Stockholm School of Economics
Departament of Economics
5350 Master Thesis in Economics
Academic Year 2014-2015

Forecasting the U.S. Unemployment Using Google Trends

Rokas Narkus (40607)

Abstract

Aim: To analyze whether Google Trends data can be used as a leading predictor to forecast the U.S. monthly unemployment rate.

Methods: Selected benchmark ARIMA models based on Box-Jenkins (1976) methodology and in-sample performance (measured by information criterion). Augmented these models by adding explanatory exogenous variables: Initial jobless claims (IC) and Google Trends Index (GI). Then created out-of-sample forecasts and evaluated whether models including GI outperformed benchmark models. Performed several robustness checks to ensure validity of the results

Data: Unemployment data taken from Bureau of Labor Statistics; Google data taken from Google Trends, Initial jobless claims data taken from Federal Reserve Economic Data. Data spans from 2004-01 to 2015-03.

Conclusions: Models which included Google Trends Index outperformed benchmark model. Although improvements are modest. The "best" model included both IC and GI showing that both are useful leading indicators and contain information which does not fully overlap.

Keywords: Forecasting, Unemployment rate, Google Trends, Initial Jobless Claims
JEL Classification: C22, C53, E24, E27

Supervisor: Rickard Sandberg
Date submitted: May 14, 2015
Date examined: May 27, 2015
Discussant: Emanuel Brendarou
Examiner: Kelly Ragan

Contents

List of Figures and Tables

Tables

Tables in Appendices

Figures

Figures in Appendices

1 Introduction

"Prediction is very difficult, especially if it's about the future."

Niels Bohr, Nobel laureate in Physics

Niels Bohr made this comment as to forewarn that it is relatively easy to come up with a model which fits existing data. Although perfectly fitting existing data can be nothing more than what forecasters call over-fitting problem, when despite in-sample fit model fails to predict the future. For this reason, models can only be accepted after scrutiny of out-of-sample evaluations.

This process delineates proper modeling methodology. Although proper methodology is only one side of the proverbial coin; another is the inputs which are fed to the model. In many instances process stands or falls based on quality of inputs. This thesis aims to evaluate quality of relatively new internet data - Google Trends. The data provides aggregated information about search patterns in a given geographic area. This allows to glimpse into changing patterns of internet use. Ideally, there is a link between our internet behavior and real life outcomes. Say, if people search for flu like symptoms then it is likely that they have a flu or when people search for unemployment benefits then it is likely they got laid off. Thus, this thesis tries to exploit the link between online search patterns and economic outcomes. Another interesting caveat is that Google trends data comes in weekly frequency as opposed to longer reporting lags observed in official statistics. As Choi and Varian (2009b) argue that reporting structure mismatch this allows to predict the present. One can think of using internet data to check on economic pulse of a given country.

Choice of target series is only bounded by common sense and creativity but in this thesis I chose to predict the U.S. monthly unemployment rate. The choice of target series was driven by two key reasons. First, that the series is relevant from economic and/or social perspective. Second, that the series have identifiable pathways how online search patterns can be linked to actual real life behavior. In my opinion, unemployment satisfies both criteria. Unemployment as a social phenomenon needs little justification as its one of the primary statistics used to describe country's economic and social health. Although the link between internet searches and

unemployment is "noisy". On the one hand, high unemployment can lead to increased online search activity for jobs. On the other hand, the more people search online the higher the chance is to find a job and lower the unemployment rate. Thus, there is apparent link between search data and unemployment but the pathway is not that clear. Despite this it is important to proceed with the exercise as all real life data has flaws but we have to come up with solutions to mitigate them. The choice of country was driven by other two key reasons. First, Google search engine must be the leading search engine in the country. Second, the country should have high internet penetration rate. Google is the dominant search engine worldwide and only handful of countries avoided Google's grip, namely Russia, South Korea, Japan and China (Alexa, n.d.). Moreover, most of advanced economies have high internet penetration rates. So it all boils down to that the U.S. is the biggest economy, thus has the largest impact on the rest of the world. Moreover, it has high reporting standards so it can be seen as a "lower-bound" for forecasting performance improvement. The more reliable and the more frequent the data is the lower potential for forecasting improvement. Although it is still important to look at such countries as internet queries capture information which is different form official figure and reporting lag is still an issue. As hinted above, the U.S. satisfies the criteria for choice of country. In 2014, U.S. had quite high about 85% internet penetration rate (World Bank, n.d.) and Google was the leading internet search engine with share of 67% (comScore, 2014). Thus, one can expect that Google data is representative of the U.S.

In 2008 for the first time Google made aggregated search query data publicly available by launching Google Insights (Google, 2008). The data dates back to 2004 January. Thus, research community already had a chance to get their hands dirty and test whether the data can be used to improve predictions. Ginsberg et al. (2009) published first article which successfully used Google data to forecast outbreaks of flu. Although Google data started to attract more attention only when Google's own chief economist Hal Varian along with Choi published articles showcasing Google data potential to "predict the present" (Choi & Varian, 2009a, 2009b). Following these articles various papers appeared which successfully employed Google Trends. For this thesis of particular importance are papers which aimed to forecast unemployment. Askitas and Zimmerman (2009) were first to employ Google data to forecast

unemployment in Germany. Others did so for other countries: Italy (D'Amuri, 2009), Israel (Suhoy, 2009), Turkey (Chadwick & Sengul, 2013) and the U.S. (D'Amuri & Marcucci, 2012).

The last paper is of particular interest as it forecasts unemployment in the U.S. Although there are important departures between methodologies used in this paper and in D'Amuri and Marcucci (2012). They perform a "rat race" with different models and evaluate results using the Model Confidence Set (MCS) test as described by Hansen et al. (2011). This approach allows to say whether models which include Google Trends outperform benchmark models. Although such approach fails to quantify how much better models with GI are. As well as provide little guidance how one should select a single best model or a group of best model which should be used to make forecasts. For instance, D'Amuri and Marcucci (2012) include different week of a month results for GI find that the best model is ARMAX(2,2) with lagged value of Google Indicator for week 4. This is completely sample dependent as it is hard to justify why this particular model should be selected to forecasts future. For this reason, I adopt methodology inspired by Swallow and Labbé (2013) who employed Google data to forecast car sales in Chile. Similar methodologies were used in aforementioned papers which predicted unemployment (Askitas & Zimmermann, 2009; Chadwick & Sengul, 2013). Another aspect is that this thesis has over 2 more years of observations. This is particularly important as their sample is highly effected by 2008 financial crisis.

The whole data series which spans from 2004-January to 2015-March is split into training and testing sets. Training set spans 95 periods (months) from 2004-January to 2011-November. This accounts for about 70% of all observations. Then, the best AR and ARIMA models are selected as benchmarks based on Box-Jenkins methodology and on best in-sample fit as measured by information criteria. Once benchmark models are established they are augmented to include exogenous variables which should help to improve forecasts of unemployment rate. First exogenous variable is Initial jobless claims (IC) which is shown to be a leading indicator for forecasting unemployment (Montgomery et al, 1998). Other exogenous variables include Google Index variables (GI). Two key Google series include information associated with keywords: "jobs" and "unemployment benefits". At any given analysis five models are considered. First model is base model which includes only ARIMA process (AR can be nested into ARIMA). Second model is augmented with IC. Those are benchmark models. Third is augmented with GI

for "Jobs". Fourth is augmented with GI for "Jobs" and "Unemployment benefits". Last one is *Combined* model which includes IC and GI variables. These are targets models which predictive accuracy is of interest in this thesis.

Box-Jenkins methodology and unit root pretesting shows that differentiation provides a better fit. Thus, models are estimated with integration order of 1. $ARIMA(3,1,0)$ and $ARIMA\ (3,1,1)$ are selected as benchmark models. Then models are recursively estimated and one-step-ahead forecasts are made for hold-out testing sample. Afterwards predictions are compared by computing standard error metrics: Root Mean Square Error (RMSE) and Mean Absolute Error (MAE). More formal testing is provided by applying Diebold and Mariano (1995) test.

The results indicate that models which include Google Indicators, indeed, outperformed benchmark models. Yet, the improvement is rather small from 2.4% to 2.9% depending on model specification. Model which included only GI for *Jobs* performed best in terms of RMSE while the *Combined* model performed best in terms of MAE. This shows not only that both IC and GI have some predictive power but also that information captured by these variables do not fully overlap. These results are in-line with results reported by D'Amuri and Marcucci (2012). Despite improving results it cannot be shown that there is statistically significant improvement as measured by *DM* test. These results support Wu & Brynjolfsson (2014) claim that Google data only modestly improves predictive accuracy over simple autoregressive models. These findings were robust to changing sample sizes/periods, changing estimation technique from recursive to rolling-window, and employing full-sample estimation.

The rest of the paper is organized as follows: *Section 2* provides overview of existing literature associated with unemployment forecasting and usage of Google Trends data. *Section 3* describes the data used in the paper. Also describes pre-treatments and transformations. *Section 4* describes methodology. *Section 5* provides results and interpretation of results. To ensure validity of results in *Section 6* robustness checks are performed. Lastly, *Section 7* concludes.

2 Literature review

In this section theoretical background is provided. For this thesis two important branches of literature are important, namely forecasting unemployment and employing Google Trends. First, I provide an overview of forecasting unemployment literature with focus on conventional modelling techniques. Afterwards I provide an overview of employing Google Trends and specifically in setting when forecasting unemployment.

2.1 Forecasting unemployment

As unemployment is one of the biggest social issues it is no wonder that economists attempt to forecast unemployment. Typically, unemployment rate forecasts are made using one of three approaches. The first approach is based on theoretical relationship between output growth and unemployment rate change, in economics literature known as Okun's law. The second approach is based on forecasting using labor flows. Last approach is based on historical time-series data. Time-series analysis may incorporate additional variables such as leading indicators (e.g. Google Trends can be example of leading indicator).

Okun (1962) suggested that there is relationship between unemployment and country's production, namely slowing down economy leads to higher unemployment. Okun's law has natural appeal of simplicity and it has received empirical support even in cross-country studies (Lee, 2000; Moosa, 1997). Still, it suffers from endogeneity problem as relationship causality is not clear. Say, it is likely that higher unemployment would lead to economy-wide slowdown. Knotek (2007) analyzed how useful is Okun's law. He concludes that it is a statistical relationship rather than a structural feature of the economy. Yet, the evidence suggest that despite variation the relationship is pretty robust and it can be useful as forecasting tool - provided that instability is taken into account. Despite this Okun's law is usually used as a barometer to check whether forecasts are make sense but usually central bankers and policymakers rely on other two approaches to get unemployment estimates.

Barnichon et al. (2012) shows that forecasting unemployment rate using labor force flows gives better results than conventional time-series forecasts. Authors expressed unemployment rate as a mismatch between unemployment inflows and outflows. They proposed simple analogy to explain idea behind it. Unemployment at a given time can be though as the amount of water in

bathtub, a stock. Given an initial water level, the level at some time in the future is determined by the rate at which water flows into the tub and the rate at which water flows out of the tub. When the flows are constant the water level remains constant. Otherwise, it changes based on which flow is stronger. Thus, the inflow rate and the outflow rate provide information about the future water level - or in this case, level of unemployment. Authors find that even using simple models where a person can be either employed or unemployment (i.e. no option to leave labor force) superior forecasts are obtained when compared to forecasts made from Survey of Professional Forecasters.

Despite shortcomings the most popular method is time-series analysis based on unemployment rate series. Typically, researcher tries to improve forecasting accuracy either by testing different model specification or by adding additional explanatory variables. Montgomery et al. (1998) forecast unemployment using time-series analysis. They showcase that monthly initial jobless claims can be used as a leading indicator to predict quarterly unemployment rate of the U.S. Similar methods are employed in papers forecasting unemployment using Google Trends. Another aspect of time-series analysis is to check whether predictive accuracy can be improved by model choice. In same study Montgomery et al. (1998) show that nonparametric models outperform linear time-series models. This is supported by Golan and Perloff (2004) who show that nonlinear, nonparametric models outperform traditional linear models. This thesis main focus is on testing the inputs rather than the methods. In other words, to test whether additional explanatory variables improve predictive accuracy. Thus, the next segment turns discussion to employing Google Trends data in research setting.

2.2 Employing Google Data

In 2008, Google made aggregated search query information publicly available for the first time by launching Google Insights (Google, 2008). Given that Google has been a dominant search engine worldwide for a while, the data has attracted attention from academia. Ginsberg et al. (2009) published first article using Google data to estimate weekly influenza activity in the U.S. They recognized that people usually search for flu-like symptoms online and used this information to predict influenza activity. Interestingly, this is one of examples when academic

community makes an impact on a corporation; as Google incorporated similar methodology to estimate selected disease activities (e.g. flu) and tracks it real-time basis[1].

Google Trends data started to attract more attention when Google's own chief economist Hal Varian along with Choi published articles showcasing Google data potential to "predict the present" (Choi & Varian, 2009a, 2009b). Choi and Varian (2009b) showcased potential of the data by predicting car sales and home sales in the U.S. Moreover, they showed how Google Trends can be separated by regions and used to predict travel destinations. For instance, Google Trends can help to predict how many visitors from Germany will come to the U.S.

Edelman (2012) surveyed literature related to using internet data for economic research. He highlights that internet data is a valuable complement to structured datasets (e.g. government agency data) as internet data has lower cost of acquisition and is available at higher frequency. Moreover, data is very broad ranging from information about prices from Amazon or Ebay to aggregated search information from Google Trends. Thus, it is no wonder that he shows that internet data has broad applications. Specifically Google Trends were used to predict social behaviors, disease activity and various economic variables.

Google data was successfully employed in predicting some social behavior. Billari et al. (2013) use web-search data related to fertility as a leading indicator to predict birth rate of the U.S. Baker and Fradkin (2014) develop a job-search activity index to analyze the reaction of job-search intensity as a response to change in unemployment benefit duration in the U.S. Vosen and Schmidt (2011) compare private consumption forecasts in U.S. between survey-based indicators and Google trends data. Wu and Brynjolfsson (2014) made an extensive evaluation of Google data quality for different forecasting applications. They re-examined Google flu data and find that improvements over benchmark autoregressive models are only modest. However, they find that Google data is very useful at predicting novel events such as the opening weekend box-office revenue for feature films, first-month sales of video games, and the rank of songs on the Billboard Hot 100 chart.

Several papers focused on forecasting unemployment using Google Trends. Askitas and Zimmermann (2009) were first to employ Google data to Germany. They identified keywords

[1] https://www.google.org/flutrends/

which should be related to unemployment. One group of keywords were related to searching for unemployment agency or office. Another group were related to searching jobs online through popular job websites (e.g. Monster Jobs). They find that models which include Google trends performed better. However, they only provide in-sample fit which is not ideal but necessary evil when time span of data is relatively small. Chadwick and Sengul (2013) were first to employ Google data to an emerging market - Turkey. They have more robust predictive accuracy testing with in-sample estimation and out-of-sample evaluation. Moreover, they use slightly more interesting approach by taking 6 different keywords related to unemployment and extract one factor by using principal component analysis. This factor is used as an explanatory variable. The model which includes this Google unemployment factor performed the best in terms of predictive accuracy. Other researchers focused on different countries: Italy (D'Amuri, 2009), Israel (Suhoy, 2009), and the U.S. (D'Amuri & Marcucci, 2012). Some papers focus on specific sub-group as youth unemployment in France (Fondeur & Karamé, 2013). All of papers report that Google data improves predictive accuracy.

As in this thesis, D'Amuri and Marcucci (2012) employed Google Trends to forecast the U.S. unemployment rate. They perform a "rat race" with different models and evaluate results using the Model Confidence Set (MCS) test as described by Hansen et al (2011). This approach allows to answer whether models which include Google Trends outperform benchmark models, although the approach fails to quantify how much better models with Google data are. For instance, D'Amuri and Marcucci (2012) include different week of a month results for Google variables find that the best model is ARMAX(2,2) with lagged value of Google variable for week 4. This is sample dependent and could not be known before. As Diebold (2013) argues that without a real out-of-sample performance testing, researchers can always perform data mining techniques to find a model which performs well in out-of-sample. Thus, this thesis aims to fill that research gap by providing a step by step model selection procedure and quantifying Google Trends potential using this procedure.

3 Data

3.1 Sources

This paper utilizes three data series. The first is unemployment rate series which is the target series to forecast. Other two series are used as exogenous variables which should help to predict the unemployment rate. One series is initial jobless claims which is shown to be leading indicator (Montgomery et al., 1998). Another is Google internet queries index which is primary interest of this paper.

Unemployment: Seasonally adjusted U.S. monthly unemployment rate (LNS14000000) released by Bureau of Labor Statistics (BLS) is used as a proxy for unemployment. Unemployment rate shows the share of people who are unemployed from the total work force. BLS defines unemployed if person is jobless, looking for job and available for work. To get the estimates BLS conducts a monthly survey called Current Population Survey (CPS). Each month during reference week (usually the week which includes the 12th of the month) Census Bureau employees contact 60,000 eligible sample households and ask about the labor force activities. Unemployment data dates back to 1967.01. As Google data unfortunately is only available from 2004.01 unemployment data is used from 2004.01 to 2015.03.

Initial Claims: Seasonally adjusted U.S. weekly initial jobless claims (IC) released by U.S. Employment and Training administration. Initial claims is a measure of the number of jobless claims filed by individuals who seek to receive state jobless benefits. The measure is sometimes used as indicator for short-term economic health. IC is often used as a leading indicator for estimating unemployment (Montgomery et al., 1998). Data used from 2004.01.10 to 2015.04.18

Google data: Not seasonally adjusted weekly Google index (GI) data related to job searches performed through the Google website is taken from Google trends.

Google trends analyzes a percentage of Google web searches to determine how many searches have been done for the terms you've entered compared to the total number of Google searches done during a given time **(t)** and geographical area **(r)**. Absolute values of the index are not publicly available. Google normalizes the index $\boldsymbol{GI_{t,r}}$ to range from 0 to 100. In the week in

which most searches were made $\boldsymbol{GI_{t,r}}$ is equal to 100. The data spans from 2004.01.10 to 2015.04.18. All queries are restricted to the U.S.

Following D'Amuri and Marcucci (2012) I choose my main keyword - "*jobs*". Google trends provide information about key word used in the search. For example, "*jobs*" would include combinations such as "*craiglist jobs*" or "*walmart jobs*". This means that "*jobs"* keyword covers a wide range of job-related activities. In order to capture only job-related searches I exclude famous non-job related queries, namely exclude "Steve Jobs" queries. Google trends also allows to restrict keyword searches for certain categories. Thus, where possible I restrict queries for *Jobs* category. The main restriction criteria is that there should be sufficient number of observations if there are not sufficient number of queries then $\boldsymbol{GI_{t,r}}$ is equal to 0.

Alternative keywords include: "*unemployment benefits*" and "*job center*". Same keywords were used as alternatives in previous research. These keywords should capture variation inherently different from "*jobs"* keyword. "*Unemployment benefits*" and "*job center*" queries are directly linked to person losing a job while searching for "*jobs"* can be seen as secondary and by-product search. For instance, if person is fired, thus unemployed then he is more likely to search online for a new job. Thus, higher *jobs* related queries can reflect higher unemployment. Although if online search is not futile then it is reasonable to assume that the more person searches for job the higher likelihood that he will find one. Thus, "jobs" impact on unemployment is not clear.

Lastly, instead of looking only at keyword for queries I look into quotation subjects. This is still a beta feature. However, the idea behind it is appealing. Quotation subjects should measure interest in topic rather than specific keywords. For analysis "*Unemployment*" and "*Employment*" quotation subjects are selected.

3.2 Descriptive statistics

Table 1 provides descriptive statistics for time series used in the analysis. There are 135 unique observations spanning from 2004 January to 2015 March. Target series is unemployment rate. Unemployment rate range from 4.40 (in March, 2007) to 10.00 (in October, 2010) with mean of 6.80. The peak of unemployment rate coincides with aftermath of 2008/09 financial crisis. Another variable of interest is Initial jobless claims (IC) ranges from 275'500 (in February, 2015) to 645'000 (in February, 2009).

	Mean	Median	Min	Max	Std.Dev	Count
unemployment rate	6.80	6.20	4.40	10.00	1.84	135
IC	378'880	347'250	275'500	654'500	822.70	135
GI "employment"	69.60	66.81	58.18	86.94	7.84	135
GI "jobs"	70.65	71.45	59.70	85.77	5.87	135
GI "unemp. benefits"	22.59	19.00	7.00	66.00	14.94	135
GI "unemployment"	33.72	32.91	15.16	72.29	15.73	135
GI "job center"	68.79	67.99	55.03	84.38	8.70	135

Table 1 Descriptive statistics for period 2004-01 to 2015-03. All variables are monthly frequency and seasonally adjusted.
Source: Bureau of Labor Statistics, Google Trends, U.S. Employment and Training Administration

The main predictive variables of interest are Google Index variables. In total 5 variables were extracted but the main focus is given to GI for *Jobs* and *Unemployment benefits*. First, of all they are intuitive and seemingly should be connected with unemployment and previous research (see D'Amuri and Marciano 2012) have used them. Other GI variables are used as robustness checks (see graph in Appx B. for visual representation of GI development).

GI for Jobs ranges from 59.70 (in January, 2006) to 85.77 (in January, 2009) with mean of 70.65. While GI for Unemployment benefits ranges from 7.00 (in January, 2006) to 66.00 (in June, 2010) with mean of 22.59. By inspecting at dates of troughs and peaks it can be seen that Google indicator variables mimic unemployment rate development. This is illustrated by Figure 1 which shows normalized values for unemployment rate and GI for Jobs. It is clear that GI for Jobs mimics unemployment rate series although the magnitude of variation is lower.

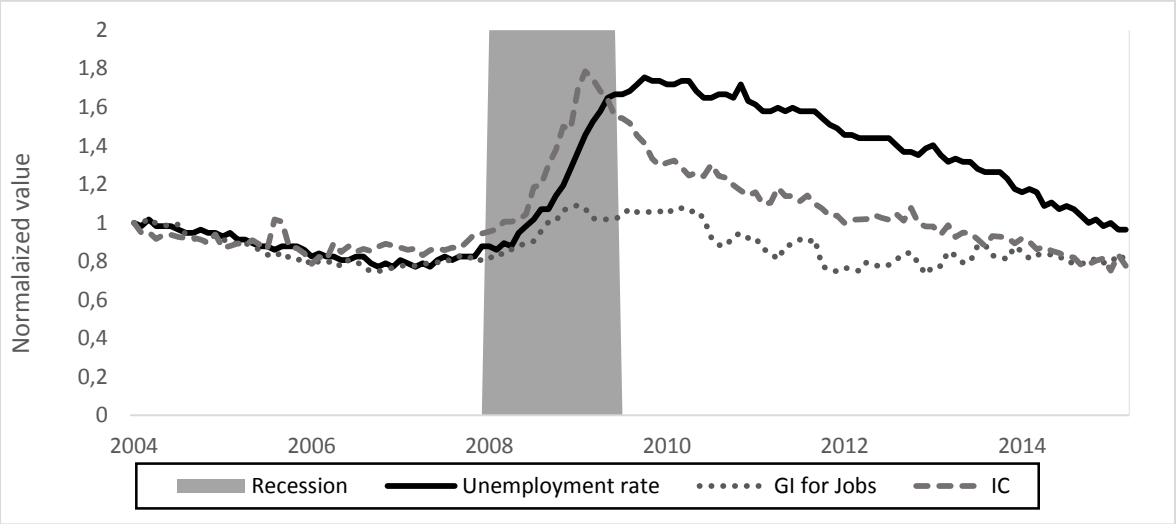


Figure 1 Normalized series for unemployment rate, GI for jobs and Initial jobless Claims for period 2004-01 to 2015-03. Series are normalized by dividing series values by 2004-01 value. Made by Author. Source: Bureau of Labor Statistics, Google Trends, U.S. Employment and Training Administration

Moreover, closer visual inspection shows that IC development seem to precede unemployment rate development. Thus, should be used as leading indicator. While GI for Jobs rather mimics the target series and has only contemporaneous effect. This affects the choice of modelling as can be seen later in methodology.

3.3 Transformations

When it comes to modelling time series data there are few key issues to take care. That data is correctly aligned (i.e. that data refers to the right date). That data is consistent (i.e. that either all data series are seasonally adjusted or not). Lastly, any data transformations which either improve fit (e.g. log transformations) or allow better inference power (e.g. difference in presence of unit root for use of ARIMA models).

Data alignment: Data alignment is determined by BLS methodology for calculating unemployment. To estimate unemployment BLS conducts a monthly survey. Sample households are contacted during reference week (usually the week which includes 12th day of a month) and are asked about their employment status. To qualify as unemployed person has to be willing to work but unable to do. Moreover, person has to actively search for work in past 4 weeks (reference week included). Thus, given week which has 12th day of a month is called a "reference" week and it represents the last week of previous month. Afterwards 3 weeks before the reference week are accounted for that month. Lastly, averages of 4 weeks are taken to get monthly figures.

For example, weekly Google indicator for jobs is released on 2004-02-14th and as weekly series started from 2004-02-08th this means that this week included 12th day of a month, thus it's the "reference" week. Next, add three weeks before the reference week (dating back to 2004-01-18) and average the results to get monthly estimate for previous month (in this case for 2004-January).

Seasonal adjustment: Seasonal adjustment is a statistical technique designed to eliminate periodic predictable swings in the series which happen due to changes in seasons. U.S. monthly unemployment rate and Initial Jobless claims series are seasonally adjusted while Google data is not. Google data must be seasonally adjusted to make variables comparable and useful for forecasting.

The monthly Google indicators are adjusted for seasonality using X-13-ARIMA-SEATS. This software was developed by United States Bureau of the Census and is used for most of seasonal adjustments made in the U.S. X-13-ARIMA-SEATS along with TRAMO-SEATS are the methodologies promoted by EU's seasonal adjustment guidelines (Eurostat, 2009). Therefore, it is best methodology to approach seasonality. Although it's still not perfect as series are considered in isolation. Thus, seasonality components differ from series to series.

Appendix A provides graphs of seasonal adjustment for selected GI variables ("jobs", "unemployment" and "job center").

Unit root: Another concern is whether data has a unit root as presence of unit root affects choice of forecasting model. When it comes to forecasting U.S. unemployment rate literature splits between approaches. For instance, Rothman (1998) induces stationarity with a log-linear de-trended transformation. Yet, Montgomery et al. (1998) model U.S. monthly unemployment rate with levels and argue that unit-root non-stationarity is hard to justify for the U.S. unemployment rate because inherently unemployment rate can only vary within limited range. Similarly, Koop and Potter (1999) argue that as unemployment rate is bounded between 0 and 1, it cannot exhibit global unit root behavior. They argue that due to bounded nature of unemployment rate series unit-root pre-testing is not necessary.

Diebold and Killian (2000) show that unit-root pretesting essentially has no drawbacks and often even improves forecasting results. Moreover, choice of unit root testing methodology has some importance for forecasting accuracy as different methodologies have different power.

In this instance power means how likely we can reject the presence of unit root. Stock (1996) show that the asymptotically more powerful DF-GLS test of Elliot et al. (1996)may further improve forecast accuracy. Thus, Augmented Dickey Fueller test (Dickey & Fuller, 1979) and DF-GLS tests (Elliott et al., 1996) are considered.

Firstly, optimal number of lags is determined by running augmented dickey-fuller test and afterwards checking whether residuals have remaining autocorrelations (tested by Box-Ljung test). Box-Ljung test shows that after adding 5 lags/augmentations model produces residuals which are not serially correlated (Box-Ljung test p-value is 0.6488). Thus, selected number of augmentations is 5. Augmented Dickey Fueller test gives test-statistic of -1.9275 which fails to reject null hypothesis (as critical value for 10% is -2.57). However, DF-GLS test gives test-statistic of -1.77 and rejects null hypothesis at 10% level (critical values for 5% and 10% are -1.94 and -1.62 respectively). Similarly, D'Amuri and Marcucci (2012) report that using alternative unit root testing: Range Unit Root test (see Aparicio et al, 2006) they fail to reject on unit root on the long sample (from 1967.01 to 2011.06) but can reject for shorter sample of 2004.01 to 2011.06.

There is no clear cut answer from unit root tests. Therefore, decision whether to integrate the series will be made on in-sample performance as measured by information criteria.

Other transformations: Explanatory series are natural log transformed as it provided best in sample in-sample fit.

3.4 Limitations

With any forecasting exercise one needs to be aware that figures are only real life approximations and errors are part of the game. There is **estimation error**. For instance, individuals who are searching for a job through the internet may not be randomly selected among job seekers. Moreover, the Google indicators capture all search activity, thus it includes searches performed by unemployed and employed.

Data alignment error comes from dealing with several data series. Different data series arrive at different points in time. Thus, aligning the series so that it makes sense is left to discretion of the researcher.

These limitations should introduce some bias in our GI; nevertheless such a bias if anything should reduce precision of the forecasts.

4 Methodology

4.1 General procedure

Methodology is inspired by methodology used in Swallow and Labbé (2013). First of all, data is split into two parts: training sample with 95 periods (~70% of all periods) and testing sample with 40 periods (~30%). The training sample is used for benchmark model selection. Afterwards these benchmark models are augmented to include exogenous variables: IC, GI or both. There are two benchmark models: the base (ARIMA) model, and ARIMAX (exogenous variable IC). Models which include GI are of particular interests as this study aims to test GI data quality. There are three GI augmented models: ARIMAX (with GI for *Jobs*), ARIMAX (with GI for *Jobs* and GI for *Unemployment Benefits*) and *Combined* model (with GI for *Jobs*, GI for *Unemployment Benefits* and *IC*). After models are established then out-of-sample forecasts are made and performance evaluated. Lastly, different specifications are used to test whether the results are robust.

Further sections provide in-depth explanations of the steps taken in the study.

4.1.1 Benchmark models

In this section I cover base benchmark model selection. Unit root pre-testing shows that there may target series may be not stationary. Obviously unemployment rate is naturally bounded as series cannot exceed certain value. Yet, forecasting is about finding a model which best fit the data. Following similar studies (e.g. Carrière-Swallow & Labbé, 2013) autoregressive integrated moving average model (ARIMA) is considered.

Box-Jenkins (1976) methodology is used to determine optimal lag length and whether to integrate the data. The steps to apply their method can be split into three parts:

1. Inspect data to check properties of given series. Main aim is to inspect whether series is stationary. Otherwise, Box and Jenkins (1976) suggest to differentiate the data series and repeat the procedure.
2. Select appropriate AR order (i.e. number of lags). This can be done by inspecting partial auto-correlation function (PACF).
3. Perform residual diagnostics to check whether residuals resemble white noise and are normally distributed. Otherwise, some information is not fully captured by specified models and hence can be improved.

It is possible to select simple AR models by just visually inspecting ACF and PACF. Although there is too much room for interpretation when models includes moving average component. Still Box-Jenkins methodology gives guidance that series should be differentiated or integration of order 1 should be used and that lag value should be 3.

In order to select best ARIMA model information criteria is employed. ARIMA models can be expressed as:

$$y_t = \alpha_{ARIMA(p,d,q)} + \sum_{i=1}^{p} \beta_{ARIMA(p,,d,q),i}\, y_{t-i} + \sum_{j=0}^{q} \theta_{ARIMA(p,d,q),j}\, \epsilon_{t-j} + \epsilon_t \qquad (1)$$

, where y_t – represents the target series at time t. $\alpha_{ARIMA(p,d,q)}$ – represents the intercept. p – represents the order of autoregressive model part. d – represents order of integration part. q – represents order of moving average pat. $\beta_{ARIMA(p,,d,q),i}$ – represents coefficient for a lagged unemployment rate at time $t-i$. $\theta_{ARIMA(p,d,q),j}$ – represents coefficient for a moving average component at $t-j$. Lastly, ϵ_t – white noise term.

For ARIMA model, the autoregressive specification order $p = 1, \dots, 3$ and moving average specification order $q = 1, \dots, 2$ are considered to determine the best fitting model based on AIC in test sample data.

4.1.2 Target models

Once base benchmark model is established exogenous variables are included to create ARIMAX models. Table 2 summarizes the models used in the study. As discussed in data section there are two series: IC and GI. Models which include GI variables are Target models as this paper aims to evaluate their performance.

Inc. variables	Benchmark		Target		
	Base (a)	IC (b)	Jobs (c)	Jobs + Benefits (d)	Combined (e)
$ARIMA$	✓	✓	✓	✓	✓
Exogenous variables					
IC_{t-1}		✓			✓
GI_Jobs_t			✓	✓	✓
$GI\_Benefits_t$				✓	✓

Benchmark ARIMAX model which includes only IC can be expressed as:

$$y_t = \alpha_2 + \sum_{i=1}^{p} \beta_{2,i}\, y_{t-i} + \sum_{i=0}^{q} \theta_{2,j}\, \epsilon_{t-j} + \eta_{2,i} IC_{t-1} + \varepsilon_t \quad (2)$$

Target models with GI only (3) and (4) can be expressed as:

Table 2 models' specifications used in the paper

$$y_t = \alpha_3 + \sum_{i=1}^{p} \beta_{3,i}\, y_{t-i} + \sum_{i=0}^{q} \theta_{3,j}\, \epsilon_{t-j} + \gamma_{3,i} GI\_Jobs_t + \varepsilon_t \quad (3)$$

$$y_t = \alpha_4 + \sum_{i=1}^{p} \beta_{4,i}\, y_{t-i} + \sum_{i=0}^{q} \theta_{4,j}\, \epsilon_{t-j} + \gamma_{4,i} GI\_Jobs_t + \kappa_{4,i} GI\_Benefits_t + \varepsilon_t \quad (4)$$

Combined target model which includes GI and IC can be expressed as:

$$y_t = \alpha_5 + \sum_{i=1}^{p} \beta_{5,i}\, y_{t-i} + \sum_{i=0}^{q} \theta_{5,j}\, \epsilon_{t-j} + +\eta_{5,i} IC_{t-1} + \gamma_{5,i} GI\_Jobs_t + \kappa_{5,i} GI\_Benefits_t + \varepsilon_t \quad (5)$$

Notation is the same as for (1) except that $\eta_{j,i}, \gamma_{j,i}$ and $\kappa_{j,i}$ refer to coefficient values for a given model j associated with IC_{t-1}, GI_Jobs_t and $GI\_Benefits_t$ respectively

As can observed from (3), (4) and (5) GI variables are estimated at time t while IC variable is estimated at time $t-1$. The reason behind is that different information is encompassed in different variables. For instance, IC represents a number of people who applied for unemployment benefits in a given month. People who apply for it would not be considered as unemployed by BLS as they still worked during that month. For this reason IC is seen as leading indicator for unemployment and lagged value provides a better fit. The story is different for Google indicators. As Google indicators represents the internet search activity and it is reasonable to assume that internet search activity during the month of interest best represents information related to unemployment.

Note that choice to estimate contemporaneous effect of Google indicators on unemployment means that auxiliary model is necessary to make GI estimates for next period. Following D'Amuri and Marcucci (2012) I choose $AR(1)$ auxiliary model to get $\widehat{GI}_{t+1}$

4.2 Testing

4.2.1 Estimation

There are two main estimation techniques in forecasting literature: recursive scheme and rolling window scheme. Recursive scheme takes advantage of whole sample. Although if underlying data series behavior changes (e.g. structural breaks), then rolling window scheme is superior as by definition it gives importance to recent data points.

Under the recursive scheme, the model parameters are estimated of R periods, where R corresponds to the length of the training sample. In this case it corresponds to period from January 2004 to December 2014. Using this estimates the first forecast is made for one-step-ahead (for period R+1). Then the model is re-estimated by expanding the sample to include the next period's information and new forecast is made. Say t represents an end date for the estimation sample, then $t \in \{R+1, \dots, T+1\}$, where $T+1$ is the number of periods in full sample. This method allows to use all the information available at time t, as such parameters estimates are expected to converge to in-sample estimates as number of periods used in estimation approaches number of periods in full sample.

Under rolling window scheme, the technique begins with same steps as aforementioned recursive scheme. The difference is that estimation sample is fixed of size P. Thus, the beginning date shifts along with the end date t, where $t \in \{R+1, \dots, T+1\}$. Usually, researchers employ a 24-month rolling window (e.g. Carrière-Swallow & Labbé, 2013 and D'Amuri & Marcucci, 2012). Chen (2005) and Giacomini and White (2006) suggest that rolling window estimation scheme should be used when parameters are unstable as this scheme forecast accuracy and test power.

To take advantage of all data available, the recursive scheme is used as the main estimation scheme. However, rolling window scheme is employed as a robustness check. Rolling window size is set to be equal to initial training sample size.

4.2.2 Evaluation

In order to evaluate the forecasts some evaluation criteria must be selected. One-step-ahead forecast error of the model is denoted by $\hat{e}_{i,t+1} = y_{i,t+1} - E_t[\hat{y}_{i,t+1}]$. In this notation i denotes a model. Since the models are nested and they are used to forecast same period then scale dependent errors can be used to measure predictive accuracy. Hyndman and Athanasopoulos (2013)suggest to use Root Mean Squared Error (RMSE) and/or Mean Absolute Error (MAE).

RMSE is commonly used as scale dependent measure of predictive accuracy. RMSE is very similar to Mean Absolute Error (MAE) metric but RMSE amplifies and severely punishes large errors. RMSE can be expressed as:

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(\hat{e}_i)^2}$$

An alternative metric MAE can be expressed as:

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|\hat{e}_i|$$

Both RMSE and MAE give good indication which model has better predictive accuracy. But as Diebold (2013) noted that even when models inherently have same predictive accuracy (implying that $RMSE_a = RMSE_b$), in a given sample one of the models could outperform the other and thus labeled as being "better" model. To avoid this test introduced by Diebold and Mariano (1995) is used. Further in the text Diebold and Mariano test is denoted as *DM* test. The alternative for the test is set to be one sided as nested models with lower RMSE should provide better forecasts than base benchmark model. Note that *DM* test provides pair-wise comparison, thus it can only be used to determine if target Model outperforms benchmark Model. *DM* test can be expressed as:

$$DM = \frac{\frac{1}{T}\sum_{t=1}^{T}\{g(e_{1,t}) - g(e_{2,t})\}}{\sqrt{\frac{2\pi\hat{f}_d(0)}{T}}}$$

, where $g(e_{i,t})$, $i = 1, 2$ denoting the loss from forecast error $e_{i,t}$ evolving from prediction model i.

The null hypothesis tested is $H_0: \sum_{t=1}^{T}\{g(e_{1,t}) - g(e_{2,t})\} = 0$. Under H_0, DM is asymptotically standard normal distributed.

4.3 Robustness checks

Full sample estimation: Diebolt (2013) argues that performing out-of-sample forecasts is redundant because it throws away significant chunk of the data. He argues that in studies similar to this out-of-sample testing can only be labelled as quasi-out-of-sample testing and can be manipulated by a researcher. This comes from the fact that research community have bias towards positive results publications and that it is possible to use data mining tricks to create a "winner". For instance, instead of pre-selecting the model in the training sample and then perform the out-of-sample testing I could select the model based on out of sample performance and then claim that it is the best the model to begin with. This would not be possible if I only had training sample data.

For this reason, Diebolt (2013) argues that it is best just to perform full sample estimation and evaluate the model. As proposed by Castle et al. (2011) the models can be evaluated based on individual coefficient significance and on information criteria (i.e. AIC, BIC).

Rolling window scheme: As discussed in 4.2.1 Estimation section – the main analysis is performed using recursive method. Alternatively rolling window estimation scheme can be used. As a robustness check I replicate main analysis using rolling window scheme.

Different sample sizes: Common issue is that results are sample dependent (Although full-sample estimation and choosing based on information criteria mitigates this). I manipulate training and testing sample sizes by -+ 10% to see whether there is significant effects.

4.4 Implementation

Validity of research depends whether it can be replicated. For this reason, in this section I provide brief overview how results were obtained.

To conduct analysis and to produce forecasts I used RStudio (version 0.98.1091). Analysis heavily relies on *forecast* package which includes ARIMA estimation and prediction. For list of packages used see Appendix H. Appendix I provides R codes for main script and user defined functions which were used to obtain the results shown in this thesis.

Moreover, as a software robustness check I have replicated most of the analysis using MATLAB and STATA software. In general, results are very similar but they slightly differ. The difference arises from different estimation techniques and potentially different underlying algorithms and/or optimization methods. For ARIMA and ARIMAX estimation, I used default in

R - "CSS-ML" method as it provided best in-sample fit and converged to a solution in all scenarios.

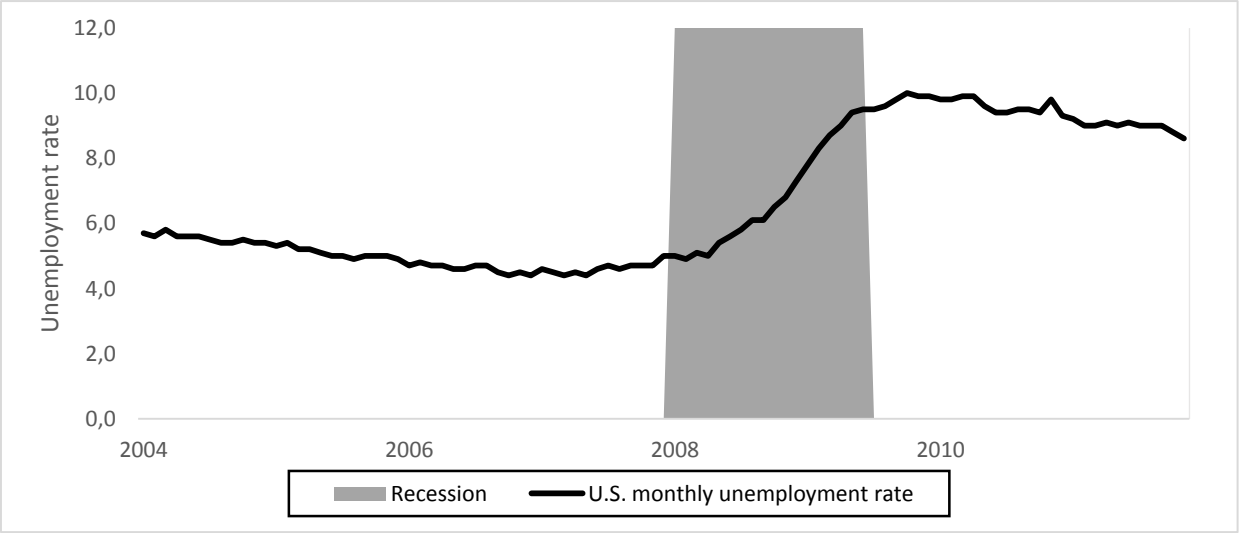
5 Results

5.1 Training sample results

5.1.1 Base benchmark model selection

In this section benchmark model is selected. As discussed in methodology Box-Jenkins (1976) method is applied to select key base benchmark ARIMA model.

All calculations are performed on training data sample. Figure 2 shows U.S. monthly unemployment rate development between 2004-01 and 2011-11. Visual inspection does not reveal whether there is time trend as the series seems bounded. Formal test using Augmented Dickey Fuller test and DF-GLS test also do not arrive at decisive conclusion. As only DF-GLS rejected null hypothesis of unit root and only at 10% significance level. Thus, I inspect auto-



correlation function (ACF) and partial auto-correlation functions (PACF) graphs

Figure 2 In-sample U.S. monthly seasonally adjusted unemployment rate spanning from 2004-01 to 2011-11. Made by Author. Source: Bureau of Labor Statistics.

Appendix C provides ACF and PACF graphs. ACF is slowly decaying which is implies that there is time trend (see Appx. C Figure 9). Box-Jenkins methodology would suggest to differentiate the series. Repeating step 1 after differentiation reveals that now the data is somewhat stationary. By inspecting PACF from Figure 10 (Appx. C) it seems that there are two

candidate AR(3) process is good potential candidate as all lagged values should be significant and its relatively parsimonious. Alternatively it could include 5 lags or even 12 lags, potentially fixing other lagged values to 0 (i.e. so that only coefficients of lagged values which were significant can have an impact on forecast). Note that Box-Jenkins methodology is not hard science hence it's left to discretion of researcher to choose how best to proceed. For this reason, I chose more parsimonious model. Otherwise, there is a possibility to run into over-fitting to the training data problem.

Box-Jenkins methodology reveals the need to differentiate the data. Although as I use ARIMA models instead of differentiating the data I chose to integrate of order 1. The results are similar yet it simplifies procedure as there is no need to back-transform results.

Based on Box-Jenkins method I would select ARIMA(3,1,0) model. Although selection of moving average part is hardly possible using this method as there is too much for interpretation. Alternatively, validation set can be introduced and model selected on model performance (e.g. say on RMSE). This is standard procedure for more advanced modelling techniques. For instance, it is in-built procedure for doing neural networks on MATLAB. Although Diebold (2013) advises against splitting data as it just throws away good chunk of data without providing any benefits. He argues that in-sample selection with information criteria is "just as good" and provides more stable coefficients (just by-product of larger sample size). For this reason, I use this approach: to use full sample size and determine best model by employing information criteria.

Recall notation – $ARIMA(p, d, q)$, where p, d, q are non-negative integers that refer to the order of the autoregressive, integrated and moving average parts of the model respectively. As noted before integration is set to 1 (integration of order 0 was tested with this method but return inferior results, results not reported). Maximum number of autoregressive order is set to 3 while maximum number of moving average part is set to 2. Then each model is estimated and information criteria numbers are obtained.

Model	AIC	BIC
ARIMA(1,1,0)	-56.01	-50.92
ARIMA(2,1,0)	-71.37	-63.74
ARIMA(3,1,0)	**-73.97**	**-63.80**

ARIMA(1,1,1)	-73.19	-65.56
ARIMA(2,1,1)	-75.09	-64.92
ARIMA(3,1,1)	**-77.96**	**-65.24**
ARIMA(1,1,2)	-74.68	-64.51
ARIMA(2,1,2)	-73.10	-60.38
ARIMA(3,1,2)	**-78.17**	-62.91

Table 3 ARIMA benchmark model selection based on information criteria (AIC and BIC) Based on calculations made by Author.

Table 3 shows ARIMA benchmark model selection using information criteria (AIC and BIC). Information criteria works in a way that it tries to penalize unnecessary complexity. Thus, the lower value the better model. Based on information from the table I select $ARIMA(3,1,1)$ as base benchmark model which will be used for predictions. $ARIMA(3,1,0)$ is considered as a robustness check.

Lastly, performing residuals diagnostics shows that residuals are normally distributed resembling white noise. As mentioned above $ARIMA(3,1,1)$ is selected as base benchmark model. So residual diagnostics will focus on this model (although reasoning and conclusion is the same for $ARIMA(3,1,0)$). Table *4* provides relevant test statistics for selected models.

	ARIMA(3,1,0)	ARIMA(3,1,1)
Mean	-0.007	-0.008
Box-Ljung test	12.41	11.01
p-value	0.41	0.53
Jarque Bera test	0.61	4.67
p-value	0.74	0.10
Shapiro test	0.99	0.98
p-value	0.47	0.28

Table 4 Residual diagnostics. Based on calculations made by Author.

Box-Ljung test tests for whether series is autocorrelated. Say, low p-value would imply that some coefficients of lagged target series are significantly different from 0, thus series is autocorrelated. In this case, p-value is 0.53 which implies that there is no remaining serial correlation in the series. Next, normality is tested. Tests for normality are somewhat controversial as they test against normality assumption. Say Shapiro-Wilk test tests the null hypothesis that "the samples come from a Normal distribution". This implies that if we can reject this null then the series is not normally distributed although failing to reject does not automatically imply that series is normally distributed. Another test Jarque-Bera test sets the

same null hypothesis but focus on testing whether series have normal distribution properties, namely skewness of zero and a kurtosis coefficient of (Jarque & Bera, 1987). Both tests have p-value far greater than conventional levels of significance. Lastly, Figure 11 (Appx. D) Q-Q plots for normal distribution. Not all observations fall on the hypothetical QQ line but the fit is pretty good. Thus, it's safe to conclude that errors have no serial correlation and are normally distributed.

5.1.2 ARIMAX model selection

In this section we look into models with exogenous variables. As discussed above $ARIMA(3,1,1)$ is selected as base benchmark model. Base model is extended by incorporating exogenous variable: Initial job claims (*IC*) and Google Index for *Jobs* and for *Unemployment Benefits*.

Table 5 provides summary of in-sample results for all five models using $ARIMA(3,1,1)$ as base model (for results using $ARIMA(3,1,0)$ see Appx. E). Obviously, the more complex model the better is in-sample fit as measured by RMSE. Castle et al. (2011) suggest that it is more important to look at individual component significance and information criteria. Table 5 shows that inclusion of lagged IC value does not improve the model as coefficient is statistically insignificant and both information criteria are lower than that of base model. For this reason prediction evaluation with Diebold and Mariano tests will be comparing target model with base benchmark model.

Inc. variables	Benchmark		Target		
	Base	IC	Jobs	Jobs + Benefits	Combined
$ARIMA(3,1,1)$	✓	✓	✓	✓	✓
$lnIC_{t-1}$		0.07			0.62*
		(0.35)			(0.36)
$lnGI\_Jobs_t$			-1.69***	-1.78***	-1.84***
			(0.64)	(0.64)	(0.64)
$lnGI\_Benefits_t$				0.17*	0.18**
				(0.09)	(0.09)
N	95	95	95	95	95
AIC	-77.96	-75.07	-76.87	-78.39	**-79.35**
BIC	**-65.24**	-59.81	-61.61	-60.59	-59.00
RMSE	0.148	0.148	0.149	0.146	0.144

Table 5 In-sample results for ARIMA(3,1,1) and ARIMAX(3,1,1). Based on calculations made by Author.

Notes: (i) *Significant at the 0.10 level
(ii) **Significant at the 0.05 level
(iii) ***Significant at the 0.01 level

When looking into results of Target models we can observe that coefficients are significant: GI for *Jobs* is significant at 1% level while GI for *Unemployment Benefits* and IC are significant at more conservative 5% to 10% significance levels. *Combined* model and target model which include GI for *Jobs* and GI for *Unemployment Benefits* have lower AIC than base

benchmark model. However, higher BIC. There are no clear cut rules which information criteria is better. Hyndman and Athanasopoulos (2013) states that BIC would be preferred if true underlying model is known or in larger data sets. None of which is true in this case as time-series data is rather small and internet search activity can be seen more as a by-product of unemployment. So taking this into consideration models with GI and especially the combined model are promising.

One interesting observation is that coefficient for GI for *Jobs* is negative. Meaning that the more people search for jobs online the lower is expected unemployment rate. While GI for *unemployment benefits* is positive. It seems that Google Indicators are capturing two sides of the unemployment coin. Say the more unemployed the more likely they are to search for unemployment benefits. Partly, they would start searching for jobs online. Although if searching for jobs online works then the more people search the more likely they are to find a job and thus become employed. This shows that Google data can be employed for inferential rather than predictive studies.

To sum up, in-sample results are promising. Coefficients in Target models are statistically significant. Several target models outperform base benchmark model on AIC. Thus, I proceed to make out-of-sample predictions.

5.2 Out-of-sample results

5.2.1 Predictions

In this section information about forecasts is provided. Firstly, I start by making one-step-ahead forecasts. Given that data series are of monthly frequency forecasting one-step-ahead make most sense. Otherwise, quarterly frequency could be used. Thus, the base scenario is one-step-ahead forecasts where estimation is done recursively.

Table 6 provides a summary statistic for the forecasts one-step-ahead forecasts. Model which includes only GI for *Jobs* has the lowest Root Mean Squared Error (RMSE) while model which includes only *IC* has the lowest Mean Absolute Error (MAE). However, in the previous section it was showcased that model with IC would not be considered as main benchmark series (as had coefficient for IC was insignificant and information criteria higher than that of base benchmark model). Combined model seems to be an attractive option as it outperforms Base benchmark model both in terms of RMSE and MAE. Although in both metrics the outperforming is rather modest 2.6% (RMSE) and 1.2% (MAE). Recall that RMSE penalizes severely large errors. This stems from the fact that RMSE by construction is squared metrics which penalizes large errors. While MAE is only addition of absolute errors. Lower RMSE but higher MAE implies that forecasted series on average would have higher absolute error but captures sudden movements of the series (as there are no big errors which would have been penalized). Thus, it seems that IC and GI both are useful for forecasting unemployment rate and that they encompass different information

Metrics	Benchmark		Target		
	Base	IC	Jobs	Jobs + Benefits	Combined
RMSE	0.156	0.152	**0.151**	0.155	0.151
% of Base	100.0%	97.9%	**97.1%**	99.6%	97.4%
MAE	0.119	**0.115**	0.121	0.123	0.118
% of Base	100.0%	**96.8%**	101.7%	103.4%	98.8%
DM statistic	-	0.52	0.70	0.11	0.50
p-value	-	0.30	0.24	0.46	0.31

Table 6 Out-of-sample one-step-ahead predictions evaluation. Based on calculations made by Author.

These results support D'Amuri and Marruci (2012) findings. They find that models which included Google indicators had the best predictive accuracy and, indeed, outperform

benchmark models. They also find that in some cases combined models (which included both GI and IC) performed best.

Overall, regardless which metrics is used for analysis the predictive accuracy improvements are modest. This is consistent with Wu and Brynjolfsson (2014) who re-examined Google flu data and find that GI provide only modest improvement over simple autoregressive models.

5.2.2 Evaluation

Using Diebold Mariano tests it can be formally checked whether the forecasts are statistically different. Although DM tests is seen as rather conservative metric and given that models are nested it is unlikely that null hypothesis would be rejected. Indeed, p-values (see Table 6) are higher than conventional statistical significance levels.

As noted in methodology I estimate GI models using contemporaneous effect. Thus, in order to construct realistic forecasts I had to predict Google indicators values for next period - $\widehat{GI}_{t+1}$. This is done in order to replicate realistic situation. For instance, if we want to know what is today's the best estimate of next month's unemployment rate. We have already estimated link between Google Indicator with unemployment rate but we still do not know what GI value will be. Thus, we have to create some estimate.

Clearly this adds variation which leads to inferior results. For instance, if we knew GI values with certainty then predictions would be better. When I re-run analysis using actual GI values instead of $AR(1)$ predictions then results are very promising. Say for model which includes GI for Jobs RMSE is 0.1374 which is about 12% better than base benchmark results and DM test shows that difference is significant at 5% level. This is consistent with what Choi and Varian (2009) called "predicting presence" or nowcasting. Although additional analysis shows that the lag cannot be very large as 2 weeks nowcasting already does not yield better results than base benchmark.

Another interesting aspect is what happens if we extend methodology to include multiple steps ahead.

5.2.3 Multiple steps-ahead

One-step-ahead forecasts including GI showed only modest improvement over base benchmark model. It is interesting to see how the predictive accuracy changes by changing forecasting horizon. In this section I look into multiple-steps-ahead forecasts.

As the data series is of monthly frequency the most relevant time-steps are 1, 3 and 6 steps ahead representing a month, a quarter and half a year respectively. Table 7 shows the RMSE results for selected multiple-steps ahead using $ARIMA(3,1,1)$ as a base model.

Model	1-steps ahead	3-steps ahead	6-steps ahead
Base	0.156	0.278	0.386
IC	0.152	0.276	0.385
Jobs	**0.151**	**0.256**	**0.363**
Jobs+Benefits	0.155	0.261	0.375
Jobs+Benefits+IC	**0.151**	0.257	0.378
N	40	38	35

Table 7 RMSE results for multiple-steps ahead forecasts using $ARIMA(3,1,1)$ as a base model. Based on calculations made by Author.

From the table we can see ARIMAX models perform better than Base benchmark model. For instance, for one-step-ahead forecasts *GI for Jobs* model were 2.9% better than *Base* benchmark model while for three-steps-ahead forecasts were 7.9% better. Figure 3 graphs selected models performance (RMSE) as forecasting steps-ahead increases. It is visible that the target model outperform others. Expressed in relative terms it is about 10% better than benchmark model which is quite large improvement. For full results see Appendix G.

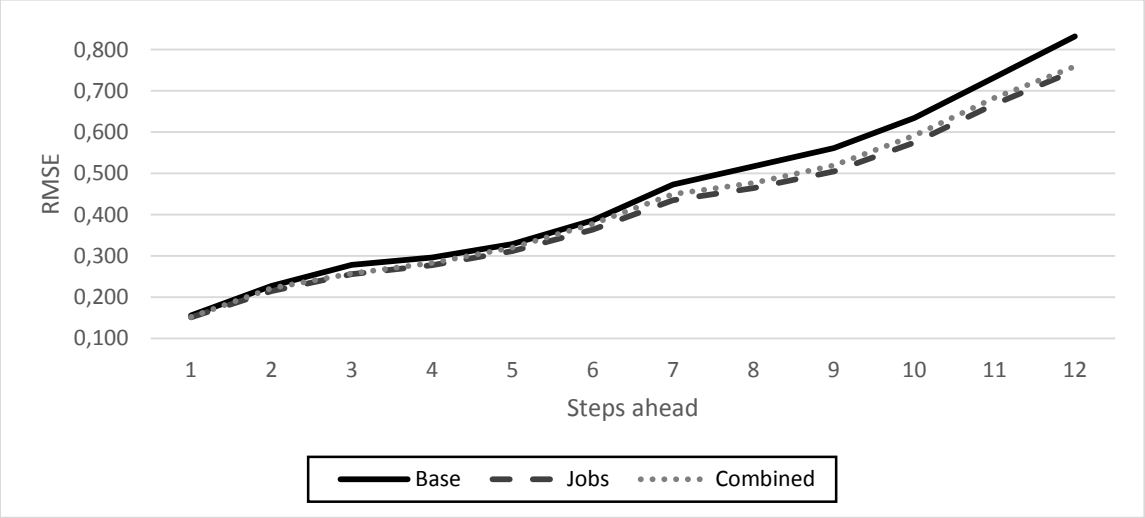


Figure 3 RMSE results for multiple steps ahead for selected models using recursive estimation scheme. Made by Author.

6 Robustness checks

6.1.1 Full-sample estimation

It seems that forecasting literature came a full circle when it comes to comparing predictive power. The usual procedure which was followed in this paper is to split the data into training and testing samples. This allows to avoid over-fitting problem as out-of-sample predictive accuracy is the most important.

Yet, Diebold (2013) argues that this quasi-out-of-sample experiment is largely redundant: it reduces power with no compensating effect (as all known procedures, including pseudo-out-of-sample procedures, can be "tricked" by data mining in finite samples). Diebold (2013) argues that in most circumstances simpler in-sample procedures like information criteria is at least as good as out-of-sample tests. For this reason as a robustness check I re-estimate the models on full sample.

Table 8 summarizes the results for full sample using $ARIMA(3,1,1)$ as kernel model. I wanted to replicate same procedure which would be carried in realistic forecasting setting, namely that we only have lagged values of GI for *Jobs* and GI for *Unemployment Benefits*. To get variable values at time t we need to create estimates. $\widehat{GI}_t$ variables are estimated with auxiliary $AR(1)$ model. Table 8 shows that even if estimates are used their coefficients are still significant and improves forecasting accuracy as measured by both information criteria.

Inc. variables	Benchmark		Target		
	Base	IC	Jobs	Jobs + Benefits	Combined
$ARIMA(3,1,1)$	✓	✓	✓	✓	✓
$lnIC_{t-1}$		0.63*			0.68**
		0.33			(0.31)
$lnGI\_Jobs_t$			-1.42***	-1.39**	-1.43***
			(0.55)	(0.54)	(0.53)
$lnGI\_Benefits_t$				0.17*	0.17*
				(0.07)	(0.07)
N	135	135	135	135	135
AIC	-112.00	-112.88	-115.53	-118.93	**-121.81**
BIC	-97.51	-95.49	-98.14	**-98.65**	**-98.62**
RMSE	0.153	0.150	0.149	0.146	0.143

Table 8 Full-sample results for ARIMA(3,1,1) and ARIMAX(3,1,1). Based on calculations made by Author.

Notes: (i) *Significant at the 0.10 level
(ii) **Significant at the 0.05 level
(iii) ***Significant at the 0.01 level

6.1.2 Rolling window scheme

In the main analysis, I used recursive estimation method, where estimation sample size increases with each additional time period. In most instances it should be preferred estimation technique as it allows to incorporate all available information. If time series did not undergo fundamental changes (e.g. structural breaks) then recursive method gives more consistent estimates. Yet, rolling window technique is used to ensure that coefficient values are not dominated by distant data points and to take into account the most recent information.

Table 9 summarizes the results for one-step-ahead forecasts with rolling window technique. It is very similar with recursive technique as target series outperform base benchmark results by about 3%. Also MAE is lower for target series. Yet, only marginally.

Metrics	Benchmark		Target		
	Base	IC	Jobs	Jobs + Benefits	Combined
RMSE	0.159	0.153	0.155	0.155	0.153
% of Base	100.0%	95.9%	97.1%	97.2%	96.0%
MAE	0.127	0.119	0.124	0.124	0.121
% of Base	100.0%	94.0%	97.9%	97.6%	95.2%
DM statistic	-	0.98	0.69	0.66	0.77
p-value	-	0.17	0.25	0.26	0.22

Figure 4 graphs results for multiple-steps-ahead. It mimics development seen in section 5.2.3, where Target series predictive accuracy improves relative to base benchmark model accuracy as number of steps increases. Improvement reaches up to 11.7% (See Appx. G).

Table 9 1-step-ahead results using ARIMA(3,1,1) model and rolling window estimation scheme. Based on calculations made by

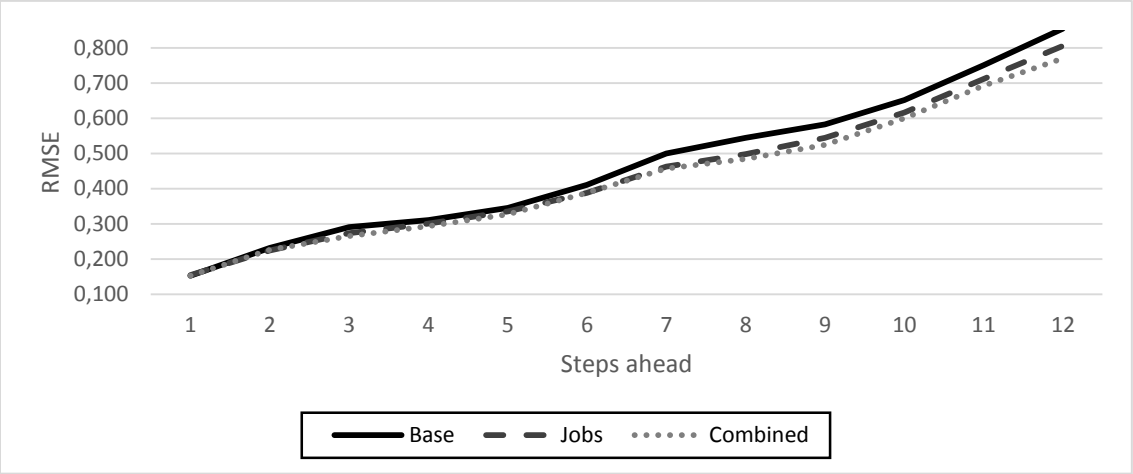


Figure 4 RMSE results for multiple steps ahead for selected models using rolling estimation scheme. Made by Author.

6.1.3 Changing sample size

Another issue with forecasts is that results may be sample dependent. As full sample estimates show that results are robust when full sample size is considered. Still in this part I provide sensitivity analysis by changing training sample size.

Table 10 summarizes results. From the table we can see that RMSE and MAE increases in both cases. Although most relevant metric which is relative performance shows that *Combined* model outperforms *Base* benchmark model. (In case of decreasing training sample by 10% *Combined* model's RMSE and MAE is about 4% lower than respective metrics of *Base* model. In case of increasing sample size by 10% results are even more dramatic. *Combined* model's RMSE is about 8% smaller while MAE is about 10% lower.

Metrics	-10%		Base (70%)		+10%	
	Base	Combined	Base	Combined	Base	Combined
RMSE	0.176	0.169	0.156	0.151	0.176	0.162
MAE	0.135	0.130	0.119	0.118	0.144	0.130
N in Training sample	81		95		108	
N in Test sample	54		40		27	

Table 10 sensitivity analysis results by changing training sample size. Based on calculations made by Author.

This sensitivity analysis shows that models which include GI, indeed, outperform base benchmark model and results vary from modest to medium improvement. Note that table

includes only results of *Combined* model but results are similar to model which only includes GI for jobs.

6.1.4 Alternative Google keywords

While testing Google predictive accuracy D'Amuri and Marcucci (2012) did falsification test which showcases that randomly selected variables (say GI for Facebook) fail to replicate positive results which were achieved with GI for Jobs. In my analysis I did not need to go as far as including unrelated variables. As even variables which are seemingly related (say "*Job center*" or "*Unemployment*" category) have little to no predictive power.

For instance, "*Unemployment*" category has over 0.95 correlation with target series but it seems that interest in unemployment follows actual unemployment, thus it is a lagging indicator and not useful for forecasting. I have experimented with all keywords mentioned in data description. I find that GI for *Jobs* is by far most relevant keyword. Inclusion of other explanatory variables do not lead to better results. However, inclusion of GI for *Unemployment Benefits* shown slight predictive accuracy improvements. Indicating that GI for *Unemployment Benefits* has some predictive information which is not fully captured by *Jobs* keyword. This stems from the fact that both keywords capture different aspects of unemployment.

Other variables were partly useful, say when constructing factors using principal component analysis. Although these improvements were rather marginal and could fall under what Diebold (2013) argues as data mining techniques which destroys the whole splitting-data and performing out-of-sample forecast analysis.

Lastly, I re-run analysis using different specifications and find that *Combined* model which includes only GI for Jobs and IC performs the best out of all models with RMSE 5% better than base benchmark (MAE 3% better).

7 Conclusions

In this thesis I suggest the use of the Google index (GI), based on the internet job searches performed through Google, as a potential leading indicator to predict the U.S. monthly unemployment rate.

As the benchmark model I use popular time series model – ARIMA. In the training sample I preselect the best ARIMA specification based on AIC and use this model as base model. This base model is augmented with leading indicators: Initial Jobless claims (IC), Google index (GI) or both (*Combined* scenario). Models which include GI indeed improved forecast out-of-sample performance not only for 1-step-ahead but also for multiple-steps-ahead. I find that best models were either model which was augmented with only GI for *Jobs* or *Combined* model. This depends on how predictive accuracy is defined. In general GI for *Jobs* model performs well if we compare RMSE while *Combined* model outperforms if we consider MAE. This suggests that GI and IC encompass different information. IC helps to make predictions more precise, thus reducing absolute error, while GI helps to avoid big errors, thus able to predict bigger changes in unemployment rate. This separation is also visible if IC model is compared with GI for Jobs model. Although as I increase forecasting steps horizon then GI for Jobs model dominates IC model. Yet, *Combined* model seems to be most robust as it always provides improvement over base benchmark model and captures most of GI for *Jobs* predictive power. Thus, should be routinely used in forecasting unemployment rate.

These results were robust to changes of estimation technique and sample relative sizes. Full sample estimation shows that coefficient are statistically significant and relevant. Results indicate that all target models have lower AIC value than benchmark models. Although *Combined* model is the best model which confirms previous conclusion. Alternative Google keywords and subject categories despite high correlation with unemployment rate did not prove to be good predictors. Thus, what researcher gets from Google data highly depends if he understands relationship between specific search term and the research topic. In my opinion, keywords should be selected as action orientated. Say, when it comes to unemployment people would either search for what happens once you're unemployed or search for a job.

Google data can improve predictive accuracy when forecasting unemployment. Yet, despite positive results I find them to be rather modest. In range of 2.4% to 5%[2], when performing 1-step-ahead forecasts. Clearly this is already a good result but using rather conservative DM test I cannot claim that the difference is statistically significant. Although it has

[2] Model which included IC and GI for Jobs provided largest improvement in terms of RMSE of about 5%. Not included in the full analysis as it was not pre-selected in Training sample.

been very consistent and robust to different specifications. To showcase that predictive accuracy is indeed better one needs larger sample size, a luxury which time-series data cannot afford. This result is consistent with Wu and Brynjolfsson (2014) who re-examined flu trends and show that utility of search data relative to ARMA models is rather modest.

Thus, despite positive results I think that strengths of Google data lies in other forecasting exercises: forecasting presence and forecasting special events. Choi and Varian (2009a, 2009b) argue that Google Trends can be used to "predict the presence" by exploiting the fact that economic series have long reporting lags. This is especially relevant for countries which do not have as strong reporting standards as U.S. as showcased by Chadwick and Sengul (2013) who forecasted unemployment rate of Turkey.

Another application is to forecast some event which has no explicit previous states. When forecasting unemployment rate economists can rely on past data series and train ARIMA models which already provide a good fit and have high short-term predictive power. Although some special events have no previous data (e.g. Wu and Brynjolfsson (2014) forecasts opening weekend box-office revenues for featured films and first-month sales figures).

To sum up, this thesis showcased Google data potential to forecast economic series, namely the U.S. monthly unemployment rate. With increasing awareness about its potential it is not hard to imagine that internet-based data will become widely used in economic research in the future.

References

Alexa. (n.d.). The top 500 sites in each country or territory. Retrieved May 14, 2015, from http://www.alexa.com/topsites/countries

Aparicio, F., Escribano, A., & Garcia, A. (2006). Range Unit-Root (RUR) Tests: Robust against Nonlinearities, Error Distributions, Structural Breaks and Outliers. *Journal of Time Series Analysis, 27*(4), 545–576.

Askitas, N., & Zimmermann, K. (2009). Google Econometrics and Unemployment Forecasting. *Applied Economics Quaterly, 55*, 107–120.

Baker, S., & Fradkin, A. (2014). The Impact of Unemployment Insurance on Job Search : Evidence from Google Search Data. Retrieved May 2, 2015, from http://dx.doi.org/10.2139/ssrn.2251548

Barnichon, R., & Nekarda, C. (2012). The Ins and Outs of Forecasting Unemployment: Using Labor Force Flows to Forecast the Labor Market. *Brookings Papers on Economic Activity*, (2), 83–131.

Billari, F., D'Amuri, F., & Marcucci, J. (2013). Forecasting births using google. In *Annual Meeting of the Population Association of America*.

Box, G. E. P., & Jenkins, G. (1976). *Time Series Analysis: Forecasting and Control.* San Francisco: Holden-Day.

Carrière-Swallow, Y., & Labbé, F. (2013). Nowcasting with Google trends in an emerging market. *Journal of Forecasting*, *32*(4), 289–298.

Castle, J. L., Doornik, J. A., & Hendry, D. F. (2011). Evaluating Automatic Model Selection. *Journal of Time Series Econometrics*, *3*(1)

Chadwick, M., & Sengul, G. (2013). *Nowcasting unemployment rate in Turkey:* Let's ask Google. Central Bank of the Republic of Turkey. Retrieved May 2, 2015, from http://goo.gl/eKJHSV

Chen, S. (2005). A note on in-sample and out-of-sample tests for Granger causality. *Journal of Forecasting*, (24), 453–464.

Choi, H., & Varian, H. (2009a). Predicting initial claims for unemployment benefits. *Economics Research Group, Google Inc*.

Choi, H., & Varian, H. (2009b). Predicting the Present with Google Trends. *Economics Research Group, Google Inc.*

comScore. (2014). comScore Releases October 2014 U.S. Desktop Search Engine Rankings. Retrieved May 2, 2015, from http://www.comscore.com/Insights/Market-Rankings/comScore-Releases-October-2014-US-Desktop-Search-Engine-Rankings

D'Amuri, F. (2009). *Predicting unemployment in short samples with internet job search query data*. Retrieved May 2, 2015, from http://mpra.ub.uni-muenchen.de/18403/

D'Amuri, F., & Marcucci, J. (2012). *The predictive power of Google searches in forecasting unemployment.* Bank of Italy Temi Di Discussione (Economic Working Papers).

Dickey, D. A., & Fuller, W. A. (1979). Distribution of the estimators for autoregressive time series with a unit root. *Journal of the American Statistical Association*, *74*(366a), 427–431.

Diebold, F. X. (2013). *Comparing Predictive Accuracy, Twenty Years Later: A Personal Perspective on the Use and Abuse of Diebold-Mariano Tests*. *Penn Institute of Econmic Research, Departament of Economics, University of Pennsylvania*.

Diebold, F. X., & Kilian, L. (2000). Unit-root tests are useful for selecting forecasting models. *Journal of Business & Economic Statistics*, *18*(3), 265–273.

Diebold, F. X., & Mariano, R. S. (1995). Comparing Predictive Accuracy. *Journal of Business & Economic Statistics*, *20*(1), 134–144.

Edelman, B. (2012). Using internet data for economic research. *Journal of Economic Perspectives*, *26*(2), 189–206.

Elliott, B. Y. G., Rothenberg, T. J., & Stock, J. H. (1996). Efficient Tests for an Autoregressive Unit Root. *Econometrica*, *64*(4), 813–836.

Eurostat. (2009). ESS Guidelines on Seasonal Adjustment. Retrieved May 2, 2015, from http://goo.gl/gqUebi

Fondeur, Y., & Karamé, F. (2013). Can Google data help predict French youth unemployment? *Economic Modelling*, *30*, 117–125.

Giacomini, R., & White, H. (2006). Tests of Conditional Predictive Ability. *Econometrica*, *74*(6), 1545–1578.

Ginsberg, J., Mohebbi, M. H., Patel, R. S., Brammer, L., Smolinksi, M., & Brilliant, L. (2009). Detecting Influenza epidemics using Search Engine Query Data. *Nature*, (457), 1012–1014.

Golan, A., & Perloff, J. (2004). Superior forecasts of the US unemployment rate using a nonparametric method. *Review of Economics and Statistics*, *86*(1), 433–438.

Google. (2008). Announcing Google Insights. Retrieved May 2, 2015, from http://adwords.blogspot.ch/2008/08/announcing-google-insights-for-search.html

Hansen, P. R., Lunde, A., & Nason, J. M. (2011). The Model Confidence Set. *Econometrica*, *79*(2), 453–497.

Hyndman, R. J., & Athanasopoulos, G. (2013). Forecasting: principles and practice. Retrieved May 1, 2015, from http://otexts.org/fpp/.

Jarque, C. M., & Bera, A. K. (1987). A test of normality of observations and regression residuals. *International Statistical Review*, *55*, 163–172.

Knotek, E. S. (2007). How useful is Okun's law? *Economic Review*, (Q IV), 73–103.

Koop, G., & Potter, S. M. (1999). Dynamic Asymmetries in U.S. Unemployment. *Journal of Business & Economic Statistics*, *3*(17), 298–312.

Lee, J. (2000). The robustness of Okun's law: Evidence from OECD countries. *Journal of Macroeconomics*, *22*(2), 331–356.

Montgomery, A., Zarnowitz, V., Tsay, R. S., & Tiao, G. C. (1998). Forecasting the US unemployment rate. *Journal of the American Statistical Association*, *93*(442), 478–493.

Moosa, I. A. (1997). A cross-country comparison of Okun's coefficient. *Journal of Comparative Economics*, *24*(3), 335–356.

Okun, A. (1962). Potential GNP: Its measurement and significance. In *American Statistical Association, Proceedings of the Business and Economic Statistics Section* (pp. 98–104).

Rothman, P. (1998). Forecasting Assymetric Unemployment Rates. *Review of Economics and Statistics*, *80*(1), 164–168.

Stock, J. H., & Watson, M. W. (1996). Evidence on structural instability in macroeconomic time series relations. *Journal of Business & Economic Statistics*, *14*(1), 11–30.

Suhoy, T. (2009). *Query indices and a 2008 downturn: Israeli data*. Research Department, Bank of Israel. Retrieved from http://goo.gl/g4dsF3

Vosen, S., & Schmidt, T. (2011). Forecasting private consumption: survey-based indicators vs. Google trends. *Journal of Forecasting*, *578*(January), 565–578.

World Bank. (n.d.). Internet users are people with access to the worldwide network. Retrieved May 2, 2015, from http://data.worldbank.org/indicator/IT.NET.USER.P2

Wu, L., & Brynjolfsson, E. (2014). The Future of Prediction: How Google Searches Foreshadow Housing Prices and Sales. In *Economic Analysis of the Digital Economy* (pp. 89-118). University of Chicago Press.

Data

Google Trends (n.d.) Retrieved April 2, 2015, from https://www.google.com/trends/

Bureau of Labor Statistics (n.d.) Current Population Survey. Labor Force Statistics. Retrieved April 2, 2015, from http://data.bls.gov/cgi-bin/surveymost?ln

U.S. Employment and Training Administration (n.d.) FRED. Initial Claims. Retrieved April 2, 2015, from https://research.stlouisfed.org/fred2/series/ICSA/

R-packages

Dragulescu, A. A. (n.d.) xlsx: Read, write, format Excel 2007 and Excel 97/2000/XP/2003 files. R package version 0.5.7, from http://CRAN.R-project.org/package=xlsx

Hyndman, R. J. (2014) forecast: Forecasting functions for time series and linear models. R package version 5.7, from http://github.com/robjhyndman/forecast

Pfaff, B. (2008) urca: Analysis of Integrated and Cointegrated Time Series with R. R package version 1.2-8, from http://CRAN.R-project.org/package=urca

Sax, C. (2013) seasonal: R interface to X-13ARIMA-SEATS. R package version 0.70.1, from http://CRAN.R-project.org/package=seasonal

Trapletti, A. & Hornik K.(2013). tseries: Time Series Analysis and Compuatational Finance. R package version 0.10-32., from http://CRAN.R-project.org/package=tseries

Appendices

7.1 Appendix A – Seasonal adjustment graphs

Figure 5 graphical representation of seasonal adjustment for GI for Jobs. Made by Author. Source: Google trends

Figure 6 graphical representation of seasonal adjustment for GI for Unemployment. Made by Author. Source: Google trends

Figure 7 graphical representation of seasonal adjustment for GI for Job center. Made by Author. Source: Google trends

7.2 Appendix B – Descriptive statistics: Google Index variables

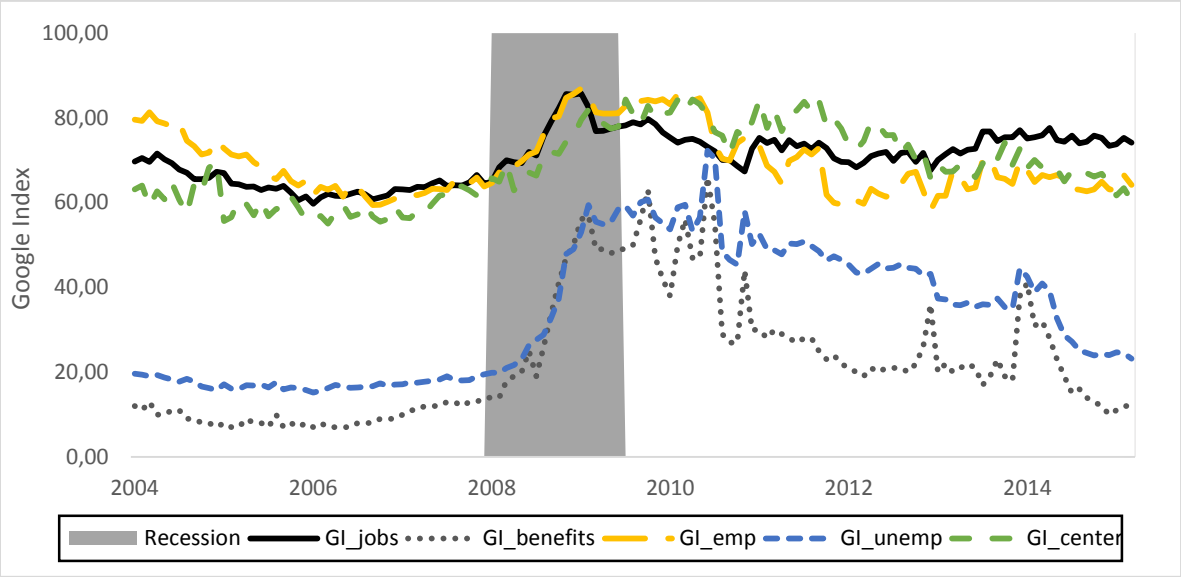


Figure 8 Google Indicator values over time. Made by Author. Source: Google trends

7.3 Appendix C – Box-Jenkins methodology (ACF, PACF)

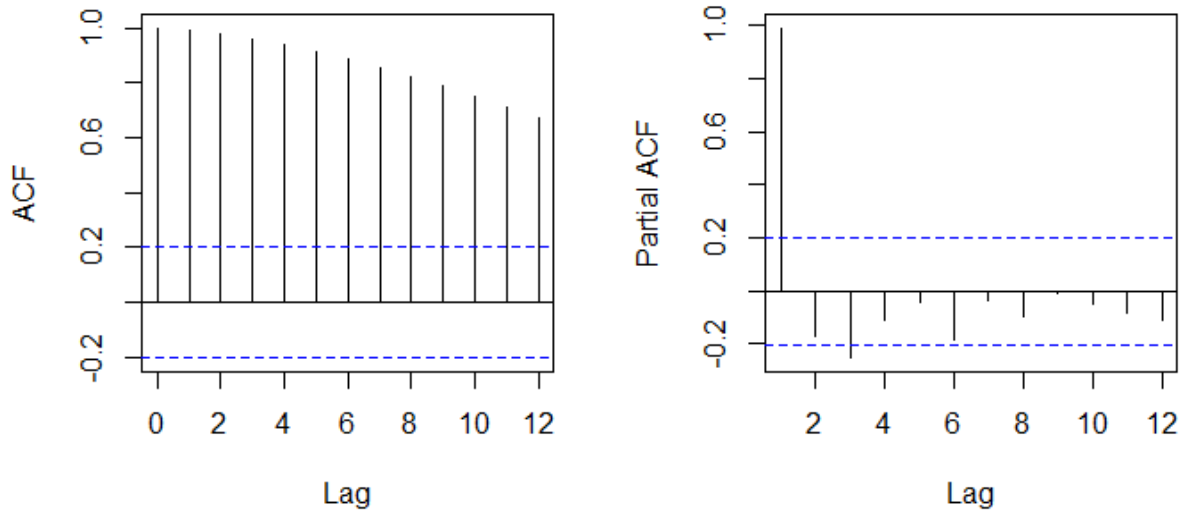


Figure 9 ACF and PACF for undifferentiated U.S. monthly unemployment rate series. Made by Author using R. Source: Bureau of Labor Statistics

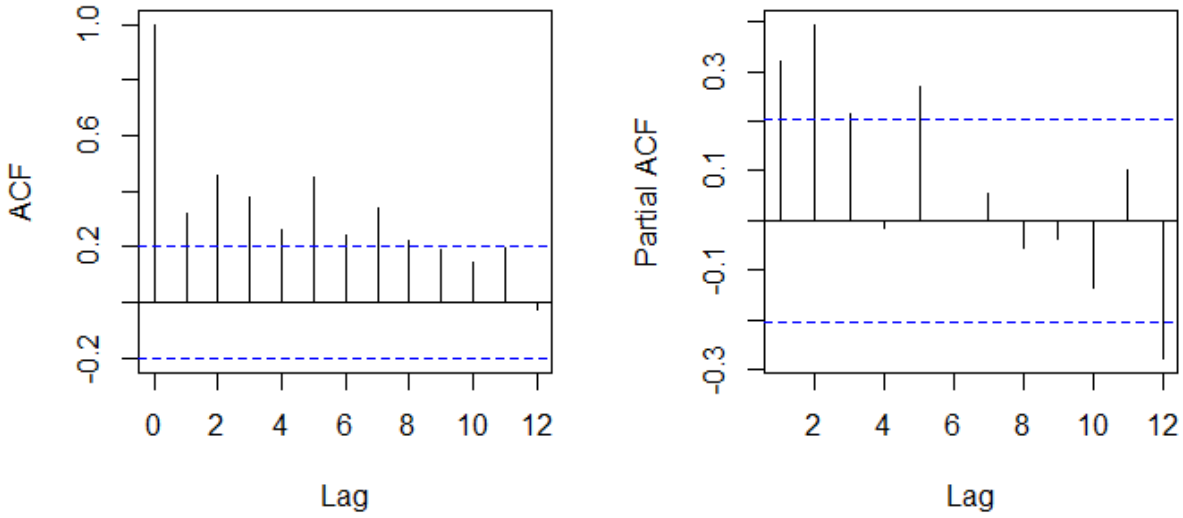


Figure 10 ACF and PACF for differentiated U.S. monthly unemployment rate series. Made by Author using R. Source: Bureau of Labor Statistics

7.4 Appendix D – Residual diagnostics

.

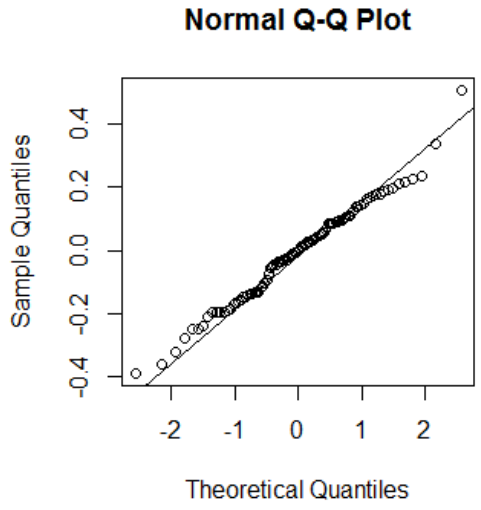


Figure 11 Normal Q-Q Plot for ARIMA(3,1,0). Made by Author using R.

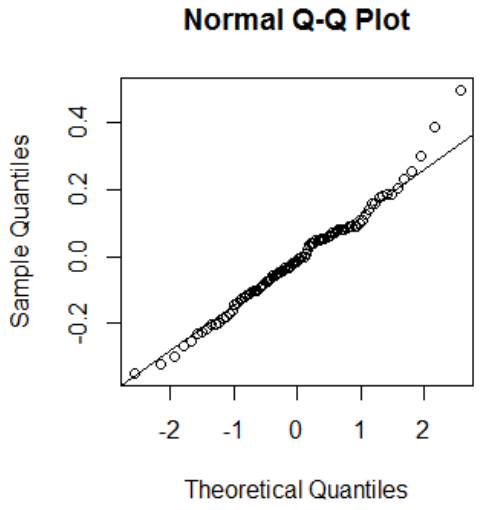


Figure 12 Normal Q-Q Plot for ARIMA(3,1,1) Made by Author using R.

7.5 Appendix E – $ARIMA(3,1,0)$ and $ARIMAX$ in sample results

Inc. variables	Benchmark		Target		
	Base	IC	Jobs	Jobs + Benefits	Combined
$ARIMA(3,1,0)$	✓	✓	✓	✓	✓
$lnIC_{t-1}$		0.44			0.47
		(0.37)			(0.35)
$lnGI\_Jobs_t$			-1.54**	-1.59**	-1.61**
			(0.65)	(0.64)	(0.64)
$lnGI\_Benefits_t$				0.16*	0.17*
				(0.09)	(0.09)
N	95	94	95	95	94
AIC	-73.97	-70.71	-76.63	-77.96	-77.82
BIC	-63.80	-58.05	-63.91	-62.70	-60.01
RMSE	0.155	0.155	0.150	0.148	0.146

Table 11 In-sample results for ARIMA(3,1,0) and ARIMAX(3,1,0). Based on calculations made by Author.

Notes: (i) *Significant at the 0.10 level
(ii) **Significant at the 0.05 level
(iii) ***Significant at the 0.01 level

7.6 Appendix F – Predictions

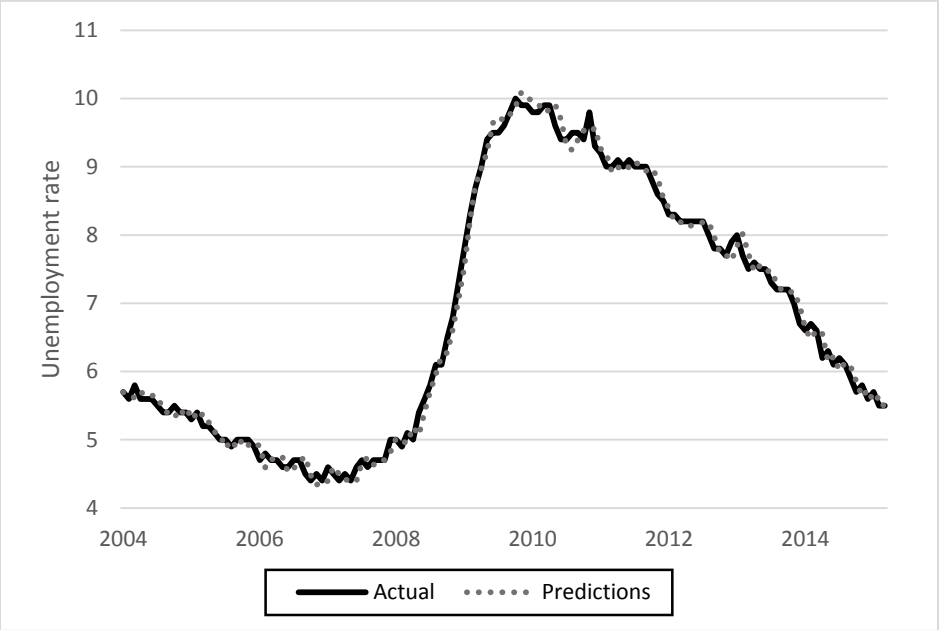


Figure 13 Forecasting performance of Combined model ($ARIMAX(3,1,1)$ with exogenous variables GI for Jobs, GI for Unemployment Benefits and IC). Made by Author. Source: Bureau of Labor Statistics

7.7 Appendix G – Multiple steps ahead out of sample results for $ARIMA(3,1,1)$

Recursive method

Steps	Benchmark		Target		
ahead	Base	IC	Jobs	Jobs + Benefits	Combined
1	0.156	0.152	0.151	0.155	0.151
2	0.227	0.224	0.214	0.222	0.221
3	0.278	0.276	0.256	0.261	0.257
4	0.296	0.293	0.277	0.284	0.282
5	0.328	0.324	0.311	0.321	0.320
6	0.386	0.385	0.363	0.375	0.378
7	0.473	0.471	0.434	0.445	0.450
8	0.517	0.515	0.464	0.472	0.476
9	0.561	0.559	0.504	0.511	0.519
10	0.634	0.632	0.574	0.582	0.591
11	0.733	0.731	0.668	0.674	0.683
12	0.832	0.831	0.749	0.750	0.759

Table 12 RMSE results for multiple steps ahead using ARIMA(3,1,1) as kernel model and recursive estimation method. Based on calculations made by Author.

Steps	Benchmark		Target		
ahead	Base	IC	Jobs	Jobs + Benefits	Combined
1	0.119	0.115	0.121	0.123	0.118
2	0.173	0.168	0.170	0.170	0.164
3	0.224	0.223	0.220	0.218	0.215
4	0.236	0.232	0.233	0.233	0.229
5	0.275	0.269	0.271	0.272	0.266
6	0.290	0.291	0.289	0.287	0.288
7	0.382	0.380	0.374	0.373	0.367
8	0.395	0.394	0.369	0.371	0.371
9	0.456	0.454	0.420	0.408	0.407
10	0.518	0.515	0.478	0.468	0.470
11	0.595	0.593	0.545	0.537	0.534
12	0.682	0.682	0.624	0.611	0.611

Table 13 MAE results for multiple steps ahead using ARIMA(3,1,1) as kernel model and recursive estimation method. Based on calculations made by Author.

Rolling window method

Steps	Benchmark		Target		
ahead	Base	IC	Jobs	Jobs + Benefits	Combined
1	0.159	0.153	0.155	0.155	0.153
2	0.241	0.231	0.224	0.227	0.226
3	0.297	0.291	0.273	0.272	0.266
4	0.325	0.310	0.301	0.299	0.293
5	0.356	0.345	0.335	0.337	0.327
6	0.421	0.411	0.388	0.393	0.388
7	0.514	0.500	0.463	0.470	0.457
8	0.564	0.544	0.498	0.506	0.484
9	0.601	0.583	0.544	0.550	0.524
10	0.674	0.652	0.616	0.626	0.600
11	0.771	0.751	0.712	0.725	0.692
12	0.874	0.854	0.805	0.813	0.770

Table 14 RMSE results for multiple steps ahead using ARIMA(3,1,1) as kernel model and rolling estimation method. Based on calculations made by Author.

Steps	Benchmark		Target		
ahead	Base	IC	Jobs	Jobs + Benefits	Combined
1	0.127	0.119	0.124	0.124	0.121
2	0.181	0.171	0.177	0.178	0.170
3	0.236	0.233	0.228	0.227	0.221
4	0.262	0.251	0.252	0.249	0.241
5	0.300	0.287	0.284	0.291	0.274
6	0.329	0.324	0.306	0.306	0.304
7	0.410	0.403	0.389	0.393	0.378
8	0.442	0.425	0.396	0.399	0.379
9	0.506	0.486	0.453	0.451	0.409
10	0.561	0.537	0.519	0.517	0.477
11	0.643	0.617	0.590	0.590	0.539
12	0.736	0.714	0.672	0.671	0.619

Table 15 MAE results for multiple steps ahead using ARIMA(3,1,1) as kernel model and rolling estimation method. Based on calculations made by Author.

7.8 Appendix H – R packages

Package	Purpose
seasonal	Seasonal adjustment
forecast	ARIMA and ARIMAX
tseries	Transforming series to time-series form
urca	Augmented dickey fuller test and DF-GLS test
xslx	Importing and exporting from/to Excel

Table 16 R packages used in the analysis.

7.9 Appendix I – R Code

```
######################

# Final Script

# Topic: Forecasting U.S. unemployment using Google Trends

# Author: Rokas Narkus

######################

#### Setting up workspace ####

# Setting working directory

setwd("C:/Users/Narkus/Dropbox/Thesis/Final/Data")

# Loading libraries

library(xlsx,quietly = T)

library(seasonal,quietly = T)

library(forecast,quietly = T)

library(tseries,quietly = T)

library(urca)


# Getting user-defined scripts later used in analysis

source("C:/Users/Narkus/Dropbox/Thesis/Final/Scripts/Get_estimates.R")

source("C:/Users/Narkus/Dropbox/Thesis/Final/Scripts/descriptive_stats_functi
on.R")

# Read data

data<-read.xlsx("Final.xlsx",1,header=TRUE)

names(data)

# [1] "Rate"        "IC"          "GI_emp"      "GI_jobs"     "GI_benefits"

# [6] "GI_unemp"    "GI_center"   "L_IC"
```

```
head(data)

#  Rate     IC   GI_emp  GI_jobs GI_benefits GI_unemp GI_center   L_IC

#1  5.7 366250 79.55984 69.73606          12 19.61982  63.15291 353250

#2  5.6 347250 79.25722 70.50945          11 19.38446  64.00399 366250

#3  5.8 348500 81.36113 69.64098          13 19.02661  59.24250 34725

### 3.2 Descriptive statistics ###

#descriptive_stats<-descriptive_stats_function(data)

#write.xlsx(descriptive_stats,file="Final_Descriptive_stats.xlsx",col.names=T
RUE,row.names=TRUE, sheetName="Descriptive Stats")

### 3.3 Transformations ###

# Unit root testing

# Augmented-Dickey Fuller test

ur.df(data[,'Rate'],lags=5,type='drift')

# Test for optimal lags => no remaining serial correlation in error term

res<-attributes(ur.df(data[,'Rate'],lags=5,type='drift'))$res

Box.test(res, lag=12, type="Ljung-Box")

summary(ur.df(data[,'Rate'],lags=5,type='drift'))

summary(ur.ers(data[,'Rate'],type="DF-GLS",model="constant",lag.max=5))


### 5.1.1 Base benchmark selection ###

# Setting up data

y=data[,'Rate']

f_size=length(y)

tr_size=ceiling(0.7*f_size)


## Box-Jenkins methodology

par(mfrow=c(1,2))

acf(y[1:tr_size],12)

pacf(y[1:tr_size],12)
```

```
# differentiate the series
d_y=y[2:f_size]-y[1:(f_size-1)]
acf(d_y[1:(tr_size-1)],12)
pacf(d_y[1:(tr_size-1)],12)

## Select model based on Information criteria
Mdl=c(3,1,1)
EstMdl=arima(y[1:tr_size],order=Mdl)
summary(EstMdl)
# select model with lowest information criteria

## Residual diagnostics
results_residuals=EstMdl$residuals
mean(results_residuals[1:tr_size])
jarque.bera.test(results_residuals[1:tr_size])
shapiro.test(results_residuals[1:tr_size])
Box.test(results_residuals[1:tr_size], lag=12, type="Ljung-Box")
qqnorm(results_residuals[1:tr_size])
qqline(results_residuals[1:tr_size])

### 5.1.2 ARIMAX selection ###
## Setting up exogenous variables
Mdl_ex_names<-list()
Mdl_ex_names[[1]]<-{}
Mdl_ex_names[[2]]<-c("L_IC")
Mdl_ex_names[[3]]<-c("GI_jobs")
Mdl_ex_names[[4]]<-c("GI_jobs","GI_benefits")
Mdl_ex_names[[5]]<-c("L_IC","GI_jobs","GI_benefits")
Mdl=c(3,1,1)
for (i in 2:length(Mdl_ex_names)){
  yTrain<-y[1:tr_size]
```

```
  xTrain<-log(data[1:tr_size,Mdl_ex_names[[i]]])

  EstMdl<-arima(yTrain,Mdl,xreg=xTrain)

  summary(EstMdl)

}

### 5.2. Out-of-sample results ###

### 5.2.1 Predictions ###

results<-data.frame()

results_residuals<-data.frame()

results_RMSE<-data.frame()

results_MAE<-data.frame()

n_ahead<-1   #no of predictions ahead (base case = 1)

tr_part<-0.7 #training sample size %

method<-"recursive" #estimation scheme

results[1:(f_size-n_ahead+1),1]<-data[1:(f_size-n_ahead+1),'Rate']

for (i in 1:length(Mdl_ex_names)){

  results[1:(f_size-n_ahead+1),(i)]<-
Get_estimates(Mdl_ex_names[[i]],Mdl,data,n_ahead,method,tr_part)

  results_residuals[1:(f_size-n_ahead+1),(i)]<-results[1:(f_size-
n_ahead+1),(i)]-data[n_ahead:(f_size),'Rate']

  results_RMSE[1,i]<-
sqrt(mean(results_residuals[(tr_size+1):(length(results_residuals[,i])),i]^2)
)

  results_MAE[1,i]<-
mean(abs(results_residuals[(tr_size+1):length(results_residuals[,i]),i]))

}

results_RMSE

results_MAE

### 5.2.2 Evaluation ###  # Diebold Mariano test

results_DM<-data.frame()

for (i in 1:length(results_residuals)){

  results_DM[1,i]<-
dm.test(results_residuals[(tr_size+1):length(results_residuals[,1]),1],result
s_residuals[96:length(results_residuals[,1]),i],h=1,alternative="greater",
power=2)$statistic
```

```
  results_DM[2,i]<-
dm.test(results_residuals[(tr_size+1)::length(results_residuals[,1]),1],resul
ts_residuals[96:length(results_residuals[,1]),i],h=1,alternative="greater",
power=2)$p.value}

results_DM


### 5.2.3 Multiple steps-ahead ###

# To calculate x-steps ahead forecasting accuracy

Mdl<-c(3,1,1)

results_steps_ahead_RMSE<-data.frame()

results_steps_ahead_MAE<-data.frame()

for (i in 1:length(Mdl_ex_names)){

  for (x in 1:12){

    n_ahead=x

    results<-data.frame()

    results_residuals<-data.frame()

    results[1:(f_size-n_ahead+1),1]<-
Get_estimates(Mdl_ex_names[[i]],Mdl,data,n_ahead,method,tr_part)

    results_residuals[1:(f_size-n_ahead+1),1]<-results[1:(f_size-
n_ahead+1),1]-data[n_ahead:(f_size),'Rate']

    results_steps_ahead_RMSE[x,i]<-
sqrt(mean(results_residuals[(tr_size+1):(length(results_residuals[,1])),1]^2)
)

    results_steps_ahead_MAE[x,i]<-
mean(abs(results_residuals[(tr_size+1):length(results_residuals[,1]),1]))

  }

}

results_steps_ahead_RMSE

results_steps_ahead_MAE
```

```
### 6. Robustness checks ###

### 6.1.1 Full Sample estimation ###

Mdl=c(3,1,1)


# Base benchmark model

summary(arima(y,Mdl))

for (i in 2:length(Mdl_ex_names)){

  yFull<-y

  xFull<-log(data[,Mdl_ex_names[[i]]])

  EstMdl<-arima(yFull,Mdl,xreg=xFull)

  summary(EstMdl)}

### 6.1.2 Rolling window scheme ###

results<-data.frame()

results_residuals<-data.frame()

results_RMSE<-data.frame()

results_MAE<-data.frame()

n_ahead<-1   #no of predictions ahead (base case = 1)

tr_part<-0.7 #training sample size %

method<-"rolling" #estimation scheme

results[1:(f_size-n_ahead+1),1]<-data[1:(f_size-n_ahead+1),'Rate']

for (i in 1:length(Mdl_ex_names)){

  results[1:(f_size-n_ahead+1),(i)]<-
Get_estimates(Mdl_ex_names[[i]],Mdl,data,n_ahead,method,tr_part)

  results_residuals[1:(f_size-n_ahead+1),(i)]<-results[1:(f_size-
n_ahead+1),(i)]-data[n_ahead:(f_size),'Rate']

  results_RMSE[1,i]<-
sqrt(mean(results_residuals[(tr_size+1):(length(results_residuals[,i])),i]^2)
)

  results_MAE[1,i]<-
mean(abs(results_residuals[(tr_size+1):length(results_residuals[,i]),i]))}

results_RMSE

results_MAE
```

```
### 6.1.3 Changing sample size ###

# change training sample % part to 0.6 and 0.8

tr_part<-0.8


results[1:(f_size-n_ahead+1),1]<-data[1:(f_size-n_ahead+1),'Rate']

for (i in 1:length(Mdl_ex_names)){

  results[1:(f_size-n_ahead+1),(i)]<-
Get_estimates(Mdl_ex_names[[i]],Mdl,data,n_ahead,method,tr_part)

  results_residuals[1:(f_size-n_ahead+1),(i)]<-results[1:(f_size-
n_ahead+1),(i)]-data[n_ahead:(f_size),'Rate']

  results_RMSE[1,i]<-
sqrt(mean(results_residuals[(tr_size+1):(length(results_residuals[,i])),i]^2)
)

  results_MAE[1,i]<-
mean(abs(results_residuals[(tr_size+1):length(results_residuals[,i]),i]))

}

results_RMSE

results_MAE
```

R Code – *Get_estimates* function

```
Get_estimates<-function(ex_names,Mdl,data,n_ahead,method,tr_part){
  ####In-Sample#####
  y<-data[,'Rate']
  f_size=length(y)
  tr_size=ceiling(tr_part*f_size)
  tt_size=f_size-n_ahead
  win_size=tr_size

  yTrain<-data[1:(tr_size),'Rate']
  xTrain<-log(data[1:(tr_size),ex_names])

  EstMdl<-arima(yTrain,order=Mdl,xreg=xTrain,method="CSS-ML")
  y_hat_train<-yTrain-EstMdl$residuals

  ####Out-of-Sample#####
  if (length(ex_names)==0){
    m<-0
    est_y<-vector()
    act_y<-y[(tr_size+n_ahead):(f_size)]
    for (i in (tr_size):(f_size-n_ahead)){
      if (method=="recursive"){win_size=i}
      m<-m+1
      yEst<-y[(i-win_size+1):i]
      EstMdl<-arima(yEst, order=Mdl, method="CSS-ML")
      est_y[m]<-predict(EstMdl,n.ahead=n_ahead)$pred[n_ahead]
    }
  }else if(length(ex_names)>1) {
    x<-log(data[,ex_names])
```

```
    m<-0
    est_y<-vector()
    act_y<-y[(tr_size+n_ahead):(f_size)]
    xNew<-
data.frame(matrix(rep(1,ncol(x)*n_ahead),nrow=n_ahead,ncol=ncol(x)))
    for (i in (tr_size):(f_size-n_ahead)){
      m<-m+1
      if (method=="recursive"){win_size=i}
      yEst<-y[(i-win_size+1):i]
      xEst<-x[(i-win_size+1):i,1:ncol(x)]
      EstMdl<-arima(yEst, order=Mdl,xreg=xEst, method="CSS-ML")
      # Create estimates of exogenous variable
      for (k in 1:ncol(x)){
        EstMdl_x<-arima(xEst[,k],order=c(1,0,0))
        if (ex_names[k]=="L_IC"){
          if (n_ahead>1){
            xNew[1,k]<-x[(i+1),"L_IC"]
            EstMdl_x<-arima(c(xEst[,k],xNew[1,k]),order=c(1,0,0))
            xNew[2:n_ahead,k]<-predict(EstMdl_x,n.ahead=(n_ahead-1))$pred
          }
          xNew[1,k]<-x[(i+1),"L_IC"]
        }else {
          xNew[1:n_ahead,k]<-predict(EstMdl_x,n.ahead=n_ahead)$pred
        }
      }
      est_y[m]<-predict(EstMdl,n.ahead=n_ahead,newxreg=xNew)$pred[n_ahead]
    }
  } else{
    x<-log(data[,ex_names])
    m<-0
    est_y<-vector()
```

```
    act_y<-y[(tr_size+n_ahead):(f_size)]
    xNew<-data.frame(rbind(rep(1,n_ahead)))
    for (i in (tr_size):(f_size-n_ahead)){
      m<-m+1
      if (method=="recursive"){win_size=i}
      yEst<-y[(i-win_size+1):i]
      xEst<-x[(i-win_size+1):i]
      EstMdl<-arima(yEst, order=Mdl,xreg=xEst, method="CSS-ML")
      # Create estimates of exogenous variable
      EstMdl_x<-arima(xEst,order=c(1,0,0))
      if (ex_names[1]=="L_IC"){
        if (n_ahead>1){
          xNew[1,m]<-x[(i+1)]
          EstMdl_x<-arima(c(xEst,xNew[1,m]),order=c(1,0,0))
          xNew[2:n_ahead,m]<-predict(EstMdl_x,n.ahead=(n_ahead-1))$pred
        }else{
          xNew[m]<-x[(i+1)]
        }
      }else {
        xNew[1:n_ahead,m]<-predict(EstMdl_x,n.ahead=n_ahead)$pred
      }
      est_y[m]<-
predict(EstMdl,n.ahead=n_ahead,newxreg=xNew[,m])$pred[n_ahead]
    }
  }
  y_hat<-c(y_hat_train,est_y)
  return(y_hat) # Returns estimated target series
}
```