invoice

March 15, 2024

[116]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns

# By Danielle and Yoshita

df = pd.read_csv('data_eda/customer.csv')
df.head(10)
df1 = pd.read_csv('data_eda/invoices.csv')
df1.head(10)
```

[116]:
```
   invoice_identifier invoice_date    due_date  amount
0              234563   01/02/2013  02/01/2013  148.80
1              234564    1/26/2013   2/25/2013  164.23
2              234565   07/03/2013  08/02/2013  175.24
3              234566   02/10/2013  03/12/2013  281.75
4              234567   10/25/2012  11/24/2012  192.24
5              234568    1/27/2012   2/26/2012  250.04
6              234569    8/13/2013  09/12/2013  198.68
7              234570   12/16/2012   1/15/2013  199.66
8              234571    5/14/2012   6/13/2012  212.99
9              234572   07/01/2013   7/31/2013  128.56
```

[117]:
```
df.dropna(subset=['country', 'customer_id', 'invoice_id'], inplace=True)
df.dropna(subset=['challenged', 'settled_date', 'days_late'], inplace=True)
df.dropna(subset=['invoice_format'], inplace=True)

df1.dropna(subset=['invoice_identifier', 'invoice_date', 'due_date'],␣
 ↪inplace=True)
df1.dropna(subset=['amount'], inplace=True)

# We were able to remove the rows with empty values in the CSVS.
```

[118]:
```
df = df[df['days_late'] >= 0]
df1 = df1[df1['amount'] >= 0]
```

```python
# We removed the negative values
```

[119]:

```python
df.drop_duplicates(subset=["invoice_id"])
df1.drop_duplicates(subset=["invoice_identifier"])

# We checked columns where duplicates may be a problem.
```

[119]:

```
      invoice_identifier invoice_date    due_date  amount
0                 234563   01/02/2013  02/01/2013  148.80
1                 234564    1/26/2013   2/25/2013  164.23
2                 234565   07/03/2013  08/02/2013  175.24
3                 234566   02/10/2013  03/12/2013  281.75
4                 234567   10/25/2012  11/24/2012  192.24
…                    …          …          …        …
2461              237024   10/18/2013  11/17/2013  211.76
2462              237025    9/19/2012  10/19/2012  101.74
2463              237026   07/02/2012  08/01/2012  179.79
2464              237027    4/27/2012   5/27/2012  141.41
2465              237028   07/04/2013  08/03/2013  182.64

[2465 rows x 4 columns]
```

[120]:

```python
result = pd.concat([df, df1], axis=1, join='inner')
result
```

[120]:

```
      country customer_id  invoice_id challenged settled_date invoice_format  \
0       391.0  0379-NEVHP      234563         No    1/15/2013          Paper
1       406.0  8976-AMJEO      234564        Yes   03/03/2013     Electronic
2       391.0  2820-XGXSB      234565         No   07/08/2013     Electronic
3       406.0  9322-YCTQO      234566         No    3/17/2013     Electronic
4       818.0  6627-ELFBK      234567        Yes   11/28/2012          Paper
…         …         …           …          …           …             …
2461    391.0  6708-DPYTF      237024         No   12/01/2013     Electronic
2462    391.0  9841-XLGBV      237025         No   10/13/2012          Paper
2463    770.0  7856-ODQFO      237026         No    7/27/2012          Paper
2464    770.0  7050-KQLDO      237027         No    5/18/2012          Paper
2465    406.0  9758-AIEIK      237028         No    7/18/2013     Electronic

      days_late  invoice_identifier invoice_date    due_date  amount
0             0              234563   01/02/2013  02/01/2013  148.80
1             6              234564    1/26/2013   2/25/2013  164.23
2             0              234565   07/03/2013  08/02/2013  175.24
3             5              234566   02/10/2013  03/12/2013  281.75
4             4              234567   10/25/2012  11/24/2012  192.24
…          …                  …          …          …       …
2461         14              237024   10/18/2013  11/17/2013  211.76
2462          0              237025    9/19/2012  10/19/2012  101.74
```

```
2463           0              237026   07/02/2012  08/01/2012  179.79
2464           0              237027    4/27/2012   5/27/2012  141.41
2465           0              237028   07/04/2013  08/03/2013  182.64

[2345 rows x 11 columns]
```

[121]:

```
corr = result.corr()
corr.style.background_gradient(cmap='coolwarm')
```

[121]:

```
<pandas.io.formats.style.Styler at 0x7fde7ef21fa0>
```

[ ]:

```
# latevformat = pd.read_csv("data_eda/customer.csv")

# plt.plot(result.amount, result.days_late)
# plt.show()
# We need to order data but did not reach

plt.xlabel('Amount of Invoice')
plt.ylabel('Days Late')
plt.title('Amount of Invoice vs Days Late')
plt.bar(result.amount, result.days_late)
```

[ ]:

```
<BarContainer object of 2345 artists>
```

[123]:

```
# We have noticed that there is a greater likelyhood
# of turning in the invoice late if it is between
# $100.00 - $250.00. We recommend sending recommendations
# of utilising a payment plan and/or sending more reminders.
```