Applied Analytics Frameworks & Methods 2
Group 10: King of Data
Phoebe Chen, Fei Qiu, Yujie Wang, Mengxi Tian

Using Google Trends Keywords to Predict U.S. Unemployment Rates

Statement of the Problem

The unemployment rate is defined as the percentage of jobless workers in the labor force, and is an indicator of hardship for U.S. families. Especially during the COVID-19 pandemic, the unemployment rate is crucial for the government to predict how many families need assistance, impacting their decisions on stimulus budget and unemployment insurance funds. However, because it is a lagging indicator, and rises or falls according to changing economic conditions, it is very difficult for the government to predict unemployment rates. In this project, we will utilize Google Trend data on certain keywords that hint at economic conditions to explore how search frequency of these keywords may be effective in predicting unemployment rates.

Since unemployment rate is a key component for the US economy, it is essential to first understand which data would act as effective predictors. Hence, our research question is whether google trends can predict unemployment rates. Specifically, this study aims to find which words or combinations of words can be best at predicting unemployment rates. In a recent study on the effect of the sentence “file for unemployment”, the word ‘unemployment’ was shown to have effect on the Initial Unemployment Insurance Claims (Goldsmith-Pinkham et al.). Thereby, the influence of other words like “unemployment”, “compensation”, “jobs”, “salary”, and “claims” on unemployment rates become the major points of research in this study.

Data

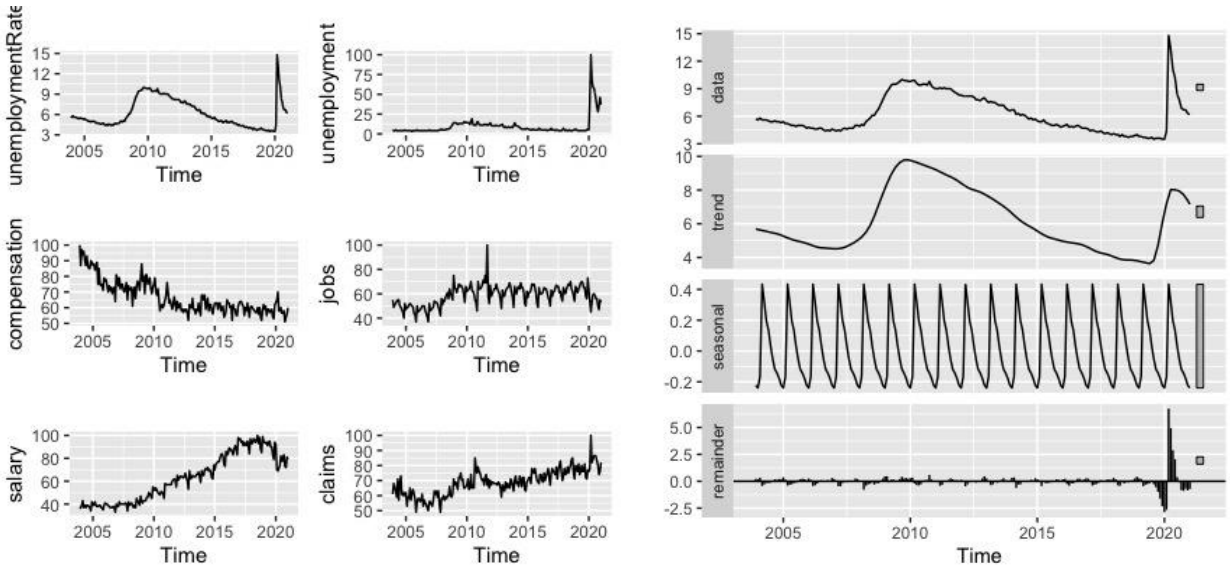
For this project, we will be using data from two sources: the federal government and Google Trends. The unemployment rate data is downloaded from the Department of Labor website as a csv file and then imported into R.

From Google Trends, we will be using monthly search frequencies of the keywords “unemployment”, “compensation”, “jobs”, “salary”, “claims”, as well as the topic “unemployment” (which contains multiple keywords related to the topic). We chose to also include a topic as a proxy to explore whether topics are better predictors than keywords. Data for these search terms and topic are available from January 2004 to the current month, but we will only be using data from January 2004 to December 2020. Monthly data were pulled directly from Google Trends website with an R API for each keyword. Rather than actual search count, Google trends data is structured as indexes assigning 100 to the period with highest search volume, and others are scaled on a range between 0 and 100 based on their volume proportional to the highest. For monthly search data, the R API automatically re-scales our dataset centered around the most searched period between January 2004 and December 2020.

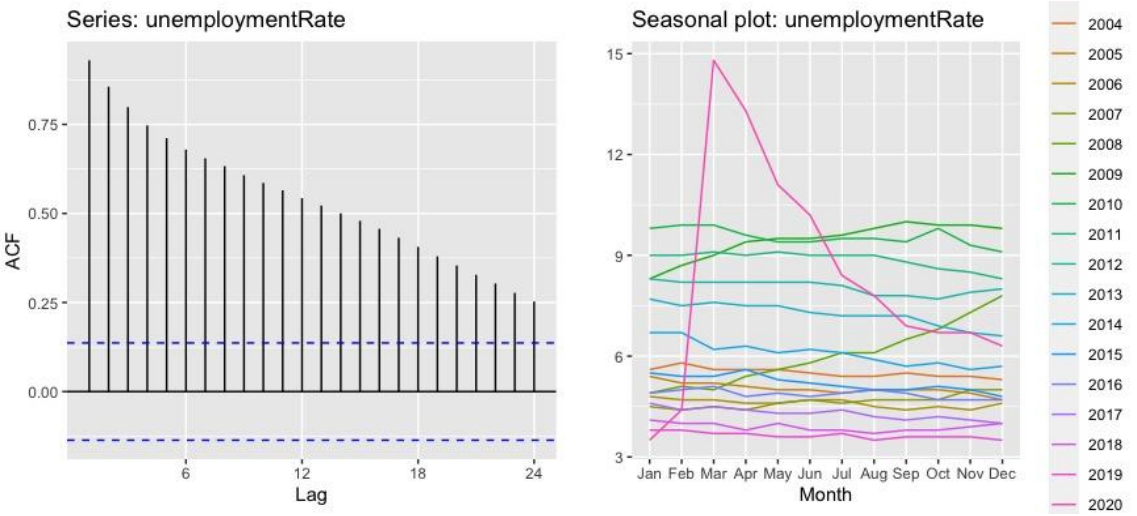
Applied Analytics Frameworks & Methods 2
Group 10: King of Data
Phoebe Chen, Fei Qiu, Yujie Wang, Mengxi Tian

Exploratory Analysis

Firstly, both unemployment rate as well as Google Trends keyword data are time series data as both datasets are indexed on time.

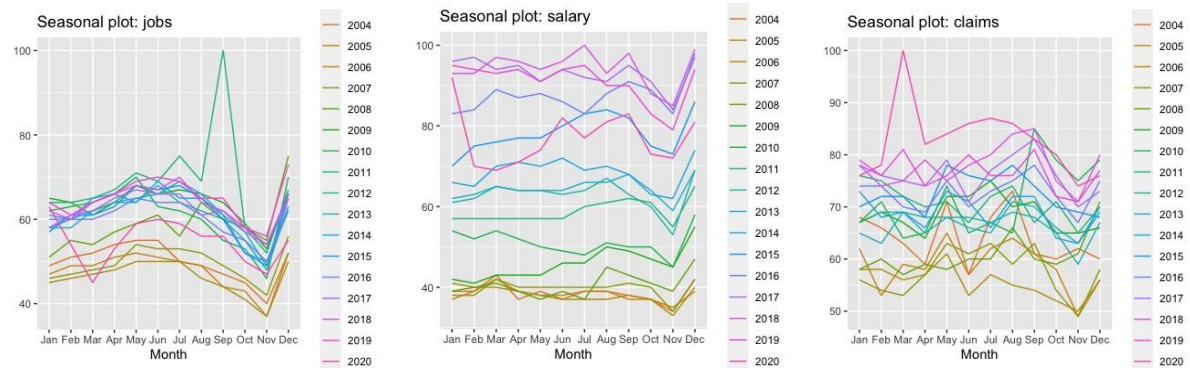


Looking specifically at unemployment rate, while the actual data may exhibit a multi-decade cyclical pattern; however, since we are looking at data from 2003 to 2020, there seems to be no seasonal or cyclical pattern. This is confirmed as the number of differences required for a seasonally stationary series (nsdiffs function) was equal to 0. On the other hand, the autocorrelations plot indicates that the maximum autocorrelation occurs at a lag of 1, suggesting that the unemployment rate is most correlated with the rate of the previous year. This is consistent with the seasonal plot, which shows that there was very low variation in unemployment rate within a year (with the exception of 2020), but high variation between years.



Looking at the plots of the keyword search terms over time, the word “unemployment” seems to be stationary before the pandemic in 2020. However, three of the keywords exhibit trends.

Specifically, “compensation” has a decreasing trend while “salary” and “claims” have an increasing trend. On the other hand, “jobs” exhibits a seasonal pattern with a spike in 2012. This is further shown in the seasonal plot for the keyword as each year (except 2012 and 2020) had very similar patterns with a large drop of search frequency in November followed with a spike in December. This pattern at the end of each year is actually also shown in the keywords “salary” and “claims”.



For these reasons, our study will primarily focus on time series. However, neural networks are also useful as algorithms are very general and adaptive. Furthermore, it doesn’t require any feature engineering as it can apply directly to raw data. Thus, while we primarily focused on time series, we also attempted to predict unemployment rates with neural networks to see if better results could be achieved.

Analytical Technique

We used several analytical techniques in time-series to manipulate, analyze and visualize the data including simple forecasting methods, exponential smoothing models, and ARIMA. To develop the model and test the accuracy of the forecast, we used roughly 76% cutoff to split the train sample, and predicted the unemployment rate on the test sample containing 24% of the data pool. As a result, we compared the RMSEs and residuals in the Ljung-Box test generated from various models to evaluate the accuracy of the prediction and stationarity of its residuals. The detailed discussion will be addressed in later sections. In the end, based on the time-series regression we developed, we can forecast the future unemployment rates. Below are the list of analytical techniques we utilized in this project.

Sampling Methods
Since our dataset is based on time, we did not split our data into train and test sets randomly. Instead, we split based on time. Since Google Trends data began from January 2004, we set our train data set for all keywords and unemployment rate to be from January 2004 to December 2016. This means that the test sets were data from January 2017 to December 2020. While unemployment data dates back to 1948, however Google Trends data only begins in

January 2004. Thus, unemployment data was also split in the same way as Google Trends data, and unemployment rates before January 2004 was not used.

Forecasting Models

- Simple Forecasting Methods

  For simple forecasting methods, we used average methods, naïve methods including seasonal naïve method, and drift method.

- Exponential Smoothing Models

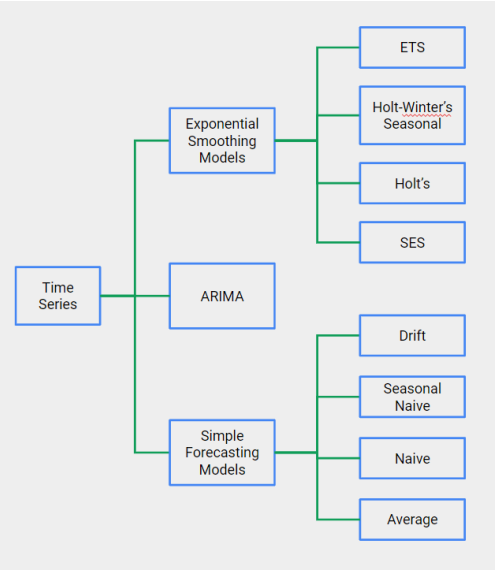
  To extend simple forecasting models, we also tested on several exponential smoothing models, including simple exponential smoothing model and those with trends such as Holt's, Holt's method with damping, Holt's seasonal method, and ETS models.

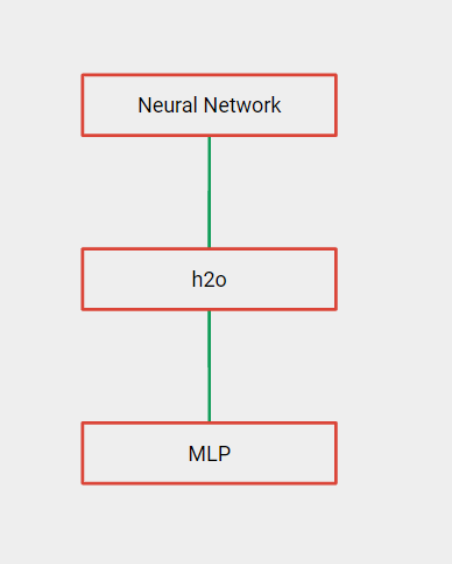
- The Autoregressive Moving Average (ARIMA) Models

  Since an ARIMA model assumes the data is stationary, the first step before applying ARIMA, is to stabilize variances by using the Box-Cox transformation. And we tested both auto and manual ARIMA to achieve the best result (lowest RMSE), and we will discuss it in the next result section.

Neural Network

As discussed in the Exploratory Analysis section, the prediction of unemployment rates could involve complex correlations hidden in the raw data, and deep learning excels at various areas. Therefore, we also decided to use the neural network, specifically multiple hidden layers perceptron from the H2O package, to test if it can help to further increase the accuracy.





Applied Analytics Frameworks & Methods 2
Group 10: King of Data
Phoebe Chen, Fei Qiu, Yujie Wang, Mengxi Tian

Analysis Results

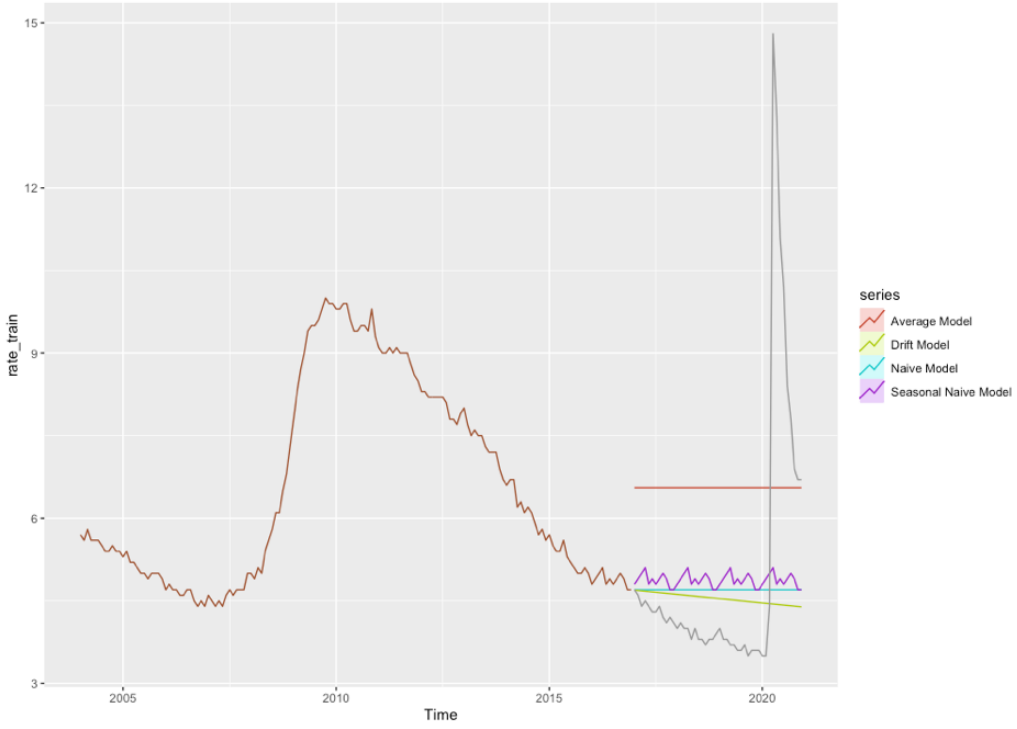
Simple Forecasting Methods

Given our goal is to forecast future unemployment rate, we used Root Mean Square Error (RMSE) as our primary metric for model comparison. For Time Series models, we also examined the error term by performing the Ljung-Box test and checking residuals plots. The best result from our simple forecasting models came from seasonal naïve model with RMSE of 2.512552. Below is a table summarizing our model results sorted by lowest RMSE, followed by a visualization of their forecasts:

Summary of accuracy:

	ME	RMSE	MAE	MPE	MAPE	MASE	ACF1	Theil's U
seasonal_naive_model	0.1333333	2.512552	1.620833	-10.988653	27.84275	1.739195	0.7955617	1.226402
naive_model	0.3083333	2.536237	1.508333	-7.039420	24.53362	1.618480	0.7949405	1.206115
drift_model	0.4663978	2.603335	1.456989	-3.660815	22.38756	1.563386	0.8008125	1.197776
average_model	-1.5467949	2.954658	2.667788	-49.288749	58.88650	2.862605	0.7949405	1.937379

Comparison with actual test data (colored in grey):



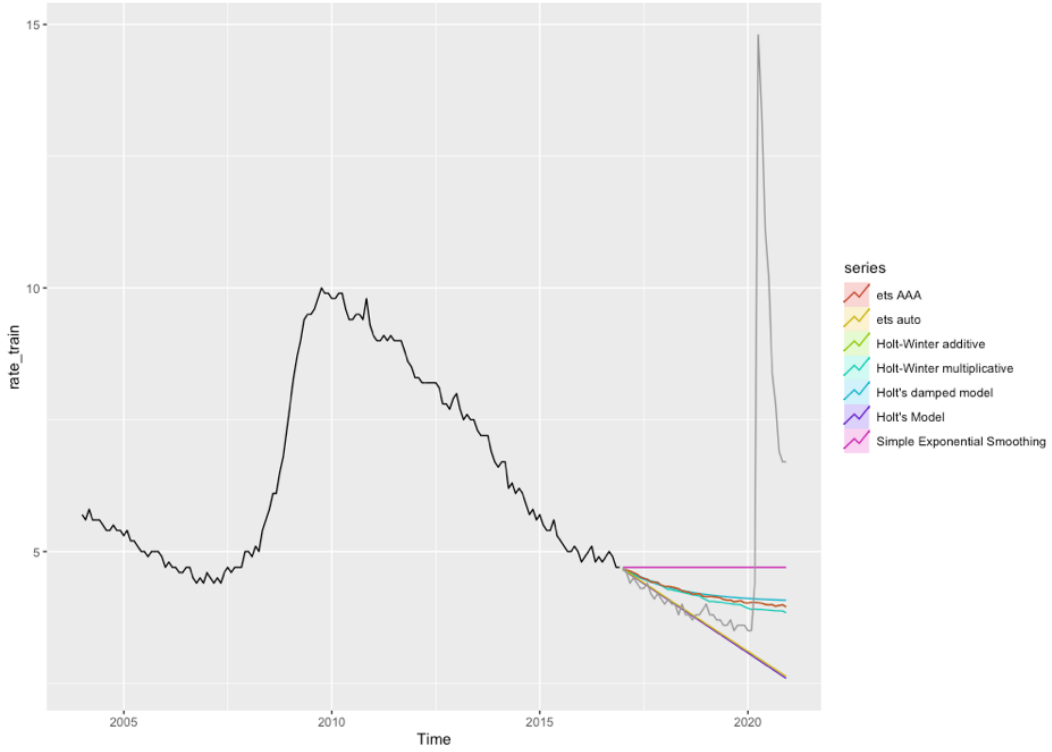
Compared with the test data colored in gray, it is sufficient to conclude none of the simple models performed particularly well in forecasting unemployment rate, even prior to the huge spike in 2020 due to the large gaps between the actual data in the test pool and the predicted rates. The Seasonal Naïve model as shown above is insufficient to capture short-term trending behavior in unemployment rate, as a result we turn to more robust models such as Holt's-Winter or ETS models.

Exponential Smoothing Models

To cover every aspect of exponential smoothing, three types of models were used. For Holt's-Winter, models with additive, multiplicative, and with damping parameters were fitted separately to capture potential trend and seasonality components. An ETS model with additive error, trend, and seasonal component was also fitted, followed by the ETS model automatically fitted by R based on the AICc score. Lastly, a Simple Exponential Smoothing model was fitted to contrast the results from all types of exponential smoothing we covered.

	ME	RMSE	MAE	MPE	MAPE	MASE	ACF1	Theil's U
ses_model	0.3083333	2.536237	1.508333	-7.039420	24.53362	1.618480	0.7949405	1.206115
holt_damped_model	0.7610265	2.682692	1.299704	3.083436	17.34675	1.394615	0.8020039	1.183088
hw_additive	0.7923728	2.715356	1.308119	3.675711	17.25015	1.403645	0.8052266	1.190977
ets_aaa	0.7923728	2.715356	1.308119	3.675711	17.25015	1.403645	0.8052266	1.190977
hw_multiplicative	0.8497975	2.753799	1.296594	4.881677	16.62371	1.391278	0.8078959	1.201877
ets_auto	1.3617833	3.160491	1.428052	15.468938	17.08619	1.532336	0.8320486	1.346522
holt	1.3843732	3.174785	1.440143	15.963827	17.32233	1.545310	0.8324910	1.352877

While Simple Exponential Smoothing model achieved the lowest RMSE among all exponential smoothing models, it is far from the best model once we look closer below:

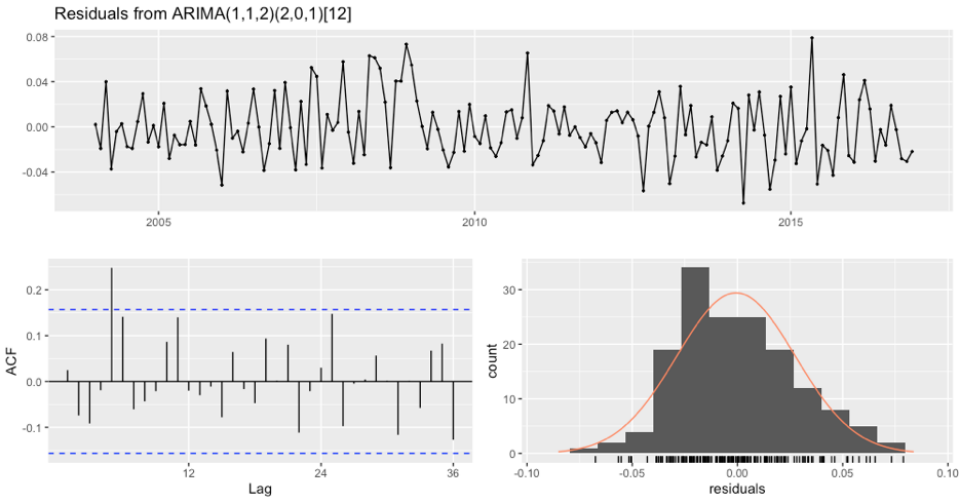


Both Holt's models and the automatically-selected ETS model perform almost identically on test data, with forecasting errors being minimal at early stages but gradually growing larger. Other exponential smoothing models with various additive errors perform similarly with their forecasts clustered together. The Simple Exponential Smoothing in fact has the worst forecasting power according to visualization above. However it was able to achieve the lowest RMSE score because the huge spike in 2020 offsets some of the errors between test data and predicted data before 2020.

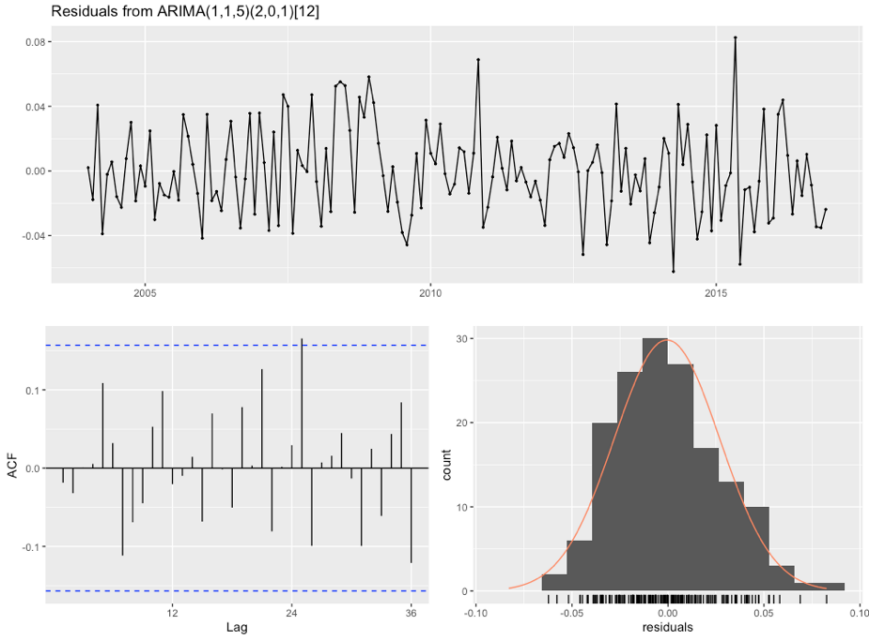
While in general exponential smoothing models achieve better forecasting results compared to simple naïve models, none of the models fitted so far reflect the jump in unemployment rate caused by the COVID-19 pandemic, with most models still trending downward. Therefore, we turned to the ARIMA models for a more rigorous approach.

The Autoregressive Moving Average (ARIMA) Models

With the help of auto.arima() function in R, we first identified the best ARIMA model as ARIMA(1,1,2)(2,0,1)[12] using the unemployment rate itself as y series and achieved an RMSE of 2.5297483. However while examining the error term of the model, Ljung-Box test returns a significant p-value of 0.04779 which indicates that the residuals do not resemble white-noise, with ACF chart showing a significant spike at lag 5.



To address this issue, we manually update the Moving Average term q to 5 and again check for residuals:



As shown above, ACF now doesn't show significant spike until lag 25, and Ljung-Box test returns a non-significant p-value of 0.4326 which indicates stationarity.

As discussed in our proposal, we decide to incorporate each keyword data obtained from Google Trends as x series in order to improve our existing ARIMA model. Additionally, the topic "unemployment" is added as a proxy for grouping multiple keywords as a single x series. For the results to be meaningful, we use a cutoff point of 5% for the p-value from Ljung-Box tests to ensure that the residuals of each model resemble white-noise, and ACF are closely examined to identify potential spikes before lag 24.

Below is a summary of the results:

	ME	RMSE	MAE	MPE	MAPE	MASE	ACF1	Theil's U
key_jobs	0.4616974	2.449776	1.300151	-2.821431	19.68844	1.395095	0.7773268	1.147862
key_salary	0.4914070	2.453036	1.280560	-2.116135	19.10642	1.374073	0.7770000	1.143281
key_compensation	0.4892639	2.457947	1.287626	-2.205320	19.25151	1.381656	0.7779112	1.145289
unemployment_rate	0.4941371	2.458282	1.284248	-2.089873	19.15705	1.378031	0.7778084	1.144709
key_claims	0.5208141	2.466255	1.271810	-1.492172	18.73181	1.364684	0.7782774	1.142783
key_topic	0.5034768	2.484308	1.296206	-2.020372	19.30567	1.390862	0.7790266	1.156537
key_unemployment	0.4958492	2.490102	1.304269	-2.237294	19.51376	1.399514	0.7788787	1.162321

"Jobs" achieved the lowest RMSE of 2.449776 by a thin margin among all keywords and topic. While keyword "unemployment" has been used in several previous studies discussed in our proposal, it performed the worst amongst the selection in terms of RMSE, and serving as our approximation for a combination of keywords, "unemployment" as a topic was the second worst regressor.

Visualization of the forecasts from all ARIMA models show that the predicted unemployment rates initially decrease but quickly rise upwards to accommodate the spike in 2020. Many keywords such as “jobs” and “salary” produce indistinguishable forecasting results as shown above. In addition, during the testing process of our ARIMA models on different machines, we noticed that their RMSEs shifted by a fractional amount even when parameters were controlled and a seed was set, which was enough to change to order of our top two keywords as the difference is minuscule. On the other hand, both models using “unemployment” as the keyword and topic initially predicted a decrease in the unemployment rate as the actual rate spikes up, and rises as the actual rate drops. As a result, we conclude that the keyword “jobs” is our best regressor closely followed by “salary”, as those models can quickly adapt to the regime shift in 2020.

Neural Network

For the neural network part, our team applied H2O to the dataset. The original dataset was cut into three parts: training, validation and test dataset (0.4,0.4,0.2 respectively). The training and validation are used to tune the model, leaving the test dataset to evaluate the performance. We firstly set all the variables as input and from the graph we can see that the rmse for the neural network model tuned at this time is 0.2165119.

```
> rmse(pred6[1],test_h2o$unemploymentRate)
[1] 0.2165119
```

However, for each time of tuning, the result of RMSE is slightly different. Yet, it is clear that the RMSE is better than the result from time series. Since neural network does not have specific weights on the variables but only weights on the layers, the performance of each word cannot be identified. Hence, neural networks can improve the performance of the model but it cannot identify the individual weights of the variables. The graph below shows the specific parameters applied in the neural network.

	layer	units	type	dropout	l1	l2	mean_rate	rate_rms	momentum	mean_weight	weight_rms	mean_bias	bias_rms
1	1	5	Input	0.00 %	NA	NA	NA	NA	NA	NA	NA	NA	NA
2	2	25	Rectifier	0.00 %	0.000090	0.000050	0.004348	0.004285	0.000000	0.006803	0.281103	0.486057	0.085359
3	3	25	Rectifier	0.00 %	0.000090	0.000050	0.007289	0.006988	0.000000	0.001019	0.200984	0.994841	0.016372
4	4	25	Rectifier	0.00 %	0.000090	0.000050	0.004880	0.006195	0.000000	0.012496	0.204167	0.998285	0.005627
5	5	25	Rectifier	0.00 %	0.000090	0.000050	0.110518	0.291907	0.000000	0.002121	0.200516	1.000430	0.003212
6	6	1	Linear	NA	0.000090	0.000050	0.087155	0.265753	0.000000	0.104530	0.213571	0.000517	0.000000

Further Analysis

One of our main challenges in reducing RMSE while achieving realistic predictions is that volatility of unemployment rate is non-constant overtime. While our ARIMA models integrated with keywords did perform better, they were unable to fully adapt to the shock incurred in 2020 responsively. To address this issue, we can consider incorporating Autoregressive conditional heteroskedasticity models to adjust the shifting variance in further analysis.

In addition, other economic data such as those closely related to unemployment can also be included as our regressor. Inflation or its inverted version for example can be our prime candidate given the series’ relationship according to the Phillips curve.

Conclusions & Recommendations

In this project, we aimed to find which words or combinations of the words that can be best at predicting unemployment rate. After exploring the data, we decided to use time series since the data was indexed on time and showed some seasonal and trending patterns. First, the team tried multiple time series models including simple forecasting and exponential smoothing. Both the simple forecasting and exponential smoothing did not show desirable RMSEs (around 2.5). Hence, we used ARIMA models, which were the best and most responsive model. Furthermore, adding the keywords greatly reduced the RMSE; and it was found that the keywords “jobs” and “salary” had the lowest RMSEs. Then, H2O neural network was applied to prove the finding. However, neural networks can only improve the performance of RMSE as there are no variable weights inside the neural networks model. In conclusion, the key word “jobs” was found to be the best predictors for the change of unemployment rates.

While our models did find that “jobs” and “salary” were the best keyword predictors; however, there may be other better keyword predictors that were not tested in this study. Our results also contradicted with the results of some previous researches. Specifically, while we found that “unemployment” was the weakest predictor in our model, a previous research mentioned in our previous literature review from Goldsmith-Pinkham and Sojourner created a model that explained up to 92.9% out-of-sample variance with this keyword (Goldsmith-Pinkham and Sojourner). Furthermore, no models will be able to predict sudden and large changes caused by, for example, the pandemic. Thus, our models will perform better with shorter forecasting periods (nowcast) in the future when the unemployment rate is less volatile.

Hence, while predicting unemployment rate will bring many benefits to the government, we would not recommend the government to make any decisions solely based on one set of models. However, if economic conditions remain relatively stable in the future, we would recommend the government to incorporate these keywords to improve their existing time-series models. Specifically, if there is an increase use of the keywords “jobs” and “salary” on public platforms such as Google, we would recommend the government to be cautious of a possible increase in unemployment rates and prepare for any changes.

References

Federal Reserve Economic Data. “Unemployment Rate.” St. Louis Fed, https://fred.stlouisfed.org/series/UNRATE. Accessed 8 March 2021.

Goldsmith-Pinkham, Paul, and Aaron Sojourner. “Predicting Initial Unemployment Insurance Claims Using Google Trends.” github.io, 2020, https://paulgp.github.io/GoogleTrendsUINowcast/google_trends_UI.html. Accessed 7 March 2021