orders  total_reorders  \
0           1                5.801836          1852            1136
1           2                9.888889            90              12
2           3                6.415162           277             203
3           4                9.507599           329             147
4           5                6.466667            15               9

   reorder_percentage  unique_users  order_first_time_total_cnt  \
0            0.613391           716                         716
1            0.133333            78                          78
2            0.732852            74                          74
3            0.446809           182                         182
```

```
4            0.600000             6                          6

   order_second_time_total_cnt  is_organic  second_time_percent  …  \
0                          276           0             0.385475  …
1                            8           0             0.102564  …
2                           36           0             0.486486  …
3                           64           0             0.351648  …
4                            4           0             0.666667  …

  aisle_reorder_percentage  aisle_unique_users                     aisle  \
0                 0.548698              54202              cookies cakes
1                 0.152391              76402           spices seasonings
2                 0.527615              53197                        tea
3                 0.556655              58749               frozen meals
4                 0.280627              32312  marinades meat preparation

   department_mean_add_to_cart_order  department_std_add_to_cart_order  \
0                           9.187743                          7.692492
1                           9.593425                          7.875241
2                           6.976699                          6.711172
3                           8.996414                          7.393502
4                           9.593425                          7.875241

   department_total_orders  department_total_reorders  \
0                  2887550                    1657973
1                  1875577                     650301
2                  2690129                    1757892
3                  2236432                    1211890
4                  1875577                     650301

   department_reorder_percentage  department_unique_users department
0                       0.574180                  174219     snacks
1                       0.346721                  172755     pantry
2                       0.653460                  172795  beverages
3                       0.541885                  163233     frozen
4                       0.346721                  172755     pantry

[5 rows x 27 columns]
```

[ ]:
```
prod_feats1.drop(['product_name', 'aisle_id', 'department_id'], axis = 1,␣
↪inplace = True)
prod_feats1.head()
```

[ ]:
```
   product_id  mean_add_to_cart_order  total_orders  total_reorders  \
0           1                5.801836          1852            1136
1           2                9.888889            90              12
2           3                6.415162           277             203
```

```
3           4               9.507599           329             147
4           5               6.466667            15               9

   reorder_percentage  unique_users  order_first_time_total_cnt  \
0            0.613391           716                         716
1            0.133333            78                          78
2            0.732852            74                          74
3            0.446809           182                         182
4            0.600000             6                           6

   order_second_time_total_cnt  is_organic  second_time_percent  …  \
0                          276           0             0.385475  …
1                            8           0             0.102564  …
2                           36           0             0.486486  …
3                           64           0             0.351648  …
4                            4           0             0.666667  …

   aisle_reorder_percentage  aisle_unique_users                      aisle  \
0                  0.548698               54202               cookies cakes
1                  0.152391               76402           spices seasonings
2                  0.527615               53197                         tea
3                  0.556655               58749                frozen meals
4                  0.280627               32312  marinades meat preparation

   department_mean_add_to_cart_order  department_std_add_to_cart_order  \
0                           9.187743                          7.692492
1                           9.593425                          7.875241
2                           6.976699                          6.711172
3                           8.996414                          7.393502
4                           9.593425                          7.875241

   department_total_orders department_total_reorders  \
0                  2887550                   1657973
1                  1875577                    650301
2                  2690129                   1757892
3                  2236432                   1211890
4                  1875577                    650301

   department_reorder_percentage  department_unique_users  department
0                       0.574180                  174219      snacks
1                       0.346721                  172755      pantry
2                       0.653460                  172795   beverages
3                       0.541885                  163233      frozen
4                       0.346721                  172755      pantry

[5 rows x 24 columns]
```

[ ]:
```
prod_feats1.shape
```

[ ]:
```
(49677, 24)
```

[ ]:
```
prod_feats1.dtypes
```

[ ]:
```
product_id                          uint16
mean_add_to_cart_order             float64
total_orders                         int64
total_reorders                       int64
reorder_percentage                 float64
unique_users                         int64
order_first_time_total_cnt           int64
order_second_time_total_cnt          int64
is_organic                           int64
second_time_percent                float64
aisle_mean_add_to_cart_order       float64
aisle_std_add_to_cart_order        float64
aisle_total_orders                   int64
aisle_total_reorders                 int64
aisle_reorder_percentage           float64
aisle_unique_users                   int64
aisle                               object
department_mean_add_to_cart_order  float64
department_std_add_to_cart_order   float64
department_total_orders              int64
department_total_reorders            int64
department_reorder_percentage      float64
department_unique_users              int64
department                          object
dtype: object
```

[ ]:
```
encoder= ce.BinaryEncoder(cols=['aisle', 'department'],return_df=True)
```

[ ]:
```
prod_feats1 = encoder.fit_transform(prod_feats1)
prod_feats1.head()
```

[ ]:
```
   product_id  mean_add_to_cart_order  total_orders  total_reorders  \
0           1                5.801836          1852            1136
1           2                9.888889            90              12
2           3                6.415162           277             203
3           4                9.507599           329             147
4           5                6.466667            15               9

   reorder_percentage  unique_users  order_first_time_total_cnt  \
0            0.613391           716                         716
1            0.133333            78                          78
```

```
2            0.732852            74                          74
3            0.446809           182                         182
4            0.600000             6                           6

   order_second_time_total_cnt  is_organic  second_time_percent  …  \
0                          276           0             0.385475  …
1                            8           0             0.102564  …
2                           36           0             0.486486  …
3                           64           0             0.351648  …
4                            4           0             0.666667  …

   department_std_add_to_cart_order  department_total_orders  \
0                          7.692492                 2887550
1                          7.875241                 1875577
2                          6.711172                 2690129
3                          7.393502                 2236432
4                          7.875241                 1875577

   department_total_reorders  department_reorder_percentage  \
0                    1657973                       0.574180
1                     650301                       0.346721
2                    1757892                       0.653460
3                    1211890                       0.541885
4                     650301                       0.346721

   department_unique_users  department_0  department_1  department_2  \
0                   174219             0             0             0
1                   172755             0             0             0
2                   172795             0             0             0
3                   163233             0             0             1
4                   172755             0             0             0

   department_3  department_4
0             0             1
1             1             0
2             1             1
3             0             0
4             1             0

[5 rows x 35 columns]
```

[ ]:
```
prod_feats1.shape
```

[ ]:
```
(49677, 35)
```

[ ]:
```
prod_feats1.columns
```

[ ]:
```
Index(['product_id', 'mean_add_to_cart_order', 'total_orders',
       'total_reorders', 'reorder_percentage', 'unique_users',
       'order_first_time_total_cnt', 'order_second_time_total_cnt',
       'is_organic', 'second_time_percent', 'aisle_mean_add_to_cart_order',
       'aisle_std_add_to_cart_order', 'aisle_total_orders',
       'aisle_total_reorders', 'aisle_reorder_percentage',
       'aisle_unique_users', 'aisle_0', 'aisle_1', 'aisle_2', 'aisle_3',
       'aisle_4', 'aisle_5', 'aisle_6', 'aisle_7',
       'department_mean_add_to_cart_order', 'department_std_add_to_cart_order',
       'department_total_orders', 'department_total_reorders',
       'department_reorder_percentage', 'department_unique_users',
       'department_0', 'department_1', 'department_2', 'department_3',
       'department_4'],
      dtype='object')
```

[ ]:
```
prod_feats1.isnull().any().any()
```

[ ]:
```
False
```

[ ]:
```
# free some memory
del aisle_feats, dpt_feats, aisles, departments
gc.collect()
```

[ ]:
```
1823
```

1.1.3 User level features

(15) User's average and std day-of-week of order

(16) User's average and std hour-of-day of order

(17) User's average and std days-since-prior-order

(18) Total orders by a user

(19) Total products user has bought

(20) Total unique products user has bought

(21) user's total reordered products

(22) User's overall reorder percentage

[ ]:
```
prior_df.isnull().any()
```

[ ]:
```
order_id                False
product_id              False
add_to_cart_order       False
reordered               False
user_id                 False
eval_set                False
```

```
order_number              False
order_dow                 False
order_hour_of_day         False
days_since_prior_order     True
product_name              False
aisle_id                  False
department_id             False
user_buy_product_times    False
dtype: bool
```

[ ]:
```
# when no prior order, the value is null. Imputing as 0
prior_df.days_since_prior_order = prior_df.days_since_prior_order.fillna(0)
```

[ ]:
```
prior_df.head()
```

[ ]:
```
   order_id  product_id  add_to_cart_order  reordered  user_id eval_set  \
0         2       33120                  1          1   202279    prior
1         2       28985                  2          1   202279    prior
2         2        9327                  3          0   202279    prior
3         2       45918                  4          1   202279    prior
4         2       30035                  5          0   202279    prior

   order_number  order_dow  order_hour_of_day  days_since_prior_order  \
0             3          5                  9                     8.0
1             3          5                  9                     8.0
2             3          5                  9                     8.0
3             3          5                  9                     8.0
4             3          5                  9                     8.0

            product_name  aisle_id  department_id  user_buy_product_times
0     Organic Egg Whites        86             16                       1
1  Michigan Organic Kale        83              4                       1
2          Garlic Powder       104             13                       1
3         Coconut Butter        19             13                       1
4      Natural Sweetener        17             13                       1
```

[ ]:
```
agg_dict4 = {
    'order_dow': [('avg_dow', 'mean'), ('std_dow', 'std')],
    'order_hour_of_day': [('avg_doh', 'mean'), ('std_doh', 'std')],
    'days_since_prior_order': [('avg_since_order', 'mean'), ('std_since_order',␣
 ↪'std')],
    'order_number': [('total_orders_by_user', lambda x: x.nunique())],
    'product_id': [('total_products_by_user', 'count'),
                   ('total_unique_product_by_user', lambda x: x.nunique())],
    'reordered': [('total_reorders_by_user', 'sum'),
                  ('reorder_propotion_by_user', 'mean')]
}
```

[ ]:
```
user_feats = prior_df.groupby('user_id').agg(agg_dict4)
user_feats.columns = user_feats.columns.droplevel(0)
user_feats.reset_index(inplace = True)
user_feats.head()
```

[ ]:

	user_id	avg_dow	std_dow	avg_doh	std_doh	avg_since_order	std_since_order	total_orders_by_user	total_products_by_user	total_unique_product_by_user	total_reorders_by_user	reorder_propotion_by_user
0	1	2.644068	1.256194	10.542373	3.500355	18.542374	10.559065	10	59	18	41	0.694915
1	2	2.005128	0.971222	10.441026	1.649854	14.902564	9.671712	14	195	102	93	0.476923
2	3	1.011364	1.245630	16.352273	1.454599	10.181818	5.867396	12	88	33	55	0.625000
3	4	4.722222	0.826442	13.111111	1.745208	11.944445	9.973330	5	18	17	1	0.055556
4	5	1.621622	1.276961	15.729730	2.588958	10.189189	7.600577	4	37	23	14	0.378378

features

(23) Average order size of a user

(24) User's mean of reordered items of all orders

[ ]:
```
agg_dict5 = {
    'reordered': [('average_order_size', 'count'),
                  ('reorder_in_order', 'mean')]
}

user_feats2 = prior_df.groupby(['user_id', 'order_number']).agg(agg_dict5)
user_feats2.columns = user_feats2.columns.droplevel(0)
user_feats2.reset_index(inplace = True)
user_feats2.head()
```

[ ]:

	user_id	order_number	average_order_size	reorder_in_order
0	1	1	5	0.000
1	1	2	6	0.500
2	1	3	5	0.600
3	1	4	5	1.000
4	1	5	8	0.625

[ ]:
```
user_feats3 = user_feats2.groupby('user_id').agg({'average_order_size' : 'mean',
                                    'reorder_in_order':'mean'})
user_feats3 = user_feats3.reset_index()
user_feats3.head()
```

[ ]:

	user_id	average_order_size	reorder_in_order
0	1	5.900000	0.705833
1	2	13.928571	0.447961
2	3	7.333333	0.658817
3	4	3.600000	0.028571
4	5	9.250000	0.377778

[ ]:
```
user_feats = user_feats.merge(user_feats3, on = 'user_id', how = 'left')
user_feats.head()
```

[ ]:

	user_id	avg_dow	std_dow	avg_doh	std_doh	avg_since_order	\
0	1	2.644068	1.256194	10.542373	3.500355	18.542374	
1	2	2.005128	0.971222	10.441026	1.649854	14.902564	
2	3	1.011364	1.245630	16.352273	1.454599	10.181818	
3	4	4.722222	0.826442	13.111111	1.745208	11.944445	
4	5	1.621622	1.276961	15.729730	2.588958	10.189189	

	std_since_order	total_orders_by_user	total_products_by_user	\
0	10.559065	10	59	
1	9.671712	14	195	
2	5.867396	12	88	
3	9.973330	5	18	
4	7.600577	4	37	

	total_unique_product_by_user	total_reorders_by_user	\
0	18	41	
1	102	93	
2	33	55	
3	17	1	
4	23	14	

	reorder_propotion_by_user	average_order_size	reorder_in_order
0	0.694915	5.900000	0.705833
1	0.476923	13.928571	0.447961
2	0.625000	7.333333	0.658817

3	0.055556	3.600000	0.028571
4	0.378378	9.250000	0.377778

features

(25) Percentage of reordered itmes in user's last three orders

(26) Total orders in user's last three orders

Last 3 orders of a user

[ ]:
```
last_three_orders = user_feats2.groupby('user_id')['order_number'].nlargest(3).
 ↪reset_index()
last_three_orders.head()
```

[ ]:

	user_id	level_1	order_number
0	1	9	10
1	1	8	9
2	1	7	8
3	2	23	14
4	2	22	13

[ ]:
```
last_three_orders = user_feats2.merge(last_three_orders, on = ['user_id',␣
 ↪'order_number'], how = 'inner')
last_three_orders.head()
```

[ ]:

	user_id	order_number	average_order_size	reorder_in_order	level_1
0	1	8	6	0.666667	7
1	1	9	6	1.000000	8
2	1	10	9	0.666667	9
3	2	12	19	0.578947	21
4	2	13	9	0.000000	22

[ ]:
```
last_three_orders['rank'] = last_three_orders.
 ↪groupby("user_id")["order_number"].rank("dense", ascending=True)
```

[ ]:
```
last_order_feats = last_three_orders.pivot_table(index = 'user_id', columns =␣
 ↪['rank'], \
                                                  values=['average_order_size',␣
 ↪'reorder_in_order']).\
                                                 reset_index(drop = False)
last_order_feats.columns = ['user_id','orders_3', 'orders_2', 'orders_1',␣
 ↪'reorder_3', 'reorder_2', 'reorder_1']
last_order_feats.head()
```

[ ]:

	user_id	orders_3	orders_2	orders_1	reorder_3	reorder_2	reorder_1
0	1	6	6	9	0.666667	1.0	0.666667
1	2	19	9	16	0.578947	0.0	0.625000
2	3	6	5	6	0.833333	1.0	1.000000

```
3        4         7         2         3   0.142857        0.0   0.000000
4        5         9         5        12   0.444444        0.4   0.666667
```

```
[ ]: user_feats = user_feats.merge(last_order_feats, on = 'user_id', how = 'left')
     user_feats.head()
```

```
[ ]:    user_id   avg_dow   std_dow    avg_doh   std_doh  avg_since_order  \
     0        1  2.644068  1.256194  10.542373  3.500355        18.542374
     1        2  2.005128  0.971222  10.441026  1.649854        14.902564
     2        3  1.011364  1.245630  16.352273  1.454599        10.181818
     3        4  4.722222  0.826442  13.111111  1.745208        11.944445
     4        5  1.621622  1.276961  15.729730  2.588958        10.189189

        std_since_order  total_orders_by_user  total_products_by_user  \
     0        10.559065                    10                      59
     1         9.671712                    14                     195
     2         5.867396                    12                      88
     3         9.973330                     5                      18
     4         7.600577                     4                      37

        total_unique_product_by_user  total_reorders_by_user  \
     0                            18                      41
     1                           102                      93
     2                            33                      55
     3                            17                       1
     4                            23                      14

        reorder_propotion_by_user  average_order_size  reorder_in_order  orders_3  \
     0                   0.694915            5.900000          0.705833         6
     1                   0.476923           13.928571          0.447961        19
     2                   0.625000            7.333333          0.658817         6
     3                   0.055556            3.600000          0.028571         7
     4                   0.378378            9.250000          0.377778         9

        orders_2  orders_1  reorder_3  reorder_2  reorder_1
     0         6         9   0.666667        1.0   0.666667
     1         9        16   0.578947        0.0   0.625000
     2         5         6   0.833333        1.0   1.000000
     3         2         3   0.142857        0.0   0.000000
     4         5        12   0.444444        0.4   0.666667
```

1.1.4 User and Product level features

(27) User's avg add-to-cart-order for a product

(28) User's avg days_since_prior_order for a product

(29) User's product total orders, reorders and reorders percentage

(30) User's order number when the product was bought last

[ ]:
```
agg_dict6 = {'reordered': {'total_product_orders_by_user':'count',
                           'total_product_reorders_by_user':'sum',
                           'user_product_reorder_percentage': 'mean'},
             'add_to_cart_order': {'avg_add_to_cart_by_user':'mean'},
             'days_since_prior_order': {'avg_days_since_last_bought':'mean'},
             'order_number': {'last_ordered_in':'max'}}
```

[ ]:
```
user_product_feats = prior_df.groupby(['user_id', 'product_id']).agg(
    total_product_orders_by_user=('reordered', 'count'),
    total_product_reorders_by_user=('reordered', 'sum'),
    user_product_reorder_percentage=('reordered', 'mean'),
    avg_add_to_cart_by_user=('add_to_cart_order', 'mean'),
    avg_days_since_last_bought=('days_since_prior_order', 'mean'),
    last_ordered_in=('order_number', 'max')
)
user_product_feats.reset_index(inplace = True)
user_product_feats.head()
```

[ ]:

	user_id	product_id	total_product_orders_by_user	total_product_reorders_by_user	user_product_reorder_percentage	avg_add_to_cart_by_user	avg_days_since_last_bought	last_ordered_in
0	1	196	10	9	0.900000	1.400000	17.600000	10
1	1	10258	9	8	0.888889	3.333333	19.555555	10
2	1	10326	1	0	0.000000	5.000000	28.000000	5
3	1	12427	10	9	0.900000	3.300000	17.600000	10
4	1	13032	3	2	0.666667	6.333333	21.666666	10

features

(31) User's product purchase history of last three orders

[ ]:
```
last_three_orders.head()
```

[ ]:

	user_id	order_number	average_order_size	reorder_in_order	level_1	rank
0	1	8	6	0.666667	7	1.0
1	1	9	6	1.000000	8	2.0
2	1	10	9	0.666667	9	3.0
3	2	12	19	0.578947	21	1.0
4	2	13	9	0.000000	22	2.0

[ ]:
```
last_orders = prior_df.merge(last_three_orders, on = ['user_id',␣
↪'order_number'], how = 'inner')
last_orders.head()
```

[ ]:

	order_id	product_id	add_to_cart_order	reordered	user_id	eval_set	\
0	7	34050	1	0	142903	prior	
1	7	46802	2	0	142903	prior	
2	14	20392	1	1	18194	prior	
3	14	27845	2	1	18194	prior	
4	14	162	3	1	18194	prior	

	order_number	order_dow	order_hour_of_day	days_since_prior_order	\
0	11	2	14	30.0	
1	11	2	14	30.0	
2	49	3	15	3.0	
3	49	3	15	3.0	
4	49	3	15	3.0	

	product_name	aisle_id	department_id	\
0	Orange Juice	31	7	
1	Pineapple Chunks	116	1	
2	Hair Bender Whole Bean Coffee	26	7	
3	Organic Whole Milk	84	16	
4	Organic Mini Homestyle Waffles	52	1	

	user_buy_product_times	average_order_size	reorder_in_order	level_1	rank
0	1	2	0.000000	2231251	2.0
1	1	2	0.000000	2231251	2.0
2	1	11	0.818182	282882	1.0
3	1	11	0.818182	282882	1.0
4	1	11	0.818182	282882	1.0

[ ]:
```
last_orders['rank'] = last_orders.groupby(['user_id',␣
↪'product_id'])['order_number'].rank("dense", ascending=True)
```

[ ]:
```
product_purchase_history = last_orders.pivot_table(index = ['user_id',␣
↪'product_id'],\
                                                   columns='rank', values =␣
↪'reordered').reset_index()
```

```
product_purchase_history.columns = ['user_id', 'product_id', 'is_reorder_3',␣
↪'is_reorder_2', 'is_reorder_1']
product_purchase_history.fillna(0, inplace = True)
product_purchase_history.head()
```

[ ]:

	user_id	product_id	is_reorder_3	is_reorder_2	is_reorder_1
0	1	196	1.0	1.0	1.0
1	1	10258	1.0	1.0	1.0
2	1	12427	1.0	1.0	1.0
3	1	13032	1.0	0.0	0.0
4	1	25133	1.0	1.0	1.0

[ ]:
```
user_product_feats = user_product_feats.merge(product_purchase_history,␣
↪on=['user_id', 'product_id'], how = 'left')
user_product_feats.head()
```

[ ]:

	user_id	product_id	total_product_orders_by_user	\
0	1	196	10	
1	1	10258	9	
2	1	10326	1	
3	1	12427	10	
4	1	13032	3	

	total_product_reorders_by_user	user_product_reorder_percentage	\
0	9	0.900000	
1	8	0.888889	
2	0	0.000000	
3	9	0.900000	
4	2	0.666667	

	avg_add_to_cart_by_user	avg_days_since_last_bought	last_ordered_in	\
0	1.400000	17.600000	10	
1	3.333333	19.555555	10	
2	5.000000	28.000000	5	
3	3.300000	17.600000	10	
4	6.333333	21.666666	10	

	is_reorder_3	is_reorder_2	is_reorder_1
0	1.0	1.0	1.0
1	1.0	1.0	1.0
2	NaN	NaN	NaN
3	1.0	1.0	1.0
4	1.0	0.0	0.0

[ ]:
```
user_product_feats.isnull().sum()
```

[ ]:
```
user_id                                0
product_id                             0
total_product_orders_by_user           0
total_product_reorders_by_user         0
user_product_reorder_percentage        0
avg_add_to_cart_by_user                0
avg_days_since_last_bought             0
last_ordered_in                        0
is_reorder_3                     8382738
is_reorder_2                     8382738
is_reorder_1                     8382738
dtype: int64
```

[ ]:
```
user_product_feats.fillna(0, inplace = True)
```

1.2 Saving all features

[ ]:
```
prod_feats1.to_pickle(root + 'product_features.pkl')
user_feats.to_pickle(root +'user_features.pkl')
user_product_feats.to_pickle(root +'user_product_features.pkl')
```

[ ]:
```
df = pd.read_pickle(root +'product_features.pkl')
df.head()
```

[ ]:
```
   product_id  mean_add_to_cart_order  total_orders  total_reorders  \
0           1                5.801836          1852            1136
1           2                9.888889            90              12
2           3                6.415162           277             203
3           4                9.507599           329             147
4           5                6.466667            15               9

   reorder_percentage  unique_users  order_first_time_total_cnt  \
0            0.613391           716                         716
1            0.133333            78                          78
2            0.732852            74                          74
3            0.446809           182                         182
4            0.600000             6                           6

   order_second_time_total_cnt  is_organic  second_time_percent  …  \
0                          276           0             0.385475  …
1                            8           0             0.102564  …
2                           36           0             0.486486  …
3                           64           0             0.351648  …
4                            4           0             0.666667  …

   department_std_add_to_cart_order  department_total_orders  \
0                          7.692492                 2887550
```

```
1                          7.875241                  1875577
2                          6.711172                  2690129
3                          7.393502                  2236432
4                          7.875241                  1875577

   department_total_reorders  department_reorder_percentage  \
0                    1657973                       0.574180
1                     650301                       0.346721
2                    1757892                       0.653460
3                    1211890                       0.541885
4                     650301                       0.346721

   department_unique_users  department_0  department_1  department_2  \
0                   174219             0             0             0
1                   172755             0             0             0
2                   172795             0             0             0
3                   163233             0             0             1
4                   172755             0             0             0

   department_3  department_4
0             0             1
1             1             0
2             1             1
3             0             0
4             1             0

[5 rows x 35 columns]
```

[ ]:
```
df = pd.read_pickle(root+'user_features.pkl')
df.head()
```

[ ]:
```
   user_id   avg_dow   std_dow    avg_doh   std_doh  avg_since_order  \
0        1  2.644068  1.256194  10.542373  3.500355        18.542374
1        2  2.005128  0.971222  10.441026  1.649854        14.902564
2        3  1.011364  1.245630  16.352273  1.454599        10.181818
3        4  4.722222  0.826442  13.111111  1.745208        11.944445
4        5  1.621622  1.276961  15.729730  2.588958        10.189189

   std_since_order  total_orders_by_user  total_products_by_user  \
0        10.559065                    10                      59
1         9.671712                    14                     195
2         5.867396                    12                      88
3         9.973330                     5                      18
4         7.600577                     4                      37

   total_unique_product_by_user  total_reorders_by_user  \
0                            18                      41
```

```
1                          102                    93
2                           33                    55
3                           17                     1
4                           23                    14

   reorder_propotion_by_user  average_order_size  reorder_in_order  orders_3  \
0                   0.694915            5.900000          0.705833         6
1                   0.476923           13.928571          0.447961        19
2                   0.625000            7.333333          0.658817         6
3                   0.055556            3.600000          0.028571         7
4                   0.378378            9.250000          0.377778         9

   orders_2  orders_1  reorder_3  reorder_2  reorder_1
0         6         9   0.666667        1.0   0.666667
1         9        16   0.578947        0.0   0.625000
2         5         6   0.833333        1.0   1.000000
3         2         3   0.142857        0.0   0.000000
4         5        12   0.444444        0.4   0.666667
```

[ ]:
```
df = pd.read_pickle(root + 'user_product_features.pkl')
df.head()
```

[ ]:
```
   user_id  product_id  total_product_orders_by_user  \
0        1         196                            10
1        1       10258                             9
2        1       10326                             1
3        1       12427                            10
4        1       13032                             3

   total_product_reorders_by_user  user_product_reorder_percentage  \
0                               9                         0.900000
1                               8                         0.888889
2                               0                         0.000000
3                               9                         0.900000
4                               2                         0.666667

   avg_add_to_cart_by_user  avg_days_since_last_bought  last_ordered_in  \
0                 1.400000                   17.600000               10
1                 3.333333                   19.555555               10
2                 5.000000                   28.000000                5
3                 3.300000                   17.600000               10
4                 6.333333                   21.666666               10

   is_reorder_3  is_reorder_2  is_reorder_1
0           1.0           1.0           1.0
1           1.0           1.0           1.0
2           0.0           0.0           0.0
```

```
3           1.0           1.0           1.0
4           1.0           0.0           0.0
```