Pythonによるドーナツの売上予測

山形東高校H8班　2年　木野紗花

研究動機　目的

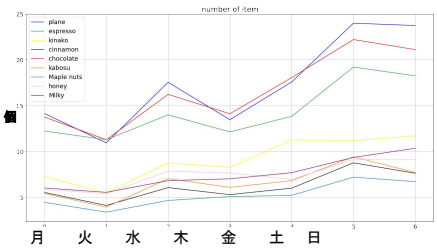
学校近くにドーナツ店があるが、私が行くころには売り切れている事が多い。需要と供給のバランスを導き出せれば私と同じような利用者にも、なおかつ御店主にも貢献できるのではないかと思い、探究活動で学習中のPythonを用い、売上予測を行っている。

研究(予測)方法

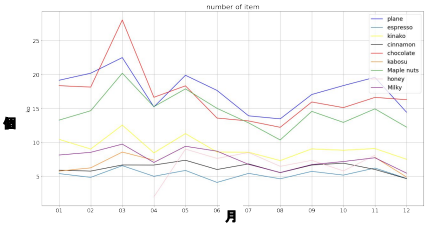
店舗は七日町商店街近くの道路沿いにあり、専用の駐車場はなく、徒歩でのお客さんが多い。客層は主婦が多いようだ。
売上は過去3年分、商品数は過去1年分のデータを加工し、また外部からもデータを取得し、Pythonのライブラリのモデルで説明変数(週や降雪量、コロナ....)と目的変数(金額)の対応関係を学習させ、一月の予測をする

データの可視化

Pythonのライブラリmatplotlib.pyplotやseabornを用いて売上と週、月、天気等との関係を調べる。

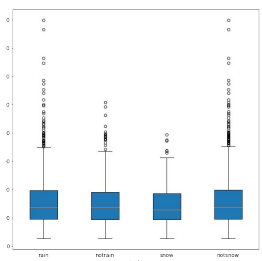


2つの折れ線グラフ
それぞれの平均をだし、グラフ化したもの。(限りがあるので一部掲載)
火曜日は定休日や閉店時間が早い日が多い。土日の売上が多いことがわかる。
商品は売上の多い「プレーン、エスプレッソ、きなこミルク、シナモン、チョコレート、はちみつかぼす、メープルナッツ、ハニーレモン、ミルキー」に絞った。
プレーン、チョコレート、メイプルナッツの順に多いことがわかる。



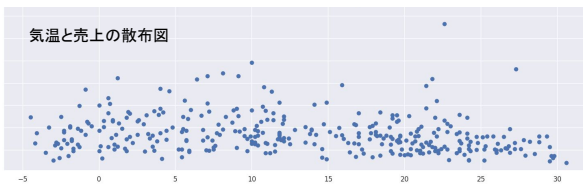
店主様から雨や雪が降るとお客さんが減るとお聞きした。店の前の道路は消雪だが、近くには消雪になっていないところも多い。

箱ひげ図
縦軸が売上(具体的な金額は載せられません)
あまり差はないが、雪が降った日がわずかに少ない。



雨○　雨×　雪○　雪×

気温と売上の散布図
相関係数0.180192
横軸が気温　縦軸が売上
あまり関係は見られなかった。
天気などの影響は日ごとではなく、時間に影響するのかもしれない。



一番上のグラフのコード ↑

```
import datetime as dt
dfa['weekday'] = dfa.index.weekday
import matplotlib.pyplot as plt
from matplotlib.pylab import rcParams
rcParams["figure.figsize"]=13,10
plt.rcParams["font.size"] = 18
fig, ax = plt.subplots()
l1,l2,l3,l4,l5,l6,l7,l8,l9="plane","espresso","kinako","cinnamon","chocolate","kabosu","Maplenuts","honey","Milky"
c1,c2,c4,c3,c8="blue","red","black","yellow" ,"pink"
ax.set_xlabel('weekday')  # x軸ラベル
ax.set_ylabel('ko')  # y軸ラベル
ax.set_title("number of item",fontname="MS Gothic")
ax.grid()  # 罫線
ax.plot(caweek.index,caweek["プレーン"],color=c1,label=l1)
ax.plot(caweek.index,caweek["エスプレッソ"],label=l2)
ax.plot(caweek.index,caweek["きなこミルク"], color=c3, label=l3)
ax.plot(caweek.index,caweek["シナモン"], color=c4, label=l4)
ax.plot(caweek.index,caweek["チョコレート"], label=l5,color=c2)
ax.plot(caweek.index,caweek["はちみつかぼす"], label=l6)
ax.plot(caweek.index,caweek["メイプルナッツ"], label=l7)
ax.plot(caweek.index,caweek["ハニーレモン"],color=c8, label=l8)
ax.plot(caweek.index,caweek["ミルキー"], label=l9,color="purple")
ax.legend(loc=0)
fig.tight_layout()  # レイアウトの設定
plt.show()
```

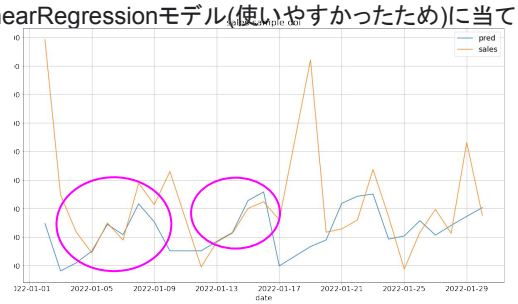
②のxgboostでのコード↗

```
import xgboost as xgb
from sklearn.metrics import log_loss
dtrain = xgb.DMatrix(tr_x, label=tr)
dvalid = xgb.DMatrix(va_x, label=va)
dtest = xgb.DMatrix(test_x1)
params = {'objective': 'reg:squarederror', 'silent': 1, 'random_state': 71,"max_depth":7}
num_round = 50
#watchlistには学習データおよびバリデーションデータをセットする
watchlist = [(dtrain, 'train'), (dvalid, 'eval')]
model = xgb.train(params, dtrain, num_round, evals=watchlist)
# バリデーションデータでのスコアの確認
va_pred = model.predict(dvalid)
score = np.sqrt(mean_squared_log_error(va_y["プレーン"], va_pred))
print(f'value rmsle: {score:f}')
predg = model.predict(dtest)
re["xgb"]=predg
```

```
import matplotlib.pyplot as plt
fig, ax = plt.subplots()
from matplotlib.pylab import rcParams
rcParams["font.size"]=18
l1,l2 = "pred","plane"
ax.set_xlabel('date')  # x軸ラベル
ax.set_ylabel('ko')  # y軸ラベル
ax.set_title("plane")
ax.grid()  # 罫線
ax.plot(re.index,re["pred"],label=l1)
ax.plot(re.index,re["プレーン"], label=l2,color="black")
ax.plot(re.index,re["xgb"],label="xgb")
ax.plot(re.index,re["xgb1"],label="xgb1 xlos")
ax.legend(loc=0)  # 凡例
fig.tight_layout()  # レイアウトの設定
plt.show()
score = np.sqrt(mean_squared_log_error(re["プレーン"], predg))
print(f'real rmsle : {score:f}')
```

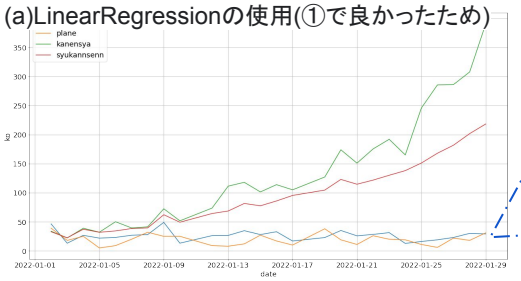
予測

①合計の売上
3年分の過去データを基に、特徴量を日、月、週、降水量、降雪量にしてLinearRegressionモデル(使いやすかったため)に当てはめ、売上予測



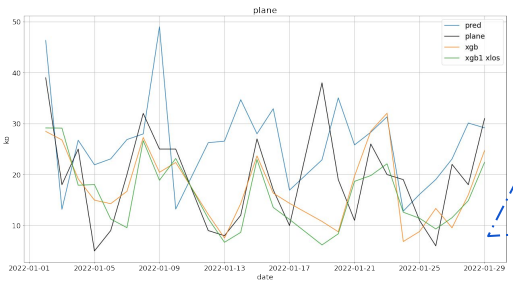
黄色が実際の売上、青が予測した売上。意外に予測出来ている。AIすごい！
しかし年始めの売上増加や、その他急激な上がり幅は予測出来なかった。(19日はイベントに参加し、売上があがったようだ)

②商品(プレーン)の売上個数
目的である個数の売上予測をするため1年分の過去データをトレーニングデータとした。(商品ごとのデータは一日ごとで3年分のデータをダウンロードする時間がなかったため)
(a)LinearRegressionの使用(①で良かったため)



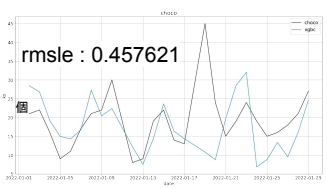
横軸は日付、縦軸は個数
黄色が実際の売上
青.赤.緑が線形モデルで予測
青は①と同じ特徴量
コロナによる影響を反映したいと思い、緑は前日のコロナ新規感染者数、赤は直近一週間の人口10万人あたりの感染者数の特徴量を足してみた。予想はしていたが良い特徴量ではなかった。

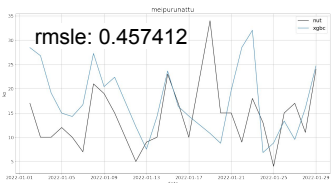
(b)XGboostの使用(不要な特徴量を追加しても影響が小さい上、精度が高いため)　トレーニングデータは過去一年とした

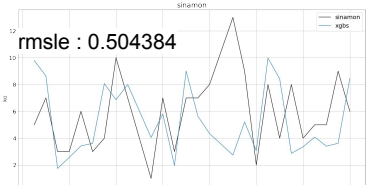


一番売上の多いプレーンを予測してみた。
黒が実際の売上個数
青は①の線形モデルでの予測
黄色と緑はxgboostを使った(二つの違いは学習データの期間の違い)
線形より正確に予測出来ていることがわかる

③ほかの商品の売上個数の予測







同じように、xgboostを使った
黒が実際の売上個数
青が予測した売上個数
scikit-learnのmetricsモジュール
mean_aquared_log_errorでRMSLEを求めた。1日、19日は例外として、全体的に惜しい。

考察　今後の展望

Pythonで売上をある程度予測できた。
しかし、これをもとにドーナツを作るとすると危険があるので、
より良い特徴量とハイパラメータを見つける必要がある。
また、天気などは今回は過去データを使ったが、実際は明日の天気予報を入力しなければならない。すべての特徴量を入力するのは大変なので、影響力の少ない特徴量を消す必要もある。一日後であったら、ARIMAモデルなども役に立ちそう。祝日や感染警戒レベル等も特徴量に加えてみたい。

参考文献
kaggleで勝つデータ分析の技術
Pythonで始めるkaggleスタートブック
SIGNATE　　Qiita

謝辞
売上データ提供していただいたnicoドーナツ山形店、助言をしてくださった先生方にに感謝の意を表します。

データ元
売上データ:nicoドーナツ山形店
気象データ:気象庁
コロナ関連:NHK