Tommaso Martorella
Data Bootcamp

**Predictive Models for Kaggle Dataset: Economic Indicators & Inflation**

The following paper will complement a co-lab notebook in which I construct four predictive regression models for a kaggle dataset. This dataset is named Economic Indicators & Inflation. It consists of data from 19 countries from the years 2010-2025, in which the correlation between economic indicators is studied. Specifically, indicators such as GDP, inflation, and unemployment are all predictors for the target of economic growth.

My research question is the following: Across a pooled panel of 19 countries (2010–2025), which core macro indicator—GDP level, inflation rate, or unemployment rate—most consistently predicts annual economic growth? I will construct a baseline model, followed by a linear regression, decision tree regressor, random forest regressor, and a neural network regressor. My goal in this analysis will be to find the model that most efficiently reduces the mean squared error, alongside other statistical measurements, in addition to determining which of the core macro indicators best predicts growth.

**Data Preliminaries:**

Loading in the data itself proved to be a bit of a challenge as you can see in the first cell. The data was structured as a .zip file and I needed to download and unzip the dataset via kaggle CLI. Then I located the csv file and converted the year to datetime, and set the date as the index. Since the dataset was downloaded via an api, I need to use .json. Kaggle also gave me an api key that I loaded using visual studio code.

**Descriptive statistics and EDA:**

The following table describes some basic statistics about the dataset.

| | count | mean | std | min | 25% | 50% | 75% | max |
|---|---|---|---|---|---|---|---|---|
| GDP (in billion USD) | 304.0 | 3383.430921 | 4283.442479 | 105.0 | 1114.25 | 2006.5 | 3625.000 | 22500.0 |
| Inflation Rate (%) | 304.0 | 4.807105 | 7.261215 | -1.2 | 1.70 | 3.3 | 5.500 | 85.5 |
| Unemployment Rate (%) | 304.0 | 6.257467 | 2.609952 | 2.1 | 4.50 | 5.5 | 7.850 | 13.7 |
| Economic Growth (%) | 304.0 | 3.060428 | 3.308527 | -14.0 | 1.60 | 2.9 | 5.125 | 11.0 |

I followed this table with a few plots, including histograms and scatter plots, that aim to show both the distribution of the data and some relationships. The histogram of the distribution of economic growth shows some problematic outliers that will later be addressed. The following scatterplot for economic growth vs. GDP level surprisingly does not show that clear of a linear

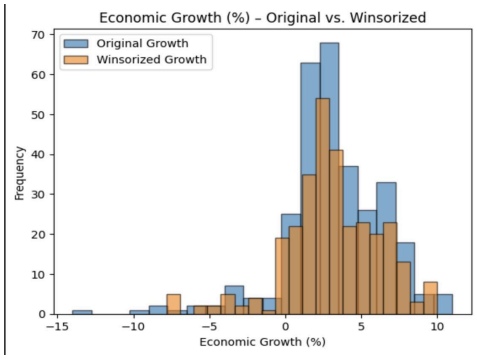
relationship(-0.055). Large economies exhibit both high and low growth. The inflation rate vs economic growth scatterplot shows a more convincing relationship but there still seem to be outliers(.158). It shows that a low inflation rate leads to higher economic growth, a predictable trend. The economic growth vs unemployment rate shows characteristics of a weak and negative linear relationship(-.205). The last EDA plot is a time series analysis of average economic growth by year. There is a notable drop in 2020 that will corrupt our results if not addressed.

Before getting into the different regression models, we need to address the notable outliers in both our target(economic_growth) and our predictors(GDP, inflation rate, unemployment rate). Within the scipy.stats.mstats library is a feature called winsorize. It will allow us to cap values beyond some quantile range, typically 1-99% so that extreme data points are not incorporated into the target. The winsorization code shows the max and min values before and after the feature was applied:

```
Before winsorization:
count    304.000000
mean       3.060428
std        3.308527
min      -14.000000
25%        1.600000
50%        2.900000
75%        5.125000
max       11.000000
Name: Economic Growth (%), dtype: float64

After winsorization:
count    304.000000
mean       3.085099
std        3.169879
min       -7.800000
25%        1.600000
50%        2.900000
75%        5.125000
max       10.000000
```

With this feature, the histogram that showed the distribution of economic growth is less distorted, as extreme values are eliminated while still retaining the integrity of the data:



Now that the target was adjusted, we also need to apply a cool feature to the predictors: Robust Scaling. Rather than centering each numeric feature on its mean, we center it on the median and divide by the interquartile range. This way, high inflation and high-unemployment do not skew the distributions. The interquartile range ignores the tails of the predictors so that recession years do not blow out the scaling factors. In addition, when we get to our MLPregressor, robust scaling will help it converge faster. This is only a preprocessing step that is applied to the predictors in order to make the inputs more numerically stable. It will not change the model's process.

Now that our target has been adjusted I labeled it as growth_winsorized. I also created an extra recession column that will provide extra clarity to some extreme values still present in our adjusted range. The goal of these models is to conduct an analysis that simple EDA cannot do. As seen above, the previous correlations seem to have weak linear relationships. The following models will combine these seemingly unrelated relationships into something with predictive power.

## Modeling and Interpretations:

### Baseline:

For our model, our X includes: country, GDP, inflation rate, unemployment rate as the predictors. Countries were added in order to distinguish relationships among different economies. Our y, the target, is growth_winsorized. Before we construct our four regression models, we need to construct a baseline that we will use as a basis for comparison. Once we defined our X and y, we train/test split the data as we have done countless times in our lecture. The baseline always predicts the mean of the training for every test point, and it has no dependence on any features. The test size equals .20, as it did in lecture. Once we fit the baseline model, we can evaluate a variety of statistical features: $R^2$(coefficient of determination), MAE(mean absolute error), RMSE(root mean square error), MSE(mean squared error)

Overall, from the baseline values for each metric( -.00237, 2.912, 3.939, 15.513) respectively, we can make some conclusions about the baseline. For $R^2$, which employs a 'predict the mean' measurement, is actually performing worse than the average growth for every observation. It captures no variation in the data which is what this metric is meant to do.

For the MAE, the baseline's prediction is off by 2.91 percentage points of growth every year. The RMSE is meant to show the large misses in the data, meaning the outliers where growth deviates strongly from the mean. The RMSE of 3.939 shows that outliers are having a stronger effect on the data than I would like. Finally the MSE of 15.513 shows that some country-year pairs sit far from the mean, due to large deviations.

The following models will harness the signals in GDP, inflation rates, and unemployment rates in order to meaningfully shrink the errors.

### Linear Regression:

The Linear Regression will now take the predictive features as inputs and produce conditional forecasts.

```
R²   = 0.389 MAE  = 1.70 RMSE = 3.074 MSE = 9.451
```

How do these values relate to the baseline? The linear model now explains about 39% of the year to year variation in winsorized growth rates. The baseline value explained zero. Out of all the ups and downs in growth across 19 countries, nearly ⅖ can be accounted for by changes in the inputs.

The MAE is cut in half by the linear model. This regression model uses the inputs as a predictive power and it reduces the MAE to 1.71.

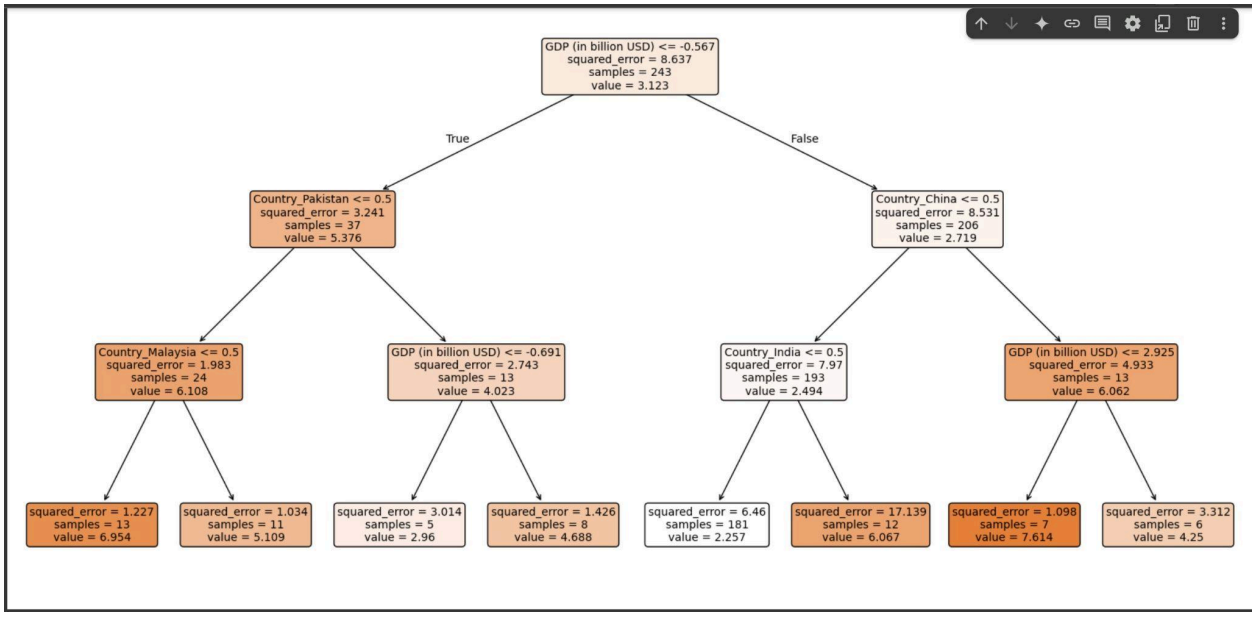
The RMSE also shows improvement, as the worst case predictors(extreme values) are now closer to actual growth, yet, occasional sharp downturns still pull the average error magnitude up to 3.074.

The mean squared error itself dropped by 39%. The linear regression definitely beats the baseline and it limits extreme misses.

In order to provide further insight on the difference in the linear models and baseline models predictions, we can conduct a t-test to determine the statistical significance of the change. From the scipy.stats library we can import ttest_rel. With an alpha of .05 as our cut off, a p-value less than that will conclude a statistical significance. A one sided hypothesis test based on squared errors of both the base and the linear models gives us a p-value of .0000401. This tells us that there is a meaningful statistical difference between the linear model and the baseline model.

**Decision Tree Regressor Model:**

How will this model differ from the linear regression? The previous model assumed that each predictor contributed in direct proportion to the outcome, meaning growth moved in a straight line under the three predictors. The decision tree employs a piecewise constant and splits the predictions based on certain parameters. It will employ conditional predictions, such as: high unemployment only hurts growth when GDP is under a certain level. While just an hypothetical example, one can already see the increased accuracy that will arise from this new model. Below is the decision tree and metrics:




How does the decision tree compare to the linear mode? The decision tree first splits the countries by GDP size, then creates more subgroups based on low performing nations compared to high performing nations. Each final leaf in the last row predicts a single growth rate. Ranging from 2.96% for smaller nations to 7.61% for mid-size Chinese GDP observations.

How do the metrics compare to the linear regression? $R^2$ increases to .412, meaning the decision tree explains about 41% of the variation in annual growth. It also cuts down MAE and RMSE, reducing big errors.

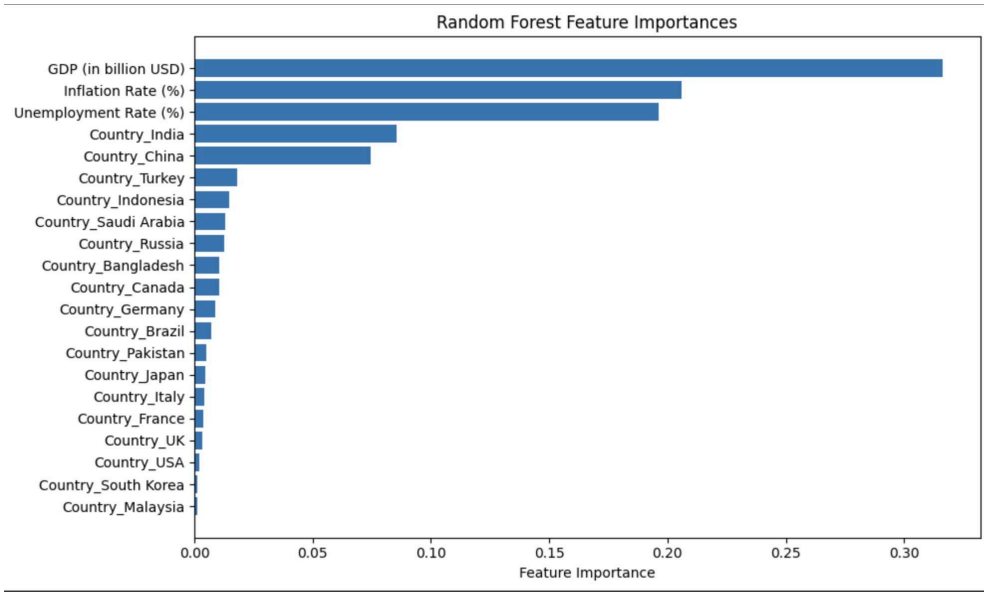
The decision tree also identifies GDP size as the strongest predictor of annual economic growth. It also takes into account the country effects next. After distinguishing based on GDP, the tree uses one-hot country indicators(Pakistan, Malaysia, India, and China) to capture country specific trends. Specifically, mid-small economies, like Malaysia and Pakistan, grow faster than other small economies. Unfortunately, inflation and unemployment are much weaker predictors of economic growth using this model.

**Random Forest Regression:**

How does this regression model differ from the previous two? It seems that the random forest regression merges the best of both worlds from the decision trees and linear model. It by default creates hundreds of smaller trees that are each trained on a random subset of the data. The final output is an average on all those tree outputs. A great benefit of the random forest regression is that it can rank each predictor in the dataset based on its reduction of error.

Before I display the rankings it is worth noting the metrics output by this model. $R^2$ performs considerably better than the linear regression and is on par with the decision tree. Overall, the metrics themselves show that the random forest regressor reduces the error of the extreme values.

The following table demonstrates the ranking of the feature importance:



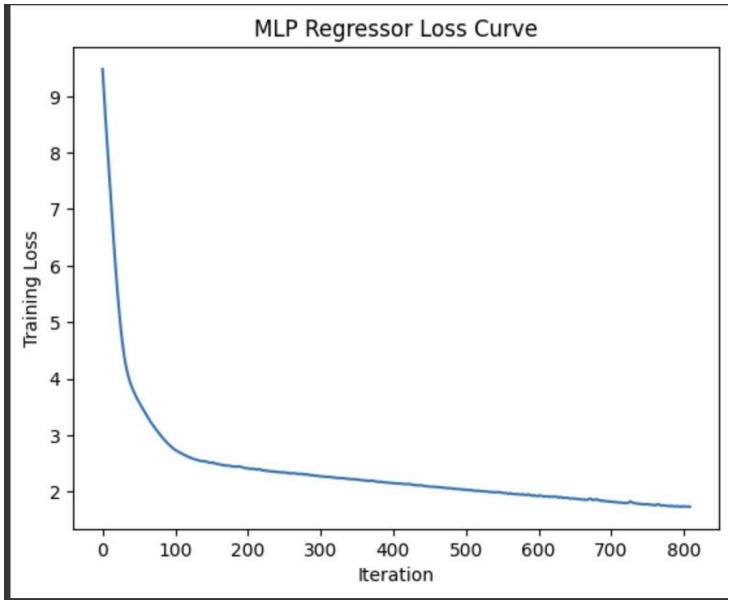

The trend is similar to the decision tree, GDP is definitely the most important feature. Yet, the random forest regressor also leverages year-to-year changes in inflation and unemployment, unlike the decision tree which ignored the other two predictors.

**Neural Network: MLP Regressor:**

Finally, how does our last model differ from our previous three? Firstly, the neural network can learn hierarchical feature combinations, which is ideal for our intention to rank the three predictors of growth. Out of all four models, the neural network brings the greatest flexibility in modeling complex non-linear patterns. However, it does require more code, increased tuning, and scaling requirements.

The metrics output display interesting results. The $R^2$ for the neural network only beats the baseline and falls behind on the other three models. It only explains 30.3% of the variation in annual-growth. The MAE also falls behind the first three models. Even the RMSE performed worst, meaning large misses in the dataset (outliers) seem to be distorting the results to a greater extent.

How can we be sure that the neural network was trained properly? After all, we want to make sure that these results are based on a well trained model that diminishes error.




We can determine this from a MLP regressor loss curve that shows a smooth decline in training error over 800 iterations. The sharp initial decline is a good sign. The training loss drops from 9 to around 2.5. The network efficiently captures the strongest signals in the data. These signals include the most straightforward relationships, including higher GDP equaling higher growth. Other trends it quickly captures are the disparity between the growth of certain countries. The smooth curve shows that the data is not overfit, as a sudden increase in the line would indicate this. This gradual decrease is typical in the MLP regressor loss curve, as further improvements in the error come with diminishing returns.

In terms of the ranking of core indicators, the network ranks GDP as the highest due to the biggest drop in loss happening immediately.

Secondly, once GDP's effect is captured, further improvements are more gradual beginning at four. This is due to the smaller, predictive bumps as a result of accounting for inflation and unemployment. Their impact is definitely present in the dataset, otherwise the loss would flatten immediately. However, they are secondary to GDP. Once you account for these three core indicators, remaining non-linear interactions barely move the test metrics. This confirms that the three predictors have the most effect.

**Conclusion:**

Now that we have analyzed four different regression models, which is the best? Random Forest Regressor emerges as the clear winner. Its statistical metrics, $R^2$, MAE, and RMSE provide the most efficiency with .42, 1.67, and 2.99 respectively.

In line with the other models, it also predicts GDP levels as the strongest, and easiest to find, predictor of year-to-year growth. In addition, it provides a clear feature ranking with inflation rate second and unemployment third. While GDP is by far the strongest, Random Forest does not forget to include the remaining two predictors for increased complexity.

While this analysis was extensive, it is important to address any next steps. Specifically for the predictors themselves, I could compute inflation volatility as a percentage. Then it could also be paired with GDP growth rate percentages. In addition, while we did omit extreme values such as the 2020 pandemic, we could conduct an analysis that aims to address whether model performances change based on pre- and post- major economic events.