feng

November 17, 2021

```
[ ]: import joblib
     import pandas as pd
     import numpy as np
     df = joblib.load('df_data.pkl')
     df.head()
```

```
[ ]:                                  id                     channel_sales  \
   0  48ada52261e7cf58715202705a0451c9  lmkebamcaaclubfxadlmueccxoimlema
   1  24011ae4ebbe3035111d65fa7c15bc57  foosdfpfkusacimwkcsosbicdxkicaua
   2  d29c2c54acc38ff3c0614d0a653813dd                                 0
   3  764c75f661154dac3a6c254cd082ea7d  foosdfpfkusacimwkcsosbicdxkicaua
   4  bba03439a292a1e166f80264c16191cb  lmkebamcaaclubfxadlmueccxoimlema

      cons_12m  cons_gas_12m  cons_last_month  date_activ    date_end  \
   0    309275             0            10025  2012-11-07  2016-11-06
   1         0         54946                0  2013-06-15  2016-06-15
   2      4660             0                0  2009-08-21  2016-08-30
   3       544             0                0  2010-04-16  2016-04-16
   4      1584             0                0  2010-03-30  2016-03-30

     date_modif_prod date_renewal  forecast_cons_12m  …  imp_cons  \
   0      2012-11-07   2015-11-09           26520.30  …     831.8
   1             NaN   2015-06-23               0.00  …       0.0
   2      2009-08-21   2015-08-31             189.95  …       0.0
   3      2010-04-16   2015-04-17              47.96  …       0.0
   4      2010-03-30   2015-03-31             240.04  …       0.0

      margin_gross_pow_ele  margin_net_pow_ele  nb_prod_act  net_margin  \
   0                -41.76              -41.76            1     1732.36
   1                 25.44               25.44            2      678.99
   2                 16.38               16.38            1       18.89
   3                 28.60               28.60            1        6.60
   4                 30.22               30.22            1       25.46

      num_years_antig                         origin_up  pow_max  churn  \
   0                3  ldkssxwpmemidmecebumciepifcamkci  180.000      0
   1                3  lxidpiddsbxsbosboudacockeimpuepw   43.648      1
   2                6  kamkkxfxxuwbdslkwifmmcsiusiuosws   13.800      0
```

```
3                6  kamkkxfxxuwbdslkwifmmcsiusiuosws   13.856      0
4                6  kamkkxfxxuwbdslkwifmmcsiusiuosws   13.200      0

   churn_retain
0        retain
1         churn
2        retain
3        retain
4        retain

[5 rows x 27 columns]
```

1 Principal component analysis

we will keep only the important features for our modelling.

```
[ ]: df['date_activ'] = pd.to_datetime(df['date_activ'], format='%Y-%m-%d')
     df['date_end'] = pd.to_datetime(df['date_end'], format='%Y-%m-%d')
```

```
[ ]: from datetime import datetime, timedelta
     df['active_dur'] = (df.date_end - df.date_activ).dt.days
     df = df[['id', 'cons_12m', 'cons_gas_12m',
              'cons_last_month', 'has_gas', 'nb_prod_act', 'num_years_antig',␣
      ↪'pow_max', 'active_dur', 'churn']]
     print(df.shape)
     df.head()
```

```
(16096, 10)
```

```
[ ]:                                  id  cons_12m  cons_gas_12m  cons_last_month  \
     0  48ada52261e7cf58715202705a0451c9    309275             0            10025
     1  24011ae4ebbe3035111d65fa7c15bc57         0         54946                0
     2  d29c2c54acc38ff3c0614d0a653813dd      4660             0                0
     3  764c75f661154dac3a6c254cd082ea7d       544             0                0
     4  bba03439a292a1e166f80264c16191cb      1584             0                0

       has_gas  nb_prod_act  num_years_antig  pow_max  active_dur  churn
     0       f            1                3  180.000      1460.0      0
     1       t            2                3   43.648      1096.0      1
     2       f            1                6   13.800      2566.0      0
     3       f            1                6   13.856      2192.0      0
     4       f            1                6   13.200      2192.0      0
```

```
[ ]: df1 = joblib.load('hist_data.pkl')
     df1.drop(['price_date'], axis=1, inplace=True)
     print(df1.shape)
     df1.head()
```

```
(16096, 13)
```

```
[ ]:                               id_x  price_p1_var  price_p2_var  price_p3_var  \
     0  038af19179925da21a25619c5a24b745      0.151367           0.0           0.0
     1  038af19179925da21a25619c5a24b745      0.151367           0.0           0.0
     2  038af19179925da21a25619c5a24b745      0.151367           0.0           0.0
     3  038af19179925da21a25619c5a24b745      0.149626           0.0           0.0
     4  038af19179925da21a25619c5a24b745      0.149626           0.0           0.0

        price_p1_fix  price_p2_fix  price_p3_fix                              id_y  \
     0     44.266931           0.0           0.0  48ada52261e7cf58715202705a0451c9
     1     44.266931           0.0           0.0  24011ae4ebbe3035111d65fa7c15bc57
     2     44.266931           0.0           0.0  d29c2c54acc38ff3c0614d0a653813dd
     3     44.266931           0.0           0.0  764c75f661154dac3a6c254cd082ea7d
     4     44.266931           0.0           0.0  bba03439a292a1e166f80264c16191cb

        churn_x churn_retain_x                                id  churn_y  \
     0        0         retain  48ada52261e7cf58715202705a0451c9        0
     1        1          churn  24011ae4ebbe3035111d65fa7c15bc57        1
     2        0         retain  d29c2c54acc38ff3c0614d0a653813dd        0
     3        0         retain  764c75f661154dac3a6c254cd082ea7d        0
     4        0         retain  bba03439a292a1e166f80264c16191cb        0

       churn_retain_y
     0         retain
     1          churn
     2         retain
     3         retain
     4         retain
```

Preparing final data

```
[ ]: df = pd.merge(left=df, right=df1, how='inner',
                   left_on='id', right_on='id')
     print(df.shape)
     df.drop(['id_x', 'id_y', 'churn', 'churn_x',        'churn_retain_x',␣
      ↪'churn_retain_y'], axis=1, inplace=True)
     df.head()
```

```
(16096, 22)
```

```
[ ]:                                 id  cons_12m  cons_gas_12m  cons_last_month  \
     0  48ada52261e7cf58715202705a0451c9    309275             0            10025
     1  24011ae4ebbe3035111d65fa7c15bc57         0         54946                0
     2  d29c2c54acc38ff3c0614d0a653813dd      4660             0                0
     3  764c75f661154dac3a6c254cd082ea7d       544             0                0
     4  bba03439a292a1e166f80264c16191cb      1584             0                0
```

```
  has_gas  nb_prod_act  num_years_antig  pow_max  active_dur  price_p1_var  \
0       f            1                3  180.000      1460.0      0.151367
1       t            2                3   43.648      1096.0      0.151367
2       f            1                6   13.800      2566.0      0.151367
3       f            1                6   13.856      2192.0      0.149626
4       f            1                6   13.200      2192.0      0.149626

   price_p2_var  price_p3_var  price_p1_fix  price_p2_fix  price_p3_fix  \
0           0.0           0.0     44.266931           0.0           0.0
1           0.0           0.0     44.266931           0.0           0.0
2           0.0           0.0     44.266931           0.0           0.0
3           0.0           0.0     44.266931           0.0           0.0
4           0.0           0.0     44.266931           0.0           0.0

   churn_y
0        0
1        1
2        0
3        0
4        0
```

has_gas can be converted to categorical by replacing t and f via dictionary or using simple get_dummies method.

```
[ ]: gas_dict = {'f': 0, 't': 1}
     df['has_gas'] = df['has_gas'].replace(gas_dict).astype('category').astype(int)
     df.rename(columns={'churn_y': 'churn'}, inplace=True)
     df.head()
```

```
[ ]:                                  id  cons_12m  cons_gas_12m  cons_last_month  \
     0  48ada52261e7cf58715202705a0451c9    309275             0            10025
     1  24011ae4ebbe3035111d65fa7c15bc57         0         54946                0
     2  d29c2c54acc38ff3c0614d0a653813dd      4660             0                0
     3  764c75f661154dac3a6c254cd082ea7d       544             0                0
     4  bba03439a292a1e166f80264c16191cb      1584             0                0

        has_gas  nb_prod_act  num_years_antig  pow_max  active_dur  price_p1_var  \
     0        0            1                3  180.000      1460.0      0.151367
     1        1            2                3   43.648      1096.0      0.151367
     2        0            1                6   13.800      2566.0      0.151367
     3        0            1                6   13.856      2192.0      0.149626
     4        0            1                6   13.200      2192.0      0.149626

        price_p2_var  price_p3_var  price_p1_fix  price_p2_fix  price_p3_fix  churn
     0           0.0           0.0     44.266931           0.0           0.0      0
     1           0.0           0.0     44.266931           0.0           0.0      1
     2           0.0           0.0     44.266931           0.0           0.0      0
     3           0.0           0.0     44.266931           0.0           0.0      0
```

```
4            0.0            0.0      44.266931           0.0            0.0       0
```

[ ]:

```
df.dtypes
```

[ ]:

```
id                    object
cons_12m               int64
cons_gas_12m           int64
cons_last_month        int64
has_gas                int64
nb_prod_act            int64
num_years_antig        int64
pow_max              float64
active_dur           float64
price_p1_var         float64
price_p2_var         float64
price_p3_var         float64
price_p1_fix         float64
price_p2_fix         float64
price_p3_fix         float64
churn                  int64
dtype: object
```

[ ]:

```
import joblib
joblib.dump(df, 'finaldf.pkl')
```

[ ]:

```
['finaldf.pkl']
```