Tabular Playground Series – Sept 2022

Objective:

Predicting sales for four items from two competing stores in six different countries is a challenge. Our goal is to find the book sales in 2021.

Inference:

The training data contains 6 columns, [row_id, date, country, store, product, num_sold].

Features of matrix: [row_id, date, country, store, product].

Label: [num_sold].

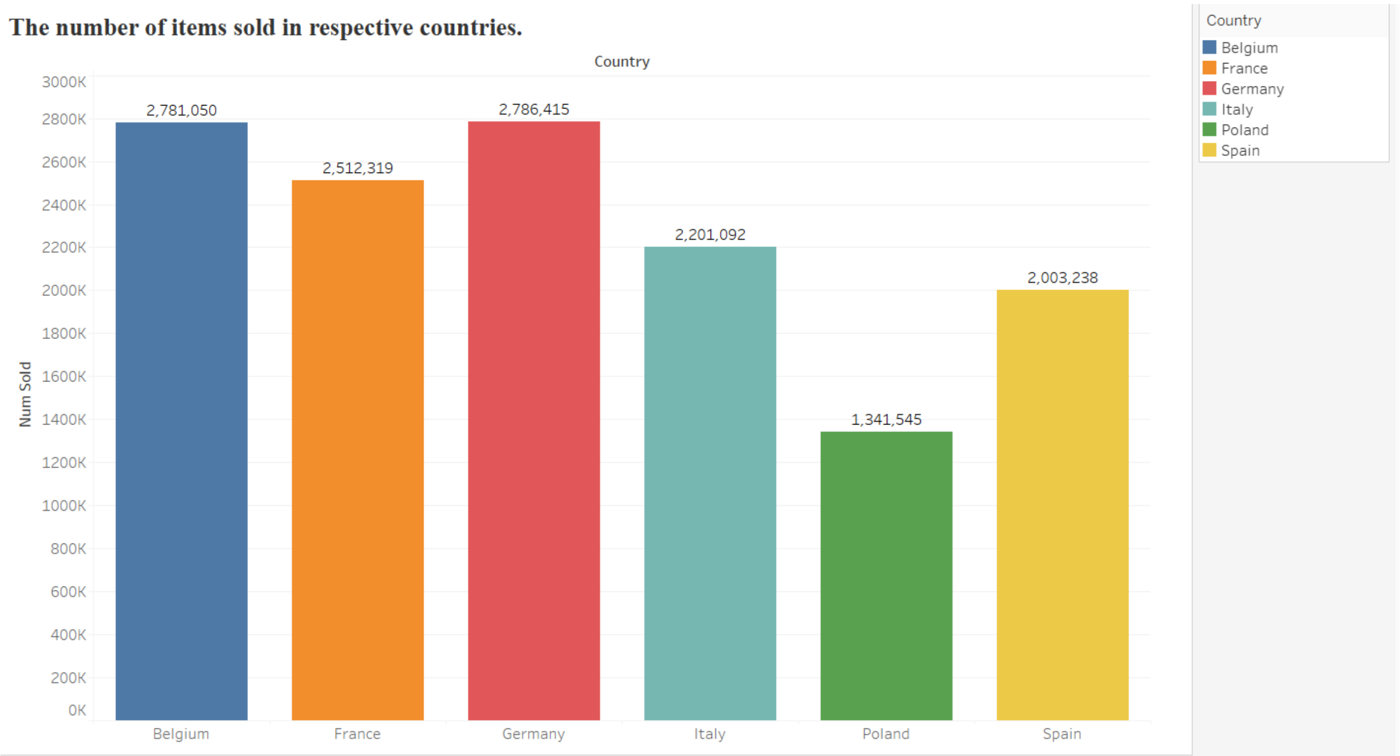
Values:

- row_id – ranges between [0 to 70127]. Increments by 1.
- date – have data from 2017 to 2020. Contains 4 years of data.
- country – [Belgium, France, Germany, Italy, Poland, Spain].
- store – [KaggleMart, KaggleRama].
- Product – [Kaggle Advanced Techniques, Kaggle for Kids: One Smart Goose, Kaggle Getting Started, Kaggle Recipe Book].
- num_sold – minimum: 19, maximum: 986.

The Goal is to predict the num_sold attribute, for each combination of [date, country, store, product] in 2021.

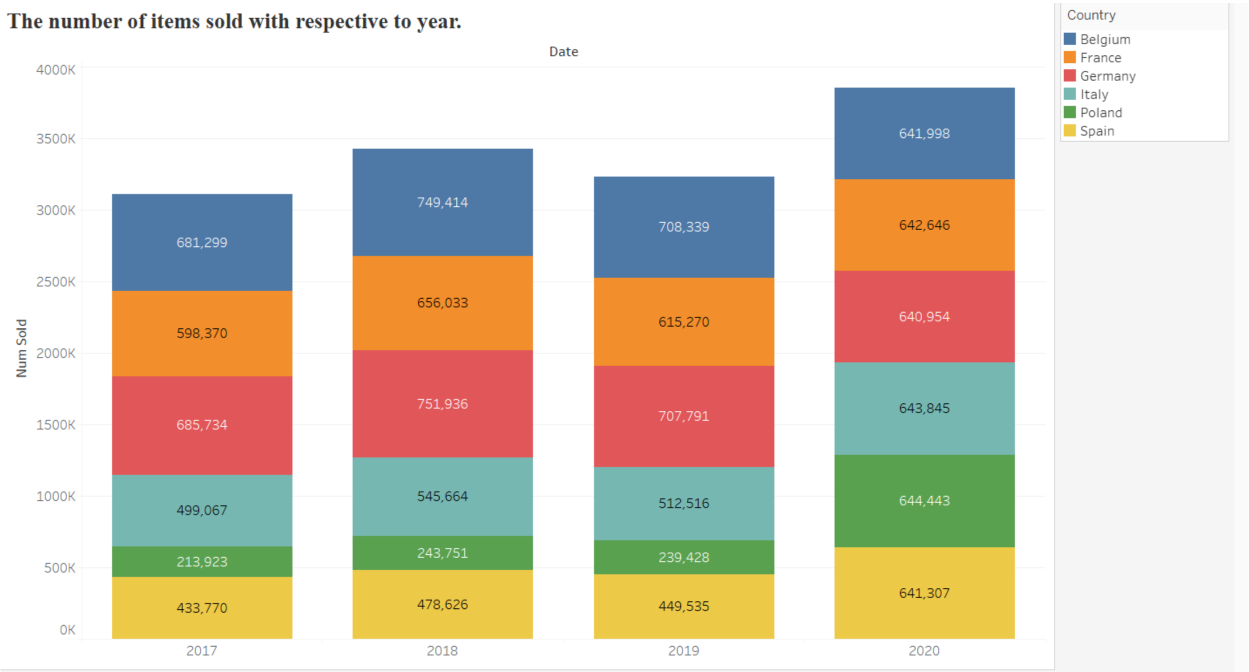
Therefore, total number of combinations = 365 * 6 * 2 * 4 = 17520 Rows.

Visualization Charts of Training Set Using Tableau:



Inference:

- The maximum number of products sold among all the countries: Germany.
- The minimum number of products sold among all the countries: Poland.
- Belgium is second leading seller among six countries, the difference between Belgium and Germany is just **5365** products.



Inference:

- The maximum number of products sold among all the years: 2020.
- The minimum number of products sold among all the years: 2017.
- The products sold in 2017 and 2019 differed slightly.

In 2017,

- The maximum number of products sold among all the countries: Germany.
- The minimum number of products sold among all the countries: Poland.

In 2018,

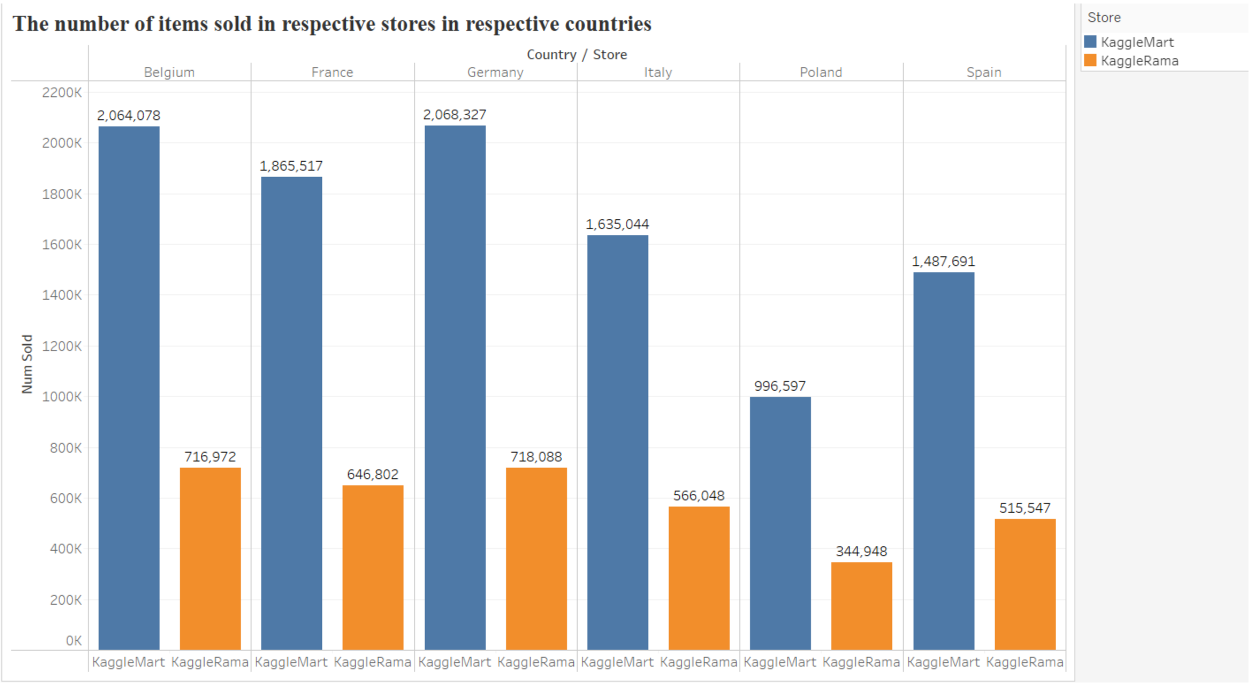
- The maximum number of products sold among all the countries: Germany.
- The minimum number of products sold among all the countries: Poland.

In 2019,

- The maximum number of products sold among all the countries: Belgium.
- The minimum number of products sold among all the countries: Poland.

In 2020,

- The maximum number of products sold among all the countries: Poland.
- The minimum number of products sold among all the countries: Germany.



Inference:

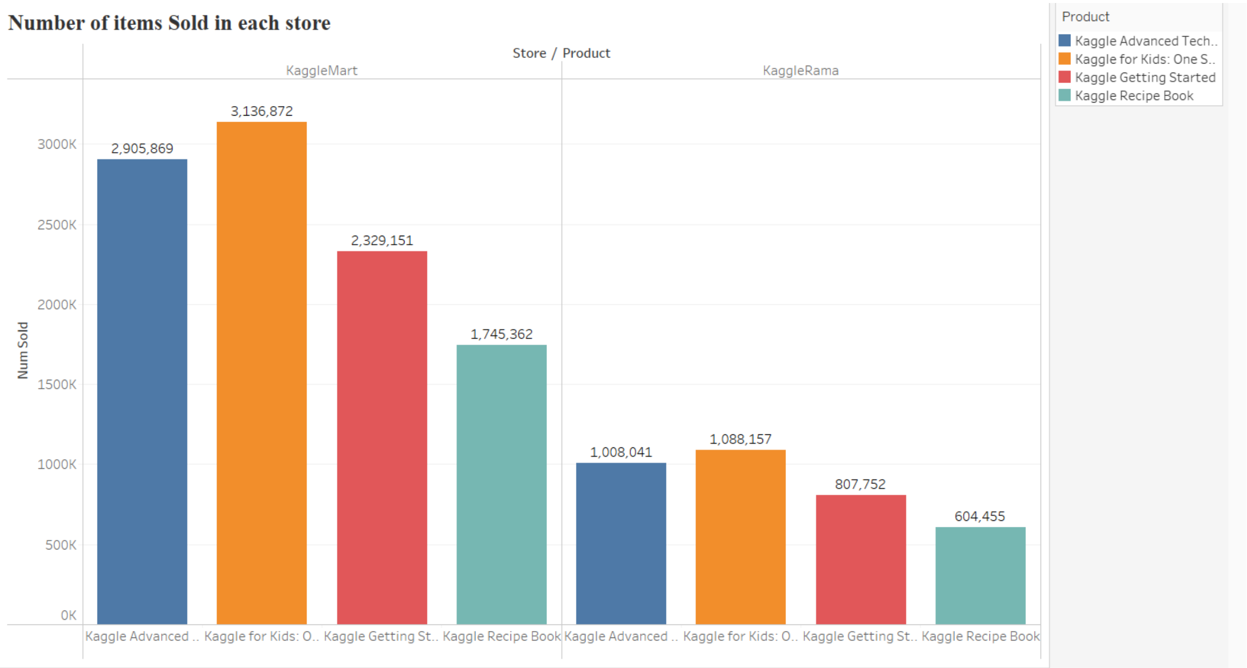
- In all countries KaggleMart sold high number of products.

In KaggleMart,

- The maximum number of products sold among all the countries: Germany.
- The minimum number of products sold among all the countries: Poland.
- The products sold in Germany and Belgium differed slightly.

In KaggleRama,

- The maximum number of products sold among all the countries: Germany.
- The minimum number of products sold among all the countries: Poland.
- The products sold in Italy and Spain differed slightly.



Inference:

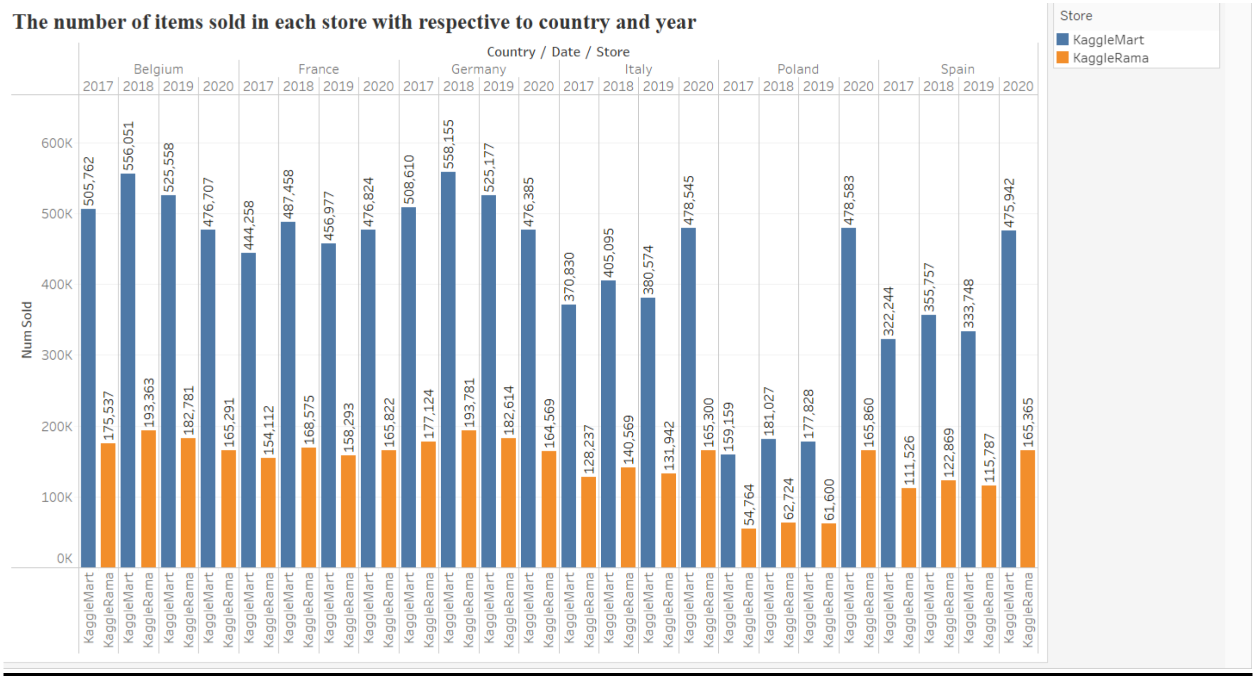
In KaggleMart,

- The maximum number of products sold among all the stores: Kaggle for Kids.
- The minimum number of products sold among all the stores: Kaggle Recipe Book.

In KaggleRama,

- The maximum number of products sold among all the stores: Kaggle for Kids.
- The minimum number of products sold among all the stores: Kaggle Recipe Book.

- Overall, the Maximum number of products sold**: Kaggle for Kids**. And the Minimum number of products sold: **Kaggle Recipe Book.**



Inference:

In **every year and in every country**, the store **KaggleMart sells higher number of products than KaggleRama.**

In Belgium,

- The maximum number of products sold among all the years: 2018.
- The minimum number of products sold among all the years: 2020.

In France,

- The maximum number of products sold among all the years: 2018.
- The minimum number of products sold among all the years: 2017.

In Germany,

- The maximum number of products sold among all the years: 2018.
- The minimum number of products sold among all the years: 2020.

In Italy,

- The maximum number of products sold among all the years: 2020.
- The minimum number of products sold among all the years: 2017.

In Poland,

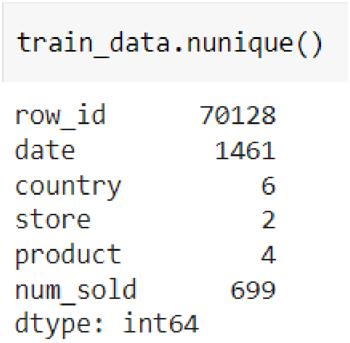
- The maximum number of products sold among all the years: 2020.
- The minimum number of products sold among all the years: 2017.

In Spain,

- The maximum number of products sold among all the years: 2020.
- The minimum number of products sold among all the years: 2017.

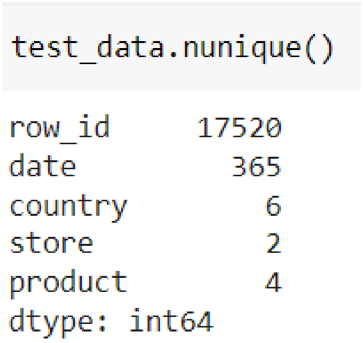
Code Walkthrough:

- The total number of unique values exists in both the training and valuation sets.



```
train_data.nunique()
```

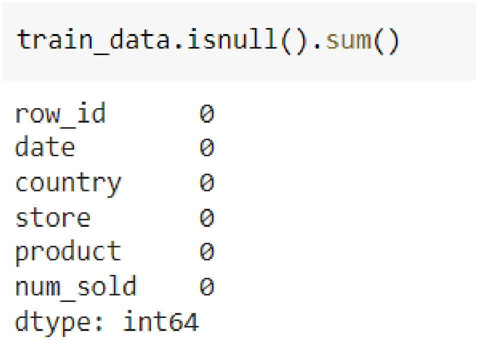
```
row_id       70128
date          1461
country          6
store            2
product          4
num_sold       699
dtype: int64
```



```
test_data.nunique()
```

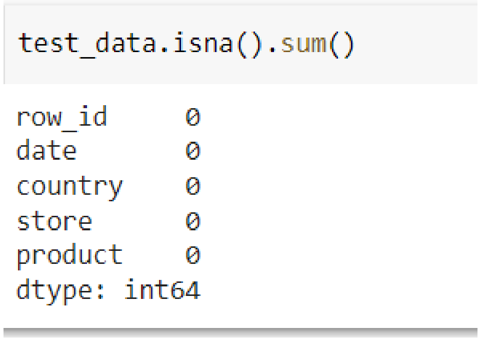
```
row_id      17520
date          365
country         6
store           2
product         4
dtype: int64
```

- The total number of missing values in both the training and evaluation sets is null.



```
train_data.isnull().sum()
```

```
row_id      0
date        0
country     0
store       0
product     0
num_sold    0
dtype: int64
```

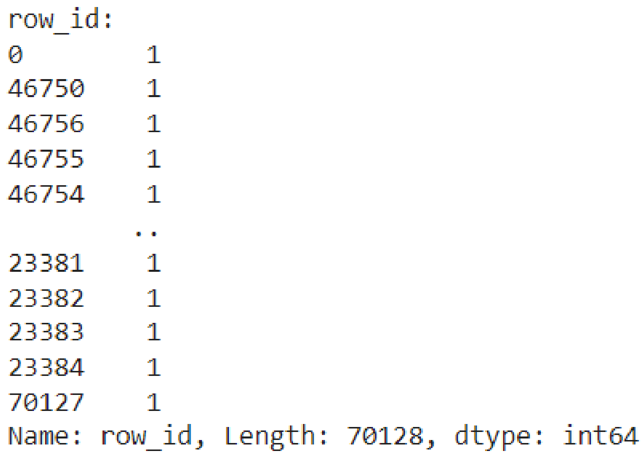


```
test_data.isna().sum()
```

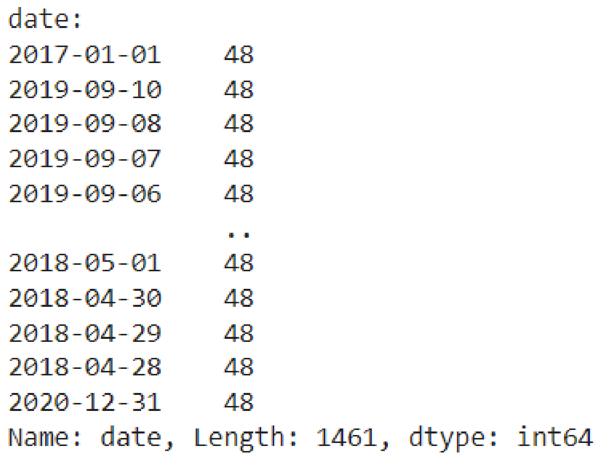
```
row_id     0
date       0
country    0
store      0
product    0
dtype: int64
```

- In the given dataset, out of 6 attributes (except row_id in features of matrix remaining is categorical variables), so we can’t perform outlier treatment.

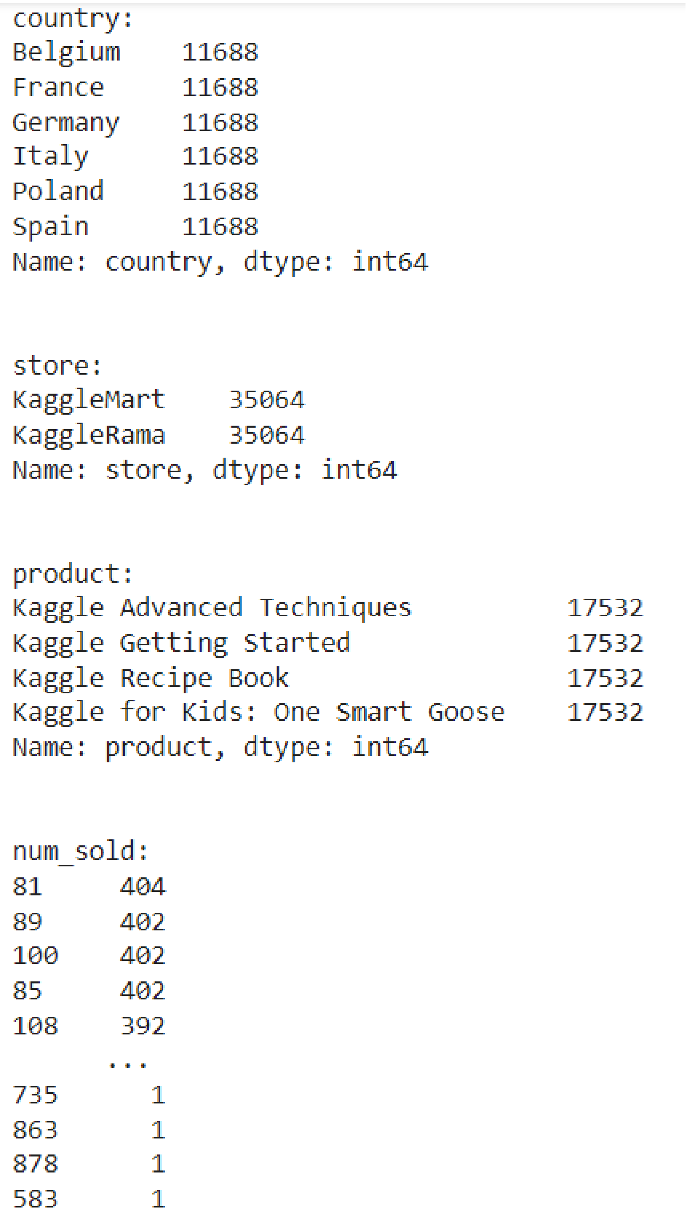
- In the training set, the number of distinct values and their counts,



```
row_id:
0        1
46750    1
46756    1
46755    1
46754    1
        ..
23381    1
23382    1
23383    1
23384    1
70127    1
Name: row_id, Length: 70128, dtype: int64
```



```
date:
2017-01-01    48
2019-09-10    48
2019-09-08    48
2019-09-07    48
2019-09-06    48
              ..
2018-05-01    48
2018-04-30    48
2018-04-29    48
2018-04-28    48
2020-12-31    48
Name: date, Length: 1461, dtype: int64
```



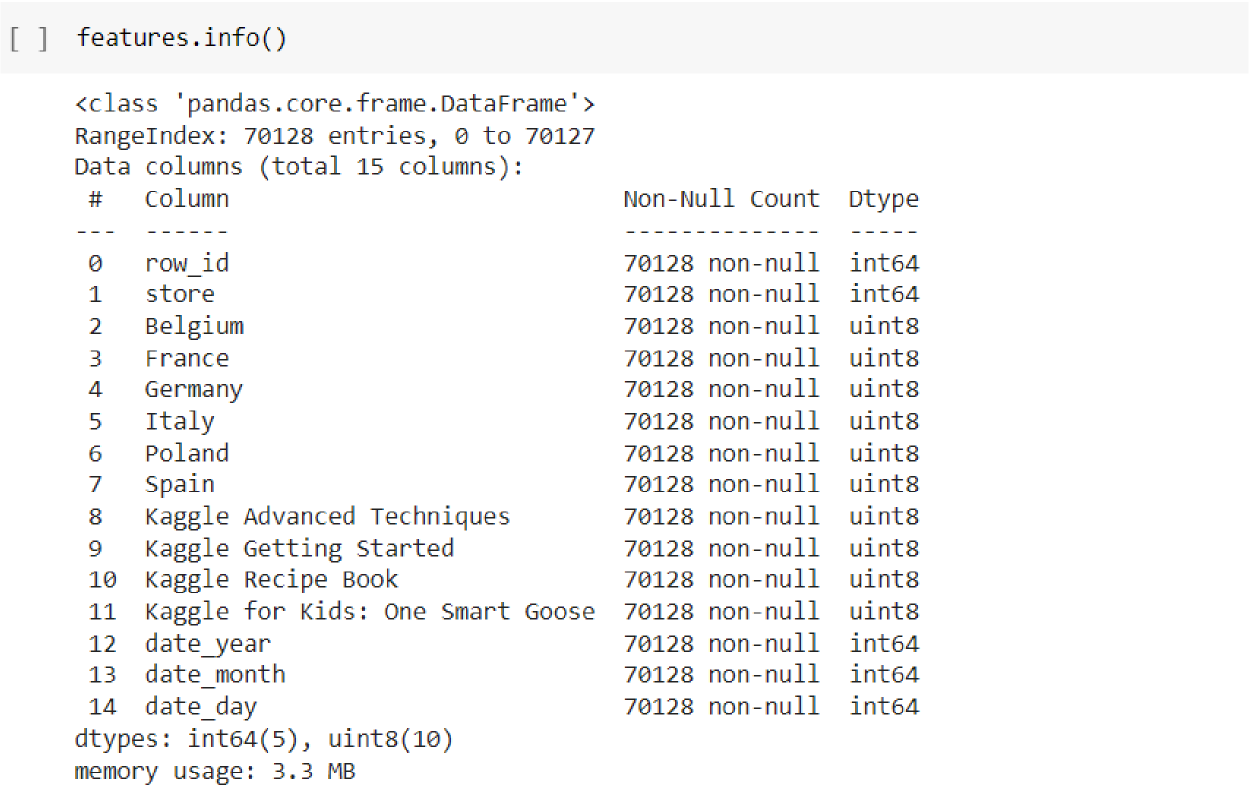
```
country:
Belgium     11688
France      11688
Germany     11688
Italy       11688
Poland      11688
Spain       11688
Name: country, dtype: int64


store:
KaggleMart    35064
KaggleRama    35064
Name: store, dtype: int64


product:
Kaggle Advanced Techniques          17532
Kaggle Getting Started              17532
Kaggle Recipe Book                  17532
Kaggle for Kids: One Smart Goose    17532
Name: product, dtype: int64


num_sold:
81     404
89     402
100    402
85     402
108    392
      ...
735      1
863      1
878      1
583      1
```

While encoding, only the columns are increased, the number of rows remains the same.

OneHotEncoding is used for multi classified value attributes. (product, country, date_year)

LabelEncoding is used for binary classified value attributes. (store).

- Now to train the model, the Categorical variables must be encoded,



```
[ ]  features.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 70128 entries, 0 to 70127
Data columns (total 15 columns):
 #   Column                           Non-Null Count  Dtype
---  ------                           --------------  -----
 0   row_id                           70128 non-null  int64
 1   store                            70128 non-null  int64
 2   Belgium                          70128 non-null  uint8
 3   France                           70128 non-null  uint8
 4   Germany                          70128 non-null  uint8
 5   Italy                            70128 non-null  uint8
 6   Poland                           70128 non-null  uint8
 7   Spain                            70128 non-null  uint8
 8   Kaggle Advanced Techniques       70128 non-null  uint8
 9   Kaggle Getting Started           70128 non-null  uint8
 10  Kaggle Recipe Book               70128 non-null  uint8
 11  Kaggle for Kids: One Smart Goose 70128 non-null  uint8
 12  date_year                        70128 non-null  int64
 13  date_month                       70128 non-null  int64
 14  date_day                         70128 non-null  int64
dtypes: int64(5), uint8(10)
memory usage: 3.3 MB
```

The datatype of the attributes, which is uint8 is encoded by OneHotEncoder and LabelEncoder.

And the total number of records is 70128.

No missing values exist in the current dataset.

Once the data is structured, the model must be trained.

Model Selection:

The best model is selected based on accuracy and precision, once the model is trained.

Both the linear and Non-linear models are used.

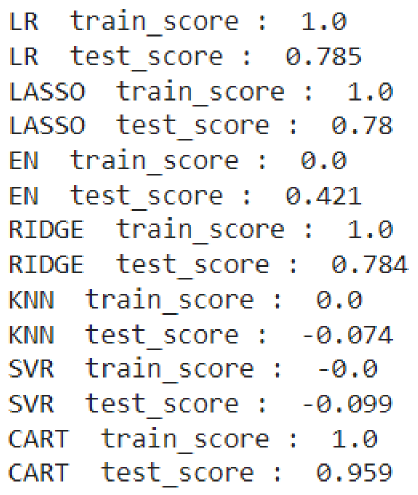
Linear Models – Linear Regression, Lasso, Ridge, Elastic Net.

Non-Linear Models – SVM, Decision tree, KNN.

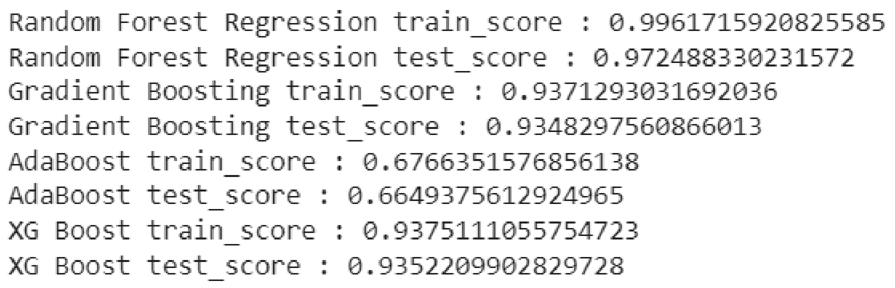
- Ensemble Models – RandomForestRegressor, GradientBoostingRegressor, AdaBoostRegressor, XGBoost.

Since it's a time series dataset, the **time series models are ideal** to use. But the above non-linear models, produce high accuracy in these types of datasets.

To identify overfitting models, both the test and training score of a model are calculated,



```
LR  train_score :  1.0
LR  test_score :  0.785
LASSO  train_score :  1.0
LASSO  test_score :  0.78
EN  train_score :  0.0
EN  test_score :  0.421
RIDGE  train_score :  1.0
RIDGE  test_score :  0.784
KNN  train_score :  0.0
KNN  test_score :  -0.074
SVR  train_score :  -0.0
SVR  test_score :  -0.099
CART  train_score :  1.0
CART  test_score :  0.959
```



```
Random Forest Regression train_score : 0.9961715920825585
Random Forest Regression test_score : 0.972488330231572
Gradient Boosting train_score : 0.9371293031692036
Gradient Boosting test_score : 0.9348297560866013
AdaBoost train_score : 0.6766351576856138
AdaBoost test_score : 0.6649375612924965
XG Boost train_score : 0.9375111055754723
XG Boost test_score : 0.9352209902829728
```

Linear Regression, LASSO, RIDGE, CART produces 100 percent accuracy while training and Elastic Net, KNN, SVR produces 0 percent accuracy while training the model,

Therefore LR, LASSO, RIDGE, CART – Overfitting Models.

And EN, KNN, SVR – Underfitting Models.

So, to overcome overfitting, the process to be undergone,

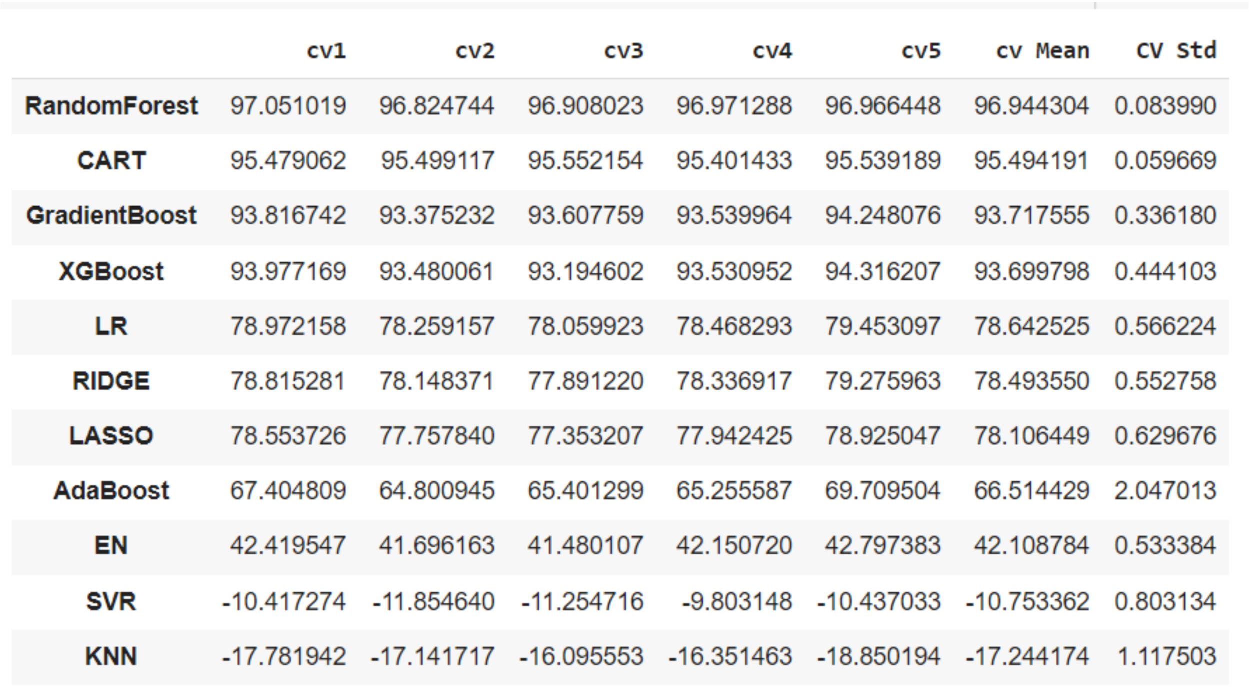
- Regularization
- Early Stopping
- Pruning
- Feature Selection
- Dimensionality Reduction

The following regularization models (Lasso, Ridge, and Elastic Net) didn't have any effect on the dataset.

The correlation values among the attributes are worse.

Therefore, the GradientBoostingRegressor produces 93.7 percent in training and 93.4 in test dataset. (<u>Performance is good</u>).

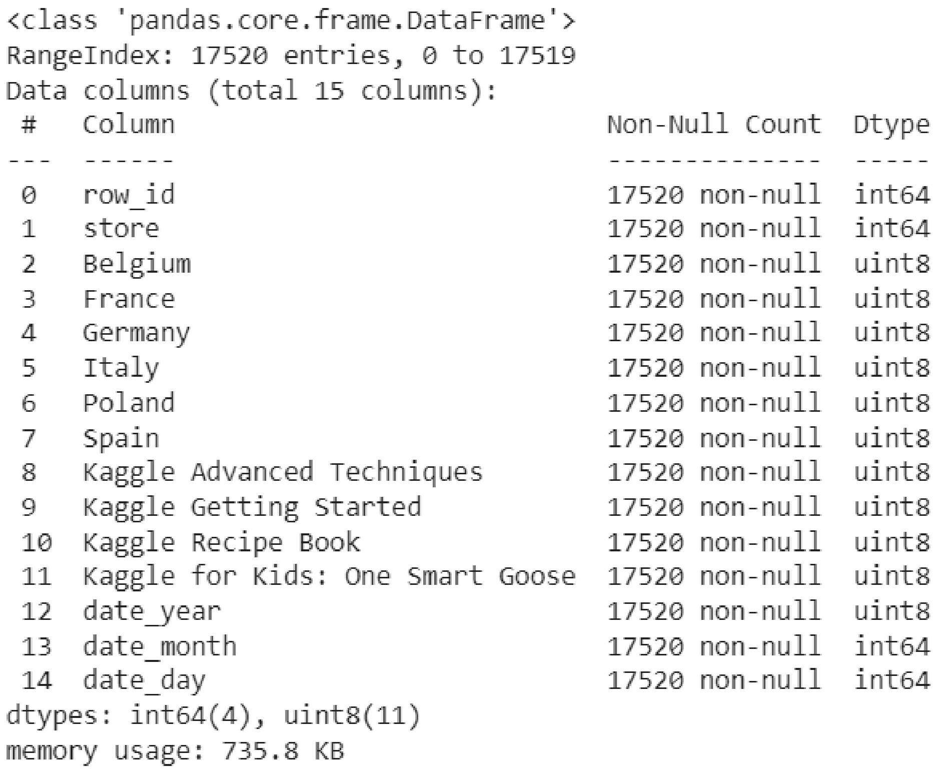
<u>CROSS VALIDATION RESULTS:</u>



	cv1	cv2	cv3	cv4	cv5	cv Mean	CV Std
RandomForest	97.051019	96.824744	96.908023	96.971288	96.966448	96.944304	0.083990
CART	95.479062	95.499117	95.552154	95.401433	95.539189	95.494191	0.059669
GradientBoost	93.816742	93.375232	93.607759	93.539964	94.248076	93.717555	0.336180
XGBoost	93.977169	93.480061	93.194602	93.530952	94.316207	93.699798	0.444103
LR	78.972158	78.259157	78.059923	78.468293	79.453097	78.642525	0.566224
RIDGE	78.815281	78.148371	77.891220	78.336917	79.275963	78.493550	0.552758
LASSO	78.553726	77.757840	77.353207	77.942425	78.925047	78.106449	0.629676
AdaBoost	67.404809	64.800945	65.401299	65.255587	69.709504	66.514429	2.047013
EN	42.419547	41.696163	41.480107	42.150720	42.797383	42.108784	0.533384
SVR	-10.417274	-11.854640	-11.254716	-9.803148	-10.437033	-10.753362	0.803134
KNN	-17.781942	-17.141717	-16.095553	-16.351463	-18.850194	-17.244174	1.117503

Since the training and model selection are completed, the accuracy and precision must be configured for the test dataset.

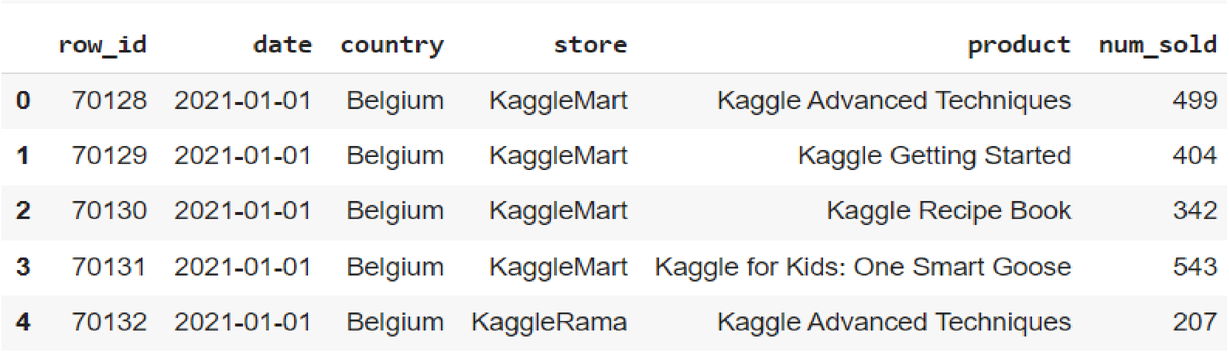
And the **test dataset must be configured the same as the training set**, to predict the target. Since the dimensions or the datatypes have been mismatched, the error is thrown.



```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 17520 entries, 0 to 17519
Data columns (total 15 columns):
 #   Column                            Non-Null Count  Dtype
---  ------                            --------------  -----
 0   row_id                            17520 non-null  int64
 1   store                             17520 non-null  int64
 2   Belgium                           17520 non-null  uint8
 3   France                            17520 non-null  uint8
 4   Germany                           17520 non-null  uint8
 5   Italy                             17520 non-null  uint8
 6   Poland                            17520 non-null  uint8
 7   Spain                             17520 non-null  uint8
 8   Kaggle Advanced Techniques        17520 non-null  uint8
 9   Kaggle Getting Started            17520 non-null  uint8
 10  Kaggle Recipe Book                17520 non-null  uint8
 11  Kaggle for Kids: One Smart Goose  17520 non-null  uint8
 12  date_year                         17520 non-null  uint8
 13  date_month                        17520 non-null  int64
 14  date_day                          17520 non-null  int64
dtypes: int64(4), uint8(11)
memory usage: 735.8 KB
```

The dimensions and the order remain the same as the training set.

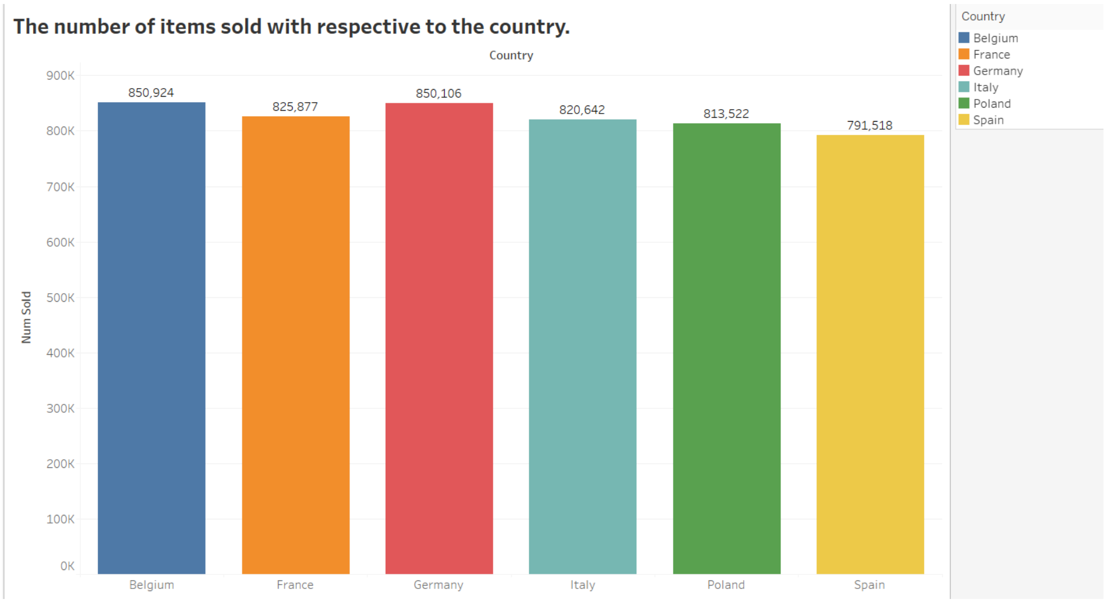
<u>Prediction:</u> **Test set size: (17520,2)**



	row_id	date	country	store	product	num_sold
0	70128	2021-01-01	Belgium	KaggleMart	Kaggle Advanced Techniques	499
1	70129	2021-01-01	Belgium	KaggleMart	Kaggle Getting Started	404
2	70130	2021-01-01	Belgium	KaggleMart	Kaggle Recipe Book	342
3	70131	2021-01-01	Belgium	KaggleMart	Kaggle for Kids: One Smart Goose	543
4	70132	2021-01-01	Belgium	KaggleRama	Kaggle Advanced Techniques	207

<u>Inference:</u> Therefore, the number of Kaggle Advanced Techniques in KaggleMart sold in Belgium on 1 Jan 2021 is <u>499.</u>

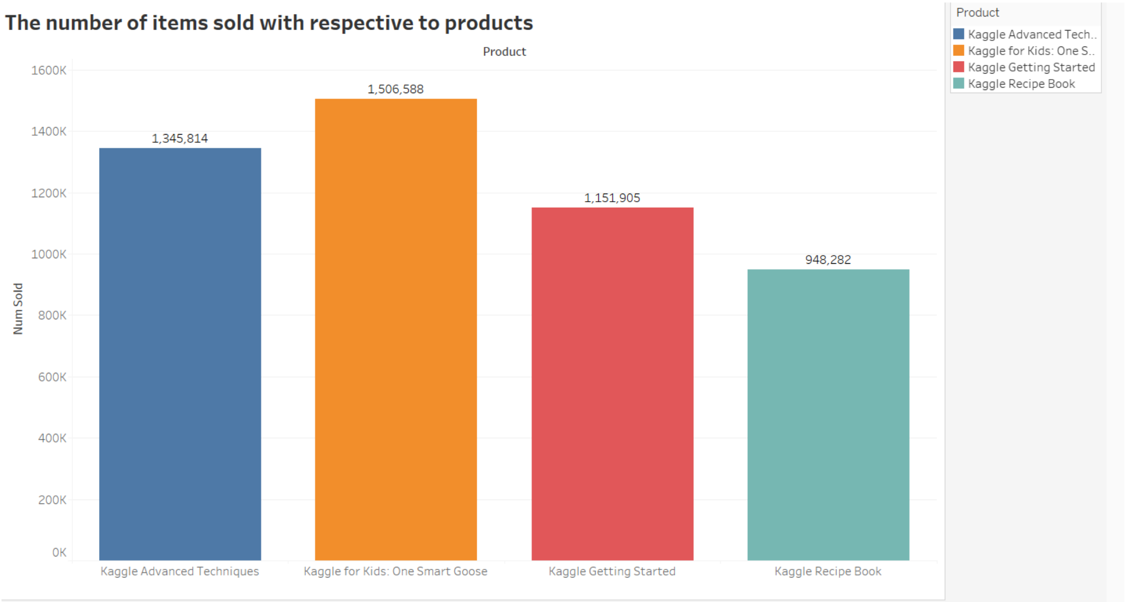
Visualization:



The number of items sold with respective to the country.

Inference:

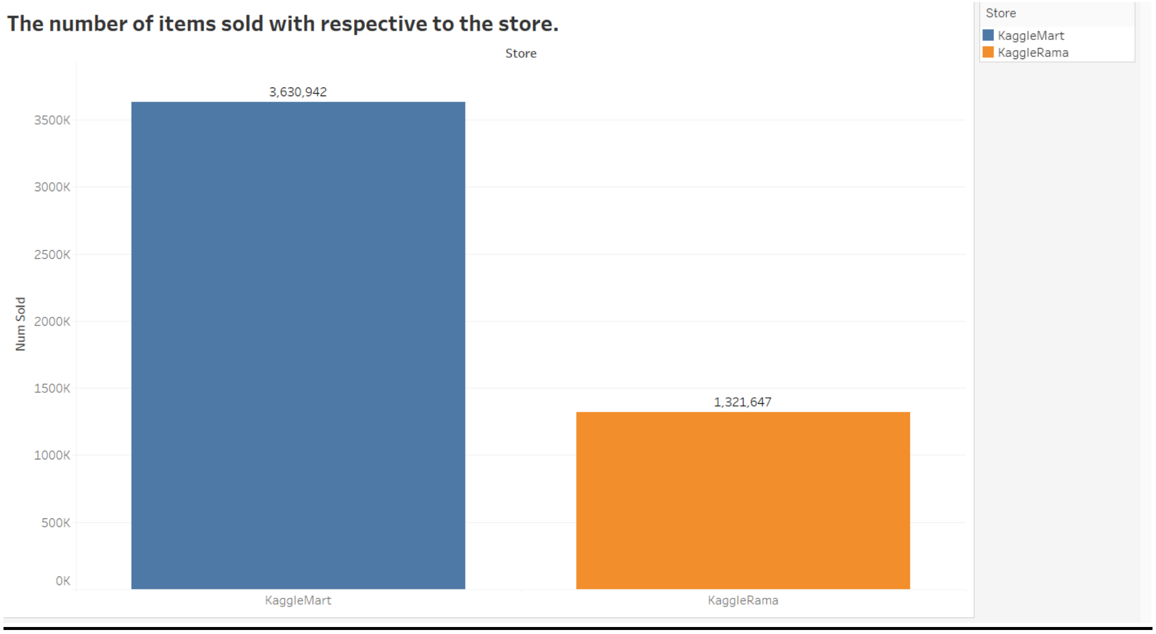
The maximum number of items sold among all countries: Belgium



The number of items sold with respective to products

Inference:

The maximum number of items sold among all products: Kaggle for kids.



The number of items sold with respective to the store.

<u>Inference:</u>

The maximum number of items sold among all store: KaggleMart.