01-Import

February 1, 2022

1 Churn — Import

Dataset is in two csv (churn and states). Main issues are:

- Poor column names - spaces and punctation
- Inconsistent labels for boolean columns
- Unique identifier column `Phone`

1.1 Imports and Setup

[1]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns

from IPython.display import display, Markdown
plt.style.use("seaborn-darkgrid")
pd.set_option('display.max_columns', None)

import os
for d in ['orig','data','output']: os.makedirs(d, exist_ok=True)

DEBUG = False
SEED = 72
```

1.2 Datasets

1.2.1 Churn

[2]:
```
import os
for d in ['orig','data','output']: os.makedirs(d, exist_ok=True)

for filename in ['churn.csv','states.csv']:
    source = f"https://datamining2-202122.github.io/live/topics/
 ↪02-Feature_Engineering/03-Practical_02_-_Churn_-_Baseline_Model/files/
 ↪{filename}"
    target = f"orig/{filename}"

    if not os.path.isfile(target):
```

```
        print (f"Downloading remote file {filename}", sep="")
        import urllib.request
        urllib.request.urlretrieve(source, target)
    else:
        print(f"Using local copy of {filename}")
```

```
Using local copy of churn.csv
Using local copy of states.csv
```

[3]:
```
df_churn = pd.read_csv("orig/churn.csv")
print(df_churn.shape)
df_churn.head(1)
```

```
(3333, 21)
```

[3]:

	State	Account Length	Area Code	Phone	Int'l Plan	VMail Plan	VMail Message	Day Mins	Day Calls	Day Charge	Eve Mins	Eve Calls	Eve Charge	Night Mins	Night Calls	Night Charge	Intl Mins	Intl Calls	Intl Charge	CustServ Calls	Churn?
0	KS	128	415	382-4657	no	yes	25	265.1	110	45.07	197.4	99	16.78	244.7	91	11.01	10.0	3	2.7	1	False.

[4]:
```
columns = df_churn.columns
columns
```

[4]:
```
Index(['State', 'Account Length', 'Area Code', 'Phone', 'Int'l Plan',
       'VMail Plan', 'VMail Message', 'Day Mins', 'Day Calls', 'Day Charge',
       'Eve Mins', 'Eve Calls', 'Eve Charge', 'Night Mins', 'Night Calls',
       'Night Charge', 'Intl Mins', 'Intl Calls', 'Intl Charge',
       'CustServ Calls', 'Churn?'],
      dtype='object')
```

[5]:
```
df_churn.columns = [c.replace(" ", "_").replace("'", "").replace("?", "") for c␣
↪in columns]
```

[6]:
```
df_churn.head(1)
```

[6]:

	State	Account_Length	Area_Code	Phone	Intl_Plan	VMail_Plan	VMail_Message	Day_Mins	Day_Calls	Day_Charge	Eve_Mins	Eve_Calls
0	KS	128	415	382-4657	no	yes	25	265.1	110	45.07	197.4	99

```
   Eve_Charge  Night_Mins  Night_Calls  Night_Charge  Intl_Mins  Intl_Calls  \
0       16.78       244.7           91         11.01       10.0           3

   Intl_Charge  CustServ_Calls   Churn
0          2.7               1  False.
```

[7]:
```
df_churn.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 3333 entries, 0 to 3332
Data columns (total 21 columns):
 #   Column          Non-Null Count  Dtype
---  ------          --------------  -----
 0   State           3333 non-null   object
 1   Account_Length  3333 non-null   int64
 2   Area_Code       3333 non-null   int64
 3   Phone           3333 non-null   object
 4   Intl_Plan       3333 non-null   object
 5   VMail_Plan      3333 non-null   object
 6   VMail_Message   3333 non-null   int64
 7   Day_Mins        3333 non-null   float64
 8   Day_Calls       3333 non-null   int64
 9   Day_Charge      3333 non-null   float64
 10  Eve_Mins        3333 non-null   float64
 11  Eve_Calls       3333 non-null   int64
 12  Eve_Charge      3333 non-null   float64
 13  Night_Mins      3333 non-null   float64
 14  Night_Calls     3333 non-null   int64
 15  Night_Charge    3333 non-null   float64
 16  Intl_Mins       3333 non-null   float64
 17  Intl_Calls      3333 non-null   int64
 18  Intl_Charge     3333 non-null   float64
 19  CustServ_Calls  3333 non-null   int64
 20  Churn           3333 non-null   object
dtypes: float64(8), int64(8), object(5)
memory usage: 546.9+ KB
```

[8]:
```
df_churn.Intl_Plan.unique()
```

[8]:
```
array(['no', 'yes'], dtype=object)
```

[9]:
```
for c in [c for c in df_churn.columns if "Plan" in c]:
    if df_churn[c].dtype =="object":
        df_churn[c] = df_churn[c].map( {"no":0, "yes":1} )
```

[10]:
```
df_churn.head(10)
```

[10]:

	State	Account_Length	Area_Code	Phone	Intl_Plan	VMail_Plan
0	KS	128	415	382-4657	0	1
1	OH	107	415	371-7191	0	1
2	NJ	137	415	358-1921	0	0
3	OH	84	408	375-9999	1	0
4	OK	75	415	330-6626	1	0
5	AL	118	510	391-8027	1	0
6	MA	121	510	355-9993	0	1
7	MO	147	415	329-9001	1	0
8	LA	117	408	335-4719	0	0
9	WV	141	415	330-8173	1	1

	VMail_Message	Day_Mins	Day_Calls	Day_Charge	Eve_Mins	Eve_Calls
0	25	265.1	110	45.07	197.4	99
1	26	161.6	123	27.47	195.5	103
2	0	243.4	114	41.38	121.2	110
3	0	299.4	71	50.90	61.9	88
4	0	166.7	113	28.34	148.3	122
5	0	223.4	98	37.98	220.6	101
6	24	218.2	88	37.09	348.5	108
7	0	157.0	79	26.69	103.1	94
8	0	184.5	97	31.37	351.6	80
9	37	258.6	84	43.96	222.0	111

	Eve_Charge	Night_Mins	Night_Calls	Night_Charge	Intl_Mins	Intl_Calls
0	16.78	244.7	91	11.01	10.0	3
1	16.62	254.4	103	11.45	13.7	3
2	10.30	162.6	104	7.32	12.2	5
3	5.26	196.9	89	8.86	6.6	7
4	12.61	186.9	121	8.41	10.1	3
5	18.75	203.9	118	9.18	6.3	6
6	29.62	212.6	118	9.57	7.5	7
7	8.76	211.8	96	9.53	7.1	6
8	29.89	215.8	90	9.71	8.7	4
9	18.87	326.4	97	14.69	11.2	5

	Intl_Charge	CustServ_Calls	Churn
0	2.70	1	False.
1	3.70	1	False.
2	3.29	0	False.
3	1.78	2	False.
4	2.73	3	False.
5	1.70	0	False.
6	2.03	3	False.
7	1.92	0	False.
8	2.35	1	False.
9	3.02	0	False.

[11]:
```
if False and 0 not in df_churn.Area_Code.unique():
    df_churn.Area_Code = df_churn.Area_Code.map( {415:0, 510:1,408:2 } )
```

[12]:
```
if df_churn.Churn.dtype == "object":
    df_churn.Churn = df_churn.Churn.map( {"False.":0, "True.":1} )
```

[13]:
```
df_churn.dtypes
```

[13]:
```
State              object
Account_Length      int64
Area_Code           int64
Phone              object
Intl_Plan           int64
VMail_Plan          int64
VMail_Message       int64
Day_Mins          float64
Day_Calls           int64
Day_Charge        float64
Eve_Mins          float64
Eve_Calls           int64
Eve_Charge        float64
Night_Mins        float64
Night_Calls         int64
Night_Charge      float64
Intl_Mins         float64
Intl_Calls          int64
Intl_Charge       float64
CustServ_Calls      int64
Churn               int64
dtype: object
```

[14]:
```
df_churn['Area'] = df_churn.Phone.apply(lambda s: s.split('-')[0])
```

[15]:
```
df_churn['Area'].value_counts()
```

[15]:
```
405    53
408    48
352    47
406    47
417    46
       ..
342    24
421    24
412    23
422    19
327    19
Name: Area, Length: 96, dtype: int64
```

[16]:
```
df_churn.to_csv("data/churn.csv", index=False)
```

1.2.2 States

[17]:
```
df_state = pd.read_csv("orig/states.csv")
print(df_state.shape)
df_state.head(1)
```

```
(52, 4)
```

[17]:

	state	latitude	longitude	name
0	AK	63.588753	-154.493062	Alaska

[18]:
```
df_state.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 52 entries, 0 to 51
Data columns (total 4 columns):
 #   Column     Non-Null Count  Dtype
---  ------     --------------  -----
 0   state      52 non-null     object
 1   latitude   52 non-null     float64
 2   longitude  52 non-null     float64
 3   name       52 non-null     object
dtypes: float64(2), object(2)
memory usage: 1.8+ KB
```

[19]:
```
df_state.columns = [c.title() for c in df_state.columns]
```

[20]:
```
df_state.to_csv("data/states.csv", index=False)
```

[ ]:
```

```

[ ]:
```

```