



Airline



Vueling Tech Hack



Guillermo Seoane | IT academy [February 2023]

BOARDING PASS

● FLIGHT

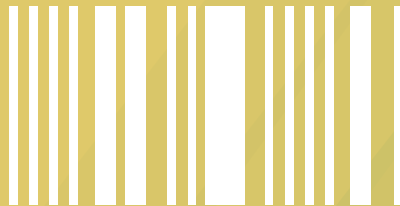
B345

● GATE

D8

● SEAT

29E





Airline



PLAN YOUR JOURNEY

Cleaning and transforming of the data used by the machine learning algorithm

PREPROCESSING

Best fit model for prediction, finding the right algorithm

MODEL EVALUATION

Confirm results on Test set

PREDICTION

TRAIN/VALID

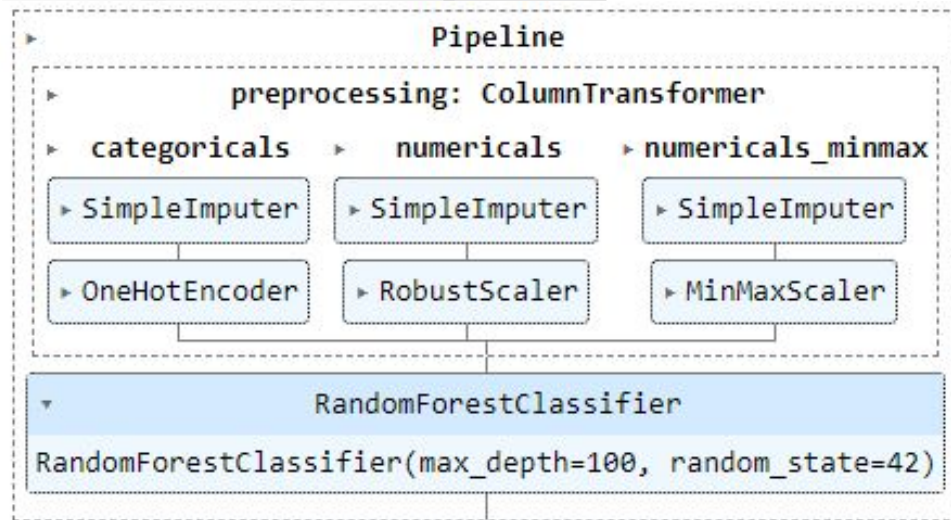
Evaluating Model Performance Using Validation Dataset

HYPER-PARAMETERS

Evaluating the performance of a model with GridSearch



Pipeline



| categoricals

```
['Origin Country']
```

| numericals

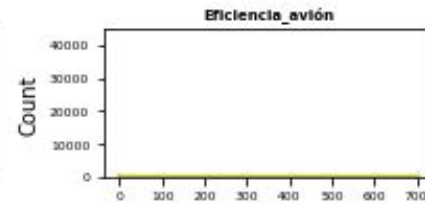
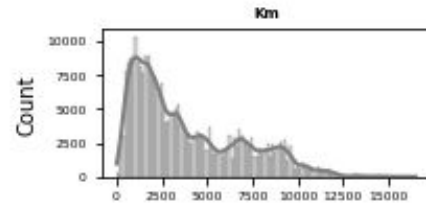
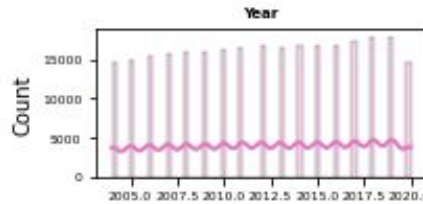
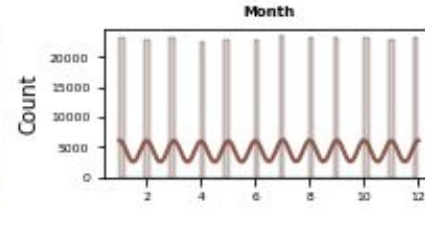
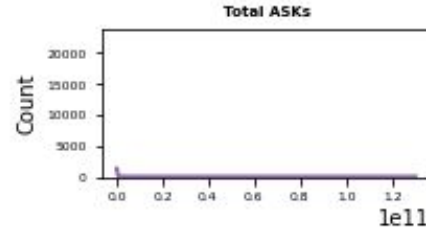
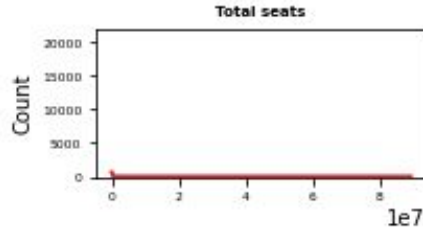
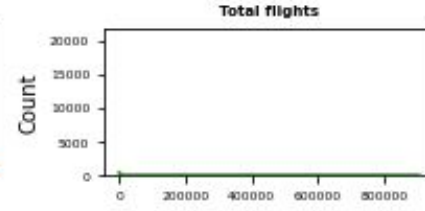
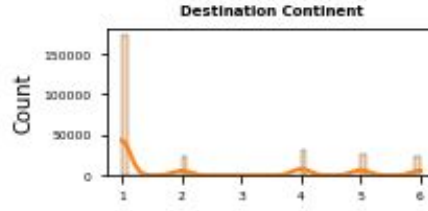
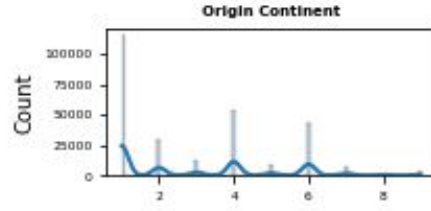
```
['Total flights', 'Total  
seats', 'Total ASKs',  
'Km', 'Eficiencia']
```

| numericals_minmax

```
['Year', 'Origin  
Continent',  
'Destination  
Continent']
```

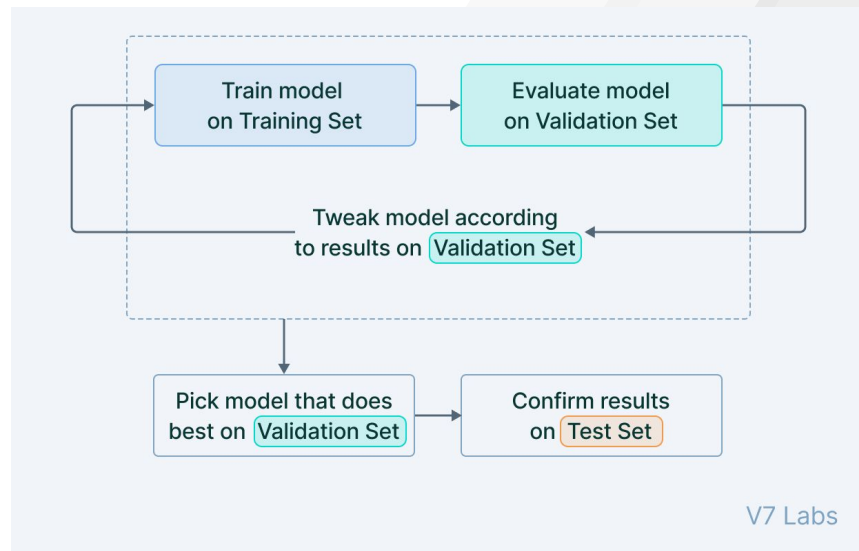


Distribución variables numéricas





Train/Valid



*statistical approach to deal with Sampling

- Evaluation with Cross-Validation

Model	f1_cv	f1_std_cv
knc	0.78189	0.00270
dtc	0.91649	0.00388
rfc	0.91302	0.00372
lr	0.06016	0.00393

```
knc = KNeighborsClassifier()  
dtc = DecisionTreeClassifier()  
rfc = RandomForestClassifier()  
lr = LogisticRegression()
```



Hyper Parameters (GridSearchCV)

```
pipeline = Pipeline(  
    [  
        ('preprocessing', preprocessor),  
        ('model', RandomForestClassifier(random_state = 42))  
    ]  
)  
  
params = {  
    'model__criterion': ('gini', 'entropy'),  
    'model__max_depth': [0.1, 10, 100]  
}  
  
rskf = RepeatedStratifiedKFold(n_splits = 5, n_repeats = 2, random_state = 42)  
cv = GridSearchCV(pipeline, params, cv = rskf, scoring = ['f1_macro'], refit = 'f1_macro')  
  
cv.fit(X, y)  
  
print(f'Best F1-score: {cv.best_score_:.3f}\n')  
print(f'Best parameter set: {cv.best_params_}\n')  
print(f'Scores: {classification_report(y, cv.predict(X))}')
```

- Cross validation

```
from sklearn.metrics import f1_score  
  
pipeline = Pipeline(  
    [  
        ('preprocessing', preprocessor),  
        ('model', RandomForestClassifier(criterion = 'gini',  
                                         max_depth = 100,  
                                         random_state = 42))  
    ]  
)  
  
score = cross_val_score(pipeline, X, y, cv=5, scoring='f1_macro')  
print('F1 score: {0:.2f}'.format(score.mean()))
```

F1 score: 0.91



Prediction

```
# Define the target variable and the features
X_train = df_train.drop(['Destination Country'], axis=1) #features
y_train = df_train['Destination Country'] #target

X_test = df_test
```

- Train/Test division

```
rfc = RandomForestClassifier(criterion = 'gini', max_depth = 100, random_state = 42)
```

- Create a Model

```
## Pipeline
pipeline = Pipeline([
    ('preprocessing', preprocessor),
    ('model', rfc)
])
```

- Pipeline & ColumnTransformer

```
pipeline.fit(X_train, y_train)
```

```
y_pred = pipeline.predict(X_test)
```

```
df_prediccioness = pd.DataFrame({'target': y_pred})
```

- Prediction



Airline



THANKS!



NOTE:

Do you have any questions?
seoaneg@gmail.com
www.linkedin.com/in/guilleseoane/
IT Academy de Barcelona Activa



BOARDING PASS

● FLIGHT

B345

● GATE

D8

● SEAT

29E

