Tools and Techniques For Data Science Project

- Problem Statement

  Predict the sales of clothing items

- Data Acquisition

  Dataset Name: Sales of summer clothes in E-commerce Wish

  Source: Kaggle

  (https://www.kaggle.com/datasets/jmmvutu/summer-products-and-sales-in-ecommerce-wish)

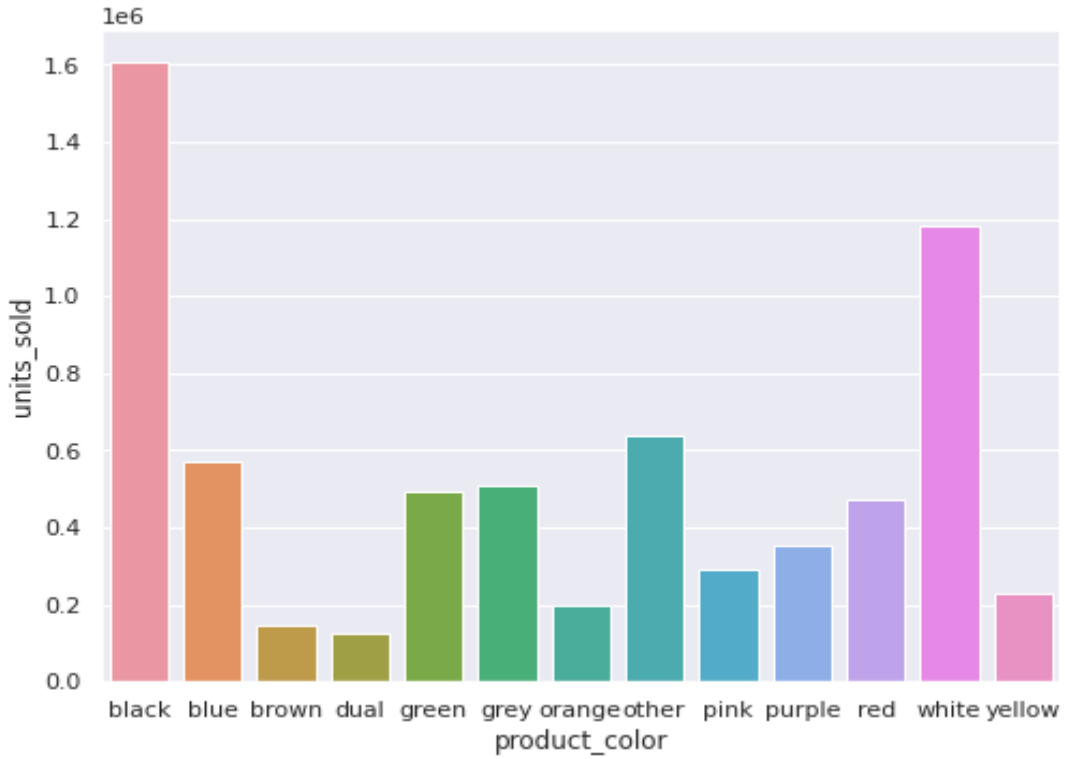
  Data Size: 1573 rows, 43 features

  Target Variable: units_sold

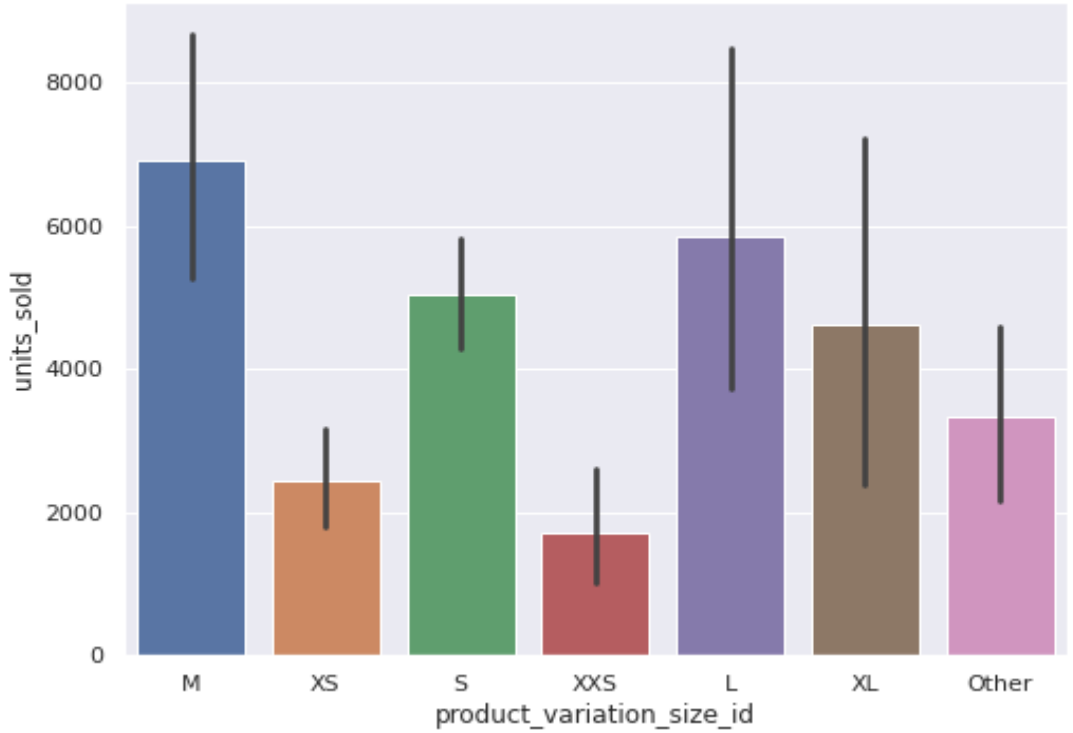
  Technology stack: NumPy , pandas, seaborn, matplotlib, sklearn, Microsoft Azure

Data Insights

Most frequently bought color: black

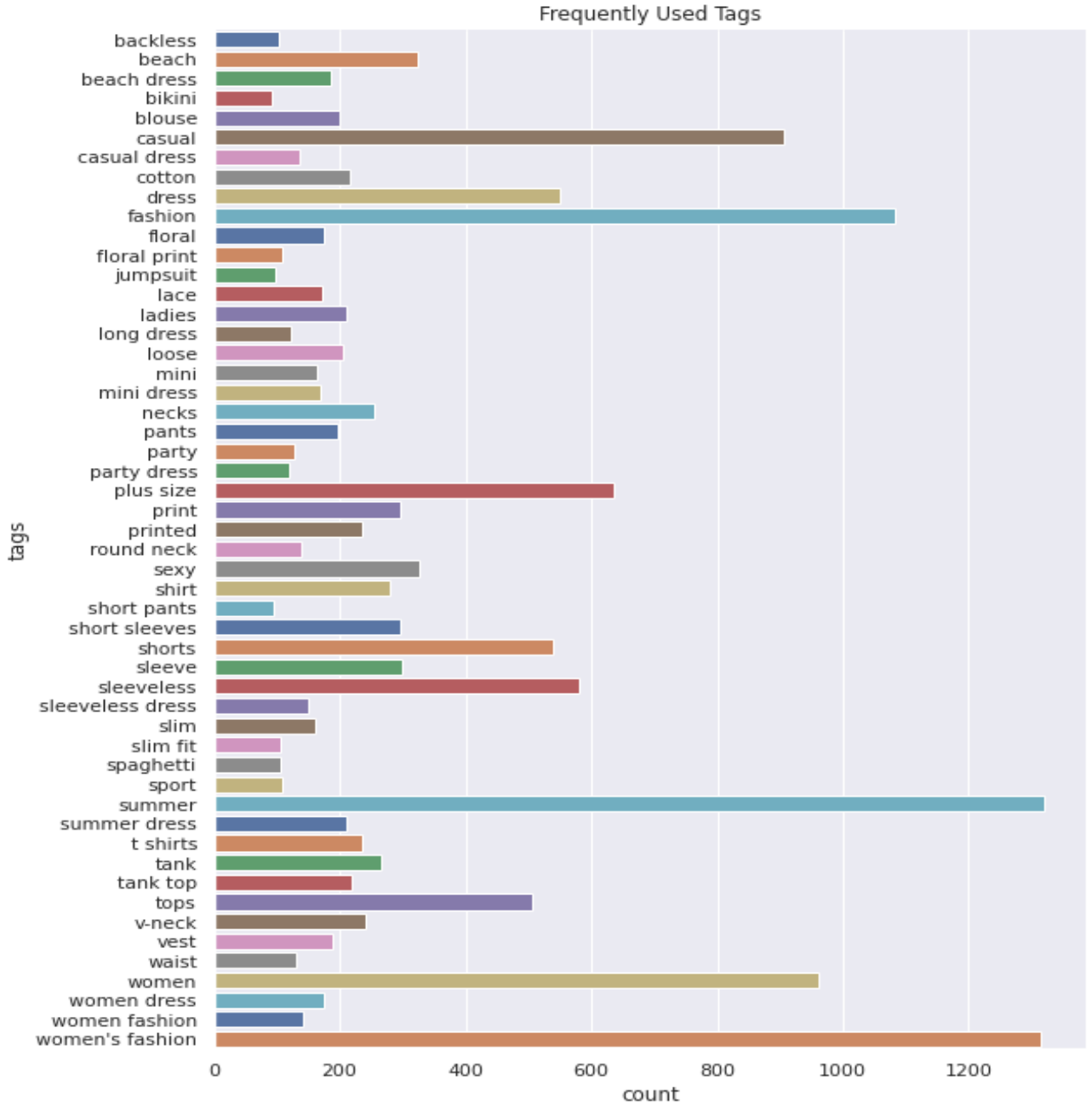


Most frequently bought size: M



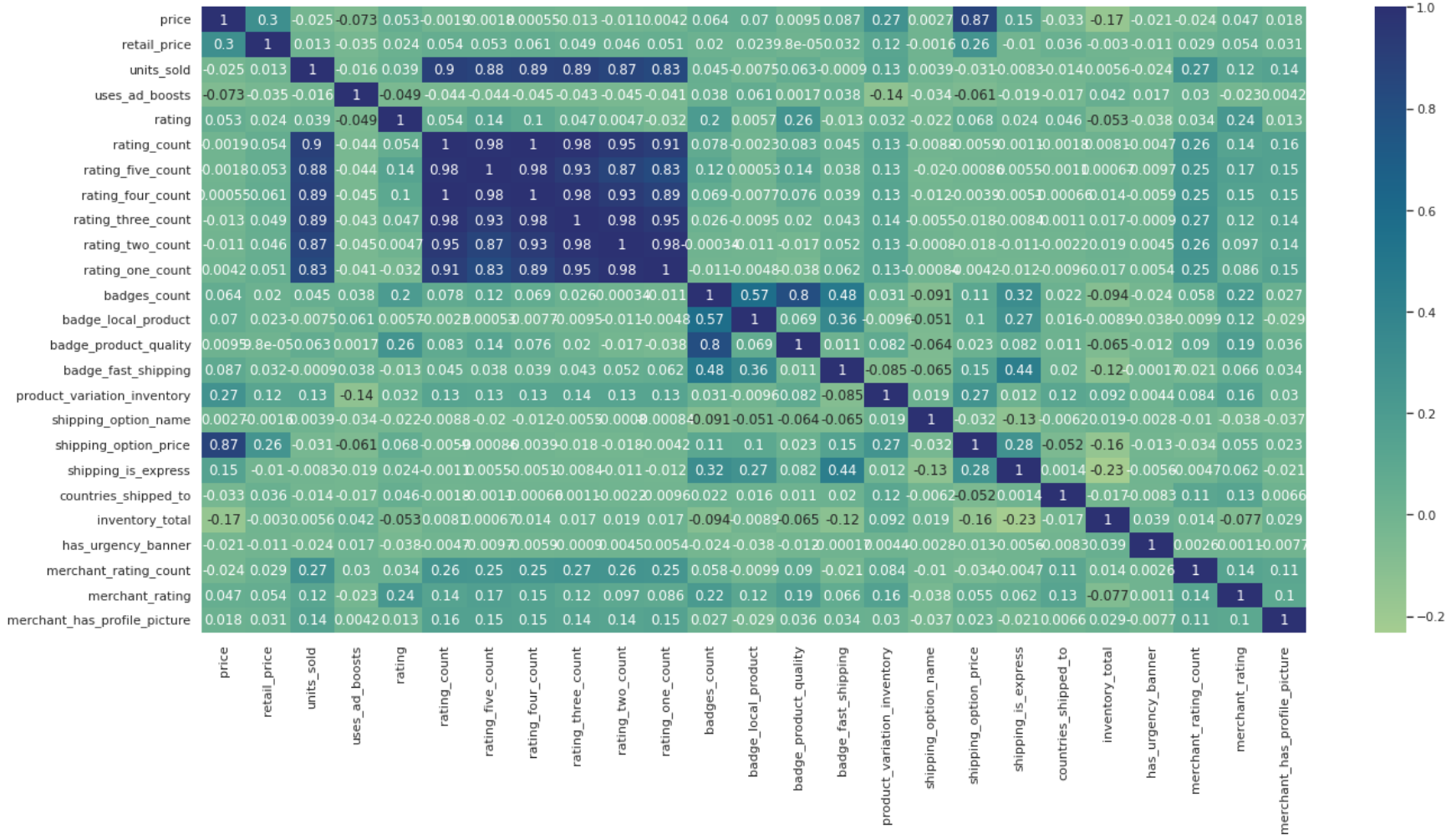
Some of the most used tags used by sellers

- Most frequent tag: summer



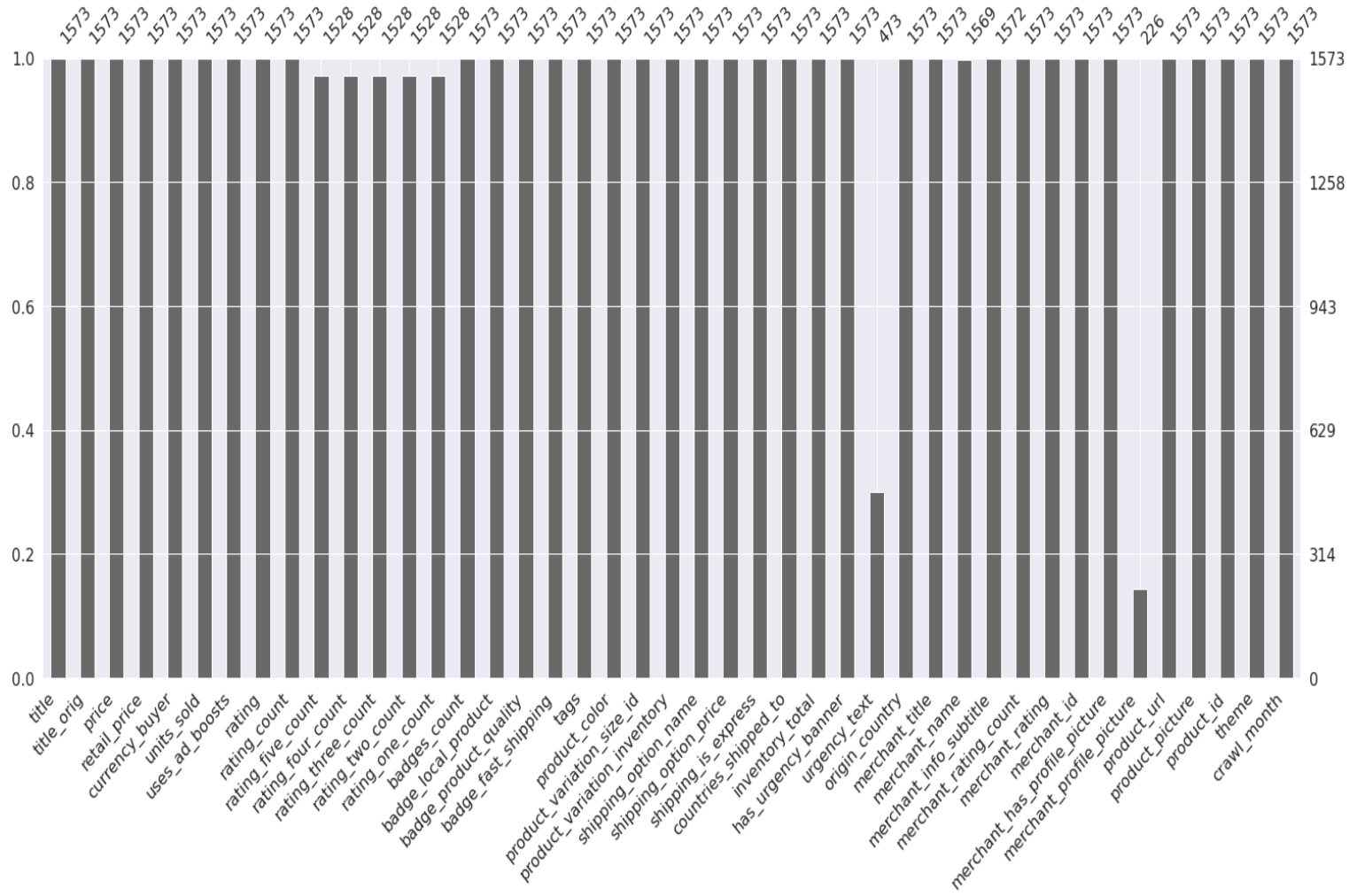
Feature Selection

- Removed features that contained only one unique values – e.g. currency_buyer
- Removed columns that were highly collinear with each other – e.g. rating_five_count, rating_four_count, urgency_text, etc.



Feature Selection

- Removed features with irrelevant information – e.g. product_url, product_id
- Also removed features with too many missing values



Reducing Categories in Target Variable

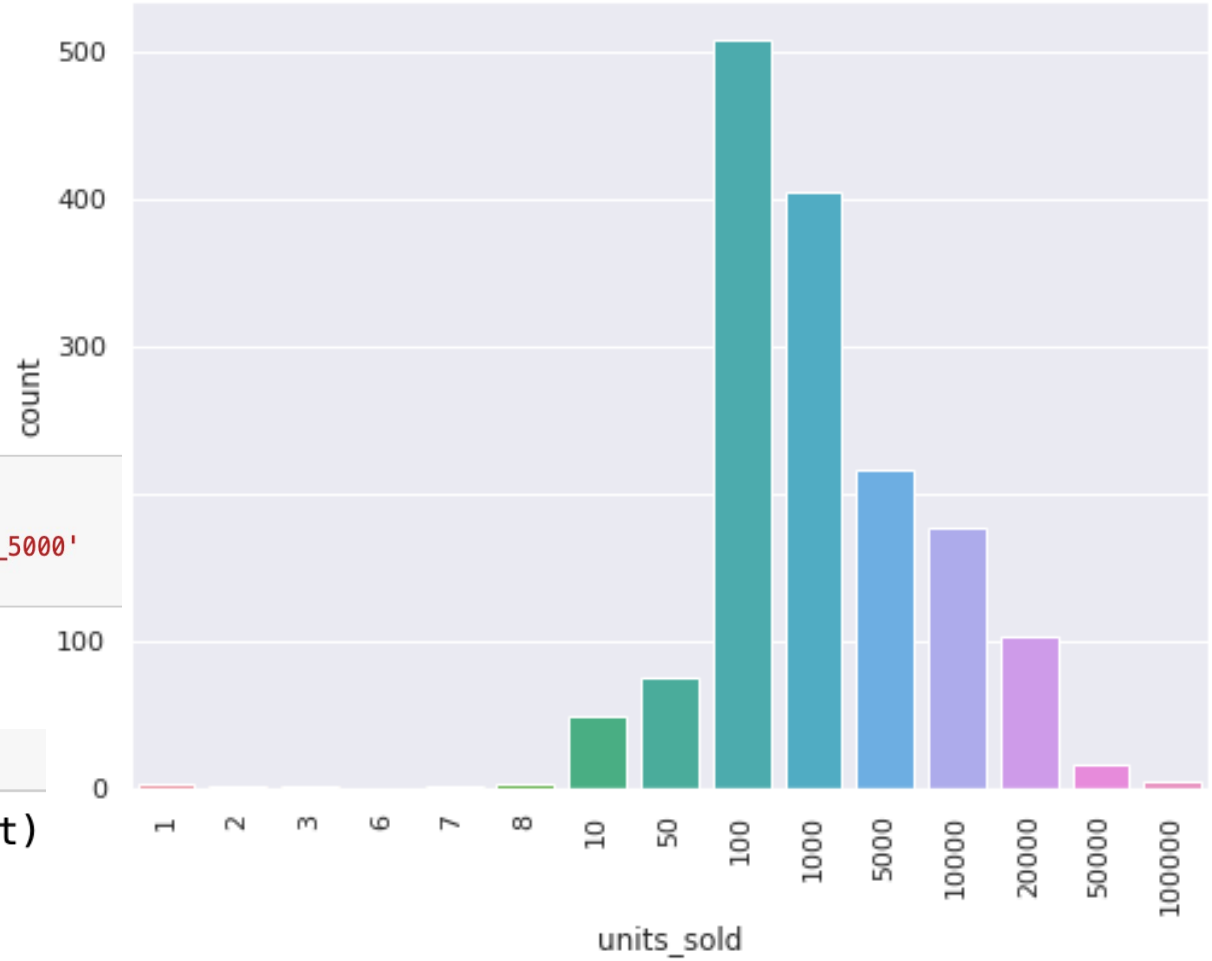
```
sales_df['units_sold'].unique()
```

```
array([   100,  20000,   5000,     10,  50000,   1000,  10000, 100000,
            50,      1,      7,      2,      3,      8,      6])
```

```
y_data = data_df['units_sold'].copy()
y_data[data_df['units_sold'] <= 100] = '<=100'
y_data[(data_df['units_sold'] > 100) & (data_df['units_sold'] <= 5000)] = '100+_to_5000'
y_data[data_df['units_sold'] > 5000] ='5000+'
```

```
y_data.unique()
```

```
array(['<=100', '5000+', '100+_to_5000'], dtype=object)
```



Unnecessary Features

- ‘rating_five_count’, 'rating_four_count’,
- 'rating_three_count’,
- 'rating_two_count’,
- 'rating_one_count’,
- 'merchant_id’,
- 'product_id',
- 'title’,
- 'title_orig’,
- 'currency_buyer’,
- 'urgency_text’,
- 'merchant_title’,
- 'merchant_info_subtitle’,
- 'merchant_profile_picture',
- 'product_url’,
- 'product_picture’,
- 'theme', 'crawl_month’,
- 'merchant_name'

```
sales_df.shape
```

```
(1573, 43)
```

Necessary Features

- 'price’,
- 'retail_price’,
- 'rating’,
- 'rating_count’,
- 'badges_count’,
- 'shipping_option_price', 'merchant_rating_count', 'merchant_rating’]
- uses_ad_boosts’,
- 'product_color’,
- 'product_variation_size_id’,
- 'shipping_is_express’,
- 'countries_shipped_to’,
- 'has_urgency_banner’,
- 'origin_country

```
X_data.shape
```

```
(1573, 17)
```

Machine Learning Solution and Results

- Categorical features

  Simple Imputer: Replace missing values with most frequent value

  OneHotEncoder

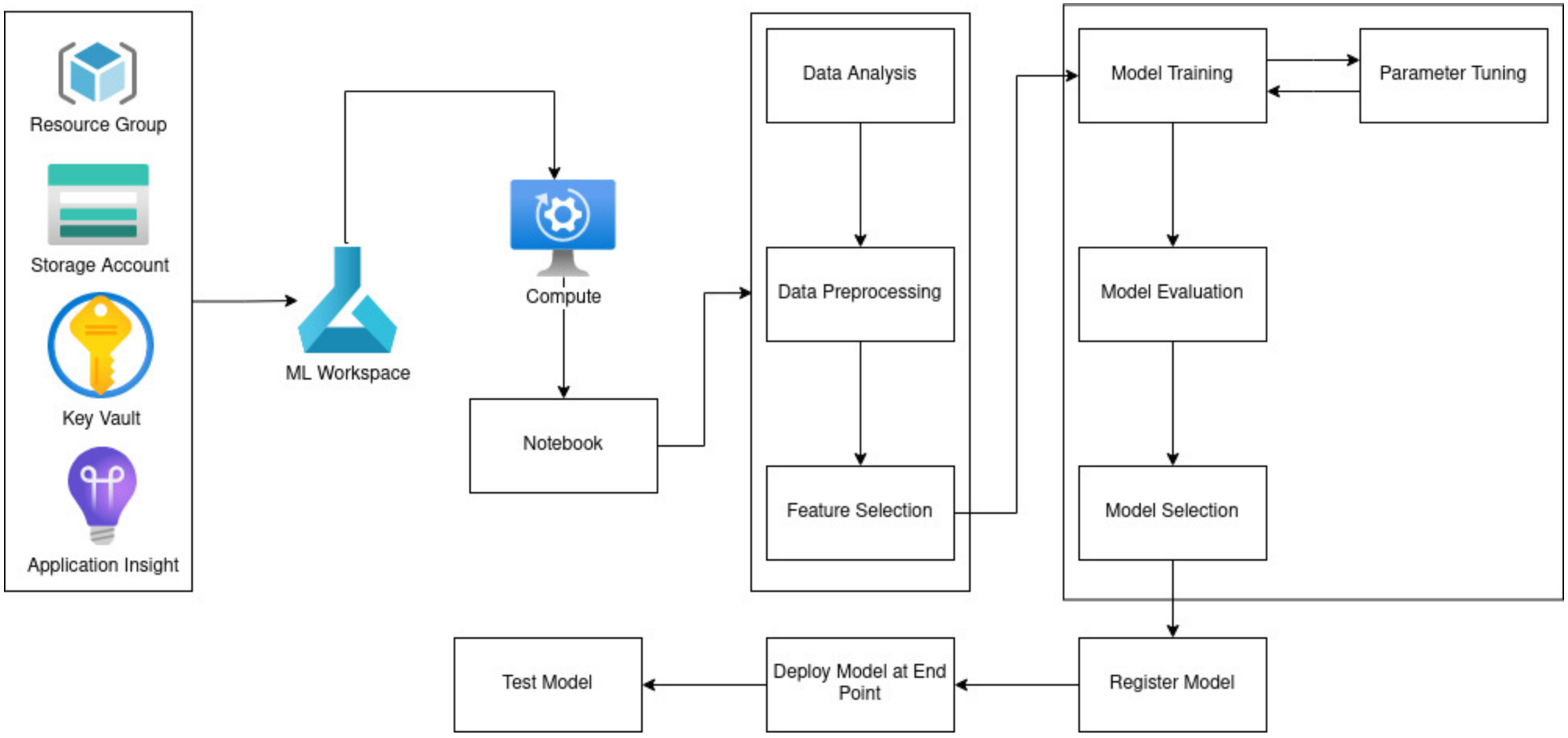
- Numerical features

  Simple Imputer: Replace missing values with mean

  StandardScaler

- Train-Test Split: 85/15

Architectural Diagram



Classifiers Used

- KNeighborsClassifier
- DecisionTreeClassifier
- RandomForestClassifier
- GradientBoostingClassifier
- AdaBoostClassifier
- LogisticRegression
- SVM

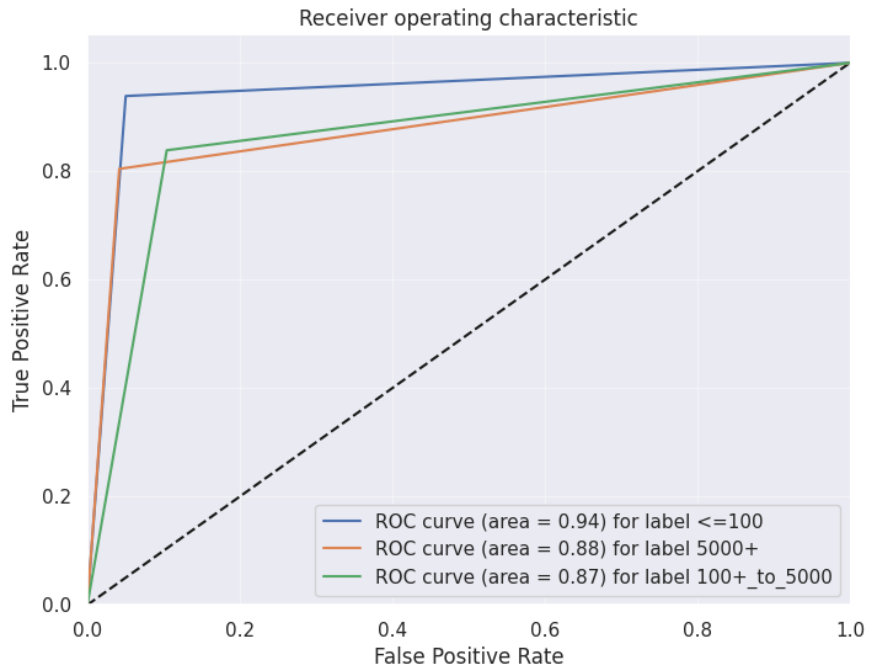
GridSearchCV for hyperparameter tuning

SVM

Default Accurancy = 0.75

```
param_grid = {
    'model__C':[0.1,1,100,1000],
    'model__kernel':['rbf','poly','sigmoid','linear'],
    'model__degree':[1,2,3,4,5,6]
}
```

```
{'model__C': 100, 'model__degree': 1, 'model__kernel': 'poly'}
Accuracy Score 0.8734177215189873
Balanced Accuracy 0.8606110045701311
F1 Score 0.8734177215189873
              precision    recall  f1-score   support

100+_to_5000       0.84      0.84      0.84        93
       5000+       0.82      0.80      0.81        46
       <=100       0.93      0.94      0.93        98

    accuracy                           0.87       237
   macro avg       0.86      0.86      0.86       237
weighted avg       0.87      0.87      0.87       237
```

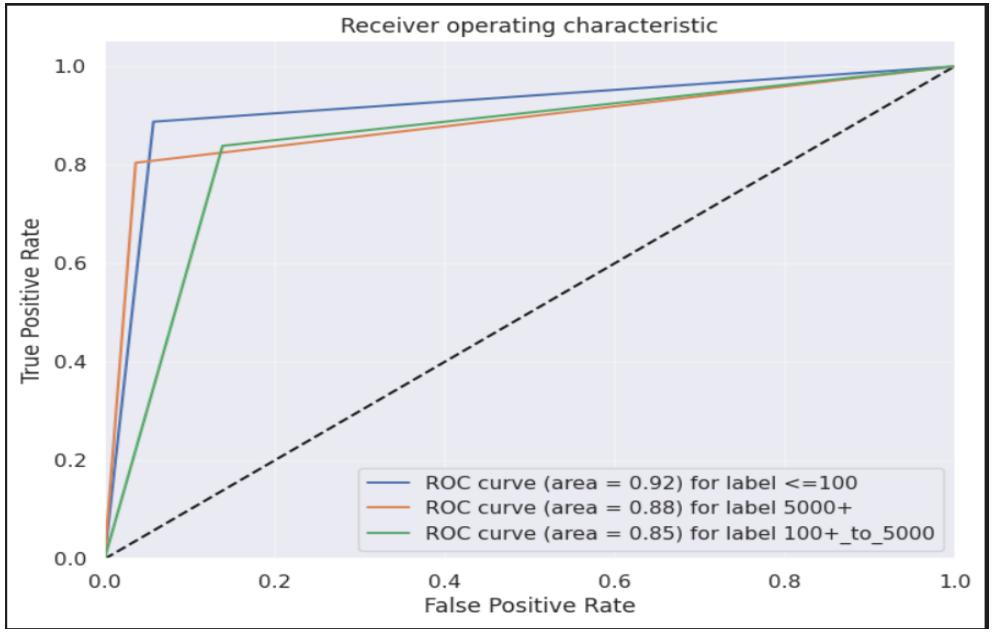


Decision Tree

- default accuracy is 0.8818565400843882

```
param_grid = {
    'model__criterion': ['gini','entropy'],
    'model__splitter': ['best','random'],
    'model__max_depth': [2, 3, 5, 10, 20],
    'model__min_samples_leaf': [5, 10, 20, 50, 100],
    'model__min_samples_split':[8,10,12,18,20,16]
}
```

```
{'model__criterion': 'gini', 'model__max_depth': 5, 'model__min_samples_leaf': 10, 'model__min_samples_split': 20, 'model__splitter': 'best'}
Accuracy Score 0.8523206751054853
Balanced Accuracy 0.8436042018490427
F1 Score 0.8523206751054853
              precision    recall  f1-score   support

100+_to_5000       0.80      0.84      0.82        93
       5000+       0.84      0.80      0.82        46
       <=100       0.92      0.89      0.90        98

    accuracy                           0.85       237
   macro avg       0.85      0.84      0.85       237
weighted avg       0.85      0.85      0.85       237
```

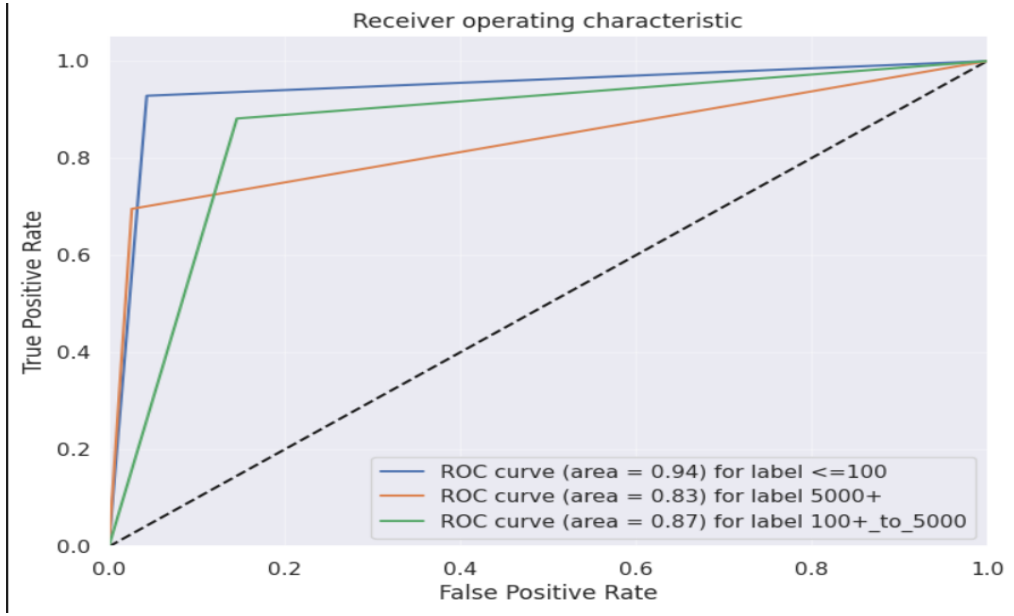


Random Forest

- default accuracy is 0.827004219409282 7

```
param_grid = {
    'model__n_estimators' : [5,20,50,100],
    'model__criterion': ['gini','entropy'],
    'model__max_features': ['auto','sqrt','log2'],
}
```

```
{'model__criterion': 'gini', 'model__max_features': 'auto', 'model__n_estimators': 100}
Accuracy Score 0.8649789029535865
Balanced Accuracy 0.8353146775306662
F1 Score 0.8649789029535865
              precision    recall  f1-score   support

100+_to_5000       0.80      0.88      0.84        93
       5000+       0.86      0.70      0.77        46
       <=100       0.94      0.93      0.93        98

    accuracy                           0.86       237
   macro avg       0.87      0.84      0.85       237
weighted avg       0.87      0.86      0.86       237
```

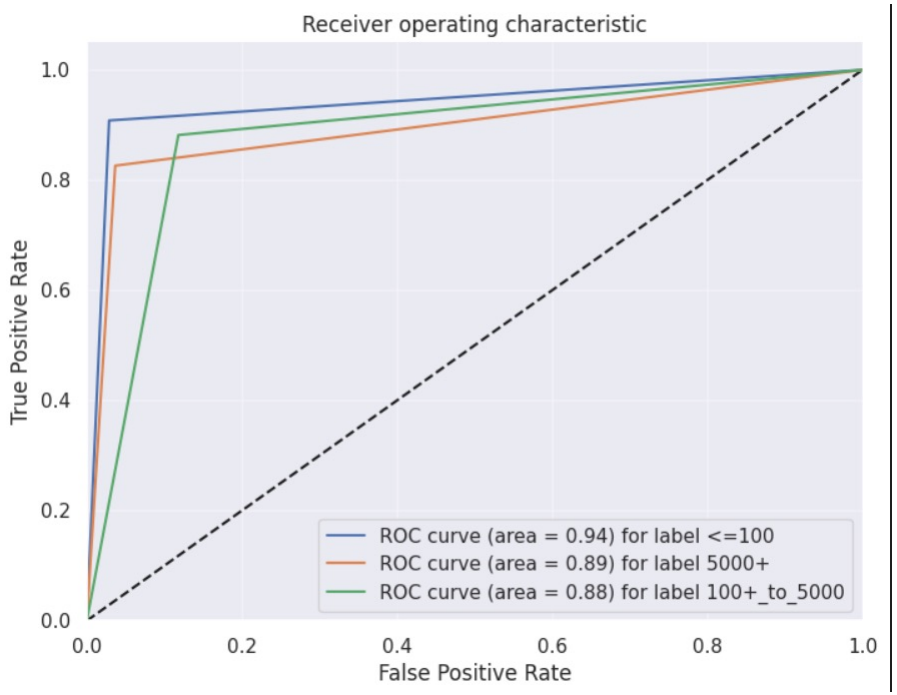


Gradient Boosting Classifier

- default accuracy is 0.8860759493670886

```
param_grid = {
    'model__n_estimators': [90,100,110,120,130],
    'model__max_depth': [1,3,5,7]
}
```

```
{'model__max_depth': 5, 'model__n_estimators': 130}
Accuracy Score 0.8818565400843882
Balanced Accuracy 0.871990217311796
F1 Score 0.8818565400843882
              precision    recall  f1-score   support

100+_to_5000       0.83      0.88      0.85        93
       5000+       0.84      0.83      0.84        46
       <=100       0.96      0.91      0.93        98

    accuracy                           0.88       237
   macro avg       0.88      0.87      0.87       237
weighted avg       0.88      0.88      0.88       237
```

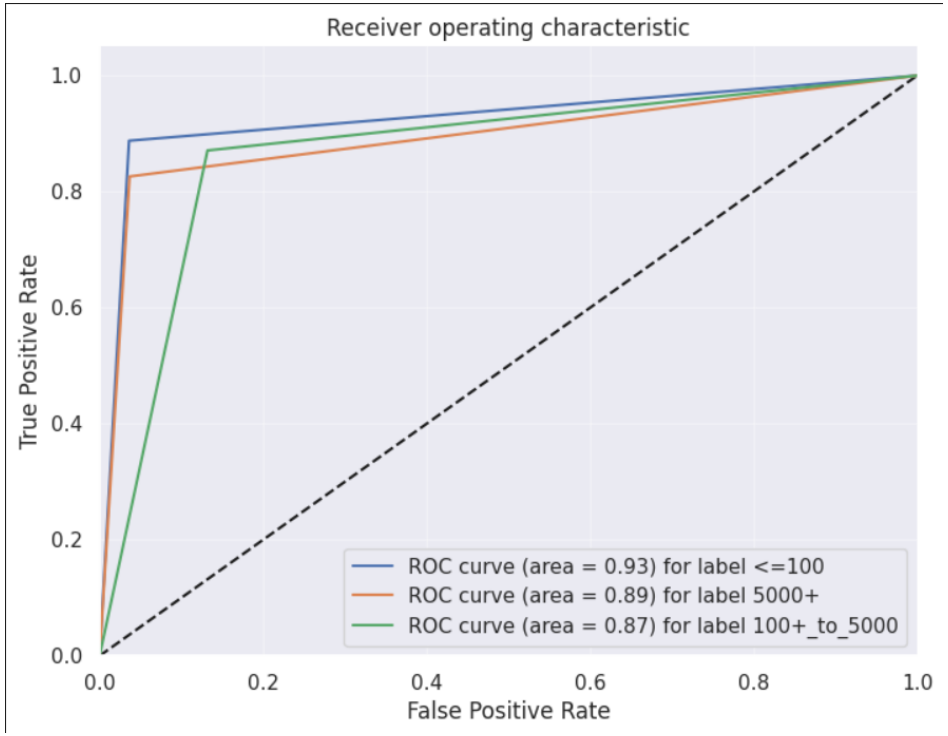


AdaBoost Classifier

- default accuracy is 0.645569620253164

```
param_grid = {
    'model__n_estimators': [3,5,7,9,11,15],
    'model__learning_rate': [0.01,0.1],
    'model__algorithm': ['SAMME','SAMME.R']
}
```

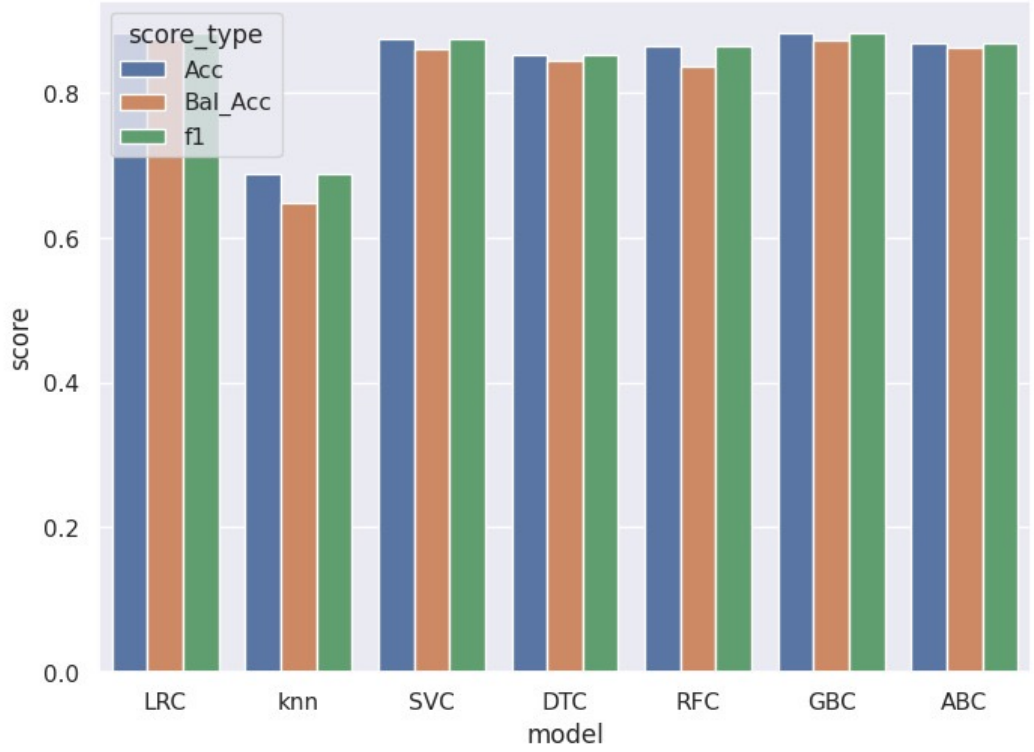
```
{'model__algorithm': 'SAMME.R', 'model__learning_rate': 0.1, 'model__n_estimators': 3}
Accuracy Score 0.869198312236287
Balanced Accuracy 0.8616032668326797
F1 Score 0.869198312236287
              precision    recall  f1-score   support

100+_to_5000       0.81      0.87      0.84        93
       5000+       0.84      0.83      0.84        46
       <=100       0.95      0.89      0.92        98

    accuracy                           0.87       237
   macro avg       0.87      0.86      0.86       237
weighted avg       0.87      0.87      0.87       237
```



Gradient Boosting Classifier

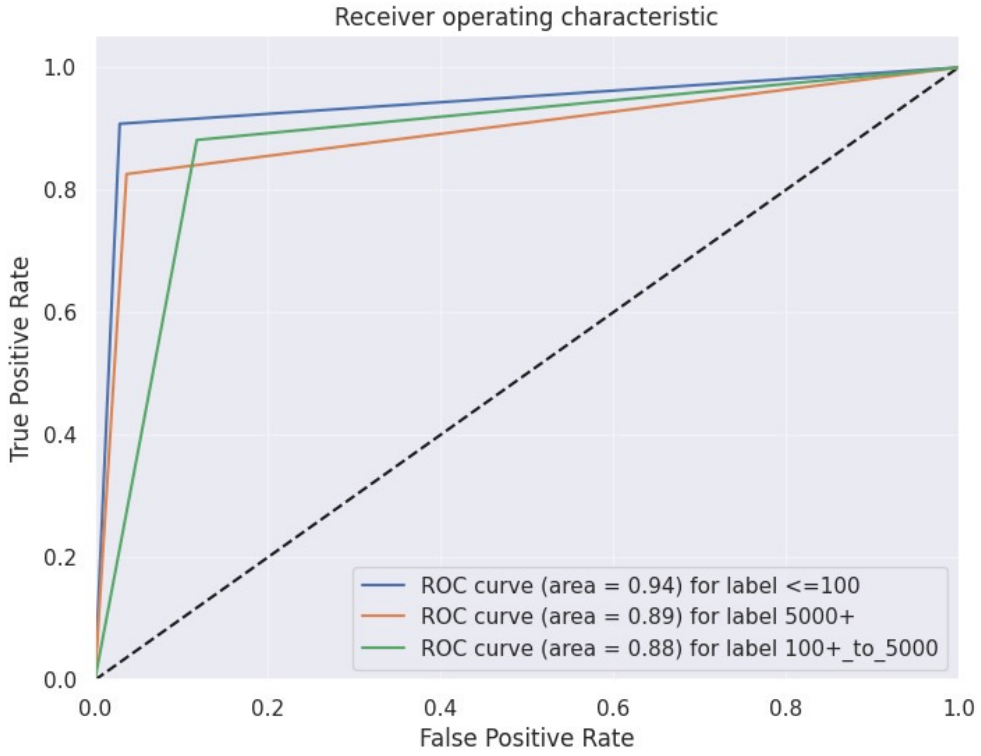
- **The best performing classifier on this dataset**

```
{'model__max_depth': 7, 'model__n_estimators': 120}
Accuracy Score 0.885593220338983
Balanced Accuracy 0.8749632500012049
F1 Score 0.885593220338983
              precision    recall  f1-score   support

100+_to_5000       0.84      0.87      0.86        93
       5000+       0.88      0.83      0.85        46
       <=100       0.93      0.93      0.93        97

    accuracy                           0.89       236
   macro avg       0.89      0.87      0.88       236
weighted avg       0.89      0.89      0.89       236
```

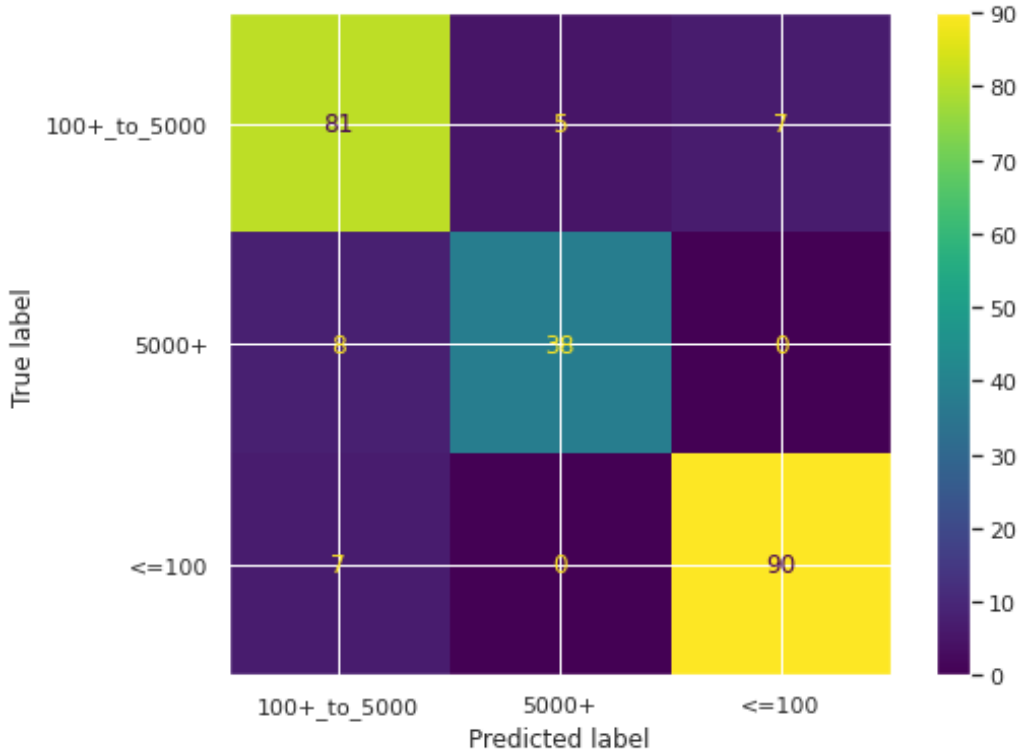


Model Results

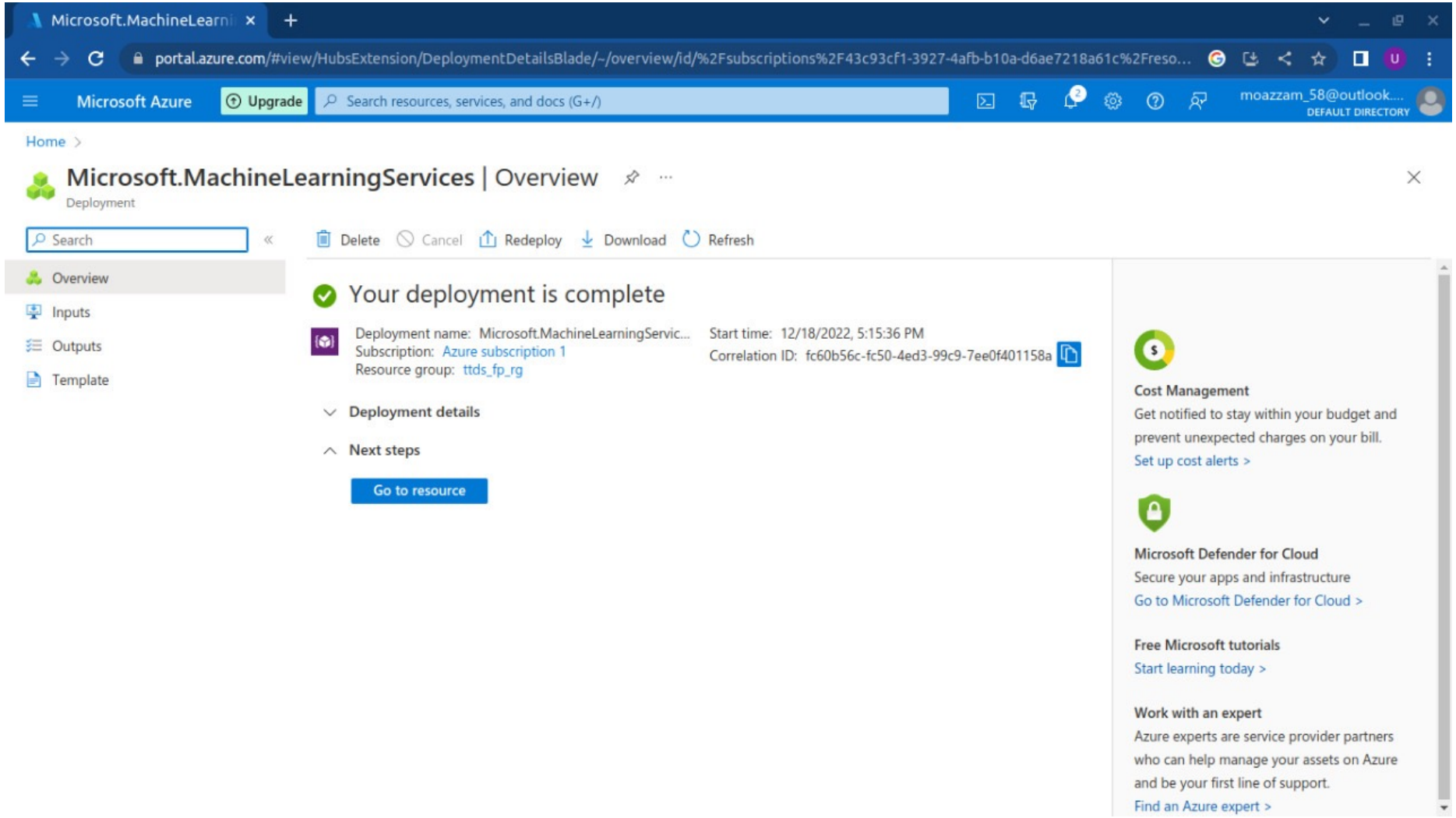
ROC Curve



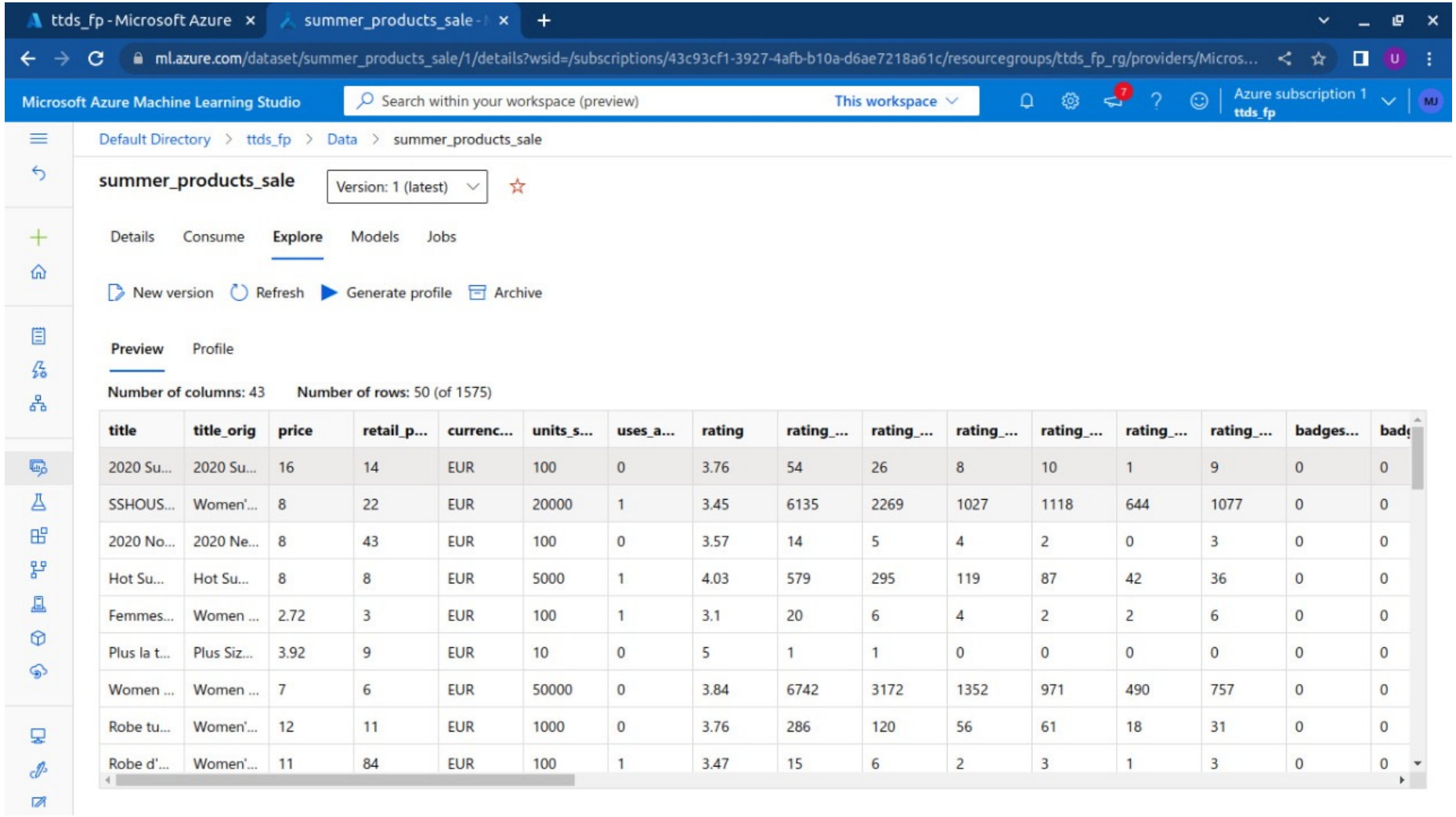
Confusion Matrix



Microsoft Azure

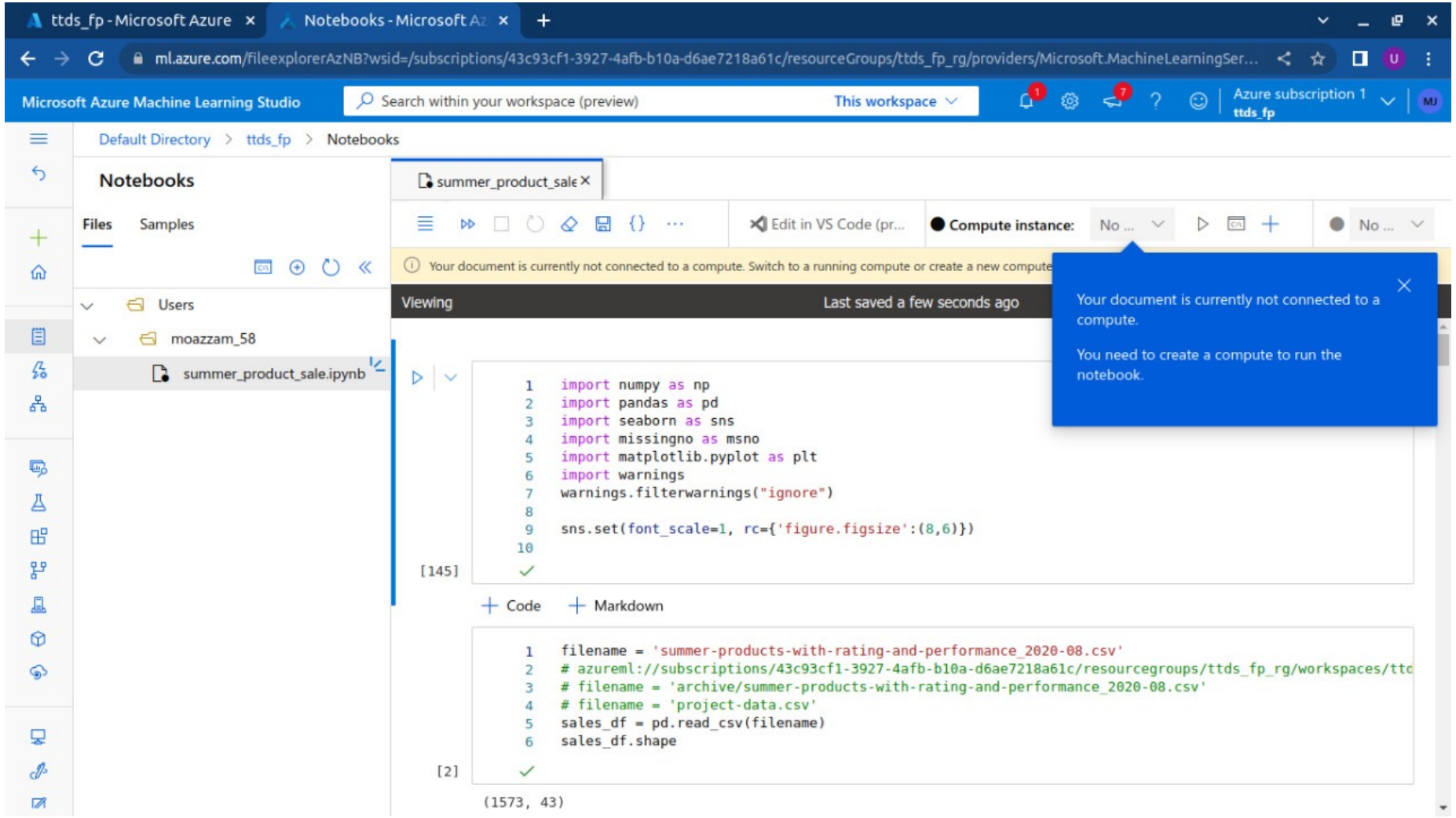


Microsoft Azure



title	title_orig	price	retail_p...	currenc...	units_s...	uses_a...	rating	rating_...	rating_...	rating_...	rating_...	rating_...	rating_...	badges...	bad
2020 Su...	2020 Su...	16	14	EUR	100	0	3.76	54	26	8	10	1	9	0	0
SSHOUS...	Women'...	8	22	EUR	20000	1	3.45	6135	2269	1027	1118	644	1077	0	0
2020 No...	2020 Ne...	8	43	EUR	100	0	3.57	14	5	4	2	0	3	0	0
Hot Su...	Hot Su...	8	8	EUR	5000	1	4.03	579	295	119	87	42	36	0	0
Femmes...	Women ...	2.72	3	EUR	100	1	3.1	20	6	4	2	2	6	0	0
Plus la t...	Plus Siz...	3.92	9	EUR	10	0	5	1	1	0	0	0	0	0	0
Women ...	Women ...	7	6	EUR	50000	0	3.84	6742	3172	1352	971	490	757	0	0
Robe tu...	Women'...	12	11	EUR	1000	0	3.76	286	120	56	61	18	31	0	0
Robe d'...	Women'...	11	84	EUR	100	1	3.47	15	6	2	3	1	3	0	0

Microsoft Azure



Thanks