Predicting Future Real GDP in the US

By Daniel Yu and Noah Galpert

Motivation and Problem Statement

Real GDP is the measure of a country's gross domestic product accounting for inflation. In other words, it's the value of all the goods and services produced by a country in a specific time period. As such, it's a good indication of a country's economy since having a high GDP value means it's goods and services are worth a lot. Thus people would want to know whether one could predict future GDP in order to determine a country's economic state and future growth. There are outside factors that can affect real GDP, such as the COVID-19 pandemic and the 2009 recession that may affect a country's economy. However, it is impossible to predict these occurrences so we will solely look at the data for trends.

In Math 104C, we learned about approximation theory where we "fit functions to given data and find the 'best' function in a certain class to represent the data." This is perfect for analyzing our real GDP data as we can find a model for the given data and project it, hence predicting what future real GDP would be. Looking at the different types of approximation functions we learned in Math 104C, we determined that a least squares approximation would be the best fit for our data, as we could compare linear and exponential polynomials.

The reason we want to find this information is so that politicians and investors can create policies and mandates that will ensure our economy stays strong and know when is a good time to invest their money into businesses and products. We hope that we can create a trustworthy function that can predict recessions and inflation.

Methods and Validation

For our project, Noah and I took data of real US GDP values for the past 75 years for every quarter of that year from https://fred.stlouisfed.org/, a trustworthy economic data site. This data came in the form of an excel file giving us a data and the value of the US GDP at the time. We then converted the data in the excel file into an array using python. As a result, we had approximately 297 points of data to work with from January 1, 1947 to January 1, 2021. We planned to use this data to find polynomials with a high degree using least-squares regression while minimizing our least squares error.

Since least squares approximation can be used to approximate linear and exponential regressions, we had to calculate both and find out which model was a better fit. We first looked at linear least squares regression. Using gauss elimination, we can obtain the coefficients of the polynomial function, which we can then use to calculate the error between the data and the polynomial. We wrote code that allowed us to obtain a polynomial of any degree and plot it with the original data, so we could compare how similar our function looked. We then coded a function to calculate which degree gave us the smallest error for any degree up to the 50th as python was unable to calculate anything higher. Through this method we obtained our linear least squares approximation with minimal error.

Then for exponential regression, we calculated the polynomial for two equations, $y = be^{ax}$ and $y = bx^a$ as these are the two forms that the approximating exponential function can have. To calculate the values for a and b, we used the normal equations

where $b = a_0$ and $a = a_1$ and coded an error function for exponential functions. Finally, we

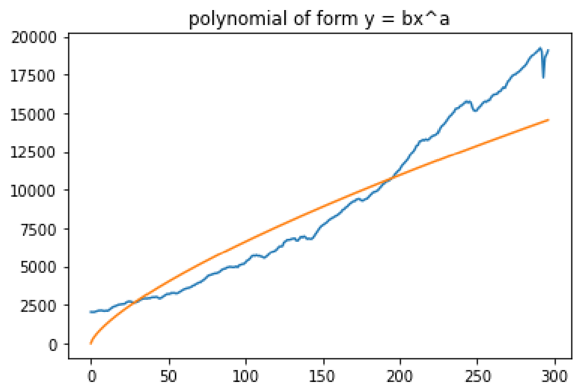
$$a_0 = \frac{\sum_{i=1}^{m} x_i^2 \sum_{i=1}^{m} y_i - \sum_{i=1}^{m} x_i y_i \sum_{i=1}^{m} x_i}{m\left(\sum_{i=1}^{m} x_i^2\right) - \left(\sum_{i=1}^{m} x_i\right)^2}$$

$$a_1 = \frac{m \sum_{i=1}^{m} x_i y_i - \sum_{i=1}^{m} x_i \sum_{i=1}^{m} y_i}{m\left(\sum_{i=1}^{m} x_i^2\right) - \left(\sum_{i=1}^{m} x_i\right)^2}$$

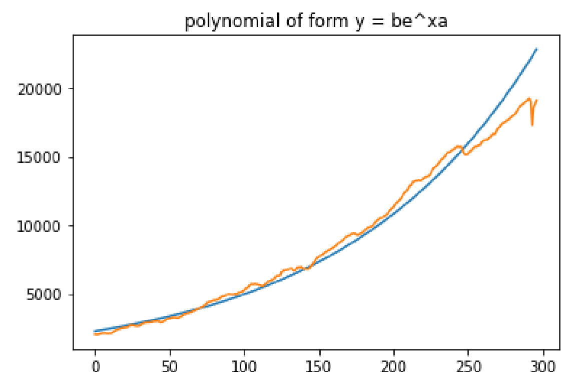
compared our total and mean errors to find out which model returned the smallest one.

Results

As mentioned above, we looked at linear and exponential least squares regressions. In the following graphs, the blue line represents the polynomial function while the orange graph represents the given data. We will give two errors for our functions, total error and mean error.

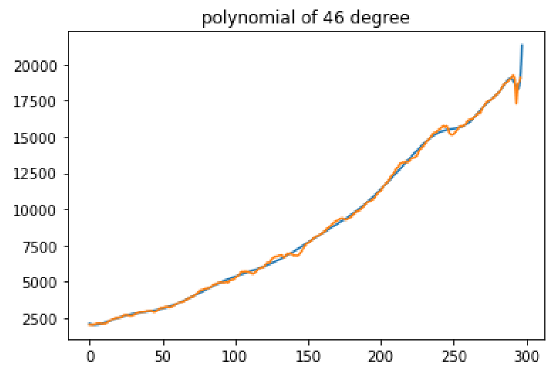


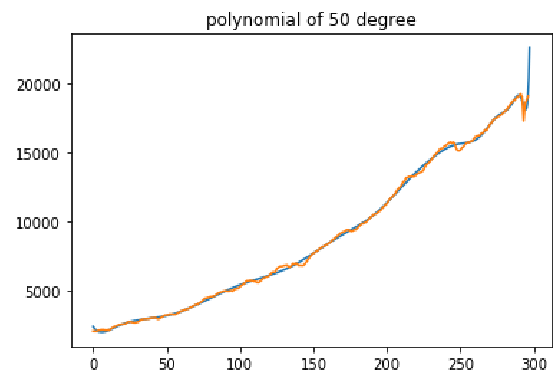
Total error is the sum of the errors at each point while mean error is the average of the errors at each point. For our exponential regressions, clearly $y = bx^a$ is not a good match for our data as it does not follow the trend at all, resulting in a large error. Thus the real GDP data is not a logarithmic function. However, $y = be^{ax}$ does follow the trend so looking at the error, we have a total error of: 264774772.78673112 and a mean error of 891497.5514704752. This may look high, but since our data itself goes up to 20,000, it is not as bad as it seems, so it is possible that the real GDP is on an exponential model.



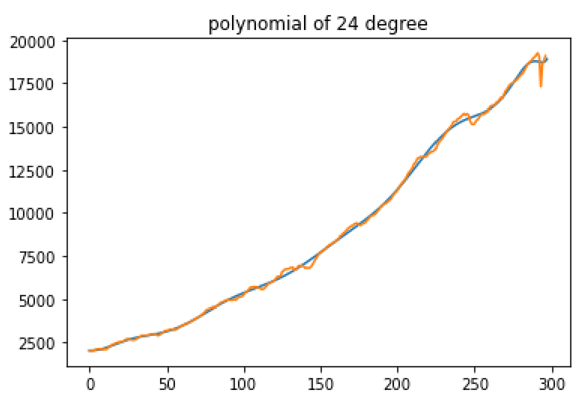
Looking at linear regression, using the fine minimal error function we found out that a polynomial of 46th degree gave the minimum error for degrees between 0 and 50. As we can see, the function follows the data tightly and the total error we get is: 7615739.369965735, while the mean error is: 108.511520735054, much smaller than the exponential function errors. This suggests that real GDP follows a linear model. One would think that having a higher degree polynomial would make the error smaller, so I added a graph for a linear polynomial of the max degree, 50. As we can see, at the end of the polynomial, the spike is higher than the polynomial of 46th degree while the beginning does not accurately follow the data. This is also shown by our errors for the 50th degree polynomial which were 8114067.207023892 for total error and 114.48242882199337 for mean error.

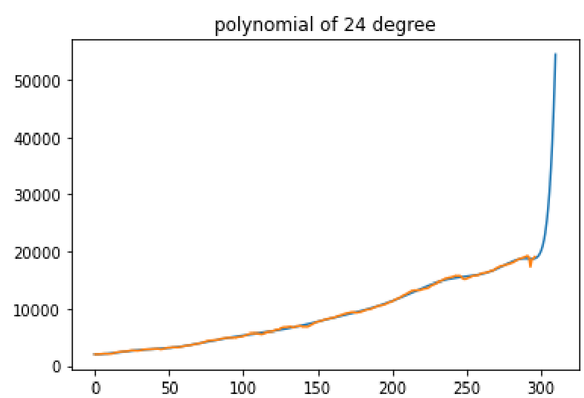
However, both the 46th and 50th degree polynomial have a steep jump at the end,





so we will analyze the polynomial with the next smallest error, which was the 24th degree. This follows the trend a little nicer than the polynomial of the 46th degree , even though it has a higher error of 9140292.902556535 for total error and 118.40322385757311 for mean error, so we will use this polynomial to predict the





future. Analyzing for future data, I expanded the graph to see the real GDP ten quarters in the future. Surprisingly, the function increases sharply, suggesting that the function is not very good for predicting the future. For example, calculating the real GDP for quarter 298 we get a value of 18774.206090927124, which is close to the 297th quarter value of 19087.568 but at quarter 305, we get a value of 27483.845664978027 which is a large increase. It seems that there is an over-fitting problem in the 24th degree polynomial model.

Thus, even though we concluded that real GDP follows a linear least squares approximation, we can't really use this approximation to predict future GDP values. It does seem like a good fit for short-term analysis perhaps for one to two quarters, but in the long run least squares analysis cannot be used to give an accurate estimate of the future. We think neural networks might be a very useful tool in this area, as it could generate strong time series analysis based on limited data via classifying, clustering and recognizing patterns.