RFM в Pandas

https://colab.research.google.com/drive/1utKQ5OiPFJ\_228s2bYHjBT5YLBd7q1mM?usp=sharing#scrollTo=T\_AsnPrKIdOI

import pandas as pd

df = pd.read\_csv(

    'https://gist.githubusercontent.com/andron23/f73a19243564d80a388db2b7b1542fd2/raw/2479b4326f6c8ae13d06060fb8510d04bcf06113/apteka.csv',

     sep=';'

)

df.head()

df['datetime'] = pd.to\_datetime(df['datetime'])

df.info()

df = df[df['card'].str.startswith('2000')]

df = df.sort\_values(['card', 'datetime'])

df

df2 = df.groupby('card').agg(

    purchase\_sum = ('summ\_with\_disc', 'sum'),

    purchase\_amount = ('summ\_with\_disc', 'count'),

    last\_purchase = ('datetime', 'last'),

).reset\_index()

df[df['card'] == '2000200150091']

df2

df2['days\_since\_last\_purchase'] = (max(df['datetime']) - df2['last\_purchase']).dt.days

df2

import seaborn as sns

sns.violinplot(df2['purchase\_sum'])

import numpy as np

quantiles = [round(el, 2) for el in np.arange(0.1, 1.1, 0.1)]

df2['purchase\_sum'].quantile(quantiles)

ax = sns.barplot(

    df2['purchase\_sum'].quantile(quantiles),

    orient='h'

)

ax.bar\_label(ax.containers[-1]);

ax = sns.barplot(

    df2['purchase\_amount'].quantile(quantiles),

    orient='h'

)

ax.bar\_label(ax.containers[-1]);

ax = sns.barplot(

    df2['days\_since\_last\_purchase'].quantile(quantiles),

    orient='h'

)

ax.bar\_label(ax.containers[-1]);

quantiles = [0.33, 0.66]

df2['days\_since\_last\_purchase'].quantile(quantiles)

df2

def set\_score(val, var, perc\_33, perc\_66):

  if val < perc\_33:

    return 3 if var != 'R' else 1

  elif val < perc\_66:

    return 2

  else:

    return 1 if var != 'R' else 3

recency\_quantiles = df2['days\_since\_last\_purchase'].quantile(quantiles)

df2['R'] = df2['days\_since\_last\_purchase'].apply(set\_score, args=('R', recency\_quantiles.iloc[0], recency\_quantiles.iloc[1]))

frequency\_quantiles = df2['purchase\_amount'].quantile(quantiles)

df2['F'] = df2['purchase\_amount'].apply(set\_score, args=('F', frequency\_quantiles.iloc[0], frequency\_quantiles.iloc[1]))

monetary\_quantiles = df2['purchase\_sum'].quantile(quantiles)

df2['M'] = df2['purchase\_sum'].apply(set\_score, args=('M', monetary\_quantiles.iloc[0], monetary\_quantiles.iloc[1]))

df2

df2['RFM'] = df2.apply(lambda row: f"{row['R']}{row['F']}{row['M']}", axis=1)

df2

import plotly.express as px

df3 = df2.groupby('RFM')['RFM'].agg({'count'}).reset\_index()

df3

px.treemap(df3, path=['RFM'], values='count')