BREWCAST

FORECASTING SUCCESS IN THE COFFEE BUSINESS

Sara Almarzooqi - 202004091
Saleha Alameri - 202008912
Safa Baalfaqih 202051107

CONTENT

PROBLEM

Meet Maryam, an aspiring entrepreneur venturing into the world of coffee shops. As she embarks on this journey, Maryam faces challenges typical of small business owners. The critical task at hand is optimizing resources, staffing, and inventory to meet customer demand, especially during favourable weather conditions and peak times. Our solution? A tailored predictive model designed to help Maryam anticipate successful business days based on a blend of weather and time factors. With this tool, Maryam gains the foresight needed to make strategic decisions, ensuring her small coffee shop not only weathers challenges but thrives in the dynamic landscape of the coffee business.

GOALS AND OBJECTIVES

Objective n° 1

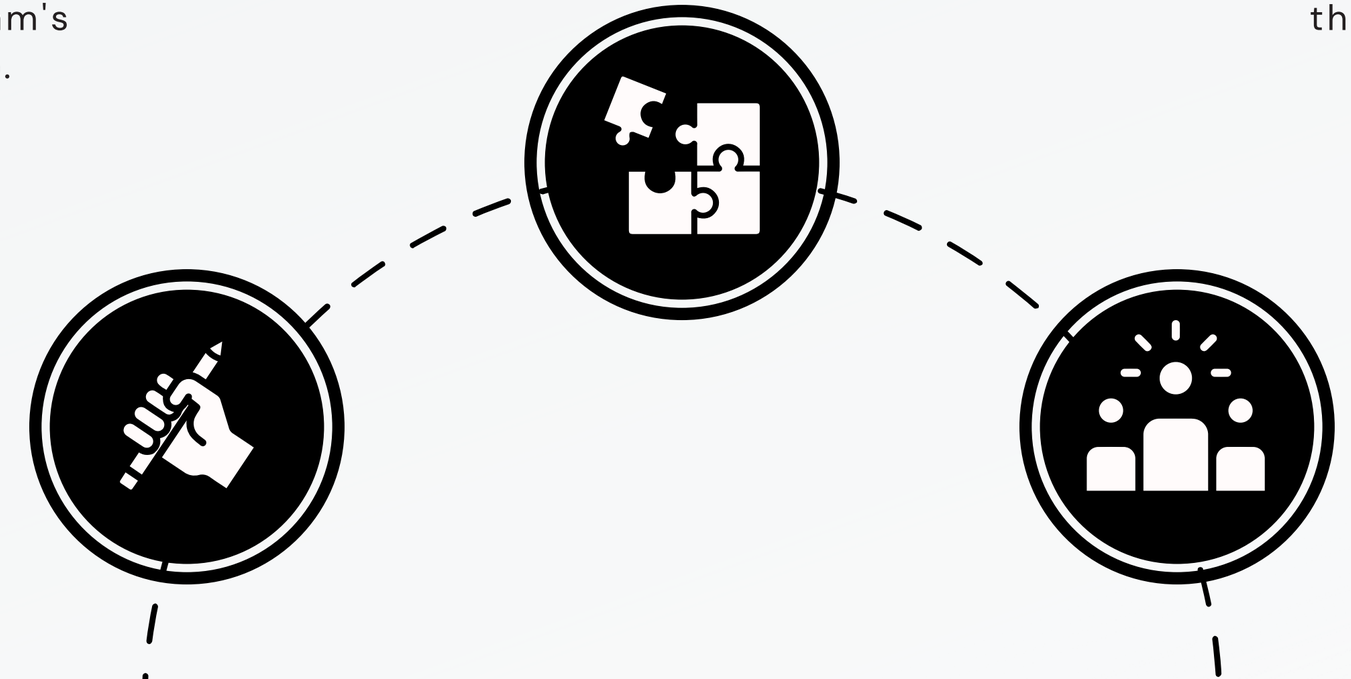
Develop a robust machine learning model to predict successful business days for Maryam's coffee shop.

Objective n° 2

Provide actionable insights enabling Maryam to optimize staffing, inventory, and offer dates.

Objective n° 3

Enhance overall business efficiency and profitability by making informed decisions through data-driven forecasting.



DATA PREPROCESSING

Mission

2/22	19.59	1	0	0	200	162	36	SUCCESS
3/22	19.58	1	0	0	171	110	60	SUCCESS
4/22	13.41	0	0	0	126	103	18	SUCCESS
5/22	14.69	0	1	1	130	86	40	SUCCESS
6/22	21.75	0	0	0	172	91	78	SUCCESS
7/22	17.69	0	1	0	94	44	48	NOT SUCCESSFUL
8/22	20.7	0	1	0	88	49	37	NOT SUCCESSFUL
9/22	24.42	1	1	0	172	110	60	SUCCESS
0/22	20.09	1	1	0	133	74	54	SUCCESS
1/22	19.12	0	1	1	245	186	58	SUCCESS
01/2022	14.58	0	1	1	339	182	152	SUCCESS
02/2022	19.36	0	0	1	221	100	118	SUCCESS
03/2022	15.89	0	1	0	69	45	21	NOT SUCCESSFUL
04/2022	22.06	0	0	0	93	74	18	NOT SUCCESSFUL
05/2022	14.97	1	1	1	140	98	38	SUCCESS
06/2022	21.76	1	0	0	308	200	104	SUCCESS
07/2022	23.59	0	1	0	92	86	2	NOT SUCCESSFUL
08/2022	16.03	0	1	0	74	48	22	NOT SUCCESSFUL
09/2022	22.81	0	0	0	77	65	9	NOT SUCCESSFUL

- Collected sales from Maryam for the past two years.
- Collected weather updates from the National Meteorological Center (NMC) and Weather Underground to incorporate external factors.
- Success rate is determined from Maryam, and it is based on approximately how many coffees should she sell to achieve profit, afford rent and staff bills.

- Addressed missing values, outliers, and inconsistencies in the dataset.
- Conducted feature engineering to enhance the model's ability to capture relevant patterns.
- Ensured data compatibility and consistency between the coffee shop and weather datasets.

Vision

Date	Avg_Temp	Is_Weeker	Humid	Rainy	Num_Cust	Hot_Drink	Cold_Drink	Succcessfull or not
01/10/2022	20.72	0	1	0	88	56	31	NOT SUCCESSFUL
01/11/2022	21.83	0	0	1	210	116	90	SUCCESS
01/12/2022	19.02	0	0	0	98	53	44	NOT SUCCESSFUL
1/13/22	21.4	0	0	0	125	106	14	SUCCESS
1/14/22	16.39	0	0	0	68	54	12	NOT SUCCESSFUL
1/15/22	21.25	1	0	0	194	136	56	SUCCESS
1/16/22	18.72	1	0	1	167	130	34	SUCCESS
1/17/22	18.19	0	1	1	110	74	34	SUCCESS
1/18/22	23.02	0	1	0	78	38	37	NOT SUCCESSFUL
1/19/22	14.82	0	1	0	127	89	29	SUCCESS
1/20/22	16.4	0	1	1	191	84	106	SUCCESS
1/21/22	18.55	0	1	1	207	128	78	SUCCESS
1/22/22	19.59	1	0	0	200	162	36	SUCCESS
1/23/22	19.58	1	0	0	171	110	60	SUCCESS
1/24/22	13.41	0	0	0	126	103	18	SUCCESS
1/25/22	14.69	0	1	1	130	86	40	SUCCESS
1/26/22	21.75	0	0	0	172	91	78	SUCCESS
1/27/22	17.69	0	1	0	94	44	48	NOT SUCCESSFUL
1/28/22	20.7	0	1	0	88	49	37	NOT SUCCESSFUL
1/29/22	24.42	1	1	0	172	110	60	SUCCESS
1/30/22	20.09	1	1	0	133	74	54	SUCCESS
1/31/22	19.12	0	1	1	245	186	58	SUCCESS
02/01/2022	14.58	0	1	1	339	182	152	SUCCESS
02/02/2022	19.36	0	0	1	221	100	118	SUCCESS
02/03/2022	15.89	0	1	0	69	45	21	NOT SUCCESSFUL
02/04/2022	22.06	0	0	0	93	74	18	NOT SUCCESSFUL
02/05/2022	14.97	1	1	1	140	98	38	SUCCESS
02/06/2022	21.76	1	0	0	308	200	104	SUCCESS
02/07/2022	23.59	0	1	0	92	86	2	NOT SUCCESSFUL
02/08/2022	16.03	0	1	0	74	48	22	NOT SUCCESSFUL
02/09/2022	22.81	0	0	0	77	65	9	NOT SUCCESSFUL
02/10/2022	17.37	0	0	1	247	230	12	SUCCESS
02/11/2022	24.44	0	0	0	55	53	1	NOT SUCCESSFUL
02/12/2022	22.49	1	1	0	181	80	98	SUCCESS
2/13/22	18.66	1	1	0	223	104	118	SUCCESS

MODEL SELECTION

- Chose a Random Forest due to its ability to capture non-linear relationships and provide interpretability.
- Random Forest are well-suited for this scenario, as they can effectively represent the complex interactions between weather conditions, time of day, and business success.

DEVELOPMENT AND RESULTS

[1]:

```
# 1. Import Libraries

from numpy import mean
from sklearn.model_selection import KFold
from sklearn.model_selection import cross_val_score
from sklearn.model_selection import train_test_split
from sklearn.model_selection import KFold
from sklearn.ensemble import RandomForestClassifier
import pandas as pd
import numpy as np
```

+ Code + Markdown

[2]:

```
# 2. Loading the dataset

df= pd.read_csv('/kaggle/input/coffeeshopp/coffee_shop_sales_dataset_updated.csv')
df = df.drop('Date', axis=1)
df.head()
```

[2]:

```
# 2. Loading the dataset

df= pd.read_csv('/kaggle/input/coffeeshopp/coffee_shop_sales_dataset_updated.csv')
df = df.drop('Date', axis=1)
df.head()
```

[2]:

	Avg_Temperature	Is_Weekend	Humid	Rainy	Num_Customers	Hot_Drinks_Ordered	Cold_Drinks_Ordered	Suc
0	20.72	0	1	0	88	56	31	SUCCESS
1	21.83	0	0	1	210	116	90	SUCC
2	19.02	0	0	0	98	53	44	SUCCESS
3	21.40	0	0	0	125	106	14	SUCC
4	16.39	0	0	0	68	54	12	SUCCESS

[3]:

```
# 3. Independent And dependent features

X = df.drop('Success', axis=1)  # 'Success' is the target column
y = df['Success']

#X=X.dropna(axis=1)

df['Success'] = df['Success'].replace({'SUCCESS': 1, 'NOT SUCCESSFUL': 0})
```

[4]:

```
# 4. Splitting the dataset into train and test

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.30, random_st
```

[5]:

```
# 5. Define folds to test the values of k in the given range
```

```python
from sklearn.model_selection import GridSearchCV

param_grid = {
    'n_estimators': [50, 100, 200],
    'max_depth': [None, 10, 20],
}

grid_search = GridSearchCV(RandomForestClassifier(random_state=42), param_grid, cv=5)
grid_search.fit(X_train, y_train)

best_params = grid_search.best_params_
print("Best Hyperparameters:", best_params)

# Use the best model for evaluation
best_model = grid_search.best_estimator_
print(best_model)
```

```
Best Hyperparameters: {'max_depth': None, 'n_estimators': 50}
RandomForestClassifier(n_estimators=50, random_state=42)
```

[7]:

```
# 6. Evaluating the model using a given test condition

from sklearn.model_selection import GridSearchCV

def evaluate_model(cv):
    X = df.drop(['Success'], axis=1) #independent features
    y = df['Success']  # Dependent feature


    # Create a Random Forest classifier
    model = RandomForestClassifier(random_state=42)


    # Perform cross-validation
    scores = cross_val_score(best_model, X_train, y_train, scoring='accuracy', cv=cv, n_jobs=-1)
    # Return scores
    return scores.mean(), scores.min(), scores.max()

cv_results = evaluate_model(cv=5)
print("Mean Accuracy:", cv_results[0])
print("Min Accuracy:", cv_results[1])
print("Max Accuracy:", cv_results[2])
```

```
Mean Accuracy: 0.9849228611500701
Min Accuracy: 0.978494623655914
Max Accuracy: 0.989247311827957
```

```python
for k in folds:
    # define the test condition
    cv = KFold(n_splits=k, shuffle=True, random_state=10)
    # record mean and min/max of each set of results
    k_mean, k_min, k_max = evaluate_model(cv)
    # report performance
    print('-> folds=%d, accuracy=%.3f (%.3f,%.3f)' % (k, k_mean, k_min, k_max))
```

```
-> folds=2, accuracy=0.989 (0.987,0.991)
-> folds=3, accuracy=0.983 (0.974,0.987)
-> folds=4, accuracy=0.987 (0.974,1.000)
-> folds=5, accuracy=0.991 (0.978,1.000)
-> folds=6, accuracy=0.985 (0.974,1.000)
-> folds=7, accuracy=0.987 (0.970,1.000)
-> folds=8, accuracy=0.989 (0.966,1.000)
-> folds=9, accuracy=0.991 (0.962,1.000)
-> folds=10, accuracy=0.985 (0.957,1.000)
-> folds=11, accuracy=0.989 (0.952,1.000)
-> folds=12, accuracy=0.989 (0.947,1.000)
-> folds=13, accuracy=0.989 (0.944,1.000)
-> folds=14, accuracy=0.989 (0.939,1.000)
-> folds=15, accuracy=0.989 (0.968,1.000)
-> folds=16, accuracy=0.989 (0.931,1.000)
-> folds=17, accuracy=0.987 (0.926,1.000)
-> folds=18, accuracy=0.987 (0.923,1.000)
-> folds=19, accuracy=0.987 (0.917,1.000)
-> folds=20, accuracy=0.989 (0.913,1.000)
-> folds=21, accuracy=0.983 (0.909,1.000)
-> folds=22, accuracy=0.985 (0.905,1.000)
-> folds=23, accuracy=0.987 (0.900,1.000)
-> folds=24, accuracy=0.989 (0.947,1.000)
-> folds=25, accuracy=0.989 (0.944,1.000)
-> folds=26, accuracy=0.987 (0.889,1.000)
-> folds=27, accuracy=0.987 (0.882,1.000)
-> folds=28, accuracy=0.987 (0.938,1.000)
-> folds=29, accuracy=0.991 (0.938,1.000)
```

CONCULSION

In conclusion, our machine learning model achieves an impressive 98.5% validation accuracy in predicting successful days for Maryam's coffee shop. This accuracy empowers Maryam to strategically time promotional offers and optimize labour resources and enhancing operational efficiency. The model serves as a valuable tool for Maryam to make informed decisions, ultimately contributing to the overall success of her business.