Project 2022

Big Data Computing

Prof: G. Tolomei

Giovanni Pica
1816394

Introduction

- In these slides there is an explanation of the project that I made for the Big Data Computing course of the A.Y. 2021/2022.
- The project is based on a **Clustering** and a **Regression** problem applied on a dataset called **PREDICT 2021**. (More details here http://data.europa.eu/89h/26d3d1c0-cf84-4f1f-86ad-4ff28c8f7a77)
- This dataset provides indicators in a wide variety of topics of the major countries of the EU plus other countries such as China, USA and so on. Those topics include value added, employment, labour productivity and business R&D expenditure (BERD), distinguishing fine grain economic activities in ICT industries, media and content industries and at a higher level of aggregation for all the other industries in the economy. It also produces data on Government financing of R&D in ICTs, and total R&D expenditure. Nowcasting of more relevant data in these domains is also performed until a year before the reference date, while time series go back to 1995.
- In this project the topics that are covered are the costs in millions of euros for the regression task and full time equivalent for the clustering task that corresponds to one year's work by one person.

Dataset

- The features of the dataset are 12 and they are:
  - **variablecode**: is a code that describe what are the economic variables;
  - **unit**: is the unit of the label value for example millions of euros or full time equivalent;
  - **dataset_type**: which dataset contains this information, because there were some nowcasted dataset or old ones;
  - **country_code**: is the abbreviation of the country;
  - **country**: the country name;
  - **classification**: is a code that could be NACE Rev. 2. for example and is used for define which type of work sector we have;
  - **classificationcode**: it defines the classification code;
  - **sectorcode**: it defines the code of the specific sector where the operations were done;
  - **definition**: which type of sector;
  - **description:** is a descrition of what kind of operation was maded;
  - **year**: is the year;
  - **value**: is the value based on which unit we choose.

Preliminar steps

- Some interesting preliminar steps:
    - Unzip the dataset with the **ZipFile** library and save it to Google Drive after the mount;
    - Check for NULL values and drop them. I decide to drop them because some estimations or measurements maybe were lost. So to find the correct cost for example in the 1997 of some country is not really simple.
    - Split the dataset in two ways one with the unit in **Millions of current euros (also PPS that is an artificial currency unit more details here** https://ec.europa.eu/eurostat/statistics-explained/index.php?title=Glossary:Purchasing_power_standard_(PPS)**)** and **Full Time Equivalent** respectively for the Regression and Clustering task.

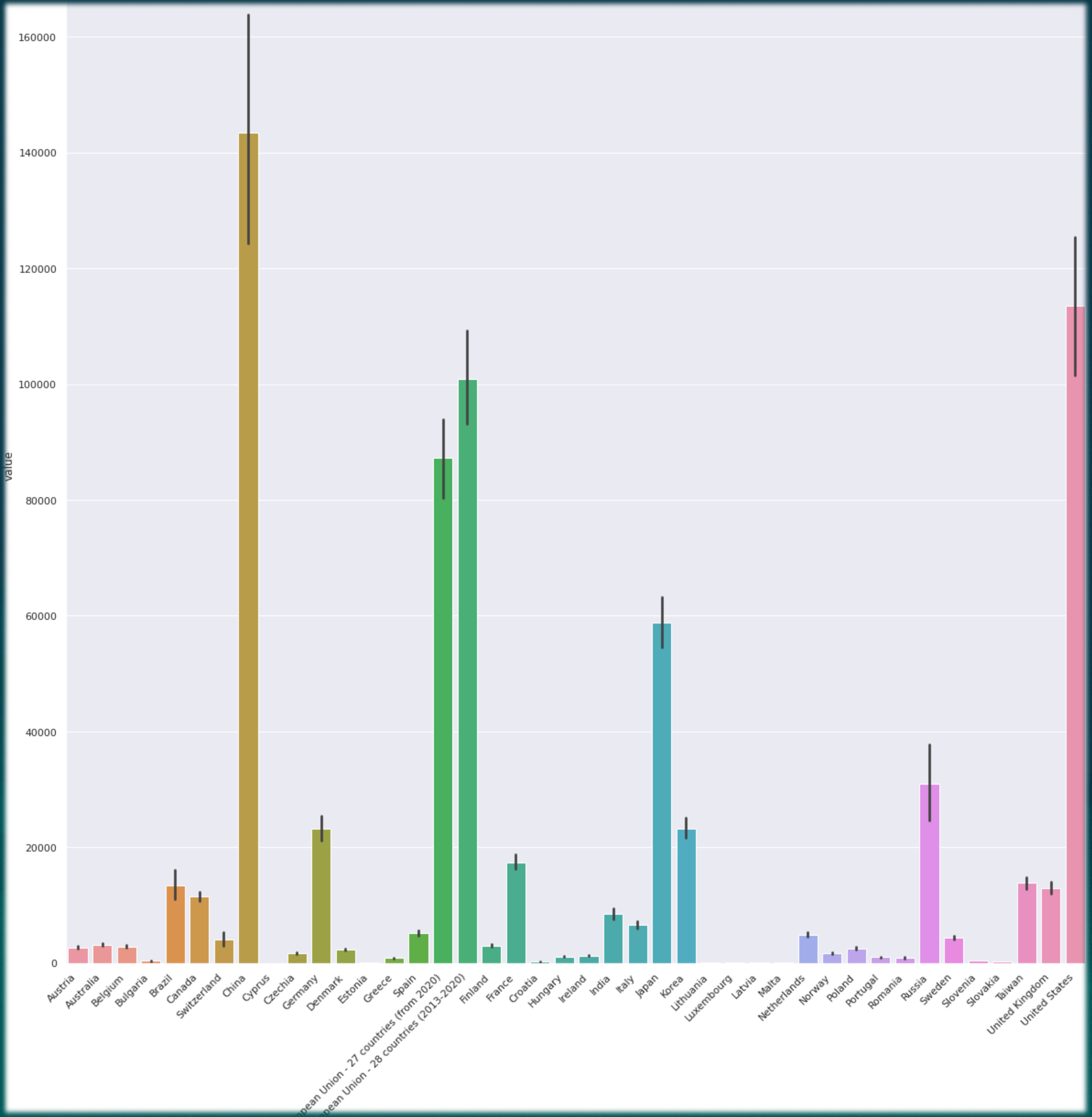
Clustering Task

Definition

- The features that I use for this task are the country and the value because I’m interested to find the similarity between countries based on the Full Time Equivalent.
- The model that I used for define the clusters is the **K-Means Clustering** that is a partition based algorithm.
- Evaluation of the Clustering algorithm are computed using the **Silhouette Coefficient** and for the correct choice of K the **Elbow Method**.

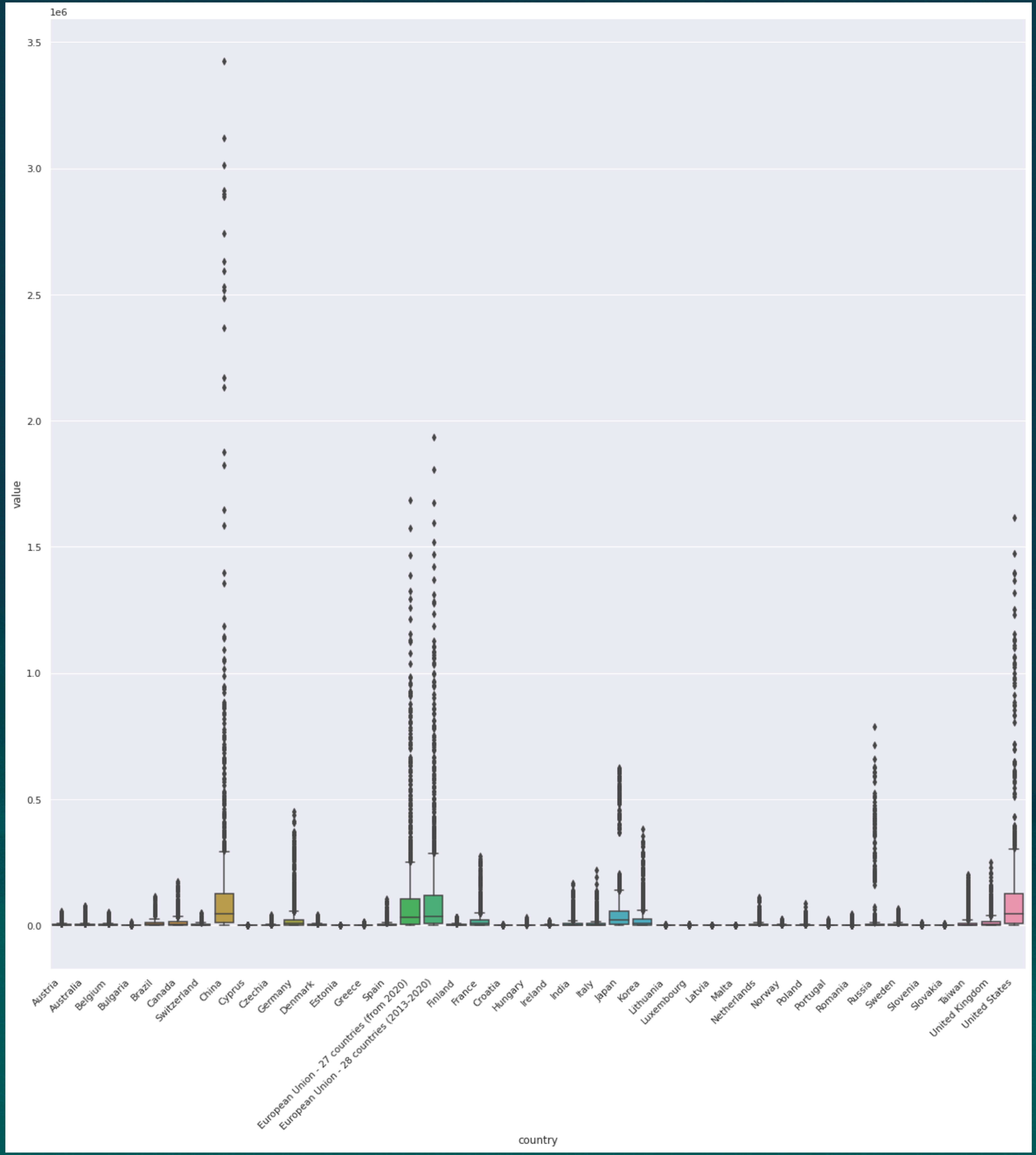
Data Exploration 1

- A nice barplot between value and country.
- In this plot there are some interesting facts, for example that **China** is the country that has the highest FullTimeEquivalent so in this country a man in average works more than other countries.



Data Exploration 2

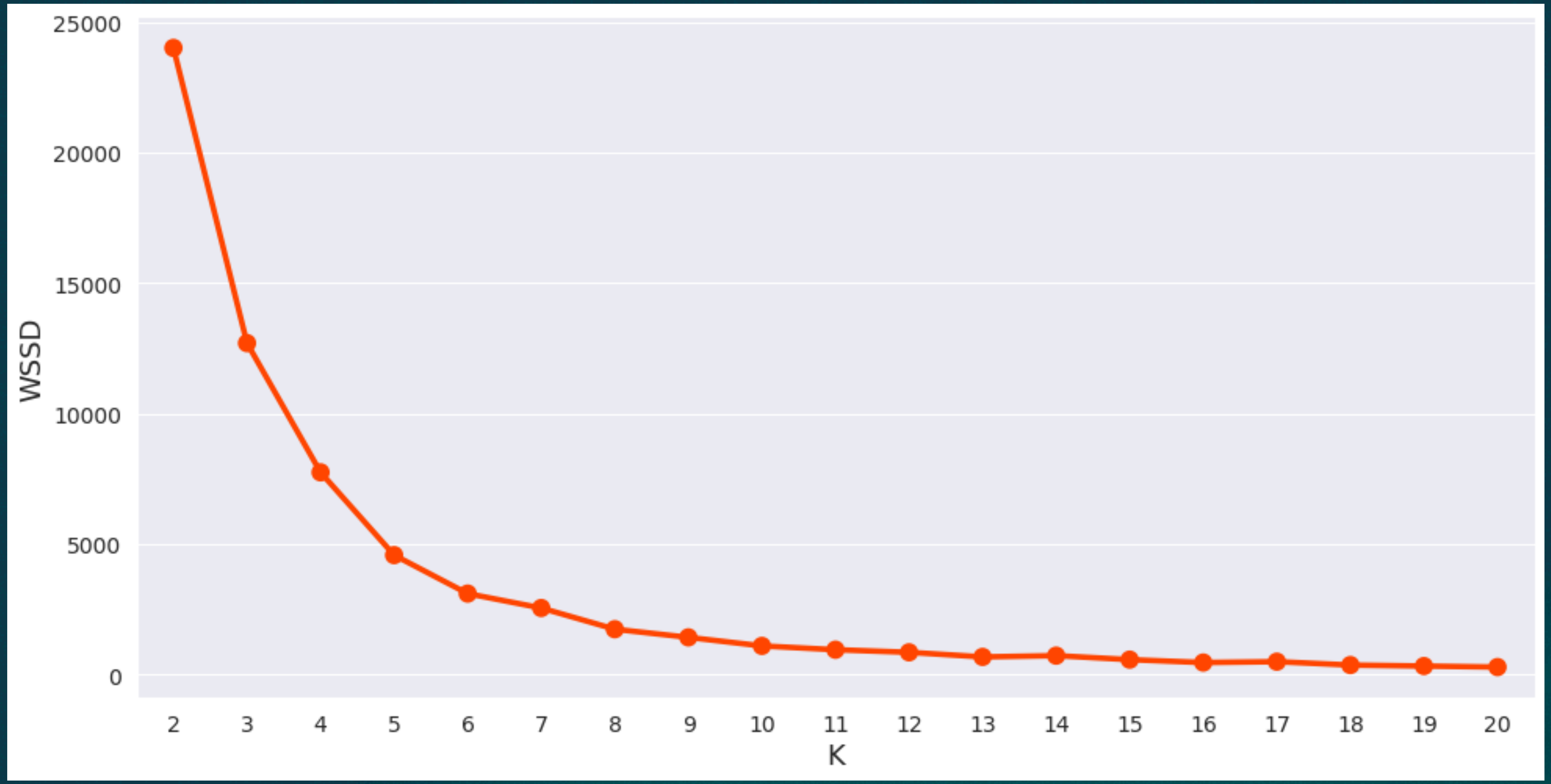
- A nice boxplot between value and country.
- Another interesting fact is in the boxplot because it draws some outliers so those must be handled.



Data Preprocessing

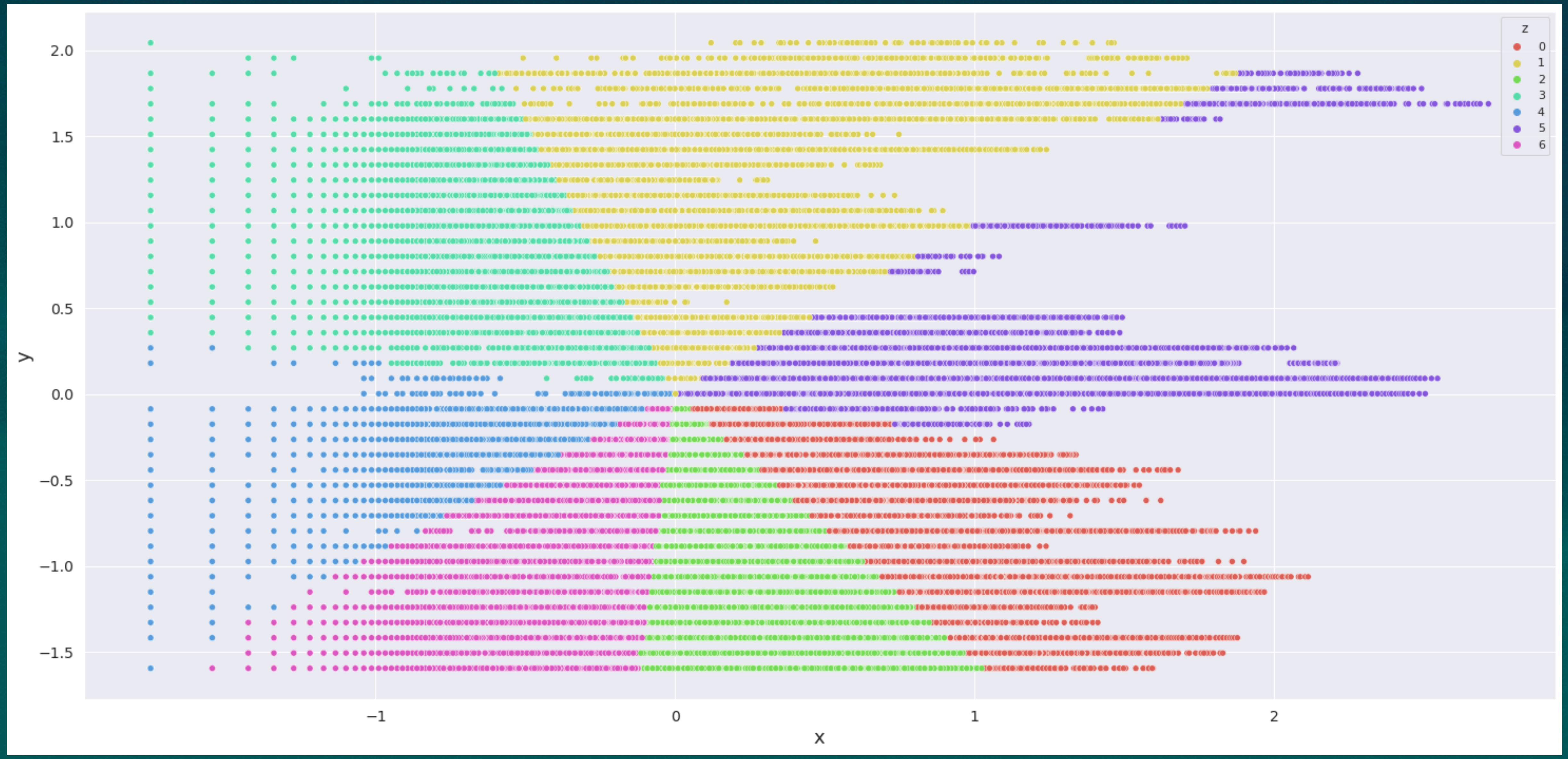
- In clustering tasks we deserve to use numerical data, data should have no noises or for example we could remove some useless features. So we must clean our data before we start our clustering algorithm.
- Removing of the duplicates (if there are) and transform categorical data into numerical (for example country).
- Outlier detection with a function (more details here https://deepnote.com/@rajshekar-2021/Outlier-Detection-Pyspark-069e69af-2c1d-4d4d-884a-92aad276d06f) and **Winsorization** technique to transform the data (when the value is 0 the logarithm is log(x + 1)).
- Scaling of the features with the **StandardScaler** to set standard deviation and mean at True to standardize all the features to 0-mean and 1-unit of standard deviation.

Model Results



- K Elbow method with WSSD and K.

- Scatterplot of the clusters with K=7



Regression Task

Definition

- Prediction of the costs of the countries and their ICT industries.

- The models are **LinearRegression**, **GeneralizedLinearRegression** (instead of LinearRegression the output doesn’t follow a Gaussian distribution but other exponential families distributions), **DecisionTreeRegression**, **RandomForestRegression**, **GradientBoostedTreeRegression**, **LassoRegression**, **RidgeRegression**, **FMRegression** (using factorization machines more details here https://spark.apache.org/docs/latest/ml-classification-regression.html#factorization-machines).

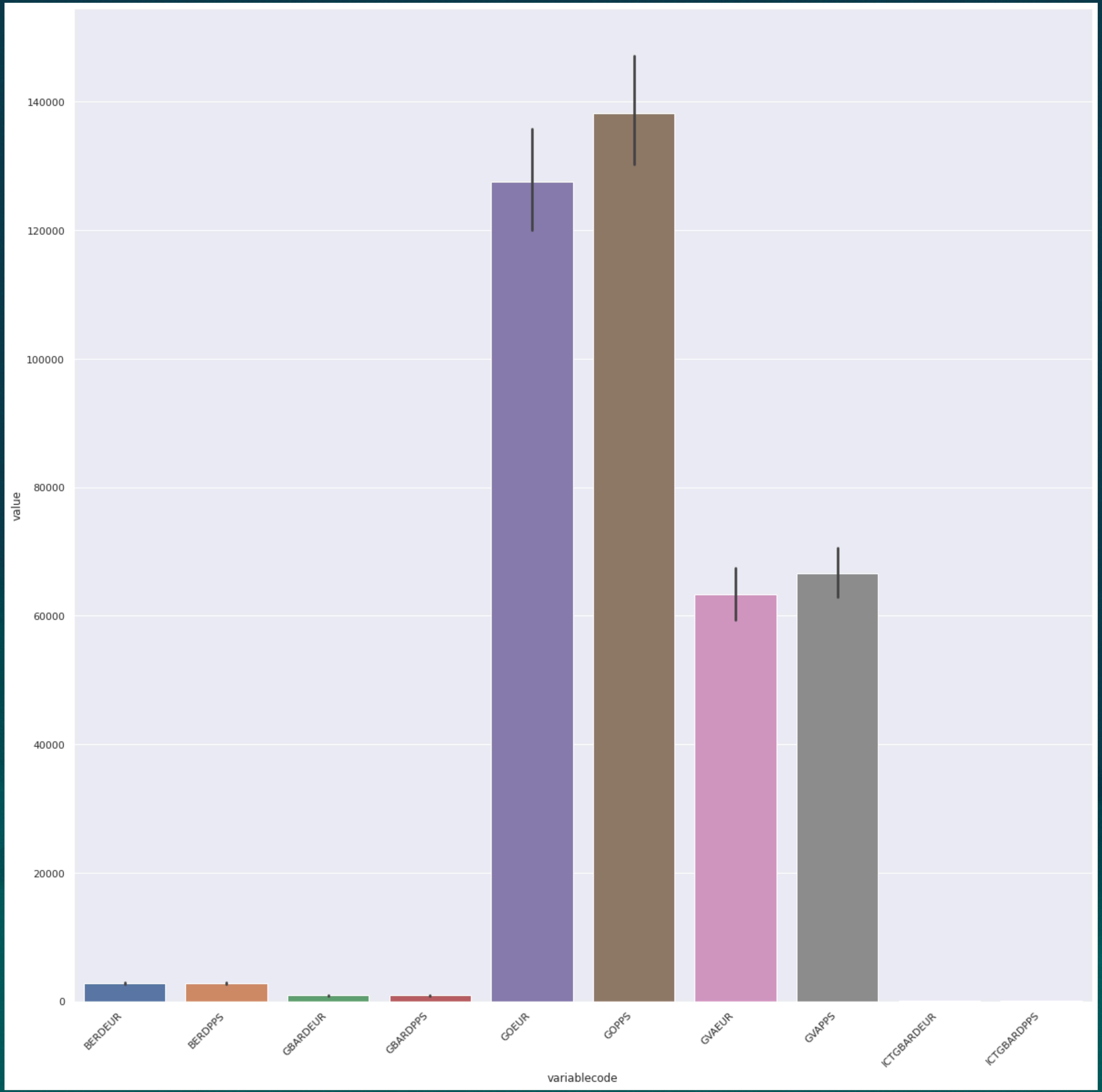
- Evaluation metrics are R^2, adjusted R^2, RMSE.

Data Exploration

- For the Data Exploration I decide to do some barplots between the features with the label “value”.

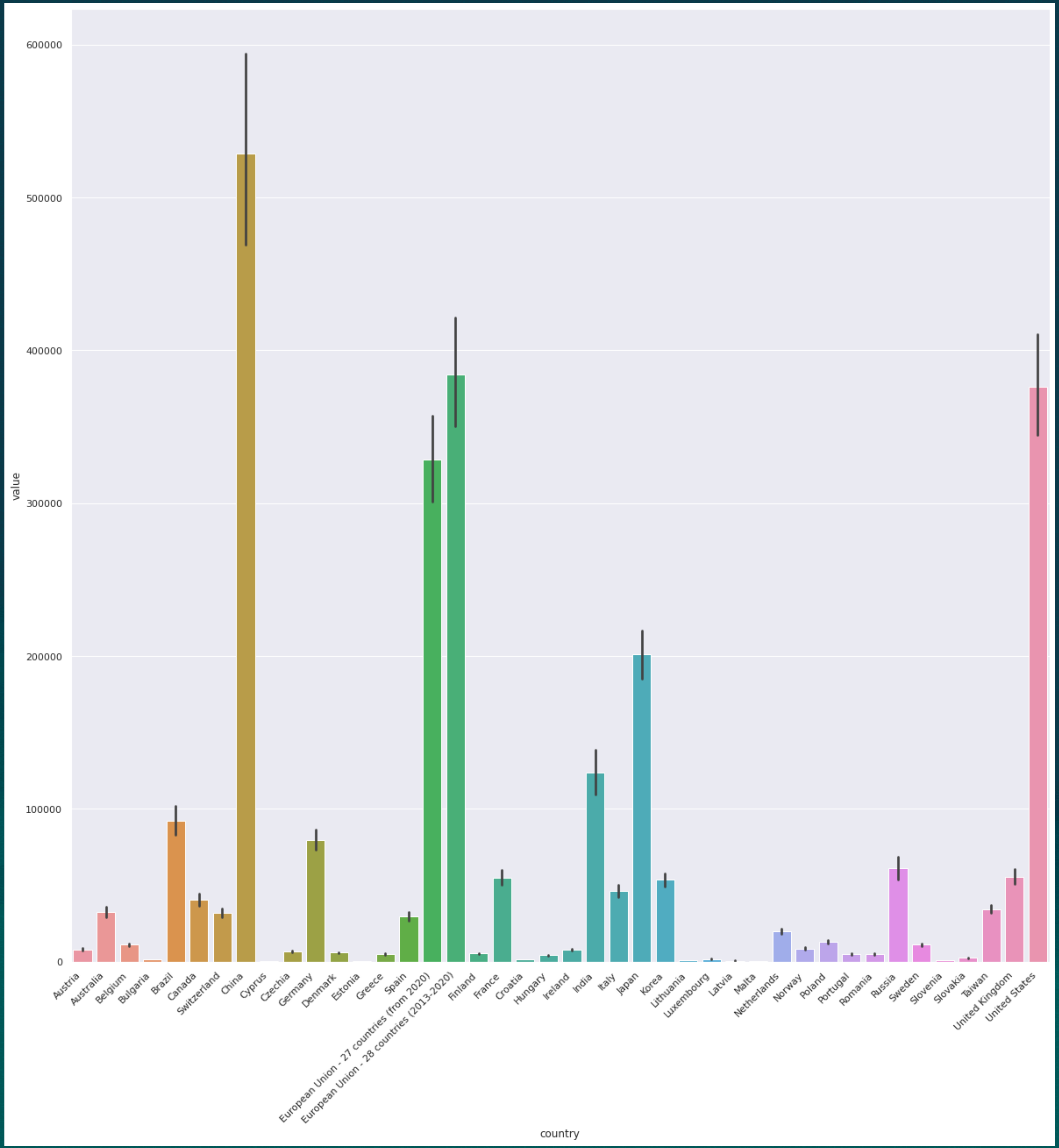
Barplot 1

Variablecode vs value



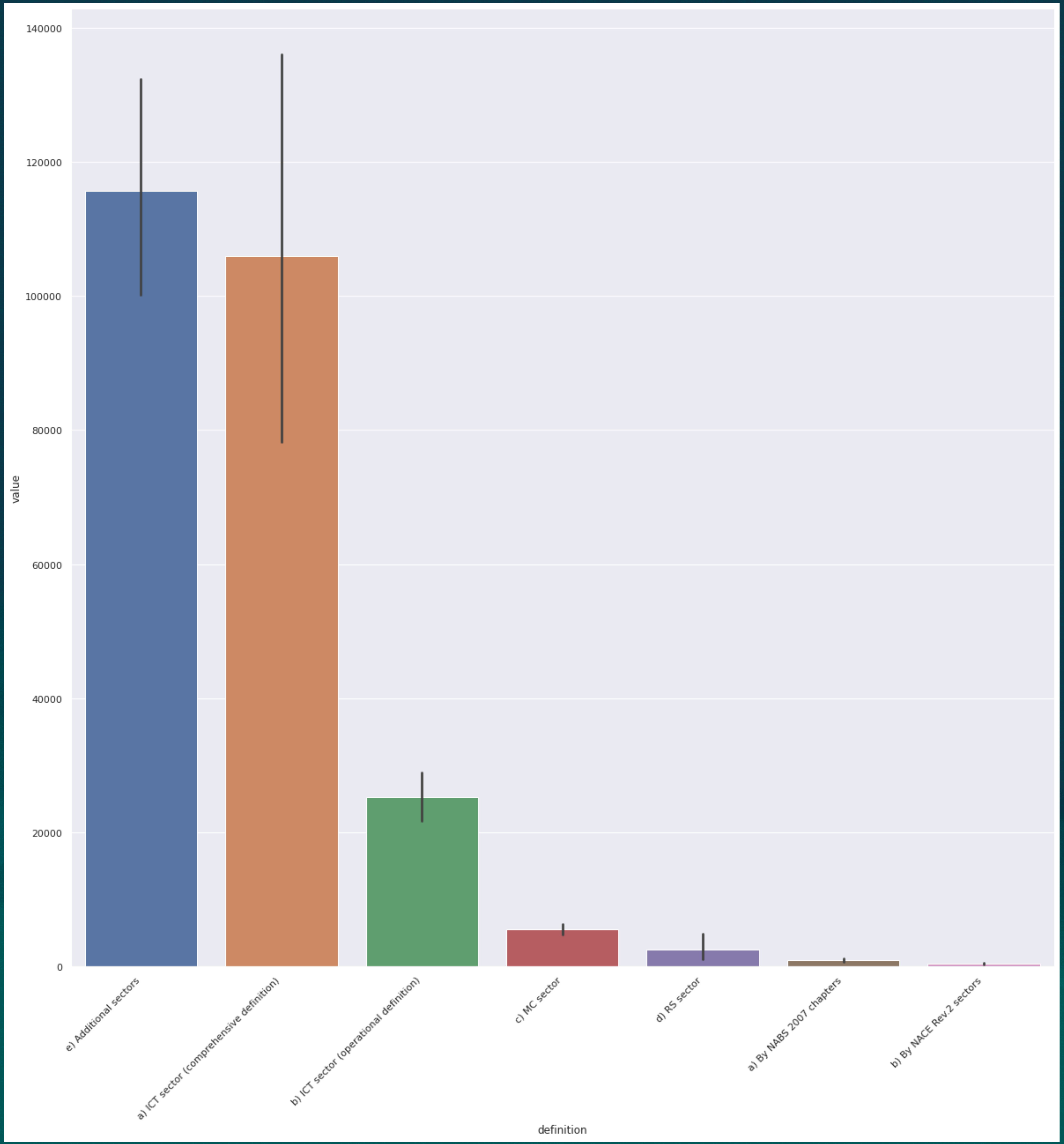
Barplot 2

Country vs value



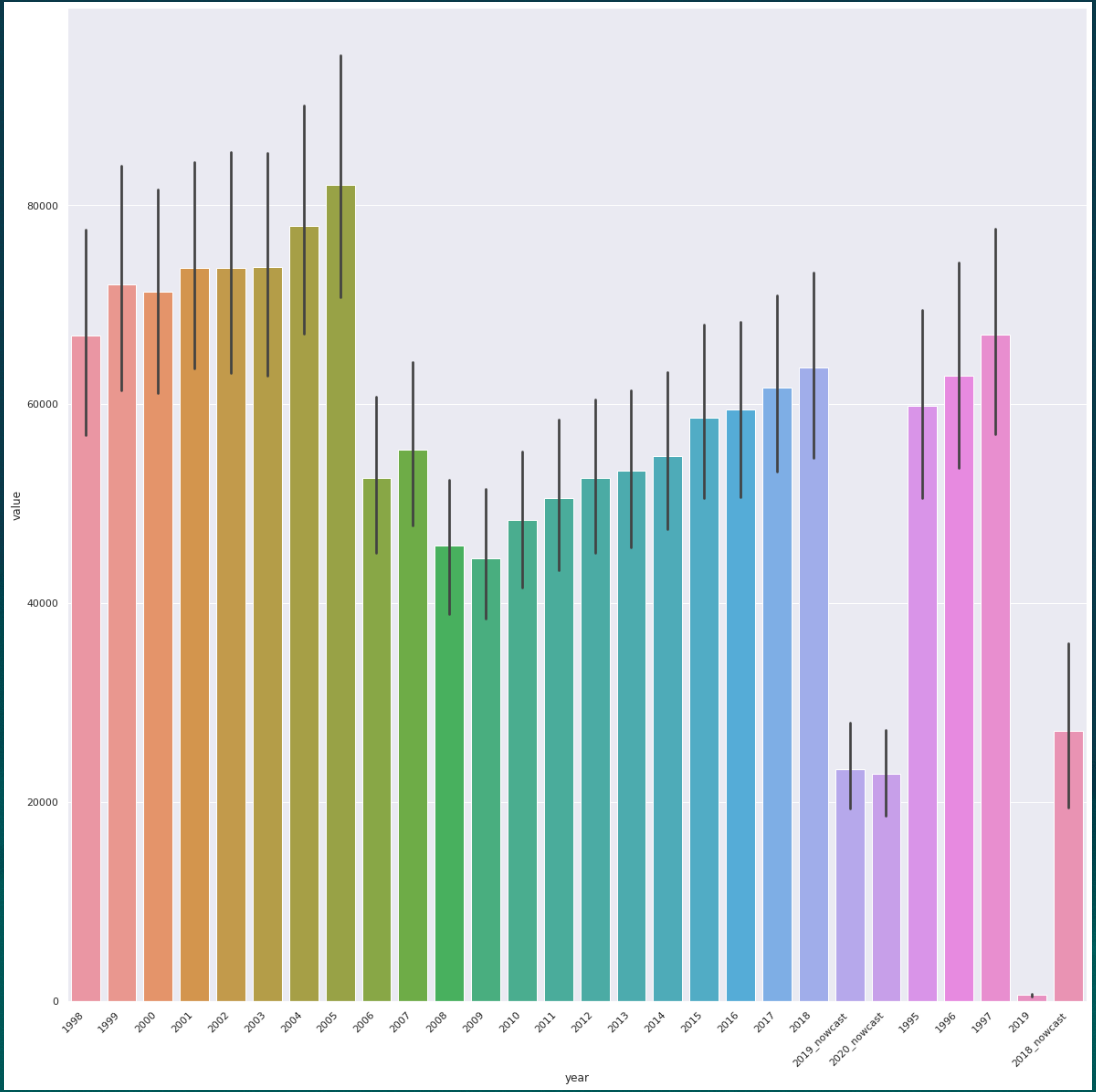
Barplot 3

Definition vs value



Barplot 4

Year vs value

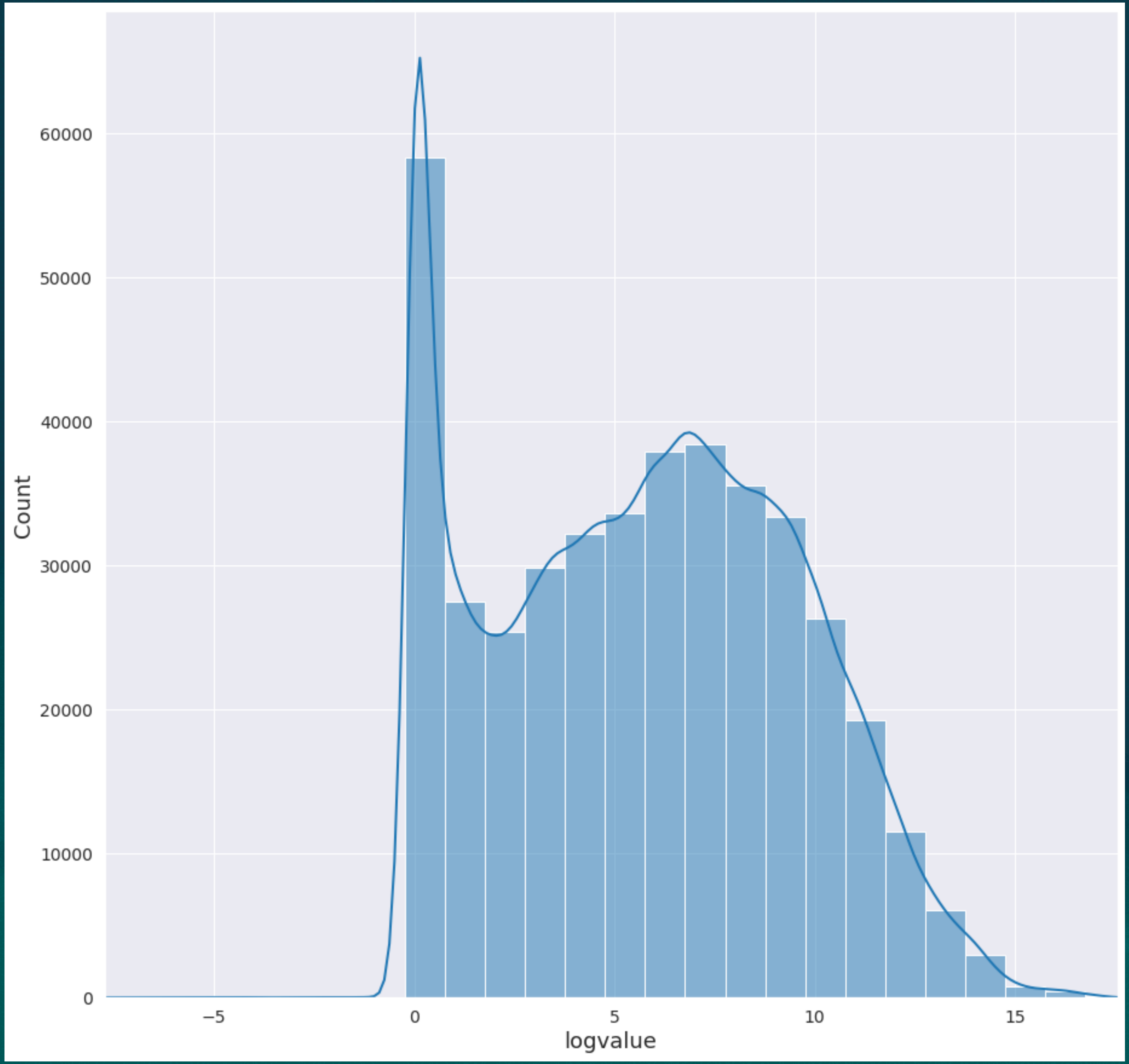


Data Preprocessing

- **StringIndexer** to transform the categorical features in numerical ones.
- **OneHotEncoder** to transform those features with the OneHotEncoding.
- Log transformation of the label because data is skewed and has a weird distribution and not a really clear plot. Log transformation is performed with this formula $sign(log(|x|+1))$ because in this way the negative values could have a logarithm and the sign gives the negative value.
- Finding outliers and apply the log transformation above.
- Heatmap and after this the calculation of the **VIF** (variance inflation factor more details here https://www.analyticsvidhya.com/blog/2020/03/what-is-multicollinearity/) for decide which features could be dropped, this features are “country”, “sectorcode” and “description”.

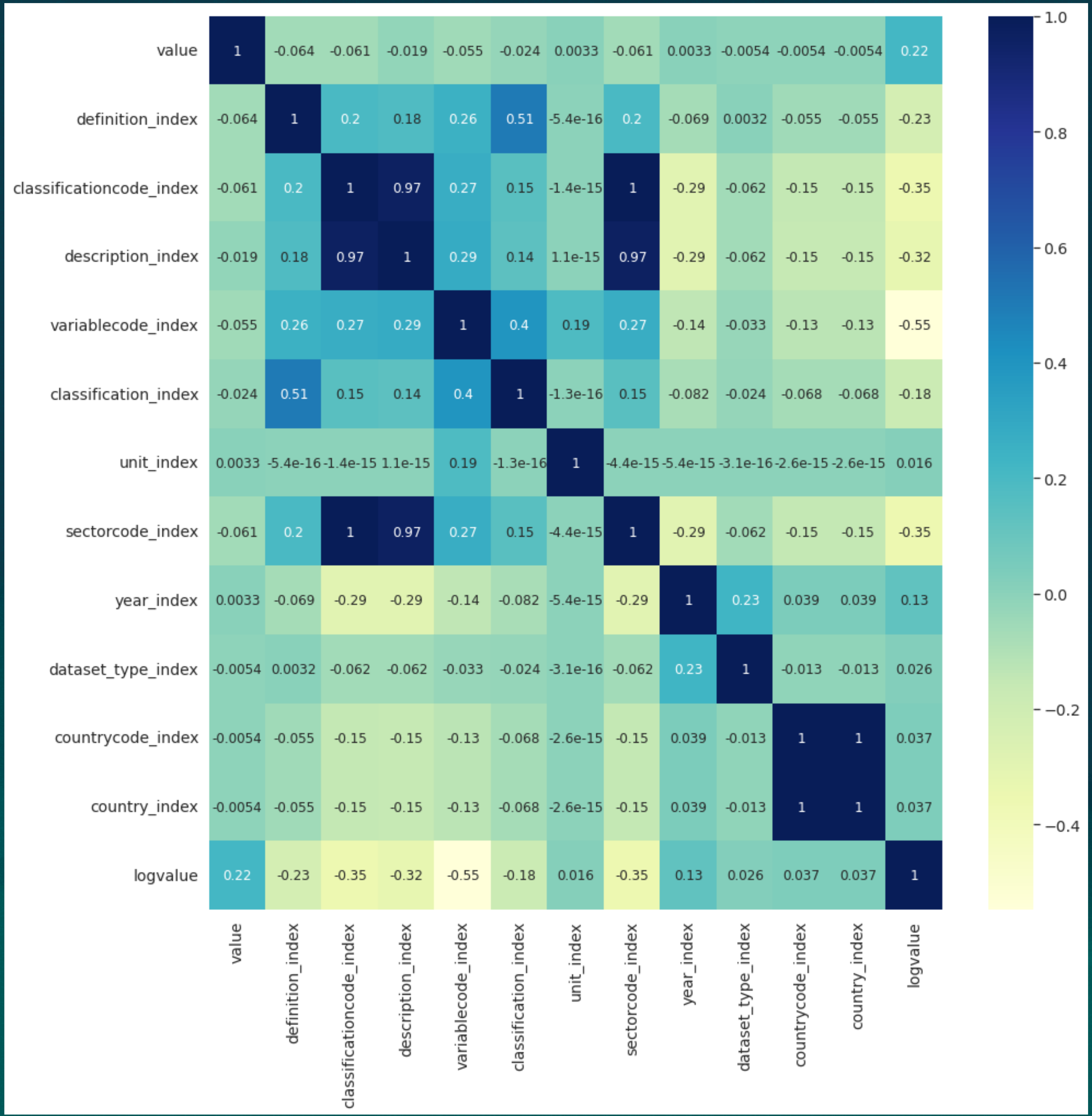
Label Distribution

After the log transformation



Heatmap

Collinearity in country, classificationcode, sectorcode, description, country_code



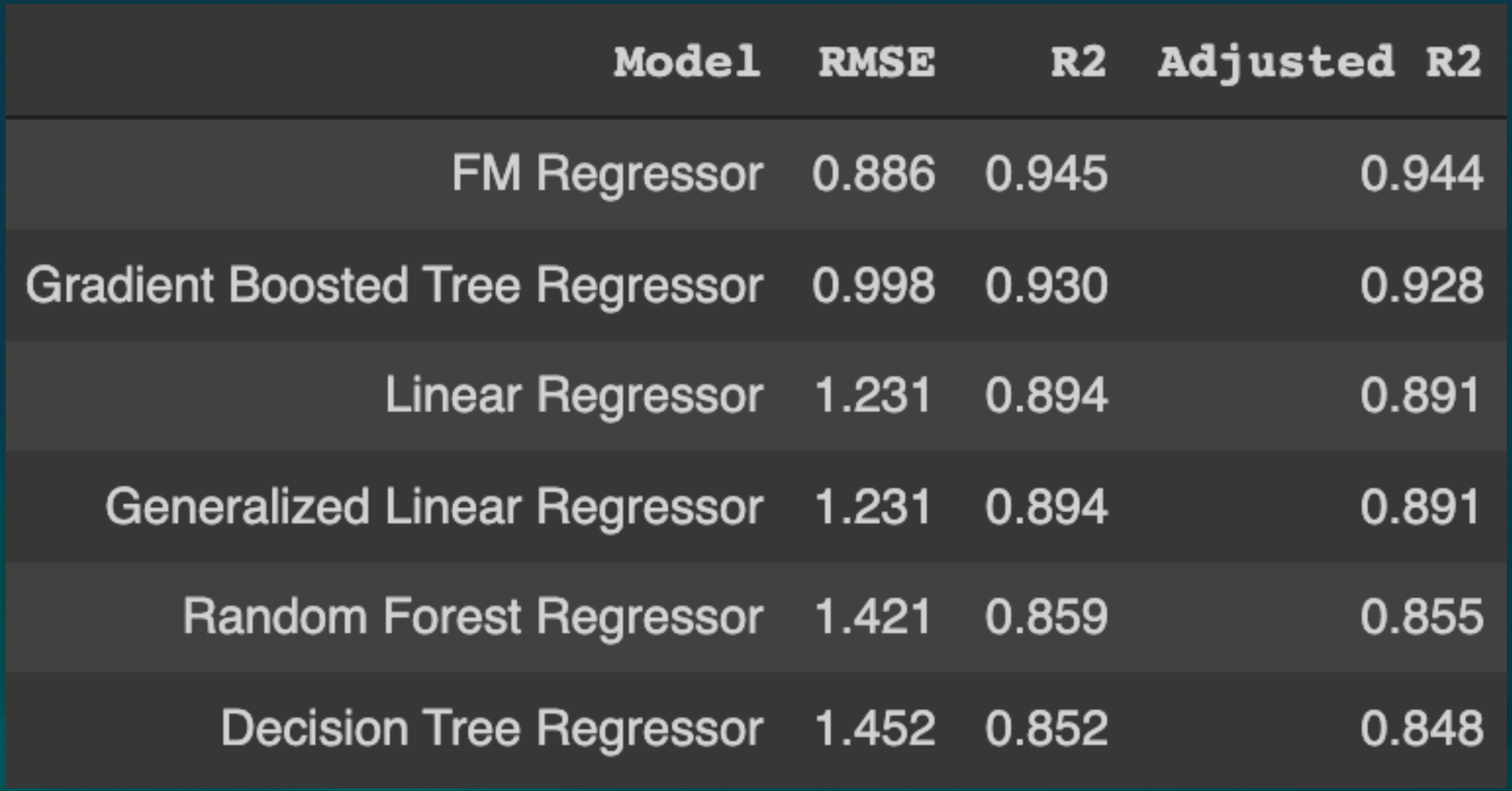
	value	definition_index	classificationcode_index	description_index	variablecode_index	classification_index	unit_index	sectorcode_index	year_index	dataset_type_index	countrycode_index	country_index	logvalue
value	1	-0.064	-0.061	-0.019	-0.055	-0.024	0.0033	-0.061	0.0033	-0.0054	-0.0054	-0.0054	0.22
definition_index	-0.064	1	0.2	0.18	0.26	0.51	-5.4e-16	0.2	-0.069	0.0032	-0.055	-0.055	-0.23
classificationcode_index	-0.061	0.2	1	0.97	0.27	0.15	-1.4e-15	1	-0.29	-0.062	-0.15	-0.15	-0.35
description_index	-0.019	0.18	0.97	1	0.29	0.14	1.1e-15	0.97	-0.29	-0.062	-0.15	-0.15	-0.32
variablecode_index	-0.055	0.26	0.27	0.29	1	0.4	0.19	0.27	-0.14	-0.033	-0.13	-0.13	-0.55
classification_index	-0.024	0.51	0.15	0.14	0.4	1	-1.3e-16	0.15	-0.082	-0.024	-0.068	-0.068	-0.18
unit_index	0.0033	-5.4e-16	-1.4e-15	1.1e-15	0.19	-1.3e-16	1	-4.4e-15	-5.4e-15	-3.1e-16	-2.6e-15	-2.6e-15	0.016
sectorcode_index	-0.061	0.2	1	0.97	0.27	0.15	-4.4e-15	1	-0.29	-0.062	-0.15	-0.15	-0.35
year_index	0.0033	-0.069	-0.29	-0.29	-0.14	-0.082	-5.4e-15	-0.29	1	0.23	0.039	0.039	0.13
dataset_type_index	-0.0054	0.0032	-0.062	-0.062	-0.033	-0.024	-3.1e-16	-0.062	0.23	1	-0.013	-0.013	0.026
countrycode_index	-0.0054	-0.055	-0.15	-0.15	-0.13	-0.068	-2.6e-15	-0.15	0.039	-0.013	1	1	0.037
country_index	-0.0054	-0.055	-0.15	-0.15	-0.13	-0.068	-2.6e-15	-0.15	0.039	-0.013	1	1	0.037
logvalue	0.22	-0.23	-0.35	-0.32	-0.55	-0.18	0.016	-0.35	0.13	0.026	0.037	0.037	1

Learning Pipeline

- **StringIndexer;**
- **OneHotEncoder;**
- **ML Model;**
- After that a **CrossValidator** used for find the best hyperparameter for each model and perform a **K-Fold Cross Validation**.
- Dataset sampled with the function sample(0.1, RANDOM_SEED) because Colab with the original dataset in models such that GBT or RandomForest crashes.

Results

FMRegression is the best model



Model	RMSE	R2	Adjusted R2
FM Regressor	0.886	0.945	0.944
Gradient Boosted Tree Regressor	0.998	0.930	0.928
Linear Regressor	1.231	0.894	0.891
Generalized Linear Regressor	1.231	0.894	0.891
Random Forest Regressor	1.421	0.859	0.855
Decision Tree Regressor	1.452	0.852	0.848

Future works

- Use other libraries and other Clustering algorithms.
- Usage of the Neural Network to predict the value.
- Feature augmentation with other future PREDICT datasets because in this dataset there were not some information about what industry spend those money (and for example their website to take other informations).
- A Recommendation System to recommend to a person for example a Researcher in which country he can go based on costs or the hour of work and based on how many km of distance of his place of birth and so on.