Recipe Site Traffic

Zean Ni Zhehan

Contents Table

1. Background
2. Goals
3. Project Summary
4. Key Findings & Results
5. Conclusion & Recommendation

Background

Background

01

Online recipes

Tasty Bytes, an online recipe startup features new recipes on their homepage website every day

02

Traffic boost

On days that they feature a popular recipe, traffic increases by as much as 40%

03

Challenge

However, it is difficult to predict ahead of time which recipes have high traffic

04

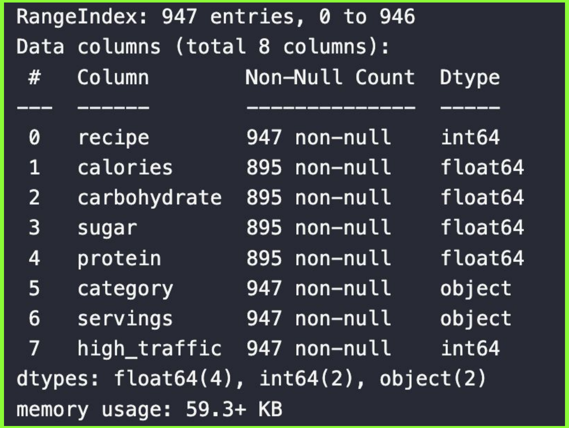
Traffic

A typical binary classification problem

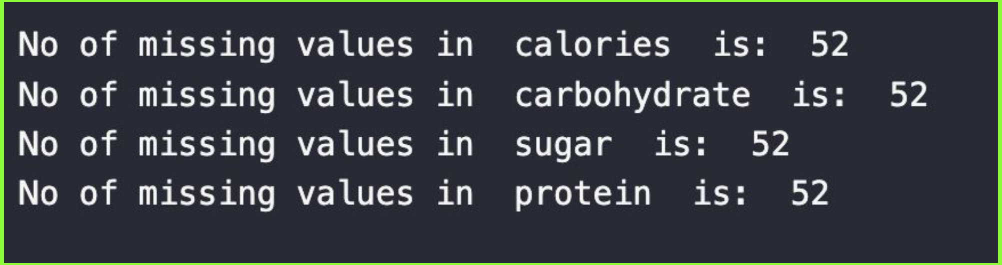
Predict whether a recipe will receive a high traffic based on the data collected from previously published recipes. The criterion for success was the correct classification of **75%** of the recipes.

✷ Project Goals ✷

Validation & EDA

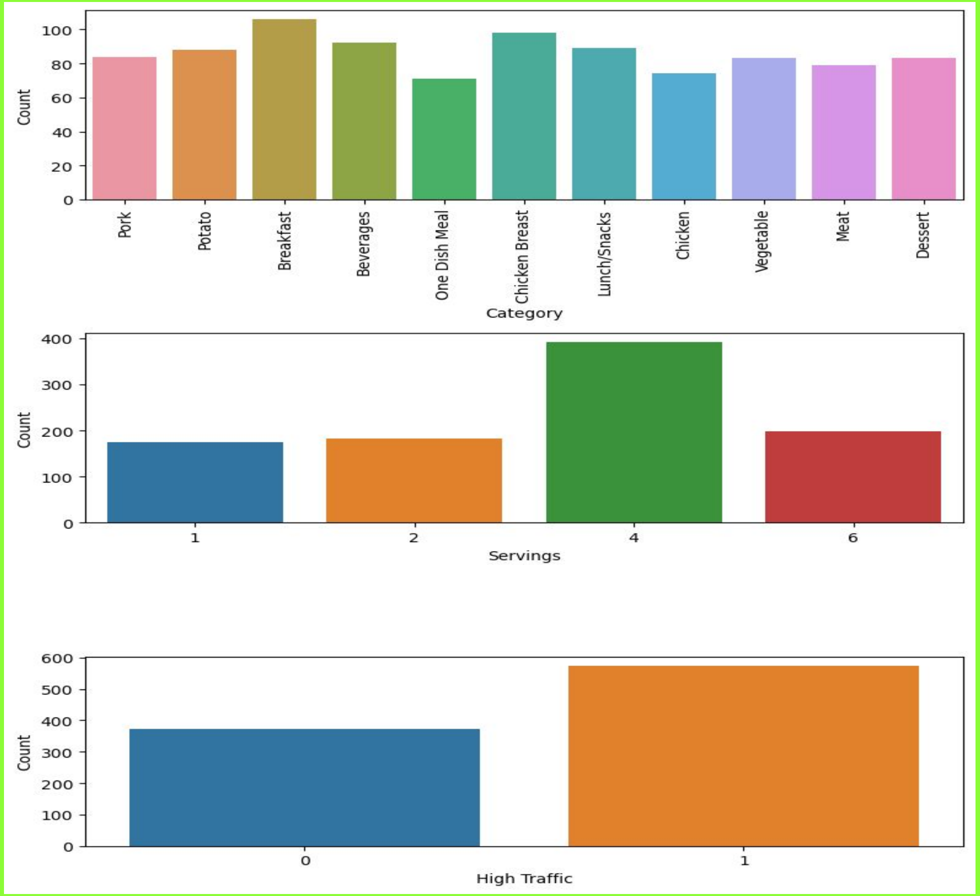


```
RangeIndex: 947 entries, 0 to 946
Data columns (total 8 columns):
 #   Column        Non-Null Count  Dtype
---  ------        --------------  -----
 0   recipe        947 non-null    int64
 1   calories      895 non-null    float64
 2   carbohydrate  895 non-null    float64
 3   sugar         895 non-null    float64
 4   protein       895 non-null    float64
 5   category      947 non-null    object
 6   servings      947 non-null    object
 7   high_traffic  947 non-null    int64
dtypes: float64(4), int64(2), object(2)
memory usage: 59.3+ KB
```

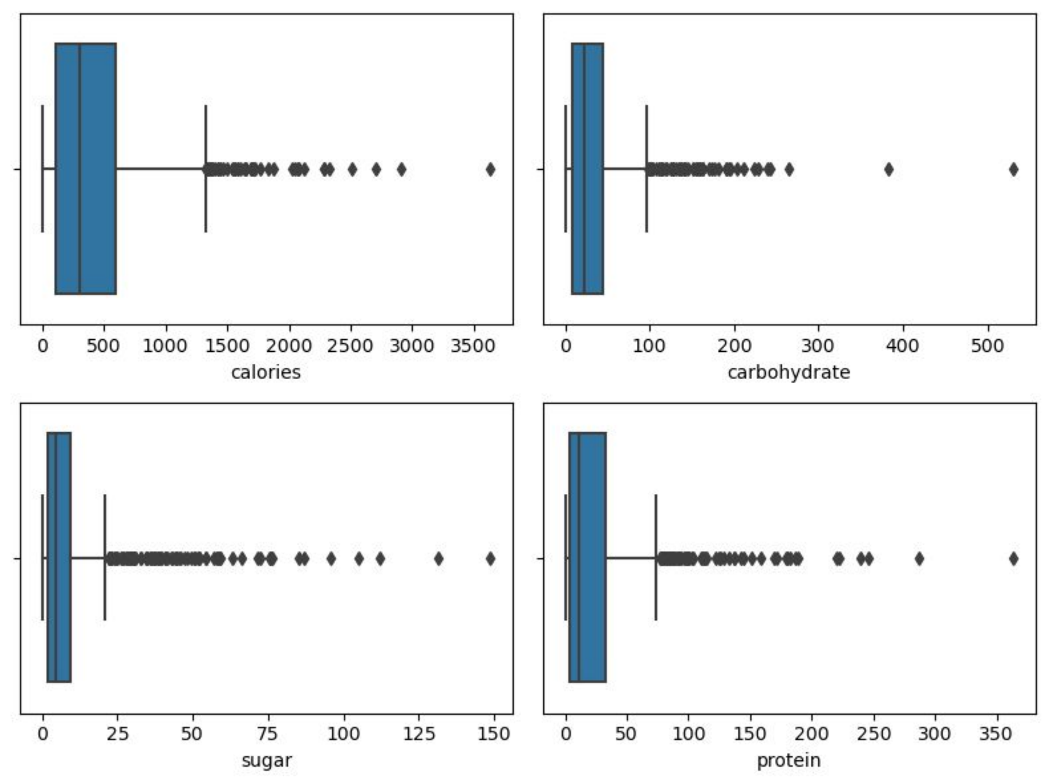


```
No of missing values in  calories  is:  52
No of missing values in  carbohydrate  is:  52
No of missing values in  sugar  is:  52
No of missing values in  protein  is:  52
```

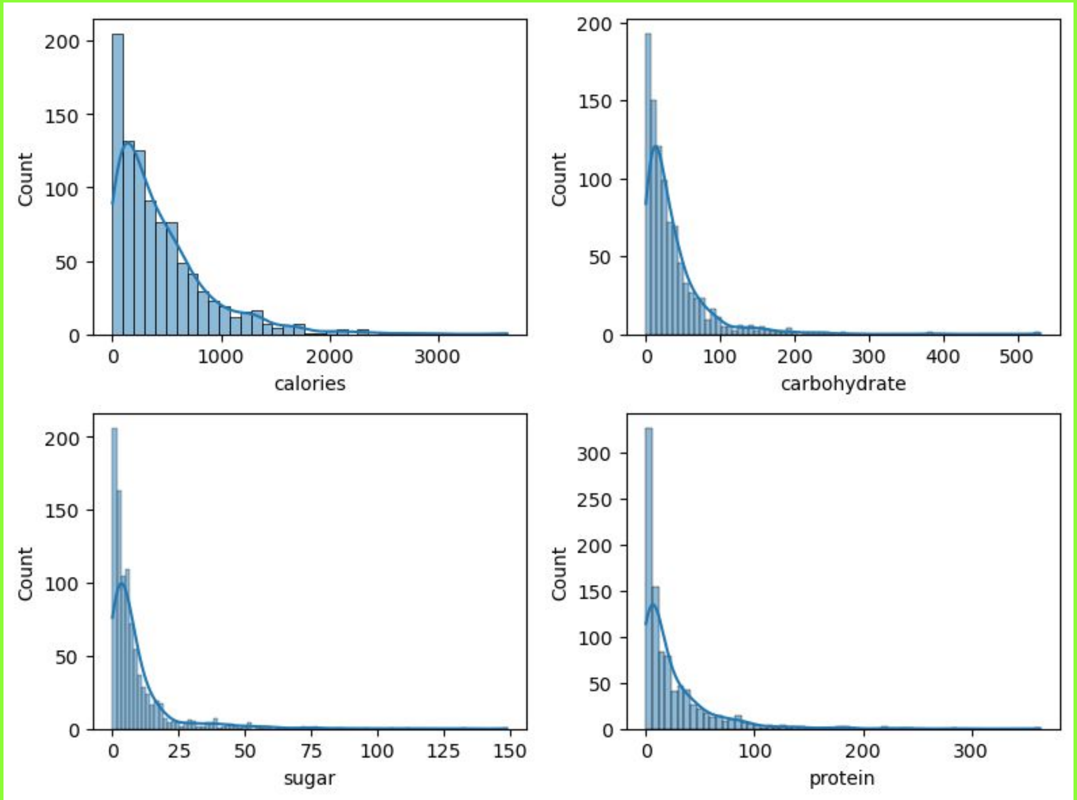
Validation & EDA



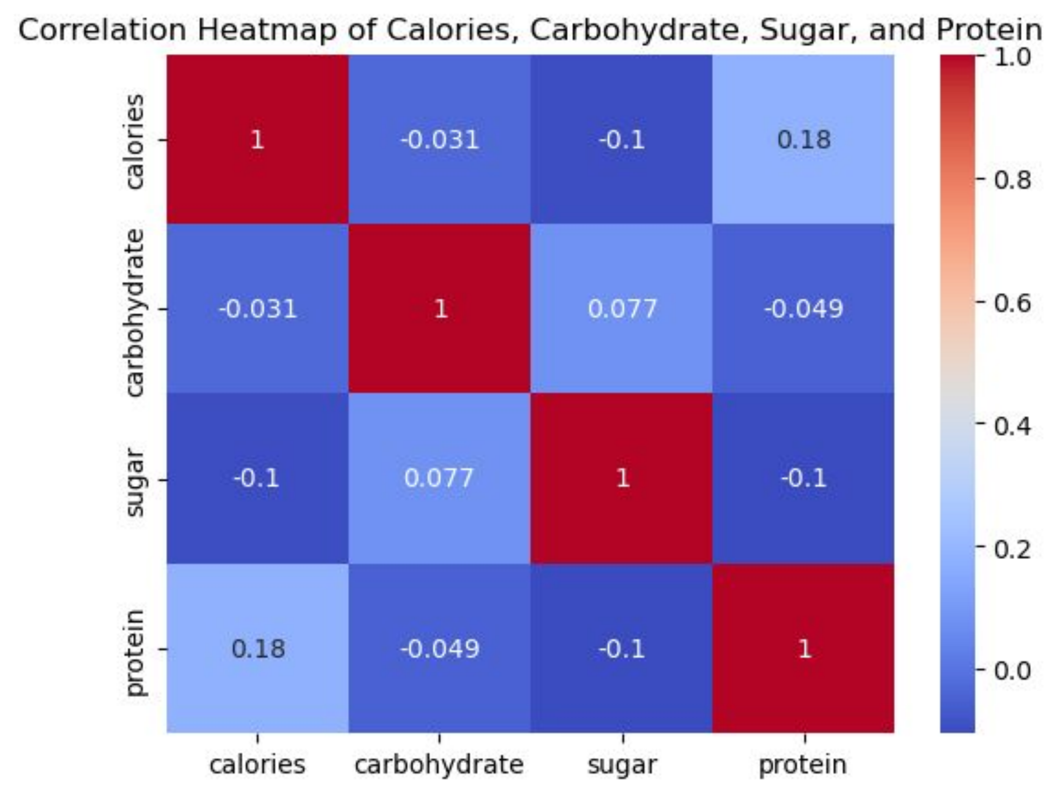
Validation & EDA



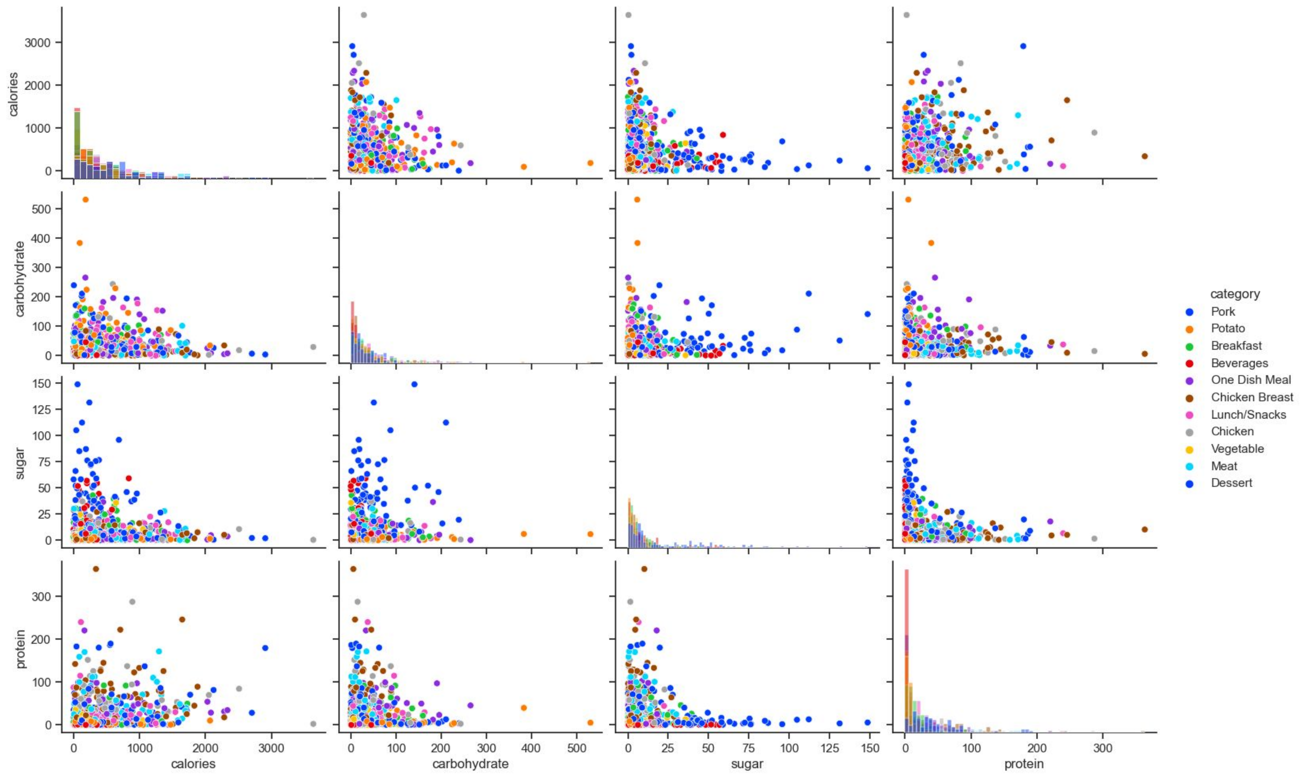
Validation & EDA



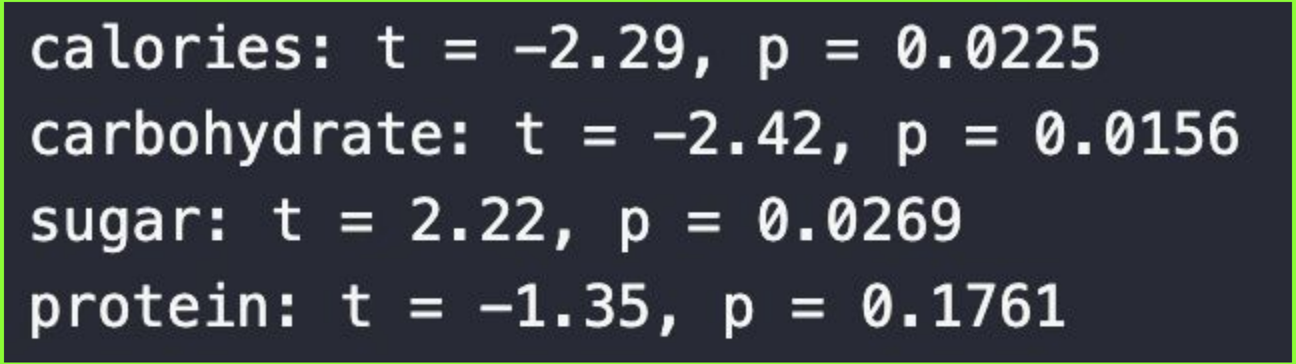
Validation & EDA



Validation & EDA

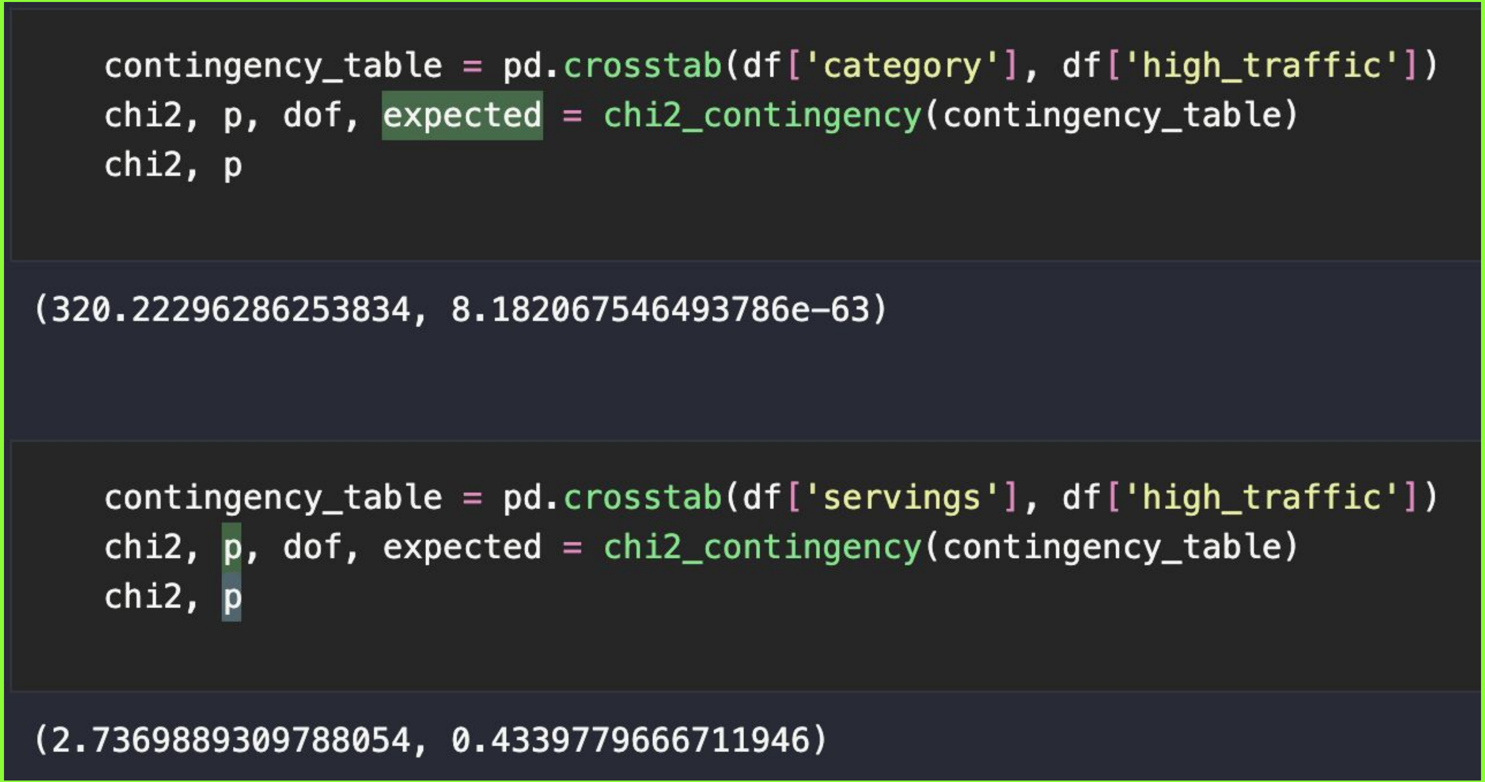


Statistical Test



```
calories: t = -2.29, p = 0.0225
carbohydrate: t = -2.42, p = 0.0156
sugar: t = 2.22, p = 0.0269
protein: t = -1.35, p = 0.1761
```

Statistical Test



```
contingency_table = pd.crosstab(df['category'], df['high_traffic'])
chi2, p, dof, expected = chi2_contingency(contingency_table)
chi2, p
```

```
(320.22296286253834, 8.182067546493786e-63)
```

```
contingency_table = pd.crosstab(df['servings'], df['high_traffic'])
chi2, p, dof, expected = chi2_contingency(contingency_table)
chi2, p
```

```
(2.7369889309788054, 0.4339779666711946)
```

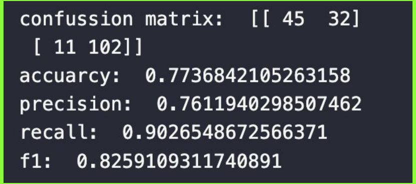
Feature Engineering

health?

Complex?

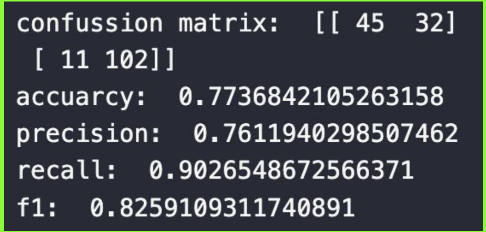
Modeling

Logistic Reg



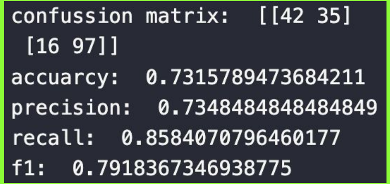
```
confussion matrix:  [[ 45  32]
 [ 11 102]]
accuarcy:  0.7736842105263158
precision:  0.7611940298507462
recall:  0.9026548672566371
f1:  0.8259109311740891
```

Random Forest



```
confussion matrix:  [[ 45  32]
 [ 11 102]]
accuarcy:  0.7736842105263158
precision:  0.7611940298507462
recall:  0.9026548672566371
f1:  0.8259109311740891
```

DNN



```
confussion matrix:  [[42 35]
 [16 97]]
accuarcy:  0.7315789473684211
precision:  0.7348484848484849
recall:  0.8584070796460177
f1:  0.7918367346938775
```

Thanks!