

# “気象データを活用した青果物育成予想モデル”

～青果物トレーの出荷予測～

気象データアナリスト

石原雄大

# アジェンダ

- ① 課題と分析ゴール
- ② 分析アプローチ
- ③ 分析/予測モデルの作成
- ④ 分析結果
- ⑤ 今後の展望





# ①課題と分析ゴール

# 1. テーマ選定の背景

古紙をリサイクルした紙製容器(パルプモールド)を国内外で製造・販売



## 需要予測がない

- 家電、食品、鶏卵、医療、物流、多岐に渡る業界で紙製容器(パルプモールド)は使用いただいている。
- 大概は発注計画表(フォアキャスト)が納品先より事前に支給され製造・納品を行う。  
ただし農作物分野では存在せず、一定の在庫を抱え、製造・納品を行う必要性がある。

# 1. 果物容器について

そもそも果物容器って何？

スーパー等で並んでいる緑色した紙の容器



果物容器 8種(🍎 🍎 🍐 🍏 🍉 🍊 🍏 🍊) →

今回は1つの果物でモデル作成  
以後、横展開する

ビジネス課題：

気象データから青果物の年間出荷数量予測を行い  
必要量を過不足なく製造する。

# 1. 果物容器の大玉・小玉

## 果物容器 過剰在庫問題について

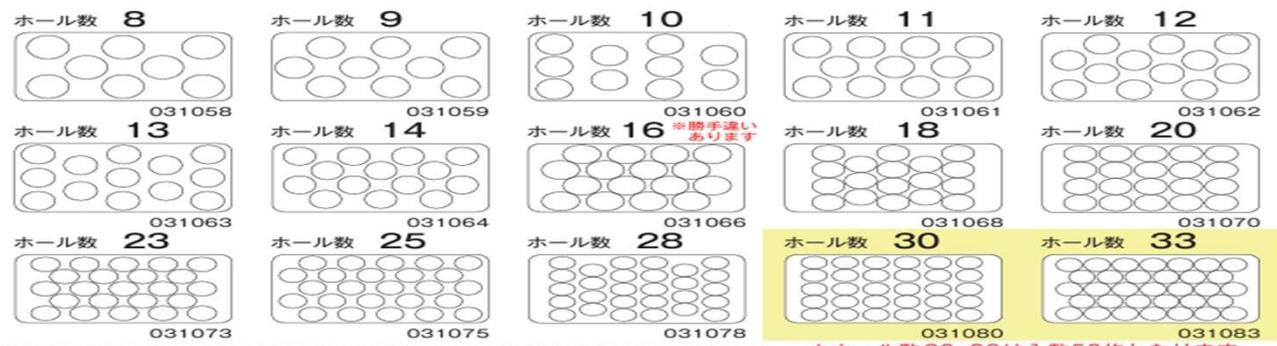
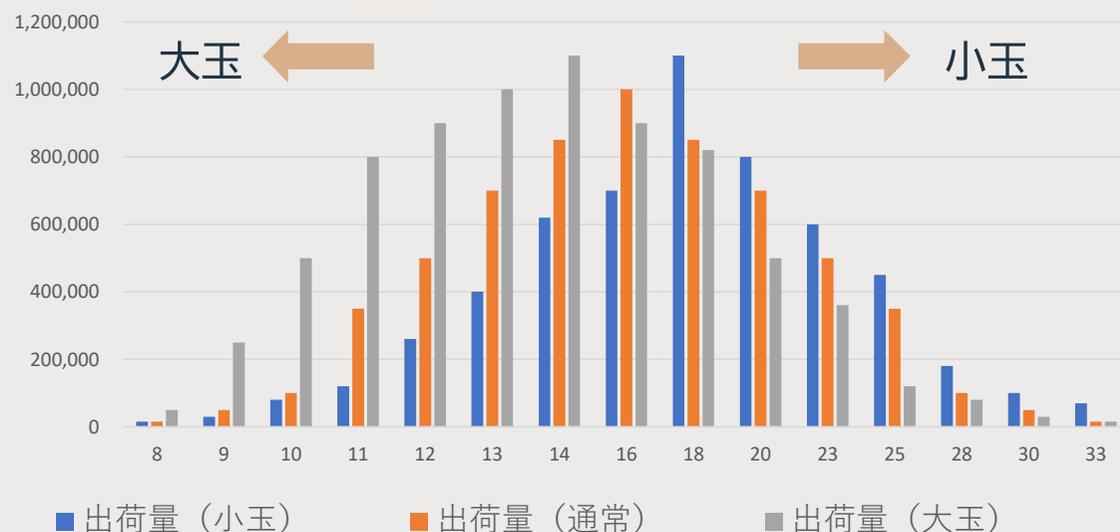
- ・気候の影響を強く受け出筋サイズのトレーが不明
- ・通常出荷想定 of 在庫を持つ
- ・大玉の年は小玉用が不動在庫(逆もしかり)

### 🍏 のホール ラインナップ

大玉の年(14Hを中心に11-13Hが出荷大)

小玉の年(18Hを中心に20-23Hの出荷大)

🍏 の出荷イメージ



ホール数 9・10・11・12・13・14・16・18・20・23・25・28  
ホールレイアウトは上記参照

# 1. 目標の設定

需要予想ターゲット製品：果物容器

目的KPI：過剰在庫の3割削減(原価削減、CF改善)

- ・ 在庫 果物容器の在庫回転期間が他の製品群の3-5倍と在庫水準が高い
- ・ 納期 シーズン性はあるが、受注から翌日~翌々日に納品を行う短納期製品
- ・ リスク 一度でも欠品があるとサプライヤーベンダーコードを失う(商圈消失)
- ・ 条件 在庫最適化プロセスに於いて、データ活用し、安全率を下げない

分析のゴール：気象データを活用し

※梨容器のホール毎の年間出荷予測

※梨は7-8月で出荷の8割が集中し、シーズン性が強い

# 1. あるべき姿

## 従来の製造方法

過去5年出荷実績から各ホールの平均値を算出

平均値を基に年間出荷量の7-8割を製造

シーズン途中で傾向を知り、必要量を追加生産

大玉であった場合、小玉の在庫は越年在庫に

出荷平均値からは安全在庫を減らせない

01

02

03

04

05

## 新しい製造方法

気象庁Webサイト 観測地点のデータをスクレイピング

データを基に時系列モデルにて青果物トレー出荷予測

年間出荷予測から出荷量を差し引いた数量 +  $\alpha$  製造

最低在庫で製造開始しており、越年在庫を最小化

複数年運用し、モデル予測と実績の乖離を最小化

## ② 分析アプローチ





## 2. 気象データ 観測地点

対象製品：梨

下記2つの納品場所で出荷量72%を占有  
A地区  
B地区



観測地点とした

観測地点一覧 気象庁Webサイトより



生データ Date: 日単位  
Pcs(枚): 目的変数y

Date	Pcs	Av.tem	Max.te	Min.tem	Precip.	Sunshi	Snowfe	Av.wind	Max.wi	Wind D	Av. vap	Av.hum	Min.se	Av.sea	Av.loca	Min. re	Av.tem	Max.te	Min.tem	Precip.	Sunshi	Snowfe	Av.wind	Max.wi	Wind D
2013/1/1	0	3.2	8.3	-1	0	7.2	0	1.5	4.8	西北西	5.6	73	1020.3	1021.5	1020.7	48	0.5	3.3	-0.7	27	0.2	31	1.4	4	南南西
2013/1/2	0	7.1	14.6	1.9	0	8.3	0	3.1	9.6	西北西	6.3	64	1020.6	1023.1	1022.3	31	2.5	5.8	-0.1	32.5	0	4	1.7	4.1	南
2013/1/3	0	2.9	4.6	1	0	5	0	5.8	9.2	西北西	3.8	51	1024.4	1026	1025.2	36	-0.7	0.5	-1.8	32.5	0.2	25	0.6	2.4	南東
2013/1/4	0	3	6.4	0.5	0	7.2	0	4.8	8.3	西北西	3.9	52	1024.6	1025.8	1025	34	-1.9	0.7	-3.4	7	0.9	11	0.7	3.3	北東
2013/1/5	0	3.8	7.5	-0.2	0	7.6	0	2.6	5.2	北西	4.8	60	1019.8	1022.3	1021.5	42	-1.7	3.1	-5.9	0	1.5	2	0.8	2.5	南南西
2013/1/6	0	5.6	9.3	2.4	0	6.5	0	4.1	7.6	北北西	4.8	54	1018.9	1020.6	1019.8	34	0.6	2.3	-0.6	3	0	3	0.6	2	南南西
2013/1/7	0	6	10.4	2.3	0	9	0	3.2	6.2	西北西	5.4	59	1018.3	1020.1	1019.3	41	1.9	7	-0.9	0.5	2.8	1	0.9	2.4	南南西
2013/1/8	0	6.6	11	3.4	0	5.7	0	3.5	6	北北西	5.5	57	1017.2	1018.9	1018.1	36	2.1	5.5	0.6	3	0.2	1	0.8	1.7	南南西
2013/1/9	0	6.2	10	3.1	0	6.9	0	4	7.6	西北西	4.8	52	1015	1017.5	1016.7	34	2	5	-0.2	6.5	0.5	1	0.8	2	南南西
2013/1/10	0	4.9	8.2	2.9	0	9.1	0	5.7	9.1	北西	3.9	45	1018.7	1021.9	1021.1	32	-0.5	1.8	-2.4	9	2.8	17	0.5	1.9	東北東
2013/1/11	0	4.2	8.5	0.8	0	8.4	0	3.5	6.3	西北西	4.6	56	1026	1027.7	1026.9	36	-0.4	3.2	-2.5	4.5	3.2	5	0.6	1.5	南
2013/1/12	0	4.9	11.5	0.2	0	6.4	0	1.6	6.1	北北西	5.9	69	1023.1	1025.2	1024.4	48	1.5	6.4	-2.7	1	2.2	0	1.4	3.2	南南西

## 2. 出荷データ 対象ホール

Pear Tray (K)	
2004-2022出荷実績	
5H	9,514
6H	27,322
7H	34,600
8H	103,000
9H	47,000
10H	344,000
11H	25,000
12H	751,000
13H	66,000
14H	841,000
15H	28,427
16H	803,000
18H	508,000
20H	233,000
22H	61,000
23H	1,200
25H	19,000
28H	300

梨容器  
ホールが多い為  
黄マーカ付けした  
8種を対象

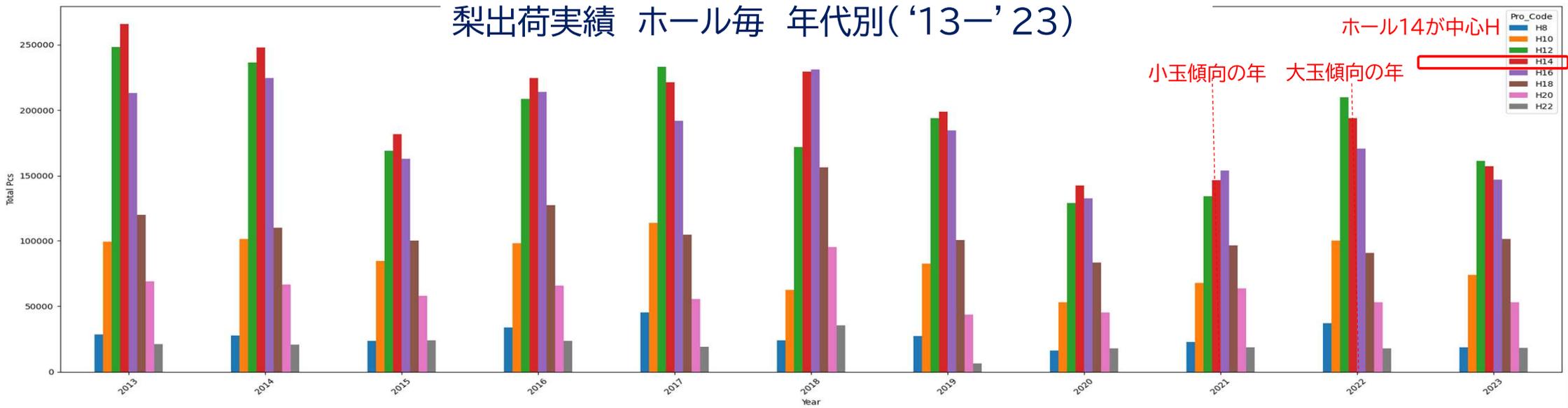
### 梨の特徴

- ・梨の花は4月上旬から中旬に咲く
- ・梨の出荷は7月頃から8月下旬でほぼ終了する
- ・1-6月の気象データを使用し、年間出荷量を予測

→モデルはホール毎に8個作成し予測する



梨出荷実績 ホール毎 年代別('13-'23)

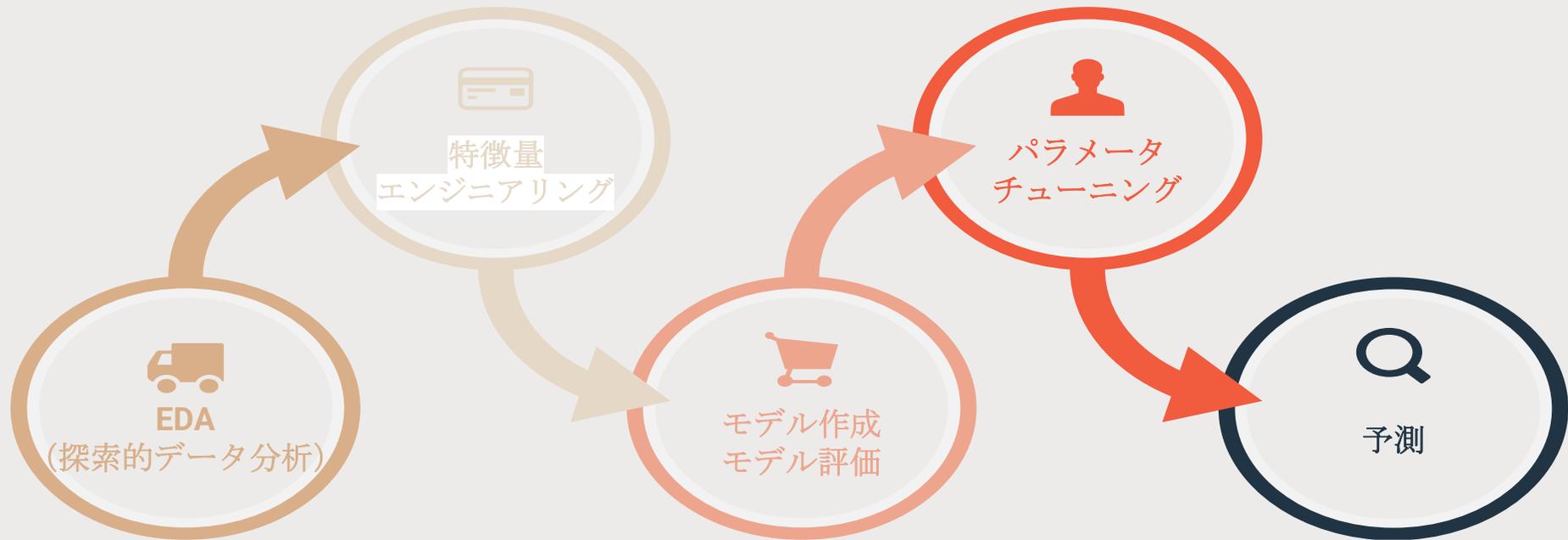


ホール14が中心H  
小玉傾向の年 大玉傾向の年

## 2. 分析フロー

- データを基に予測精度向上に必要なカラムを検討
- 青果物業界に於けるドメイン知識の活用
- 多重共線性への注意

- EDA結果を基に特徴量の活用
- 過学習に注意
- 複数の機械学習モデルし精度評価  
(時間がない時は発表後に行う)



- 目的変数 = 出荷枚数(Pcs)をyとする
- 連続値×連続値なので散布図・相関図等参照
- どの特徴量がモデルに使えるか目途をつける

- 時系列データで、シーズン性が強い
- Prophetでまずモデル作成
- 評価はR2, MAPE, RMSE, MAE  
MAPE=0.2以下を目標

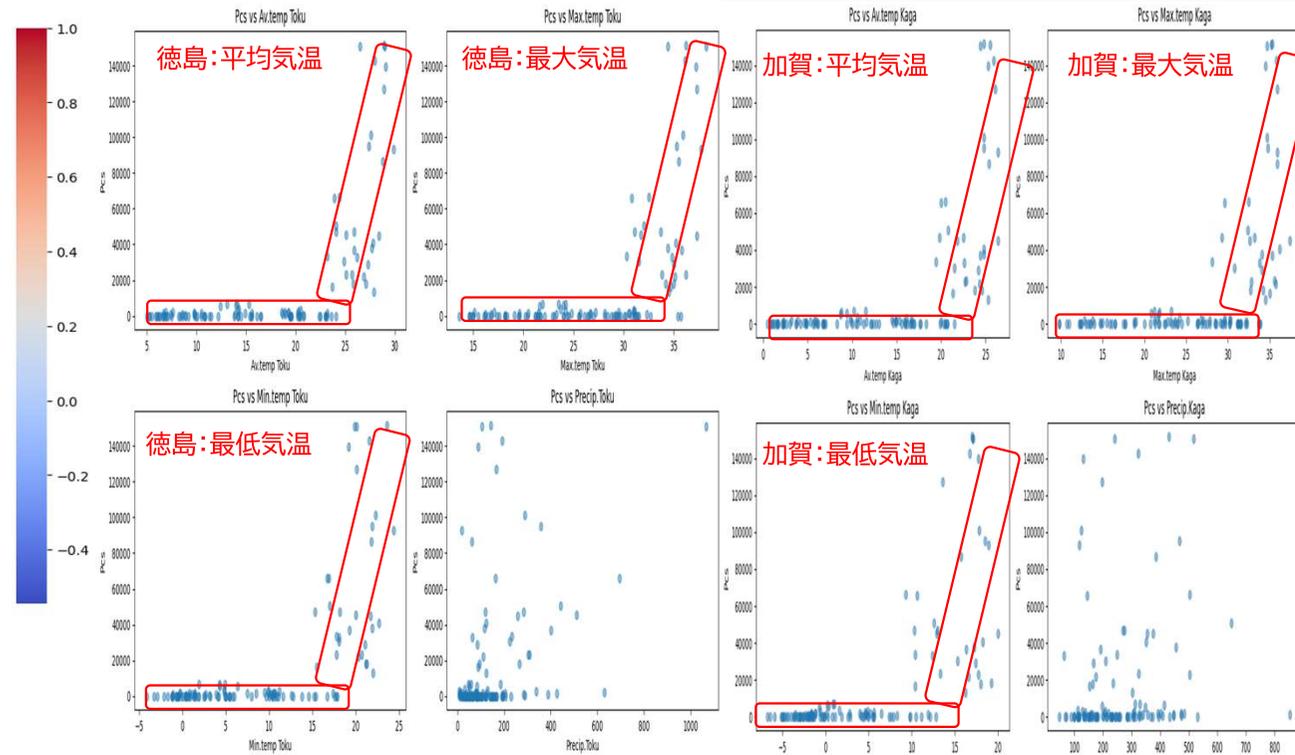
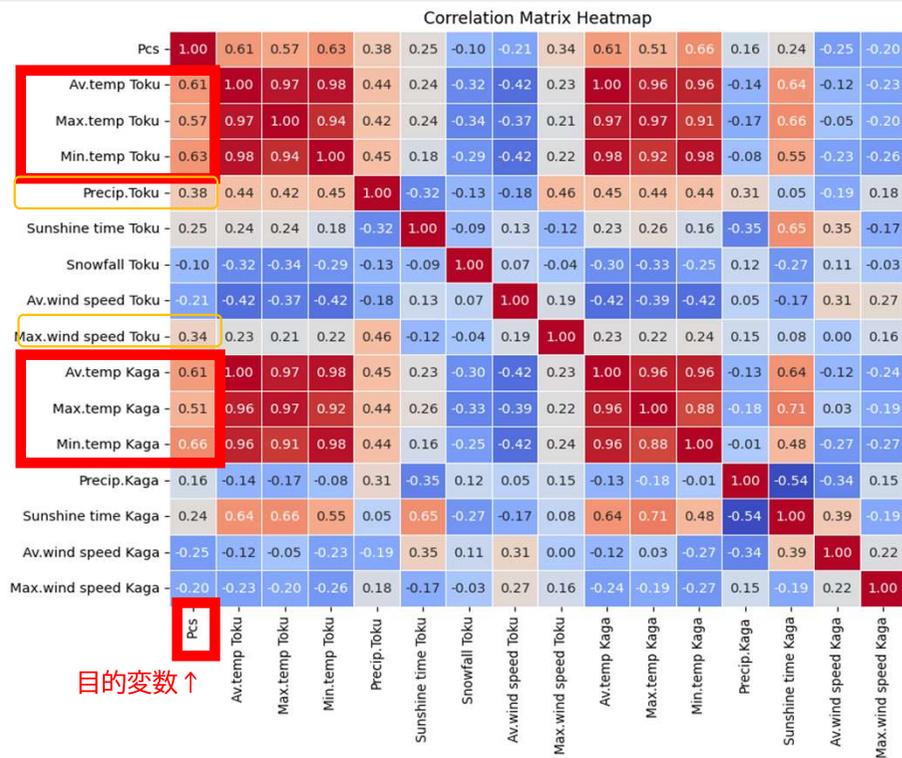
- 2024年1-6月の気象データを活用  
今年の梨の出荷枚数を予測
- 実数と比較し課題の抽出

## 2. EDA(検索的データ分析)

### EDA結果より仮説

・出荷数量(Pcs)を目的変数として、EDAを行った結果、相関係数が徳島市と加賀市とも平均気温(Av.temp)、最高気温(Max.temp)、最低気温(Min.temp)は50-60%と高い相関関係を示した。散布図より気温が上がる程、目的変数も増えている事が確認できた。

・降水量(Precip.Toku)最大風速(Max.wind speed Toku)も比較的相関が高く、特徴量の調整の際に試す価値がありそう。





### ③分析/予測モデルの作成

# 3. 外因なし 単純なProphetモデル vol.1

```
import pandas as pd
from prophet import Prophet
from sklearn.metrics import r2_score, mean_absolute_percentage_error, mean_absolute_error, mean_squared_error
import numpy as np

# 'Date' 列にNaNがある場合、除去する
PH14_train = PH14_train.dropna(subset=['Date'])

# 年月の文字列から日付を生成 ("13-Jan" の形式を処理)
# 文字列として処理できるように 'Date' を文字列に変換
PH14_train['Date'] = PH14_train['Date'].astype(str)

# 年を "20XX" 形式にし、月を適切な数値に変換
PH14_train['Year'] = PH14_train['Date'].apply(lambda x: '20' + x[:2]) # '13-Jan' -> '2013'
PH14_train['Month'] = PH14_train['Date'].apply(lambda x: x[-3:]) # '13-Jan' -> 'Jan'

# 月名を数値に変換
month_map = {
    'Jan': 1, 'Feb': 2, 'Mar': 3, 'Apr': 4, 'May': 5, 'Jun': 6,
    'Jul': 7, 'Aug': 8, 'Sep': 9, 'Oct': 10, 'Nov': 11, 'Dec': 12
}
PH14_train['Month'] = PH14_train['Month'].map(month_map)

# 'Year' と 'Month' を使って新しい 'ds' カラム (日付形式) を作成
PH14_train['ds'] = pd.to_datetime(PH14_train[['Year', 'Month']].assign(Day=1))

# 'Pcs' を 'y' として Prophet 用に準備
PH14_train['y'] = PH14_train['Pcs']

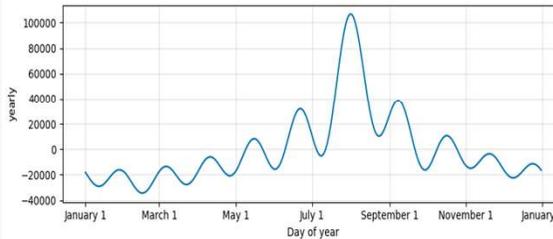
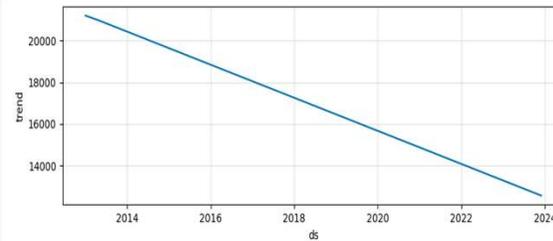
# Prophet のモデル作成とフィット
model = Prophet()
model.fit(PH14_train[['ds', 'y']])

# 予測データの作成
future = model.make_future_dataframe(periods=12, freq='M') # 12ヶ月先まで予測
forecast = model.predict(future)

# 予測結果を評価するために実際の値と予測値を比較
actual = PH14_train['y'] # 実際の値
predicted = forecast['yhat'][:len(actual)] # 予測値を実際のデータの長さに合わせる

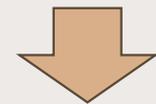
# R^2, MAPE, MAE, RMSE の計算
r2 = r2_score(actual, predicted)
mape = mean_absolute_percentage_error(actual, predicted)
mae = mean_absolute_error(actual, predicted)
rmse = np.sqrt(mean_squared_error(actual, predicted))

# 結果の表示
print(f"R^2: {r2}")
print(f"MAPE: {mape}")
print(f"MAE: {mae}")
print(f"RMSE: {rmse}")
```

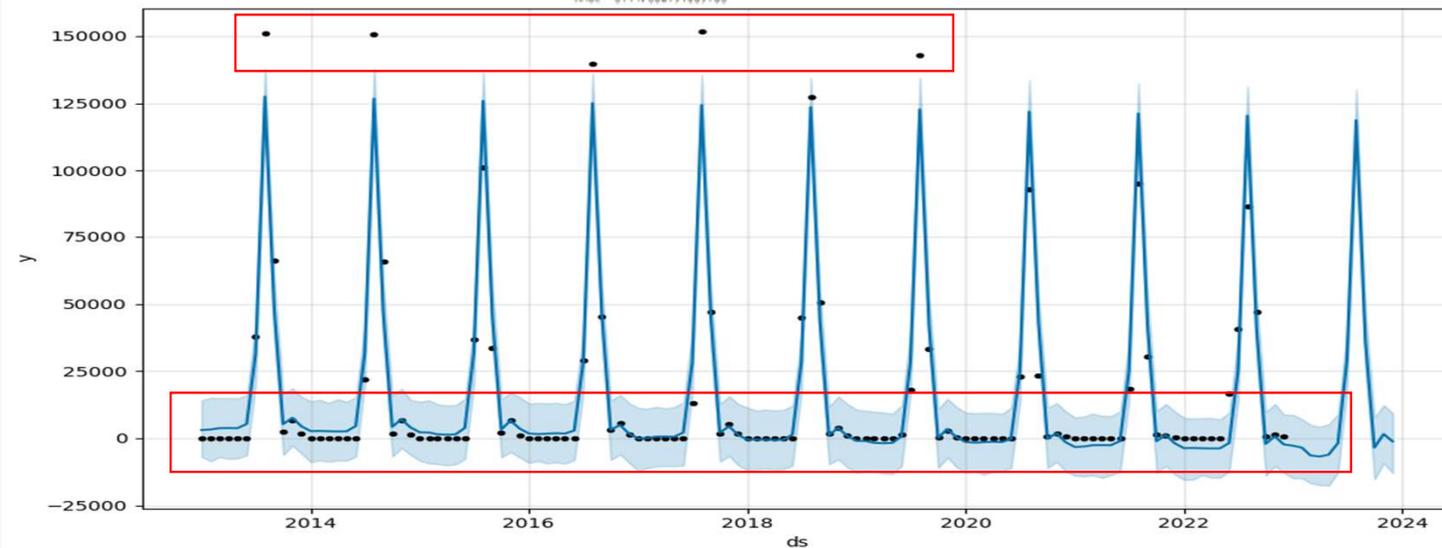


```
INFO:prophet:Disabling weekly seasonality. Run prophet wi
INFO:prophet:Disabling daily seasonality. Run prophet wi
DEBUG:cmdstancy:input tempfile: /tmp/tmp8kexfai/eq_cz87n
DEBUG:cmdstancy:input tempfile: /tmp/tmp8kexfai/c91f9oq
DEBUG:cmdstancy:idx 0
DEBUG:cmdstancy:running CmdStan, num threads: None
DEBUG:cmdstancy:CmdStan args: ['/usr/local/lib/python3.10
03:31:08 - cmdstancy - INFO - Chain [1] start processing
INFO:cmdstancy:Chain [1] start processing
03:31:08 - cmdstancy - INFO - Chain [1] done processing
INFO:cmdstancy:Chain [1] done processing
R^2: 0.9416565689582572
MAPE: 4.5507523113596006e+18
MAE: 4975.126851854652
RMSE: 8714.852191039738
```

- ピーク時の予測は少し外れてるが 予想自体はまずまず
- MAPEが超異常な数字を表示 0枚の所での誤差を拾っている



分析の見直し必要



# 3. 分析見直し



データの粒度を変更

デイリー (日) → マンスリー (月)



出荷枚数0の予測をやめる

→ 出荷がある数字のみ予測



特徴量の調整

→ 外因として平均・Max・Min気温



対数を取る (log)

→ 8月に年間の60%以上の出荷が集中  
→ 対数を取り、8月のインパクトを抑制  
→ 制度誤差を高める

Date	Pcs	Av.tem	Max.te	Min.tem	Precip.	Sunshi	Snowfa	Av.wind	Max.wi	Wind D	Av. vap	Av.hum	Min.se	Av.sea	Av.loca	Min. re	Av.tem	Max.te	Min.tem	Precip.	Sunshi	Snowfa	Av.wind	Max.wi	Wind D
2013/1/1	0	3.2	8.3	-1	0	7.2	0	1.5	4.8	西北西	5.6	73	1020.3	1021.5	1020.7	48	0.5	3.3	-0.7	27	0.2	31	1.4	4	南南西
2013/1/2	0	7.1	14.6	1.9	0	8.3								3.1	1022.3	31	2.5	5.8	-0.1	32.5	0	4	1.7	4.1	南
2013/1/3	0	2.9	4.6	1	0	5								26	1025.2	36	-0.7	0.5	-1.8	32.5	0.2	25	0.6	2.4	南東
2013/1/4	0	3	6.4	0.5	0	7.2								5.8	1025	34	-1.9	0.7	-3.4	7	0.9	11	0.7	3.3	北東
2013/1/5	0	3.8	7.5	-0.2	0	7.6	0	2.6	5.2	北西	4.8	60	1019.8	1022.3	1020.5	42	-1.7	3.1	-5.9	0	1.5	2	0.8	2.5	南南西

デイリーベース (日)

Date	Pcs	Av.tem T	Max.tem	Min.tem	Precip.To	Sunshine	Max. Snov	Snowfall	Av.wind sp	Max.wind	Max. insta	Wind Dire	Av. vapor	Av.humi T	Min.sea le	Av.sea lev	Av.local	Min. relati	Av.tem K	Max.tem	Min.tem	Precip.Ka	Sunshine	Max. Snov	Snowfall K	Av.wind sp	Max.wind	Wind Dire
14-Jul	21800	26.9	35.1	20.3	111.5	196.4	0	0	16.4	16.4	16.4	西	26.5	75	203.7	1008.7	38	38	24.3	35.7	16.6	175	161.2	0	0	1.2	5.8	西南西
14-Aug	150800	26.5	34.4	20.1	1065.5	116.3	0	0	16.4	16.4	16.4	西	26.5	75	203.7	1008.7	38	38	24.4	34.7	17.1	516.5	91.9	0	0	1.1	5.5	西南西
14-Sep	65800	23.9	30.8	16.9	163	182.4	0	0	16.4	16.4	16.4	西	26.5	75	203.7	1008.7	38	38	20	29.6	10.6	144.5	149.5	0	0	1.2	6.1	西
14-Oct	1600	19.3	28.1	10.6	427	169	0	0	16.4	16.4	16.4	西	26.5	75	203.7	1008.7	38	38	14.9	27.8	4	306	143	0	0	1.4	11.8	西南西
14-Nov	6800	14.1	21.9	4.9	48	144.5	0	0	2.7	9.4	15.4	西北西	11.1	68	1005.3	1019.5	1018.7	32	10.2	21.2	0.9	329.5	98.8	0	0	1.1	5.5	西南西
14-Dec	1250	6.7	19.6	-0.3	124.5	155.2	1	1	3.5	10.5	19	西北西	6.2	62	1000.4	1019.6	1018.8	19	2.3	18.8	-3.5	853	27.9	42	160	1.3	9.3	西南西

マンスリーベース (月)

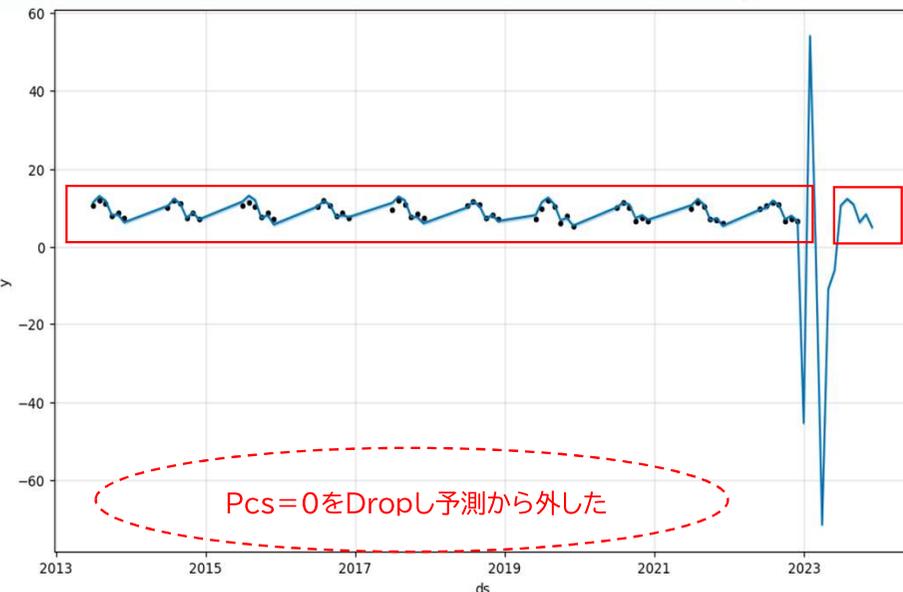
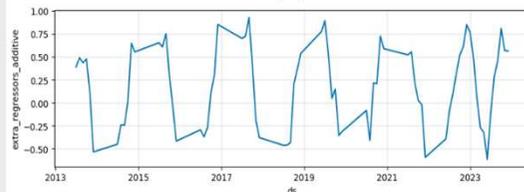
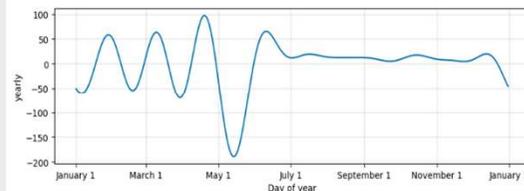
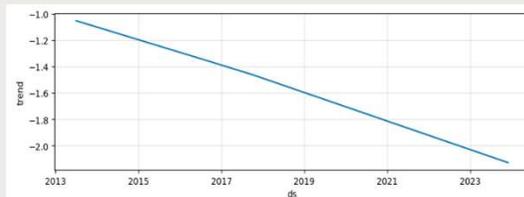
# 3. 外因あり Prophetモデル vol.2

・7-12月の出荷時期のみを予測する様になり  
7-8月のピーク出荷時も視覚的に良く予測

・MAPEが目標値0.2以下となり、理論上  
誤差が17%に

・実際2023年出荷量  
2023年のActual合計: 157,150  
2023年のForecast合計: 311,287

→特徴量のチューニングが必要と思われる

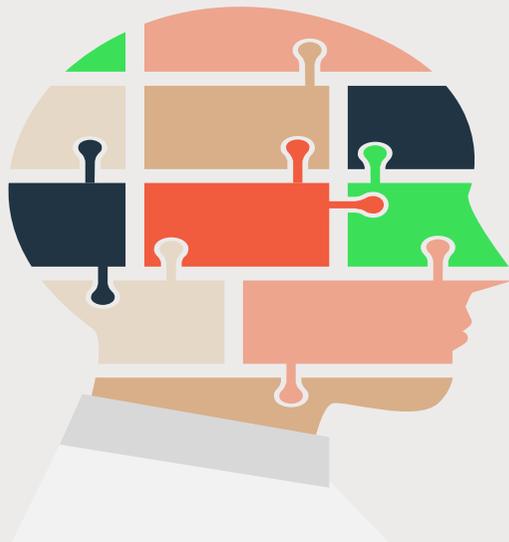
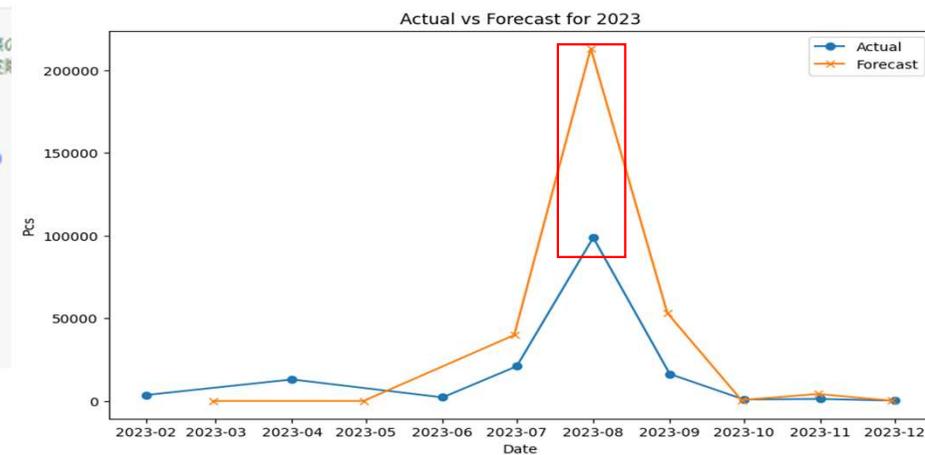


```
# 予測結果を評価するために実際の値と予測値を比較
actual = test.loc[test['ds'] >='2023-07-01','y'] # 実際の値
predicted = forecast['yhat'][:len(actual)] # 予測値をactualに合わせる

# MAPE, RMSE, MAEの計算
r2 = r2_score(actual, predicted)
mape = mean_absolute_percentage_error(actual, predicted)
rmse = np.sqrt(mean_squared_error(actual, predicted))
mae = mean_absolute_error(actual, predicted)

# 結果の表示
print(f'R²: {r2}')
print(f'MAPE: {mape}')
print(f'RMSE: {rmse}')
print(f'MAE: {mae}')

R²: 0.5388913469001129
MAPE: 0.17282834440880077
RMSE: 1.4562910913317237
MAE: 1.4166670984186496
```



# 3. 特徴量の再調整

```
# その他の列を外因として追加
model3.add_regressor('Av. temp Toku')
model3.add_regressor('Max. temp Toku')
model3.add_regressor('Min. temp Toku')
model3.add_regressor('Precip. Toku')
model3.add_regressor('Av. temp Kaga')
model3.add_regressor('Max. temp Kaga')
model3.add_regressor('Min. temp Kaga')
model3.add_regressor('Precip. Kaga')
```



```
# その他の列を外因として追加
model3.add_regressor('Av. temp Toku')
model3.add_regressor('Max. temp Toku')
model3.add_regressor('Min. temp Toku')
model3.add_regressor('Precip. Toku')
model3.add_regressor('Max. wind speed Toku')
model3.add_regressor('Av. temp Kaga')
model3.add_regressor('Max. temp Kaga')
model3.add_regressor('Min. temp Kaga')
model3.add_regressor('Precip. Kaga')
```



```
# その他の列を外因として追加
model3.add_regressor('Av. temp Toku')
model3.add_regressor('Max. temp Toku')
model3.add_regressor('Min. temp Toku')
model3.add_regressor('Precip. Toku')
model3.add_regressor('Av. temp Kaga')
model3.add_regressor('Max. temp Kaga')
model3.add_regressor('Min. temp Kaga')
```

```
# Prophetのモデル作成とフィット
# 外因変数に敏感度を調整0.07がベスト
model3 = Prophet(changepoint_prior_scale=0.07)
```



- MAPEを評価基準  
特徴量は  
- 降水量(徳島)  
- 降水量(加賀)  
のみ加えた場合  
MAPEが最小化となった

- 外因変数の感度を調整  
結果的に0.07が  
一番良いスコアをマーク

- 「最高気温と最低気温の差」  
(Temp.diff) 「1月から6月の累積降  
水量」(Precip.Toku)  
と少し複雑化した特徴量も試したが、  
良い結果が得られなかった。

# 3. 外因あり Prophetモデル vol.3



- 特徴量のチューニングし再度モデルを評価
- 引き続き7-12月の出荷時期のみを予測  
Pcs0の月は事前にDrop
- MAPEがさらに低下し理論上誤差が14%に！
- 実際2023年出荷量  
2023年のActual合計: 157,150  
2023年のForecast合計: 221,626  
とさらに予測値の精度がUp
- 2月の大きな▲が気になる(DROPして計算)

```
# 結果の表示
print(f"R2: {r2}")
print(f"MAPE: {mape}")
print(f"RMSE: {rmse}")
print(f"MAE: {mae}")
```

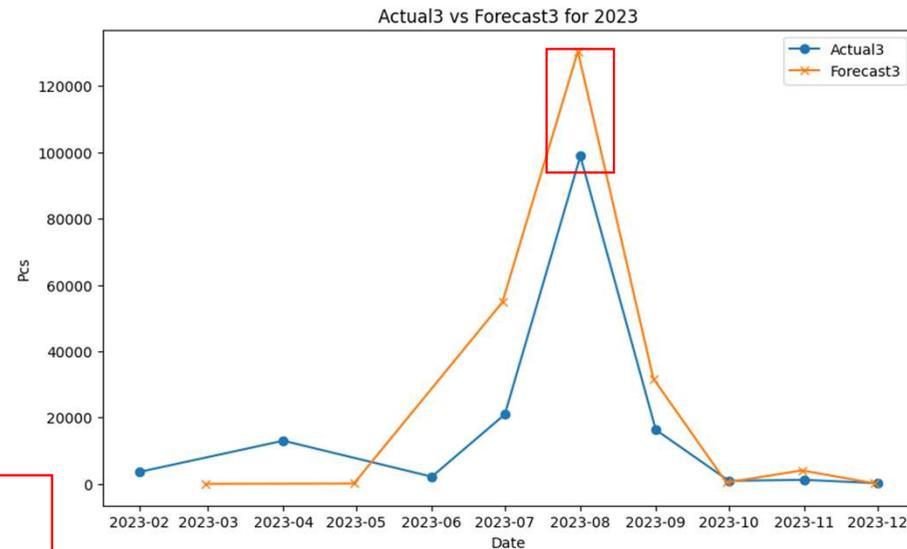
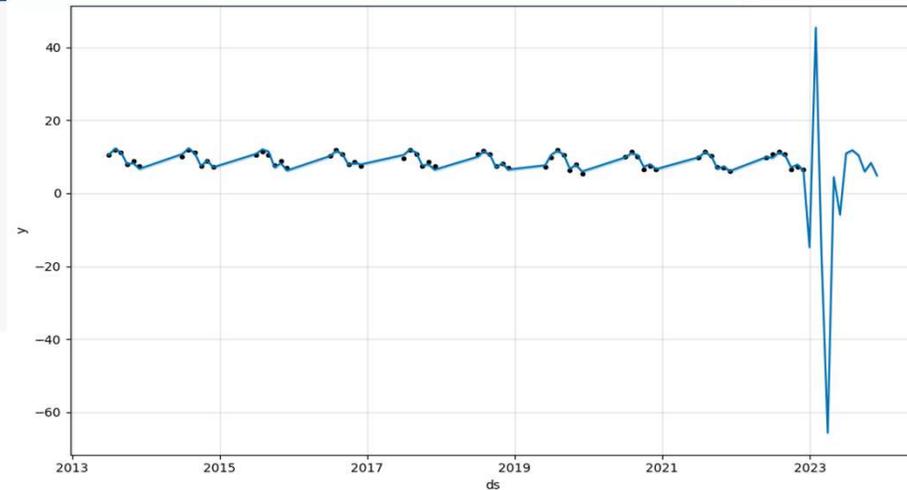
R<sup>2</sup>: 0.734711196120801  
 MAPE: 0.1413417711283651  
 RMSE: 1.1046024890053612  
 MAE: 1.045960623040685

Filtered forecast3 data:

	ds	yhat
64	2023-02-28	-16.365751
66	2023-04-30	4.396490
68	2023-06-30	10.911507
69	2023-07-31	11.778764
70	2023-08-31	10.364866
71	2023-09-30	5.950231
72	2023-10-31	8.306638
73	2023-11-30	4.860491

2023年のActual合計: 157149.99999999994

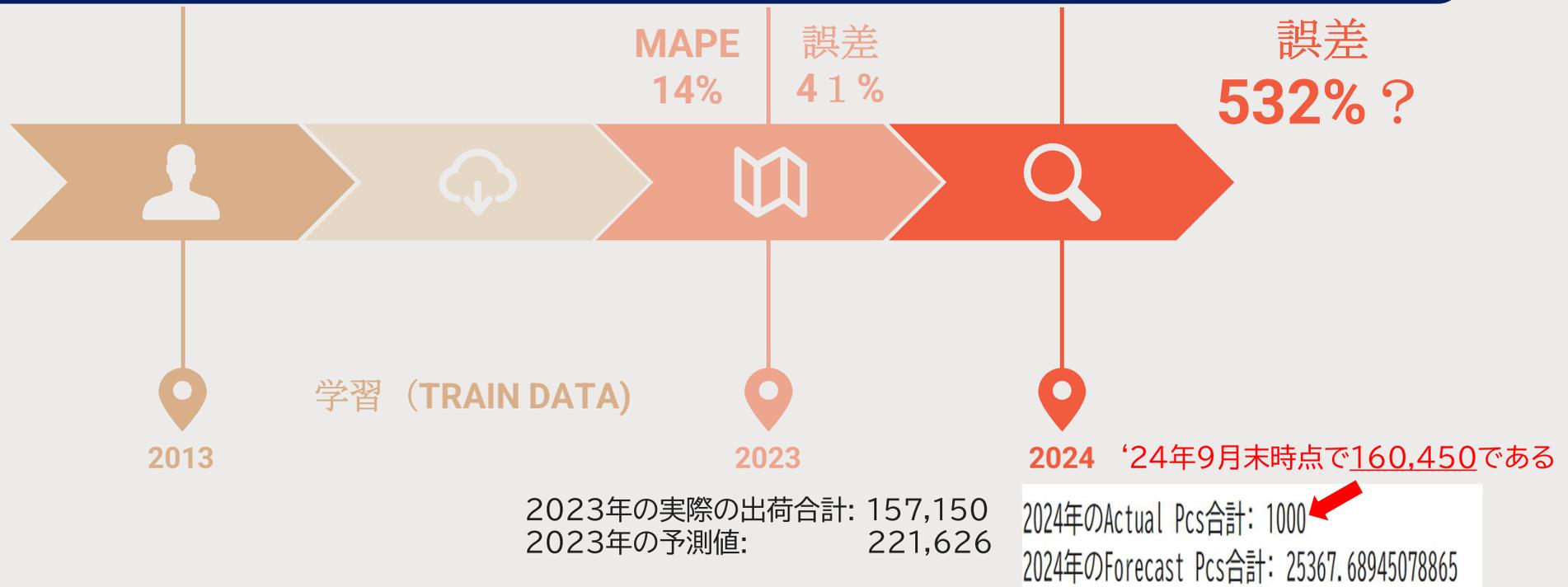
2023年のForecast合計: 221626.74890411523



A close-up photograph of several stacked food containers. The containers are made of white and brown paper or cardboard. They are arranged on a light-colored wooden tray. The lighting is soft and natural, highlighting the textures of the paper and wood. The text '④分析結果' is overlaid on the left side of the image.

## ④分析結果

## 4. 実際に2024年度を予測



### ドメイン知識の観点から仮説

MAPEを高める為でも、学習の段階では出荷(目的変数)がほぼ0の1-6月の気象データを除外している  
→学習の段階では全期間の気象データで学習し、精度評価の段階で除外し評価が望ましい???

MAPEを高める為でも、8月の出荷が6割を占めるインパクトを抑える為、対数を取った  
→実数に戻す段階で実態と乖離が大きい気がする。特に8月の対数が過小評価され、実数が少なくカウントされる?  
→モデルの評価は対数を取るべき?実際に運用する段階では対数取らず実数でモデリング、'24年を予想が望ましい???

## 4. 対数を取らない方法で3つの特徴量 実数比較

8H '24販売実数12,750枚

温度

温度と降水量

温度と降水量と敏感度

2023年のActual合計: 18900  
2023年のForecast合計: 23910.867983895903

2023年のActual合計: 18900  
2023年のForecast合計: 24025.27425715923

2023年のActual合計: 18900  
2023年のForecast合計: 22342.076880475783

2024年のActual Pcs合計: 0.0  
2024年のForecast Pcs合計: 26470.8520450206

2024年のActual Pcs合計: 0.0  
2024年のForecast Pcs合計: 26583.43767400309

2024年のActual Pcs合計: 0.0  
2024年のForecast Pcs合計: 28255.417843710402

12H '24販売実数が139,550枚

温度

温度と降水量

温度と降水量と敏感度

2023年のActual合計: 161300  
2023年のForecast合計: 171154.8409266267

2023年のActual合計: 161300  
2023年のForecast合計: 153512.2331651281

2023年のActual合計: 161300  
2023年のForecast合計: 154319.09572683915

2024年のActual Pcs合計: 1000.0  
2024年のForecast Pcs合計: 166036.78507932508

2024年のActual Pcs合計: 1000.0  
2024年のForecast Pcs合計: 140666.53277007007

2024年のActual Pcs合計: 1000.0  
2024年のForecast Pcs合計: 112397.22308613424

16H '24販売実数が153,750枚

温度

温度と降水量

温度と降水量と敏感度

2023年のActual合計: 146900  
2023年のForecast合計: 182178.9685212109

2023年のActual合計: 146900  
2023年のForecast合計: 178265.54618582988

2023年のActual合計: 146900  
2023年のForecast合計: 165880.0692946016

2024年のActual Pcs合計: 700.0  
2024年のForecast Pcs合計: 180136.12209346742

2024年のActual Pcs合計: 700.0  
2024年のForecast Pcs合計: 180276.04400097489

2024年のActual Pcs合計: 700.0  
2024年のForecast Pcs合計: 175957.35360866765

20H '24販売実数が57,050枚

温度

温度と降水量

温度と降水量と敏感度

2023年のActual合計: 53250  
2023年のForecast合計: 64195.23268922302

2023年のActual合計: 53250  
2023年のForecast合計: 59416.73680478897

2023年のActual合計: 53250  
2023年のForecast合計: 60168.38818706679

2024年のActual Pcs合計: 0.0  
2024年のForecast Pcs合計: 47183.05515121495

2024年のActual Pcs合計: 0.0  
2024年のForecast Pcs合計: 43001.88880078279

2024年のActual Pcs合計: 0.0  
2024年のForecast Pcs合計: 41380.6485235983

10H 52100が'24販売実数52,100枚

温度

温度と降水量

温度と降水量と敏感度

2023年のActual合計: 74050  
2023年のForecast合計: 77334.94322269563

2023年のActual合計: 74050  
2023年のForecast合計: 72469.8533662896

2023年のActual合計: 74050  
2023年のForecast合計: 71501.71093496557

2024年のActual Pcs合計: 450.0  
2024年のForecast Pcs合計: 52728.97788790877

2024年のActual Pcs合計: 450.0  
2024年のForecast Pcs合計: 51230.37320647025

2024年のActual Pcs合計: 450.0  
2024年のForecast Pcs合計: 51135.40166934924

14H 160450が'24販売実数160,450枚

温度

温度と降水量

温度と降水量と敏感度

2023年のActual合計: 157150  
2023年のForecast合計: 188864.155092864

2023年のActual合計: 157150  
2023年のForecast合計: 188955.32041085407

2023年のActual合計: 157150  
2023年のForecast合計: 174256.86731002765

2024年のActual Pcs合計: 1000  
2024年のForecast Pcs合計: 170840.72892288698

2024年のActual Pcs合計: 1000  
2024年のForecast Pcs合計: 150665.11028347036

2024年のActual Pcs合計: 1000  
2024年のForecast Pcs合計: 141562.7017277137

18H '24販売実数が87,950枚

温度

温度と降水量

温度と降水量と敏感度

2023年のActual合計: 101450  
2023年のForecast合計: 95294.50741174769

2023年のActual合計: 101450  
2023年のForecast合計: 95104.752160617

2023年のActual合計: 101450  
2023年のForecast合計: 92996.5242775823

2024年のActual Pcs合計: 850.0  
2024年のForecast Pcs合計: 112840.48710809338

2024年のActual Pcs合計: 850.0  
2024年のForecast Pcs合計: 91469.08884424233

2024年のActual Pcs合計: 850.0  
2024年のForecast Pcs合計: 92038.08794195956

22H '24販売実数が18,550枚

温度

温度と降水量

温度と降水量と敏感度

2023年のActual合計: 18150  
2023年のForecast合計: 20283.192327950663

2023年のActual合計: 18150  
2023年のForecast合計: 17865.899341524826

2023年のActual合計: 18150  
2023年のForecast合計: 17998.77661932129

2024年のActual Pcs合計: 0.0  
2024年のForecast Pcs合計: 9396.894913149798

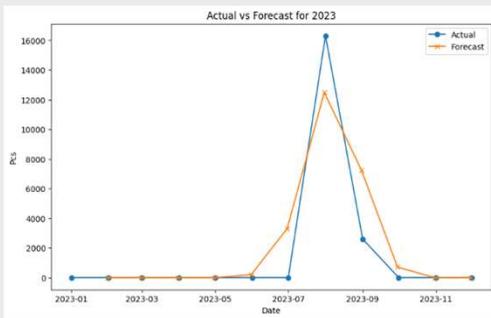
2024年のActual Pcs合計: 0.0  
2024年のForecast Pcs合計: 5060.376927329224

2024年のActual Pcs合計: 0.0  
2024年のForecast Pcs合計: 4852.705331126472

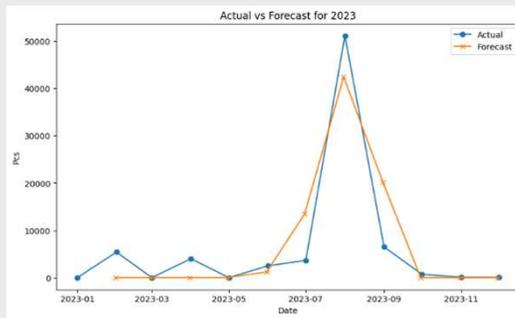
# 4. '23年実績対予測

気温 + 降水量 + 敏感度のモデル モデルは良いのだけど、'24年の予測は実際精度が落ちる

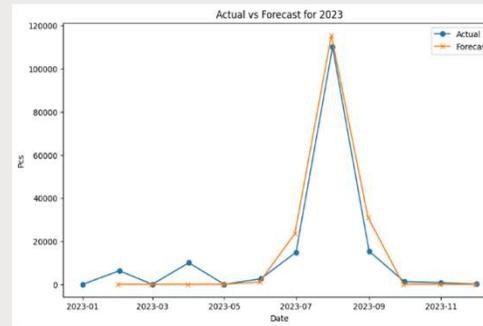
### 8H '23年実績:予測



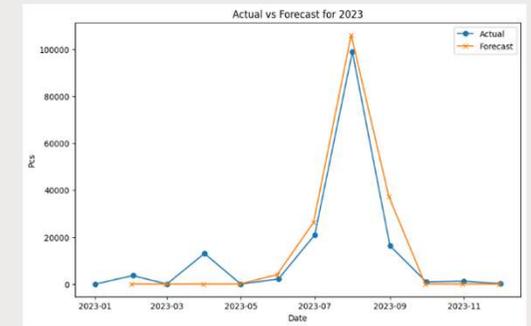
### 10H '23年実績:予測



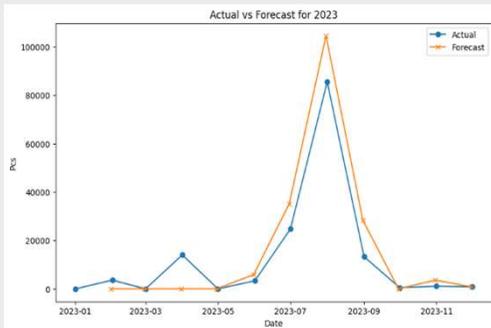
### 12H '23年実績:予測



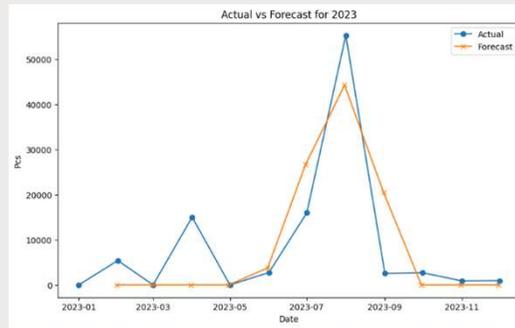
### 14H '23年実績:予測



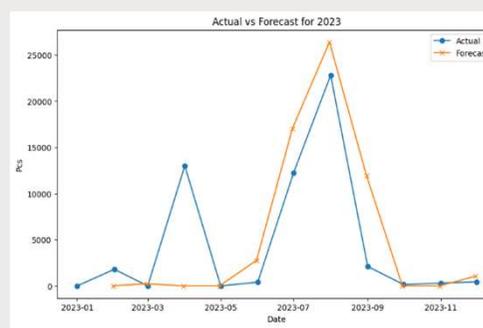
### 16H '23年実績:予測



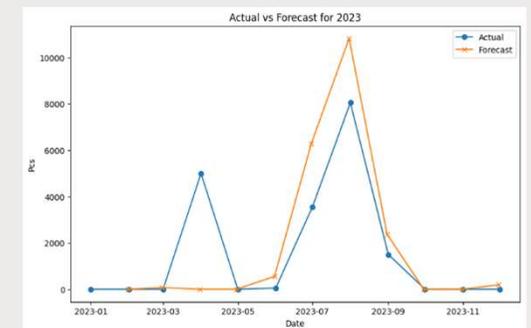
### 18H '23年実績:予測



### 20H '23年実績:予測



### 22H '23年実績:予測



# 4. 敏感度を特徴量除外 ‘24年度予測

降水量と0.7感度調整を加えないモデルの方が、実際の‘24年度の予測精度が高い

14H温度と降水量

14H温度と降水量と敏感度

2023予測モデル

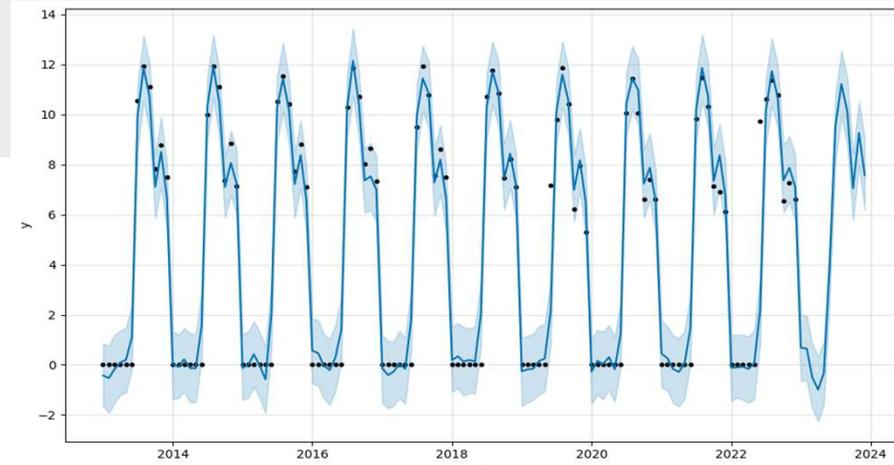
2023予測モデル

Filtered forecast data:

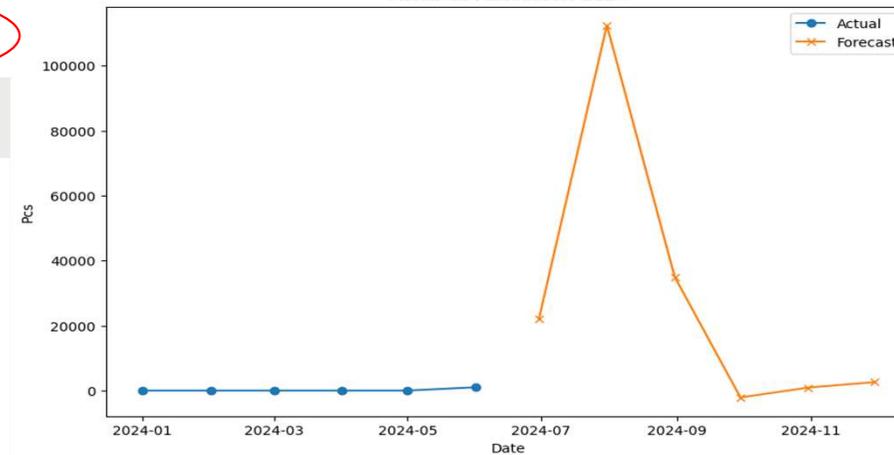
	ds	yhat
121	2023-01-31	-4057.949627
122	2023-02-28	-2058.027061
123	2023-03-31	-7191.413910
124	2023-04-30	-5855.225131
125	2023-05-31	5023.101488
126	2023-06-30	29096.286001
127	2023-07-31	115468.946395
128	2023-08-31	36067.214316
129	2023-09-30	-6288.768199
130	2023-10-31	3299.772211
131	2023-11-30	-2570.973212
2023年のActual合計:		157150
2023年のForecast合計:		188955.32041085407

Filtered forecast data:

	ds	yhat
121	2023-01-31	-3782.610709
122	2023-02-28	-4531.702049
123	2023-03-31	-9865.193896
124	2023-04-30	-8578.642856
125	2023-05-31	3558.844792
126	2023-06-30	26395.704929
127	2023-07-31	107469.966420
128	2023-08-31	36832.351168
129	2023-09-30	-6764.837438
130	2023-10-31	-3784.925320
131	2023-11-30	5443.036790
2023年のActual合計:		157150
2023年のForecast合計:		174256.86731002765



Actual vs Forecast for 2024



2024予測モデル

2024予測モデル

2024年のActual Pcs合計: 1000  
2024年のForecast Pcs合計: 150665.11028347036

2024年のActual Pcs合計: 1000  
2024年のForecast Pcs合計: 141562.7017277137

	ds	yhat
1	2024-07-31	102488.274224
2	2024-08-31	28072.345605
3	2024-09-30	-4192.993309
4	2024-10-31	2249.630730
5	2024-11-30	6563.063528

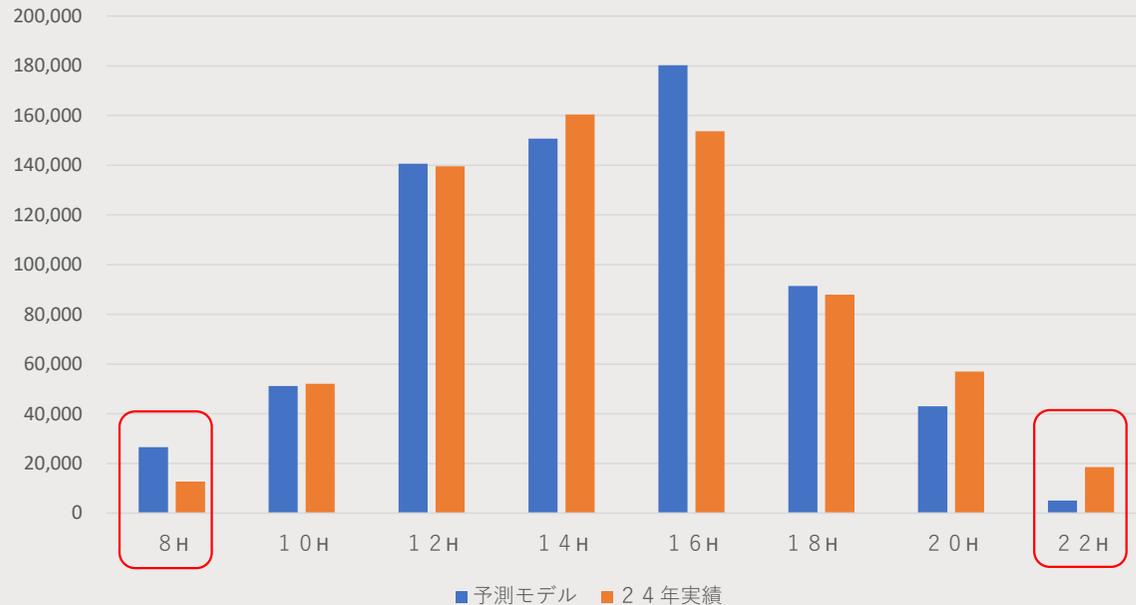
	ds	yhat
1	2024-07-31	106044.143139
2	2024-08-31	31193.408527
3	2024-09-30	-9248.830271
4	2024-10-31	-4789.374779
5	2024-11-30	-1518.633046

‘24年9月末時点  
160,450枚

## 4. '24年度予測と実績(9月末時点)

特徴量: 平均気温・最高気温・平均気温・降水量

予測モデル(ブルー) 対 24年実績(オレンジ)



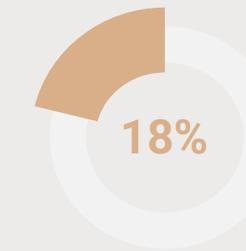
達成率



全体的には予測モデルは機能している



特徴量のチューニングが必要と感じた



8Hと22Hの予測精度に見直しの余地がある

	8H	10H	12H	14H	16H	18H	20H	22H
予測モデル	26,583	51,230	140,666	150,665	180,276	91,469	43,001	5,060
24年実績	12,750	52,100	139,550	160,450	153,750	87,950	57,050	18,550

## 4. 8Hと22H精度改善 Optuna活用

Optuna アルゴリズムTPE(Tree-structured Parzen Estimator)を使いハイパーパラメータの再調整  
‘23予測対実績の差、’24予測対実績の差 この2つを合計した値が最小化するパターンをTrial100回で検索

### 8 H OPTUNAの最適化

```
Best trial 1:  
2023年の実際の合計と予測の合計の差: 12110.878628565515  
2024年の実際の合計と予測の合計の差: 5783.343593562767  
2023年と2024年の差の合計: 17894.2222212828  
Params:  
use_Av.temp Toku: False  
use_Max.temp Toku: True  
use_Min.temp Toku: False  
use_Precip.Toku: True  
use_Sunshine.time Toku: False  
use_Max.Snowfall Toku: True  
use_Snowfall Toku: False  
use_Av.wind.speed Toku: False  
use_Max.wind.speed Toku: True  
use_Max.instantaneous.wind.speed: False  
use_Av.vapor.pressure Toku: True  
use_Av.humi Toku: False  
use_Min.sea.level.Pressure Toku: True  
use_Av.sea.level.Pressure Toku: False  
use_Av.local.pressure Toku: True  
use_Min.relative.humidity Toku: True  
use_Av.temp.Kaga: True  
use_Max.temp.Kaga: True  
use_Min.temp.Kaga: True  
use_Precip.Kaga: False  
use_Sunshine.time.Kaga: False  
use_Max.Snowfall.Kaga: False  
use_Snowfall.Kaga: True  
use_Av.wind.speed.Kaga: False  
use_Max.wind.speed.Kaga: True  
use_Wind.Direct Toku Numeric: False  
use_Wind.Direct.Kaga Numeric: False  
changeoint_prior_scale: 0.41655000479303034  
seasonality_prior_scale: 0.02705111826387107
```

### 2 2 H OPTUNAの最適化

```
Best trial 1:  
2023年の実際の合計と予測の合計の差: 438.16503496090445  
2024年の実際の合計と予測の合計の差: 15538.374692393118  
2023年と2024年の差の合計: 15976.539727354022  
Params:  
use_Av.temp Toku: True  
use_Max.temp Toku: False  
use_Min.temp Toku: False  
use_Precip.Toku: True  
use_Sunshine.time Toku: False  
use_Max.Snowfall Toku: False  
use_Snowfall Toku: True  
use_Av.wind.speed Toku: False  
use_Max.wind.speed Toku: True  
use_Max.instantaneous.wind.speed: False  
use_Av.vapor.pressure Toku: True  
use_Av.humi Toku: False  
use_Min.sea.level.Pressure Toku: True  
use_Av.sea.level.Pressure Toku: False  
use_Av.local.pressure Toku: True  
use_Min.relative.humidity Toku: True  
use_Av.temp.Kaga: False  
use_Max.temp.Kaga: True  
use_Min.temp.Kaga: True  
use_Precip.Kaga: False  
use_Sunshine.time.Kaga: False  
use_Max.Snowfall.Kaga: False  
use_Snowfall.Kaga: True  
use_Av.wind.speed.Kaga: False  
use_Max.wind.speed.Kaga: True  
use_Wind.Direct Toku Numeric: True  
use_Wind.Direct.Kaga Numeric: False  
changeoint_prior_scale: 0.004298685692807148  
seasonality_prior_scale: 0.06631315319853563
```

2023年のActual合計: 18900  
2023年のForecast合計: 24025.

2023年のActual合計: 18150  
2023年のForecast合計: 17865.

+

	8 H
予測モデル	26,583
2 4 年実績	12,750

||  
差18,958枚

+

	2 2 H
	5,060
	18,550

||  
差13,775枚

- ここでは8Hと22Hを使い実行 '24実数を入力し実行
- 残念ながら22Hの'23年度のモデルの当てはまりは±438枚と依然良いが Optuna提案のハイパーパラメータは実用的でない判断した。
- 次に機械学習も使い検証してみたい

## 4. 8Hと22H精度改善 Optuna活用

Optuna アルゴリズムTPE(Tree-structured Parzen Estimator)を使い機械学習モデルを検証  
使用した機械学習モデルは、RandomForest, XGBoost, LightGBM, CatBoostで比較  
'23予測対実績の差、'24予測対実績の差 この2つを合計した値が最小化するパターンをTrial100回で検索  
特徴量を平均気温・最高気温・平均気温・降水量とした

### 8 H OPTUNAの最適化

```
2023年の実際の合計と予測の合計の差: 439.431640625
2024年の実際の合計と予測の合計の差: 9658.505126953125
2023年と2024年の差の合計: 10097.936767578125
Params:
  model: XGBoost
  n_estimators: 169
  learning_rate: 0.0879195873927008
  max_depth: 3
2023年の実際の合計と予測の合計の差: 2910.290726010742
2024年の実際の合計と予測の合計の差: 8935.846228180166
2023年と2024年の差の合計: 11846.136954190908
Params:
  model: CatBoost
  n_estimators: 145
  learning_rate: 0.2116257387798887
2023年の実際の合計と予測の合計の差: 592.0858188745806
2024年の実際の合計と予測の合計の差: 9354.457729178057
2023年と2024年の差の合計: 9946.543548052638
Params:
  model: CatBoost
  n_estimators: 290
  learning_rate: 0.2107580273281451
2023年の実際の合計と予測の合計の差: 12481.917075627483
2024年の実際の合計と予測の合計の差: 5840.829992599369
2023年と2024年の差の合計: 18322.747068226854
Params:
  model: LightGBM
  n_estimators: 79
  learning_rate: 0.013509837648005858
  max_depth: 18
```

### 2 2 H OPTUNAの最適化

```
Best trials:
2023年の実際の合計と予測の合計の差: 4728.521484375
2024年の実際の合計と予測の合計の差: 16141.406982421875
2023年と2024年の差の合計: 20869.928466796875
Params:
  model: XGBoost
  n_estimators: 84
  learning_rate: 0.017940213327942323
  max_depth: 8
2023年の実際の合計と予測の合計の差: 4659.158837800933
2024年の実際の合計と予測の合計の差: 17256.11687729646
2023年と2024年の差の合計: 21915.275715097392
Params:
  model: CatBoost
  n_estimators: 142
  learning_rate: 0.2932362378291297
```

2023年のActual合計: 18900  
2023年のForecast合計: 24025.

2023年のActual合計: 18150  
2023年のForecast合計: 17865.

	8 H
予測モデル	26,583
2 4 年実績	12,750

	2 2 H
	5,060
	18,550

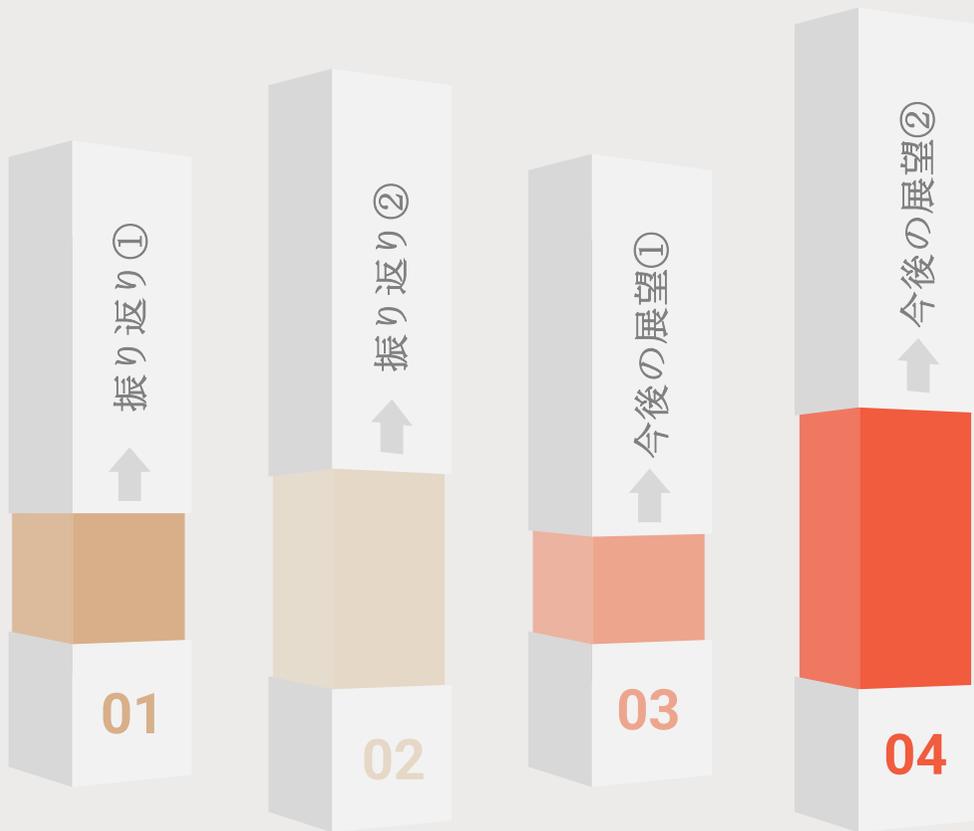
|| 差18,958枚

|| 差13,775枚

- 8 HでCatBoostモデルにより'23/24合計差は多少改善するが、'24予測は大きく外れている。  
22Hでは'23/24合計差が2万を超えるており、ここでの機械学習モデル実用的でない判断した。

## ⑤今後の展望

## 5. 振り返りと今後の展望



01

### ある程度満足したモデルが出来た

限られた時間の中では満足できるモデリングが出来た。やはり気象データを月次で捉える言に抵抗がある。モデリングは月次でないと評価に値しない事が理解できた。学習のみデイリーベースでのモデリングを再度模索したい。

02

### モデル化の際、マイナス値が気になる

マイナス値を除外して合計算出。本来は▲はおかしいと思う。実運用を通じて、改善していきたい。

Filtered forecast data:

	ds	yhat
121	2023-01-31	-3782.610709
122	2023-02-28	-4531.702049
123	2023-03-31	-9865.193896
124	2023-04-30	-8578.642856
125	2023-05-31	3558.844792
126	2023-06-30	26205.704020

03

### 機械学習モデルを展開していない

時間的な事情、及びデータが時系列な為、Prophetモデルの深化に努めた。検証として、RandomForest, CarXgboost, LightGBM等でのアンサンブル学習を通じて機械学習モデルでの検証を行ってみたい。

04

### 特徴量 一層のパラメータチューニング

最高気温と最低気温の差、1月から6月の累積降水量等を実行したが、良い結果が得られなかった。ベストはないのでベターを繰り返していきたい。



ご清聴ありがとうございました。