AAVAIL

Monthly Revenue Modelling

William Maia
March/2021

Hypothesis To Test

Model Hypothesis

Timeseries for revenue is stationary

Revenue mean between months is statistically similar

Timeseries for revenue is seasonal

Business Hypothesis

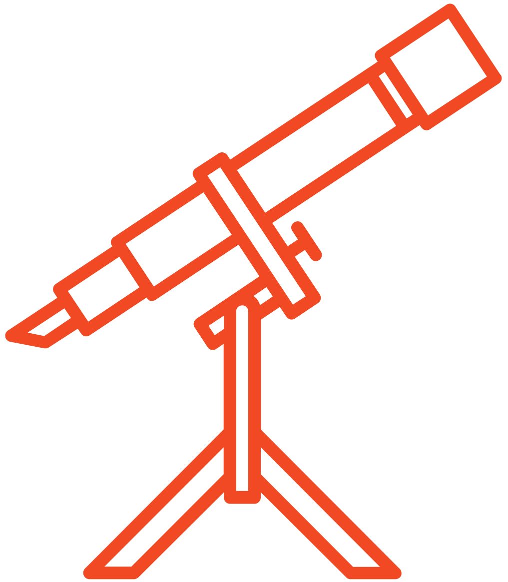
Can a Machine Learning model perform better than the managers custom methods?

Does customers behavior vary across countries, considering revenue?

EDA

Summary

- Training Dataset containing 815011 records and 9 features
- Number of different countries contained in the dataset is 43
- Number of different dates covered by the dataset is 495
- Generated Feature Invoice_date concatenating Day + Month + Year



Data Sample

	country	customer_id	day	invoice	month	price	stream_id	times_viewed	year	invoice_date
0	United Kingdom	13085.0	28	489434	11	6.95	85048	12	2017	2017-11-28
1	United Kingdom	13085.0	28	489434	11	6.75	79323W	12	2017	2017-11-28
2	United Kingdom	13085.0	28	489434	11	2.10	22041	21	2017	2017-11-28
3	United Kingdom	13085.0	28	489434	11	1.25	21232	5	2017	2017-11-28
4	United Kingdom	13085.0	28	489434	11	1.65	22064	17	2017	2017-11-28
5	United Kingdom	13085.0	28	489434	11	1.25	21871	14	2017	2017-11-28
6	United Kingdom	13085.0	28	489434	11	5.95	21523	10	2017	2017-11-28
7	United Kingdom	13085.0	28	489435	11	2.55	22350	12	2017	2017-11-28
8	United Kingdom	13085.0	28	489435	11	3.75	22349	12	2017	2017-11-28
9	United Kingdom	13085.0	28	489435	11	1.65	22195	18	2017	2017-11-28

Missing Data

- Feature Invoice_id containing 23.3% of Missing Data

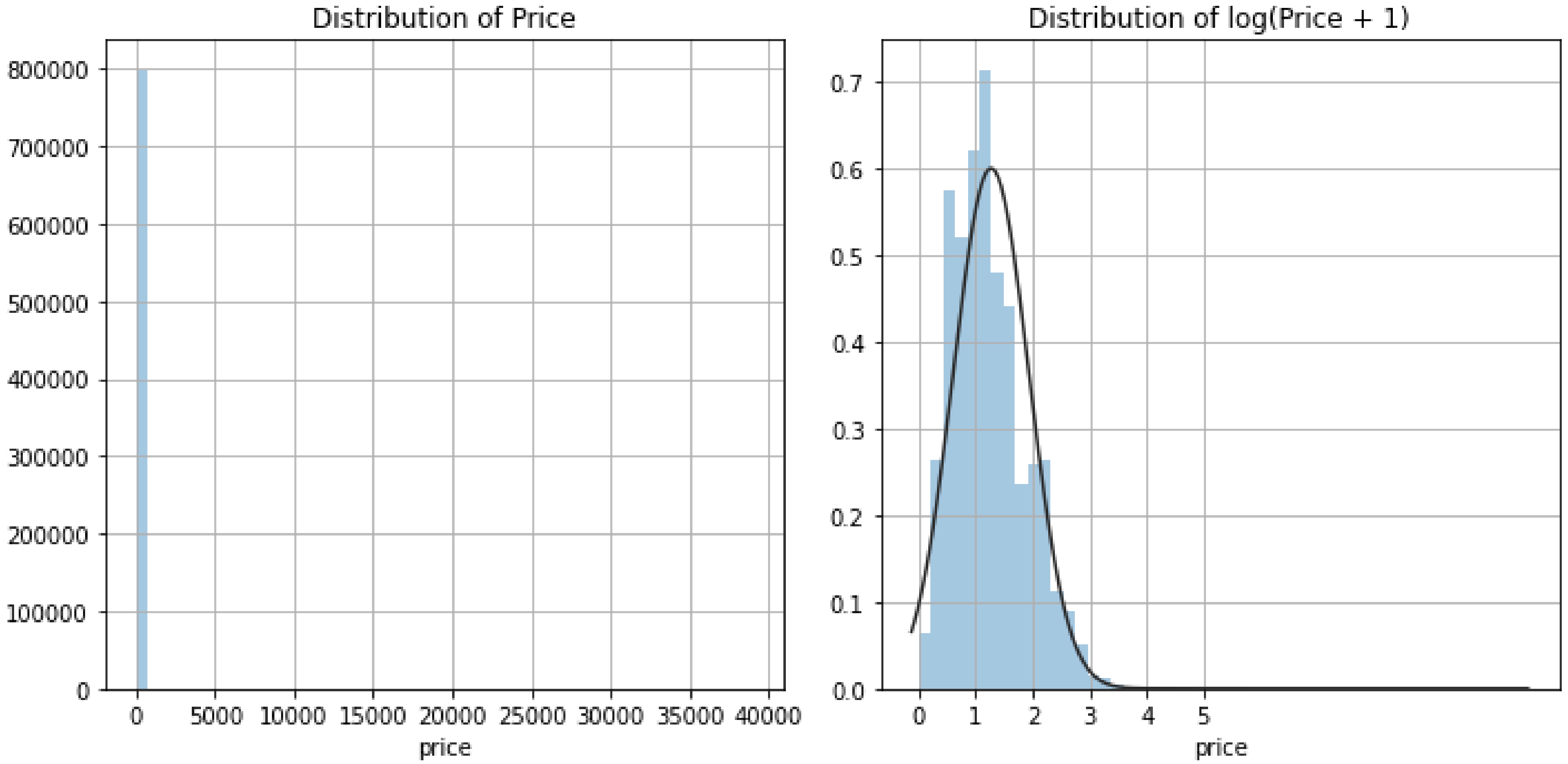
	Zero Values	Missing Values	% of Total Values	Total Zero Missing Values	% Total Zero Missing Values	Data Type
customer_id	0	189762	23.3	189762	23.3	float64

Top 10 Countries with Higher Revenues

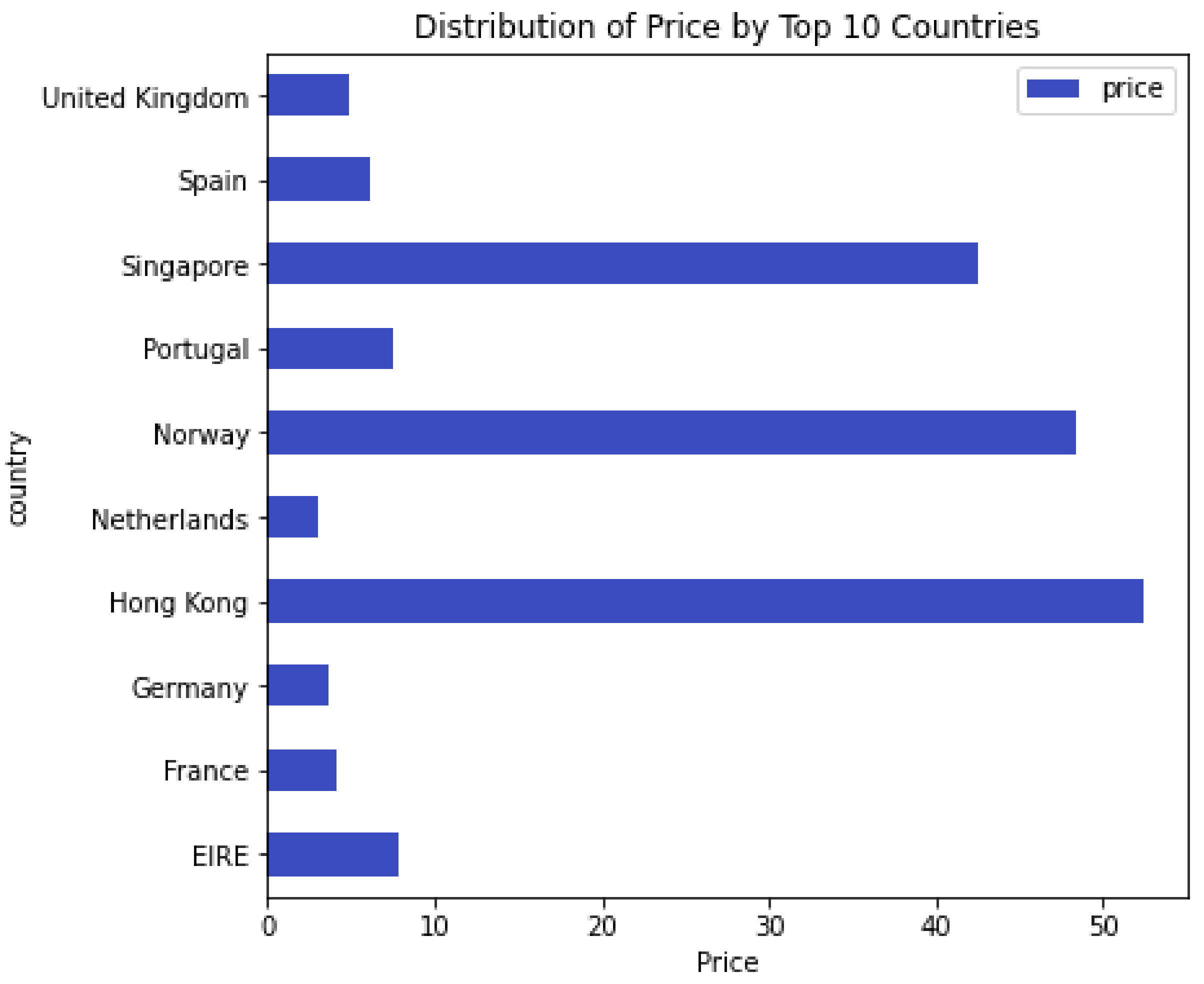
	revenue
country	
United Kingdom	3658065.525005765
EIRE	107069.20999999279
Germany	49271.820999998
France	40565.13999999977
Norway	38494.74999999956
Spain	16040.990000000262
Hong Kong	14452.57000000003
Portugal	13528.669999999951
Singapore	13175.92000000001
Netherlands	12322.800000000087

Price Distribution

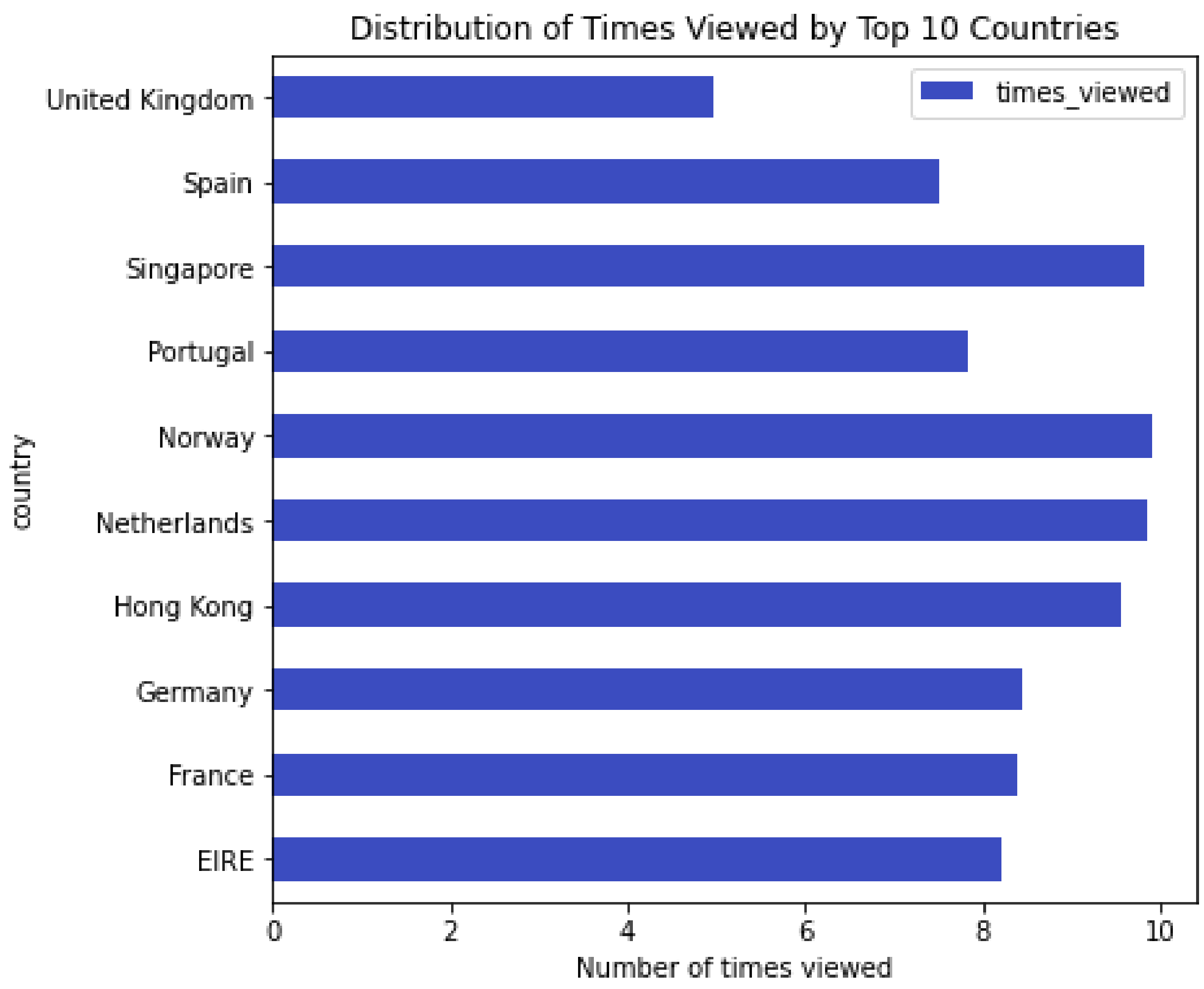
- In order to avoid high dispersion we have Applied a log transformation to the Price feature. Below we can see its benefits.



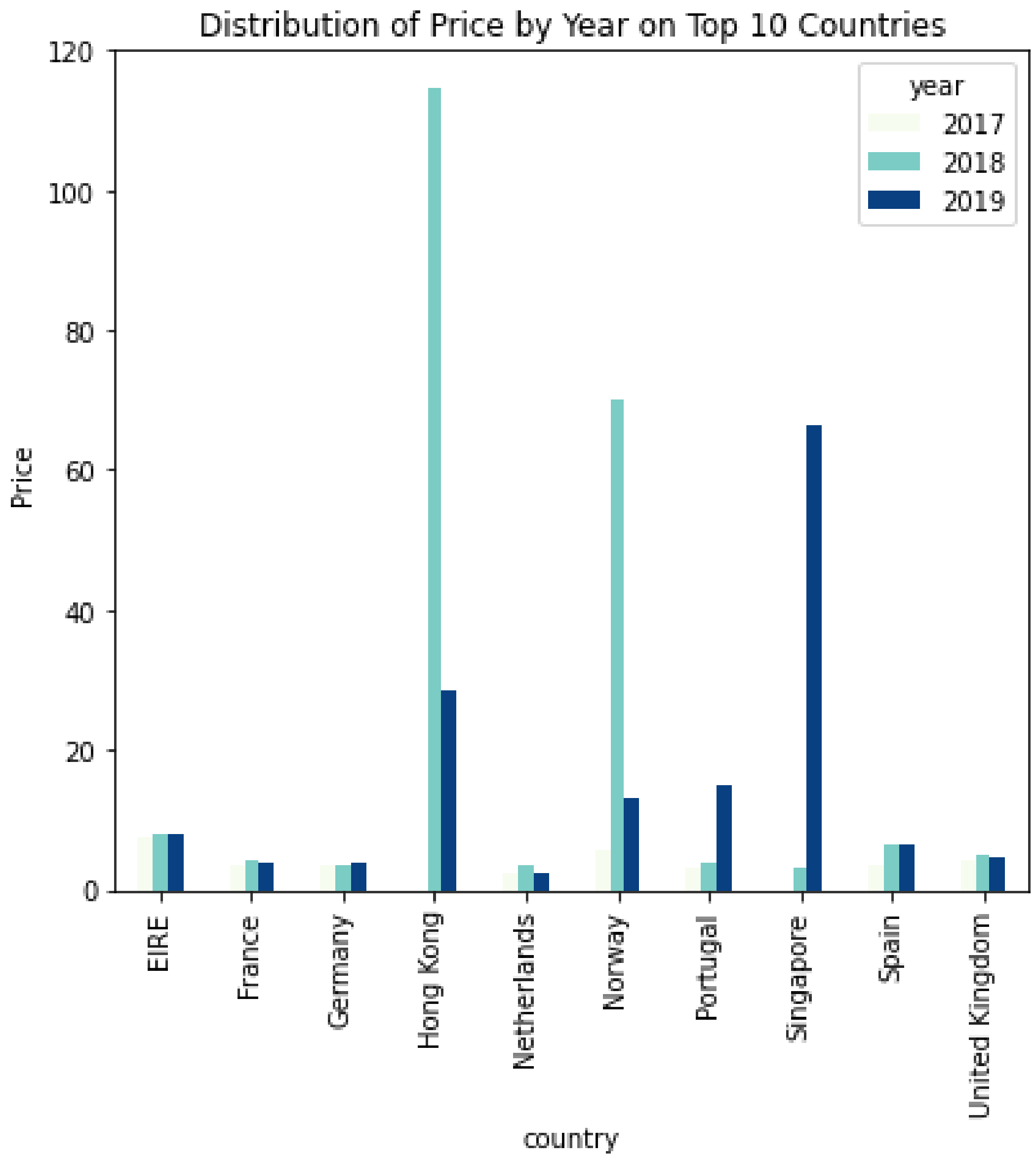
Distribution of Price by Top 10 Countries



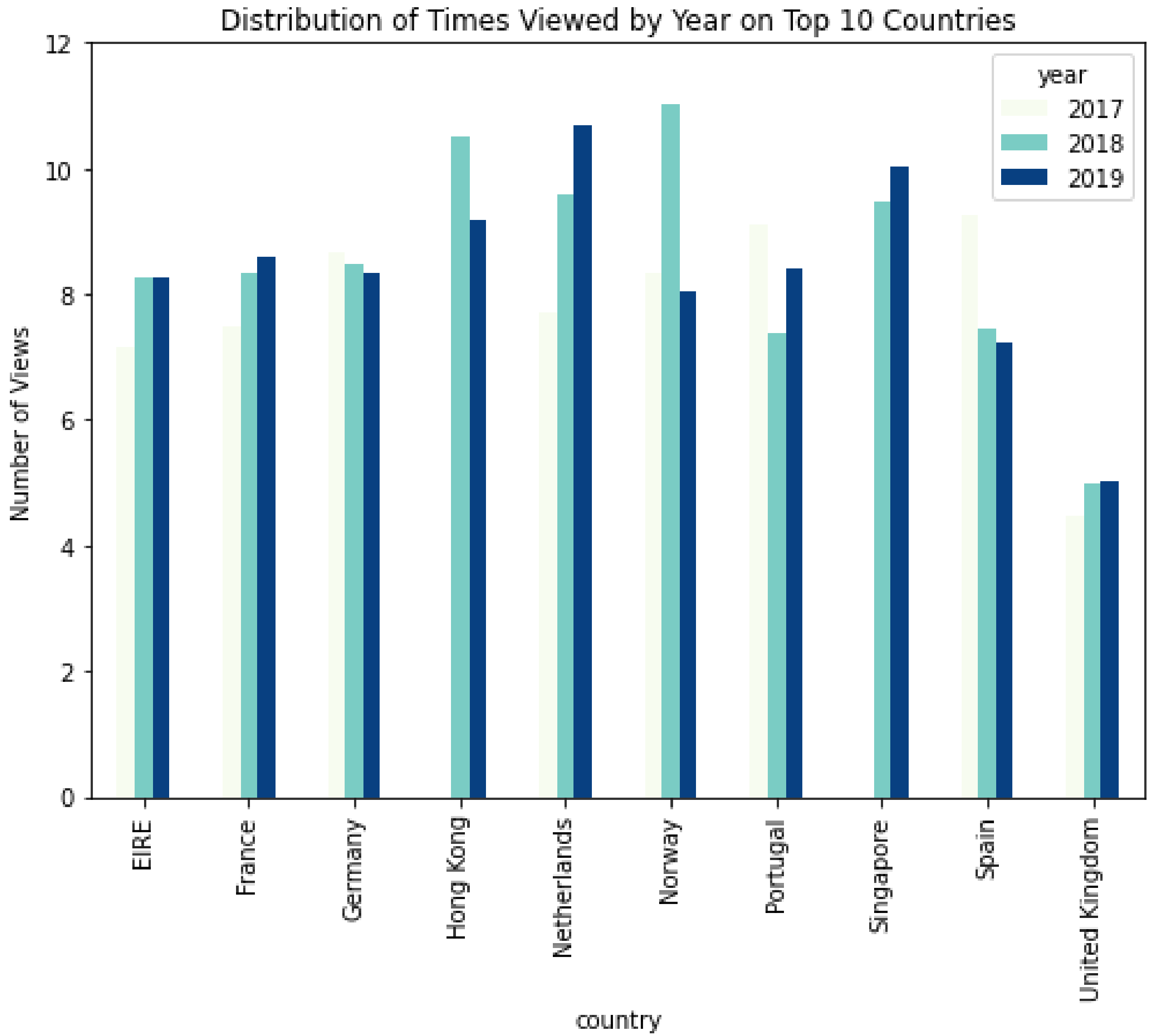
Distribution of Times_Viewed by Top 10 Countries



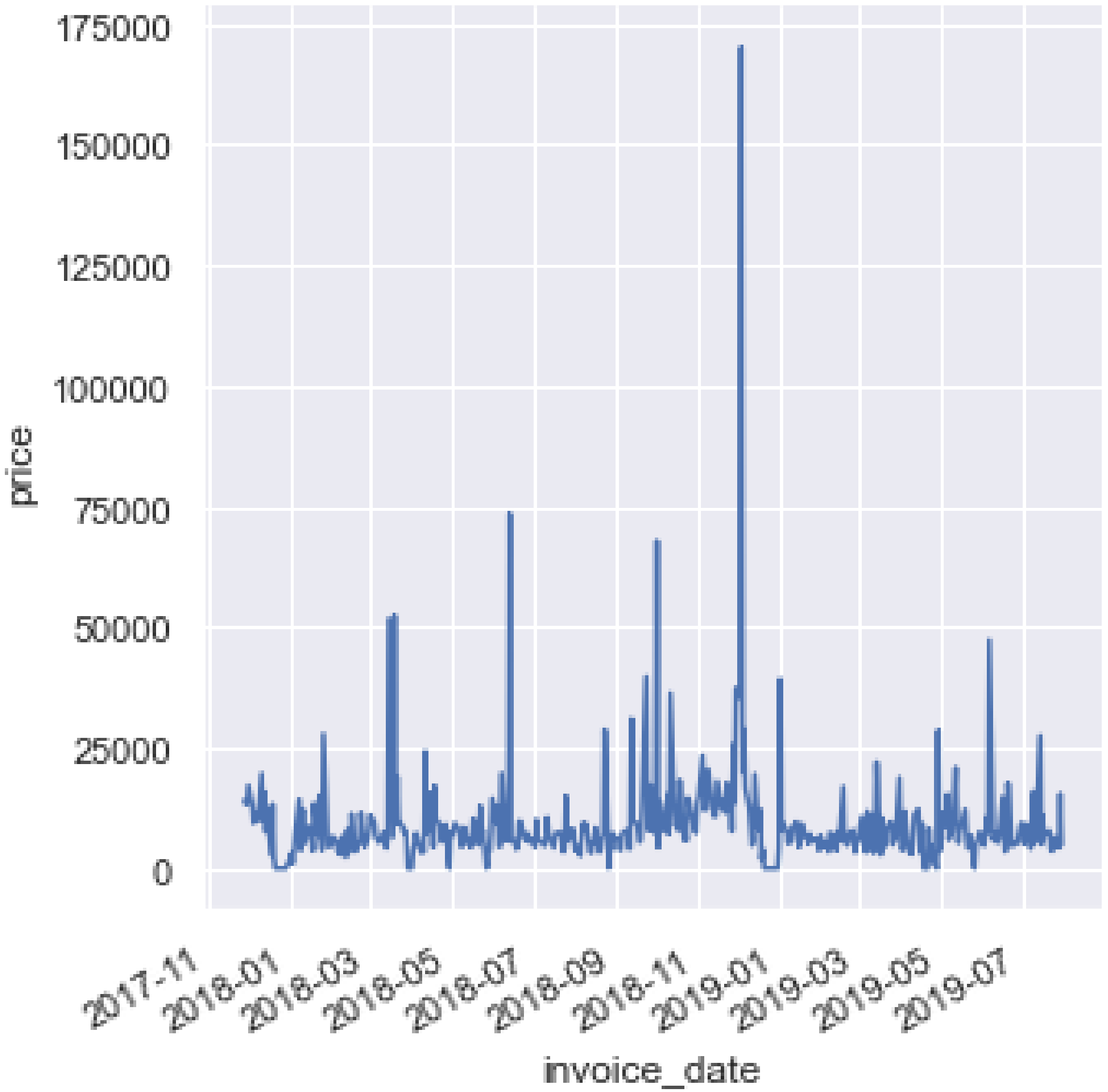
Distribution of Price by Year on Top 10 Countries



Distribution of Times Viewed by Year on Top 10 Countries



Price Behavior Through Invoice Date



Missing Data

- In order to carry out a time series analysis, record of each day should be considered and the dataframe should be in a chronological order so that forecasting models can fit and provide revenue. For example, price for the following month.

	invoice_date	times_viewed	price	country
0	2017-11-28	14948	14139.140000000087	United Kingdom
1	2017-11-29	14135	13396.920000000135	United Kingdom
2	2017-11-30	15560	13250.070000000167	United Kingdom
3	2017-12-01	12180	9517.350000000051	United Kingdom
4	2017-12-02	3101	1263.2800000000032	United Kingdom
5	2017-12-03	8421	6354.689999999959	Germany
6	2017-12-04	12350	13023.360000000022	United Kingdom
7	2017-12-05	12474	9358.970000000025	United Kingdom
8	2017-12-06	10493	11263.690000000137	United Kingdom
9	2017-12-07	11688	10816.890000000127	United Kingdom

Conclusions

EDA Conclusion

As we were able to see we have a dataset containing revenue information of several countries categorized by date.

The idea is that we can use this dataset to build a model in order to predict the revenue of the following month. It is important to mention that the dataset is imbalanced with respect to countries.

Performing transformations and aggregations we could build a structure based on the dataset in order to be used by a machine learning model. In this case we could think of approaches using Time Series Forecasting or Supervised Learning.

API/Notebooks/AAVAIL_EDA.ipynb

Model Evaluation Conclusion

When evaluating possibilities to choose, there were some approaches available, using Times Series Forecasting (such as Facebook Prophet) or a Supervised Learning approach (such as RandomForestRegressor).

So, in the end, I decided to go with a Supervised Learning Approach. So using GridSearchCV technique some models were evaluated:

- RandomForestRegressor
- GradientBoostingRegressor
- LGBMRegressor
- DecisionTreeRegressor

The one who gave best metrics was RandomForest with:

- Mean Absolute Error = 11002
- Mean Squared Error = 272711018
- r2_score = 0.958

API/Notebooks/AAVAIL_Modelling.ipynb

API Conclusion

After understanding the Bigger Picture we decided to create an API containing some main functionalities, which includes:

- Training EndPoint (Starts model training online)
- Predicting EndPoint (Starts model prediction online)
- Logging EndPoint (Enables Logging Visualization online)

Other than that we have created some batch scripts to validate the funcionalities implemented offline:

- run-model-train.py (Fires model training offline)
- run-test-predict.py (Fires massive model prediction of production dataset offline)
- run-tests.py (Runs Unit Tests implemented offline)

Deploy Conclusion

After developing and testing the API, I decided to encapsulate the solution in a Docker container:

```
# Use an official Python runtime as a parent image
FROM python:3.7.5-stretch

RUN apt-get update && apt-get install -y \
python3-dev \
build-essential

# Set the working directory to /app
WORKDIR /app

# Copy the current directory contents into the container at /app
ADD . /app

# Install any needed packages specified in requirements.txt
RUN pip install --upgrade pip
RUN pip install --no-cache-dir -r requirements.txt

# Define environment variable
ENV NAME World

# Run app.py when the container launches
CMD ["python", "app.py"]
```