Forecasting Project:

Unemployment level in the US

BAN430

Spring 2021

Total pages: 31

Candidates:

71 & 90

1. Introduction 2
1.1 Model selection information criterion: AIC, AICc, BIC and Adjusted R-squared 2
1.2 Forecasting evaluation metrics 3
2. Data 4
2.1 Unemployment level in the US 4
2.2 Descriptive statistics 5
4. Decomposition and component forecasting 7
4.1 Decomposition: season, trend and remainder components 7
4.2 Forecast of seasonal and seasonal adjusted components 7
4.3 Forecast with seasonal and seasonal adjusted components 9
5. Exponential smoothing (ETS) forecast 9
6. Autoregressive integrated moving average (ARIMA) forecast 11
7. Multivariate forecast 13
8. Dynamic forecast 15
9.Forecasting using deterministic terms 16
9.1 Deterministic trend forecast 17
9.2 Fourier terms forecast 17
10. Combinational forecast 18
11. Monte-Carlo simulation 19
12. Comparison of type-optimal forecasting methods 21
13. Conclusion 22
14. Literature 23
14. Appendix 26
14.1 26
14.2 26
14.3 27
14.4 27
14.5 28
14.6 28
14.7 29
14.8 29

1. Introduction

Unemployment level is a key indicator for the healthiness of an economy and being able to predict the unemployment levels in the future has tremendous value. The purpose of this report is to use a variety of forecasting methods on unemployment levels in the US and compare them in terms of quantifiable metrics. We will present relevant summary statistics, before proceeding to replicate, as closely as possible, the seasonally adjusted data decomposed by the U.S Labor Statistics Bureau. We will use the fable package to produce optimal ETS and ARIMA models, and briefly discuss why they are optimal. The forecast horizon is 24 months, unless otherwise stated.

We begin by reviewing the summary statistics of the unemployment training set. We will then look at the components of the unemployment time series to adjust for seasonal variation and compare this to a benchmark made by the US Bureau of Labor Statistics. To improve our forecast, we will add two additional independent variables, namely the Consumer Price Index and Export data in the US. These variables are chosen based on their assumed relationship with the level of unemployment. To test this, we will construct a multivariate VAR, VECM and several dynamic regression models. Aside from using stochastic trends, we will attempt to fit a model using deterministic trends. Specifically, we will fit an ARIMA model using linear trends, and an ARIMA model with Fourier terms. Based on our best models, as measured by our chosen performance metrics, we will perform a combinational forecast. As we will see, this serves a simple yet effective way of improving forecast accuracy. Lastly, we perform a Monte Carlo simulation of the effect of including another independent variable on the accuracy of a forecast.

Throughout this report we will emphasize our methodology and steps taken with the goal of making our findings as reproducible as possible. All our results are generated using the R statistical programming language (version 4.0.5), which is included in the submission of this report along with the data stored as an Rdata file.

1.1 Model selection information criterion: AIC, AICc, BIC and Adjusted R-squared

There are several information criteria which can evaluate the performance within the same class of forecasting models. The R-squared and its predictor adjusted counterpart, are very prevalent in statistical literature. While common, even the adjusted R-squared tends to select too many predictors (Hyndman, 2021). Another more commonly used information criterion for model selection are AICc[1], AIC[2] and BIC[3], where AIC and BIC are pursuing different goals according to Jouni Kuha (2004, p. 217). AIC attempts to select a model based on its accuracy of predicting future data entries, whereas BIC attempts to reveal an underlying causal relationship. AICc imposes a penalty based on the number of predictors included in the model, and may reduce overfitting risk incurred by including too many predictors. Considering that the goal for this report is to select the best forecast model, and not the true underlying relationship, we will use the AICc information criterion, unless otherwise stated.

1.2 Forecasting evaluation metrics

There are primarily two types of forecasting evaluation accuracy metrics: scaled and percentage error metrics. As percentage error metrics may yield unstable results, when a time series values approach zero, scaled metrics are preferable. The Mean Absolute Scaled Error (MASE) was first proposed by Hyndman and Koehler in 2005 as a scaled metric. This metric has the advantage of being more interpretable than other performance metrics. As an example, a MASE value of 2 means that the absolute scaled error is twice the size, on average, as those of seasonal naïve forecasts on the training set. We will include several forecasting evaluation metrics across this report, but the MASE evaluation metric will be emphasized.

[1] Akaike's Information Criterion corrected
[2] Akaike's Information Criterion
[3] Bayesian Information Criterion

2. Data

2.1 Unemployment level in the US

The time series of seasonal adjusted and unadjusted unemployment levels in the US is gathered by the US Bureau of Labor Statistics, made available by the Federal Reserve Bank of St. Louis. This unemployment time series comprise the total unemployment level in the US. In addition, we choose Consumer Price Index (CPI) and Export[4] of the US as variables for later use in the multivariate and dynamic regression section.

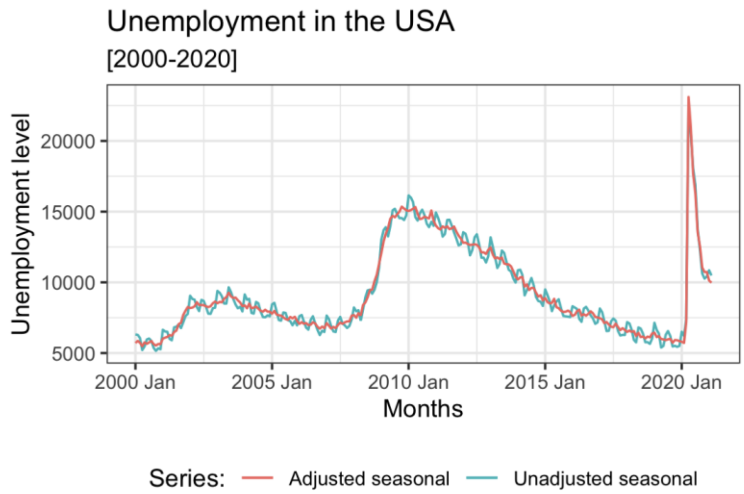


Figure 1: Unemployment level in the US Jan 2000- Dec2020

The unemployment level follows a cyclical pattern, where the length of each cycle is hard to determine on forehand. The extraordinary increase in unemployment caused by the economic consequences incurred by the COVID-19 pandemic may serve as a poor period to train and test forecasting models. Figure 1 clearly shows the anomalies of COVID-19 in 2020, where the unemployment level is skyrocketing for a few weeks. A model tested against 2020 data would, in most circumstances, underestimate the observed unemployment level. After some deliberation, we have opted to restrict the dataset to include monthly data ranging from 2000 to the end of 2019. The size of training data is 2000 to 2017, which contains 216 monthly observations, while the test set consists of 2018 to 2019 data. This is an adequate size to capture business trend-cycle effects and adequate to avoid historical biased information.

[4] More information on these variables can be found in section 7 in this report.

2.2 Descriptive statistics

In this section we will explore the unemployment time series through relevant tables and plots.

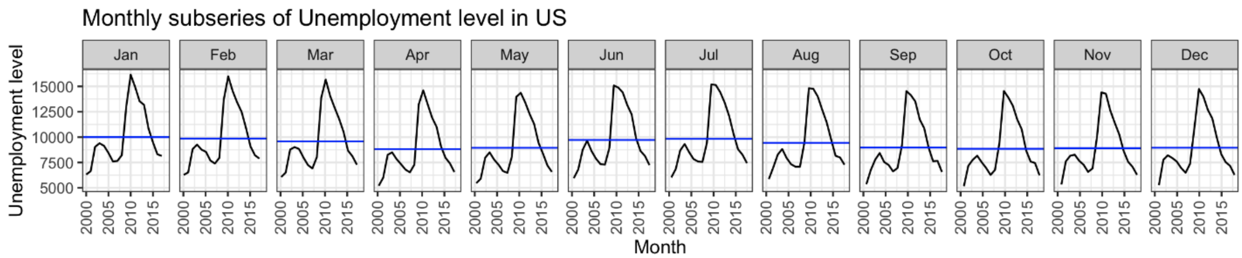


Figure 2: Monthly subseries of unemployment level in the US

The subseries of the unemployment level, in figure 2 indicates a clear tendency of seasonality in the time series mean, represented as a horizontal blue line in each subseries. From the subseries there are some signs of higher unemployment in the summer months, as well as the winter months of January and February. There is a decreasing trend of unemployment, from January 2010, where the unemployment level appears to move below the mean for each subseries. The summary statistic for each month is, in addition, represented in figure 3 with a minimum value, 25%-percentile, mean, median, 75%-percentile and maximum value.

Summary statistics of unemployment level in US

Month	Min	25%-percentil	Mean	Median	75%entil	Max
1	6316	8167.00	10005.611	9097.5	12470.50	16147
2	6284	7903.50	9861.667	8796.5	12098.25	15991
3	6069	7459.50	9575.444	8729.0	11495.50	15678
4	5212	6924.75	8822.111	7901.5	10530.25	14609
5	5460	6793.00	8947.444	8022.5	10837.25	14369
6	5959	7473.25	9721.278	8698.0	11659.25	15095
7	6028	7661.25	9828.611	8749.0	11639.00	15201
8	5863	7297.00	9437.833	8216.5	11043.25	14823
9	5359	7028.75	8975.556	7724.0	10463.50	14538
10	5153	7016.75	8850.389	7683.0	10447.00	14547
11	5336	7117.25	8909.333	7917.5	10207.00	14407
12	5264	7220.25	8954.722	7859.0	10745.25	14740

Figure 3: Monthly summary statistics

In the period between year 2000 and 2017 the minimum unemployment level was at 5,132 in 2000 and peaking January 2010 with 16.147 unemployed workers, represented in figure 4 This coincides with the global financial crisis of 2007-2010, which struck the US particularly hard. In the corresponding time interval, the average unemployment level and median were 9,324 and 8,270, respectively. From the monthly subseries plot one can see that, on average, the unemployment level is highest in January and July, and lowest in April and October.

Summary statistics of unemployment level in US

Min	25%-percentil	Mean	Median	75%-percentil	Max
5153	7287	9324.167	8270	11327.5	16147

Figure 4: Aggregate summary statistics

3. Replication of Seasonal Adjusted Data

The purpose of this section is to find a decomposition method that replicates the U.S Bureau of Labor Statistics' (referred to as the Bureau) seasonal adjusted time series of unemployment in the US, as closely as possible. By consulting the Bureau's documentation pages it states that they use X12 in data sets from 2005-2015 and X13-SEATS from 2015 (Tiller & Evans, 2018). For unemployment time series, however, the X11 is still used, and as a result using the X11 as our decomposition method we were able to achieve the closest fit.

Using the seasonal package, we perform a decomposition on the unadjusted unemployment data. We have opted to decompose our time series using three methods: SEATS, X11 and STL. As X13-SEATS have been adopted by many statistical bureaus across the world, we deemed it an apt method to decompose our time series. The STL is an acronym for "Seasonal and Trend Decomposition using Loess" developed by R. B. Cleveland, Cleveland, McRae, and Terpenning (1990) and is a widely used decomposition method, as well as being more robust to outliers (Hyndman, 2021). Figure 5 shows common evaluation metrics on the training set where we opt to compare X11, X13-SEATS and STL decomposition methods.

Evaluation metrics of decomposition methods

	ME	RMSE	MAE	MPE	MAPE
X11	0.60	86.31	63.47	0.00	0.73
X13	-0.25	101.12	77.68	-0.01	0.86
STL	0.23	115.28	79.76	0.00	0.81

Figure 5: Evaluation of decomposition methods against the Bureau's seasonally adjusted series

Note from figure 6 that although the X11 method, which uses the seasonal adjusted component, clearly has the closest fit to the Bureau's own seasonal adjusted unemployment level, it is not an exact match. The difference of our approach, in contrast to the Bureau's adjusted seasonal series, is that they adjust various industries's unemployment levels before aggregating them to total unemployment levels (Tiller & Evans, 2018). Since these adjustments are beyond the scope for the purpose of this report, we can still conclude that X11 method, from the seasonal package, is a good

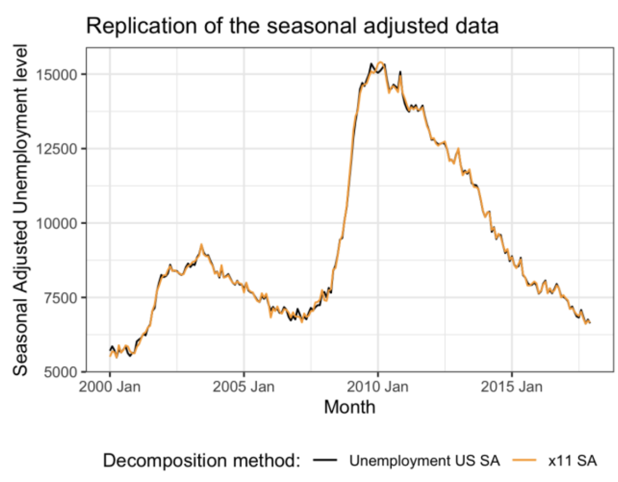


Figure 6: X11 decomposition method without alteration of data against Bureau's decomposition

replication of the seasonal adjusted time series based on the lowest RMSE in figure 5

4. Decomposition and component forecasting

4.1 Decomposition: season, trend and remainder components

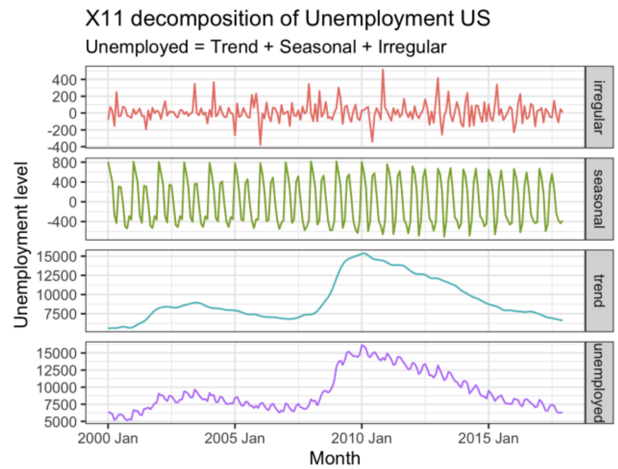


Figure 7: X11 decomposition plot

We will now use our selected decomposition method, X11, to isolate the seasonal and seasonally adjusted component and forecast using the addition of the separate forecasts. As the X11 decomposition method manages the closest fit the Bureau's own adjustment, it will be used to decompose our new training sets for the seasonal and seasonally adjusted component, where the latter is the sum of trend and irregularity (remainder). Since the magnitude of the seasonal fluctuations is stable around the unemployment level's trend-cycle, we opt to use additive decomposition (Hyndman & Athanasopoulos, 2021). The decomposition is provided in figure 7.

The trend component in figure 7 indicates two clear patterns of business trend-cycles from year 2000 to around 2007 and second from 2007 until 2017. Observing the seasonal component plot we note that the unemployment seems to be stable through the time. In more detail, there are signs that the seasonal variation is less symmetric each year from 2000 to around 2011, where typically only January has a clear unemployment level peak, while after 2011 two approximately even unemployment peaks can be observed.

4.2 Forecast of seasonal and seasonal adjusted components

The additive X11 method can be used for forecasting by adding the two main components, namely the seasonal and seasonal adjusted components (Hyndman & Athanasopoulos, 2018), where the seasonal adjusted term is simply the sum of the trend and remainder(irregularity) component.

The *seasonal component* appears to be stationary in terms of mean and variance, with little change across our training period. Hence, the seasonal naïve method may be a good forecast model. The seasonal naïve method takes the past year of the estimated component (Hyndman & Athanasopoulos, 2018). We can justify this model selection with a selection of performance metrics comparing it to three other methods such as mean, naïve and random walk with drift. The table presented as figure 8, indicates that seasonal naïve methods are performing very well, with the lowest MASE value. Considering the seasonal behavior shown in figure 9 it clearly shows that seasonal naïve fits the test set best compared to other methods.

Seasonal component forecast

Model	RMSE	MASE	MAE	MAPE	RMSSE
SNaive	55.153	2.184	45.912	12.724	2.047
Mean	447.664	20.220	425.121	99.971	16.619
Naive	587.929	21.262	447.014	114.480	21.826
Drift	628.664	22.952	482.546	126.449	23.338

Figure 8: Performance metrics for the seasonal component forecast

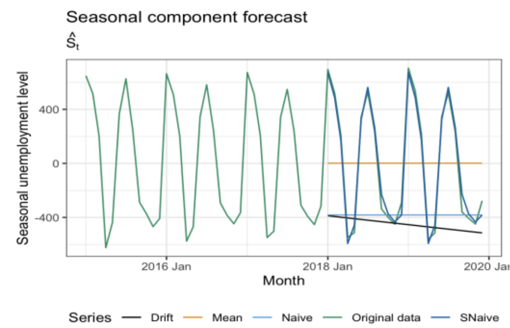


Figure 9: Forecast plot of the seasonal component (X11)

In order to forecast the *seasonal adjusted component*, one can use different forecasting methods as long as these methods are non-seasonal forecasting methods (Hyndman & Athanasopoulos, 2018). We opt to forecast this component through mean, random walk with drift, naïve, seasonal naïve, optimal ETS[5] where seasonal part is "None", and optimal ARIMA[6] adjusted with seasonal component PDQ(0,0,0). The evaluation metric table in figure 10 indicates that the optima ETS model outperforms the other methods with MASE of 0.108. This optimal model, ETS(M,Ad,N), has multiplicative error and damped additive trend components and is optimized by minimizing the AIC_c. The forecast in figure 11 shows that the model has a smooth damped fit compared to the test set (Original data).

Seasonally adjusted component forecast

Model	RMSE	MASE	MAE	MAPE	RMSSE
ETS	152.464	0.108	133.866	2.201	0.087
Arima	256.225	0.168	209.024	3.446	0.146
Naive	551.360	0.403	501.884	8.298	0.315
Drift	626.195	0.457	568.865	9.407	0.358
Mean	3172.861	2.543	3164.638	51.597	1.814

Figure 10: Performance metrics for the seasonally adjusted forecast

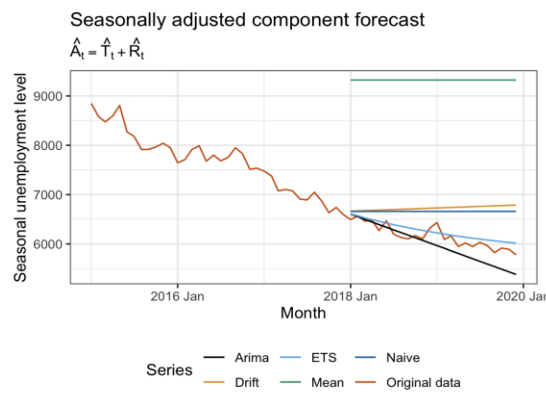


Figure 11: Forecast plot of the seasonally adjusted component (X11)

[5] More information about ETS can be found in section 5.
[6] More information about ARIMA can be found in section 6.

4.3 Forecast with seasonal and seasonal adjusted components

Using the best performing forecasting methods from the seasonal and seasonal adjusted components forecasts, one can use the additive method in order to form a forecast of unemployment level in the US on the test set. The additive forecasting formula is $\hat{y}_t = \hat{S}_t + \hat{A}_t$ where $\hat{S}_t$ is the seasonal component and $\hat{A}_t = \hat{T}_t + \hat{R}_t$ is the seasonal adjusted component comprising both trend and residual (Hyndman & Athanasopoulos, 2018). The forecasted series of $\hat{S}_t$ is derived from the seasonal naïve method, whereas the forecast of $\hat{A}_t$ is estimated by an $ETS(M, Ad, N)$. The formed forecast is displayed in the following figure 12 which has a plausible fit where the **MASE** on the unemployment test set is **0.125**.

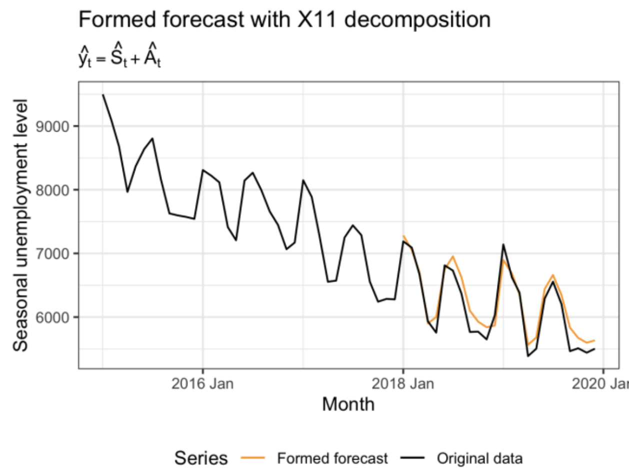


Figure 12: Forecast plot based on the formed X11 decomposition.

Model	RMSE	MASE	MAE	MAPE	RMSSE
Formed decomposition forecast	179.791	0.125	155.516	2.604	0.103

Figure 13: Performance metrics of the formed X11 decomposition forecast

5. Exponential smoothing (ETS) forecast

According to Robert Hyndman and George Athanasopoulos (2021) exponential smoothed forecasts are weighted averages of historical observations, where the more recent observations are higher weighted following an exponentially decaying weighting method. The ETS function in the fable package allows us to iterate through all possible ETS combinations, and select the best model by optimizing for the AIC_c. Our model is a seasonal version of the damped trend method as described in Roberts (1982). In this model the error component is additive, the trend model is additive damped and the seasonal part is additive ETS(A, Ad, A). Mathematically the model can be expressed as the equation provided in appendix 14.7. where the optimal coefficients of ETS(A, Ad, A) for the forecasting equations is given in figure 14:

Coefficients of ETS(A,Ad,A)

alpha	beta	gamma	phi	l	b	s0	s1	s2	s3	s4	s5	s6	s7	s8	s9	s10
0.88	0.18	0	0.93	5569.98	94.49	-387.56	-430.56	-484.45	-363.84	89.41	502.45	407.21	-377.99	-491.02	261.67	558.47

Figure 14: Coefficients of the optimal ETS model

The forecast provided in figure 15 using the estimated coefficients, is plausible because of the good fit on the test set with **MASE 0.14** but is worse than the formed decomposition forecast in section 4. The good fit may be explained by the fact that the forecast approaches the seasonal variation and the trend of the observed unemployment in the test set. We note that a dampening factor has been applied to the trend component of our optimized model. As unemployment is variable in which we can expect some reversion to the mean, dampening may seem reasonable. After a period of high unemployment, factors such as government action, investment level increases and falling wages may cause unemployment to revert. As stated in Hyndman (2021), work by Gardner & McKenzie (1985) suggests that adding a dampening term to the trend equation can resolve the issue of biased forecasting, because the dampening term "slows" the trend and makes it non-linear. The additive part of the error term can be argued for as the error term is approximately normally distributed with mean zero and constant variance, as shown in the decomposition plot of section 4.1.

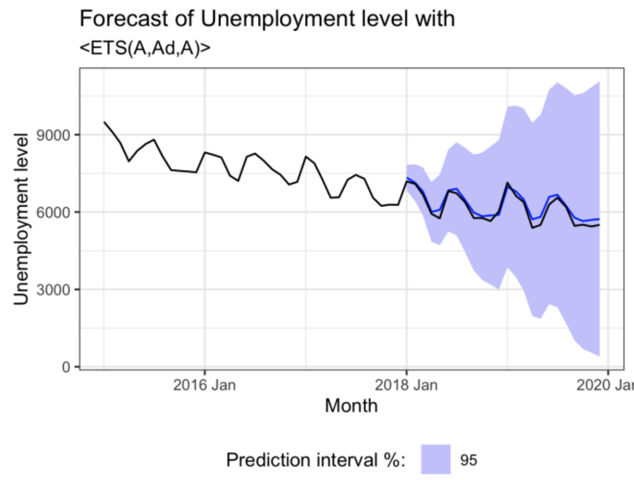


Figure 15: Optimal ETS model forecast plot

It is useful to check the residuals whether the model has adequately captured the information in the time series (Hyndman & Athanasopoulos, 2021). According to the residual plot the residuals has mean close to zero (not biased), the ACF plot does not indicate any significant correlation between the residuals, and the histogram shows approximately normally distributed residuals. This means that the residuals do not contain more information that should be extracted for better forecast, in other words, the residuals are only white noise. The prediction interval relies on normally distributed residuals, and a breach of this assumption can lead to unstable intervals. As we can see this is not a problem in our case.

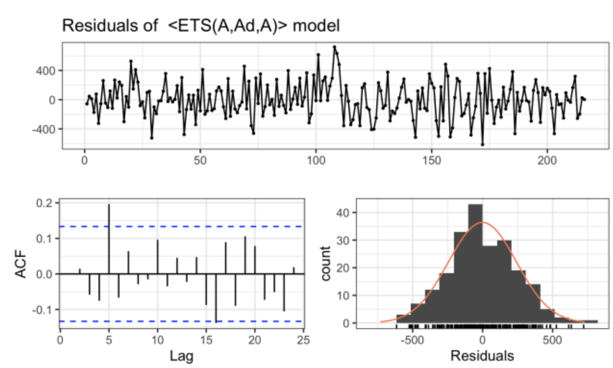


Figure 16: Residual plot of optimal ETS model

6. Autoregressive integrated moving average (ARIMA) forecast

An ARIMA model forecasts based on the autocorrelation that exists within the time series, in contrast to the exponential smoothing model which aims to only describe the trend and seasonal components (Hyndman & Athanasopoulos, 2018). It is composed of an autoregressive term, a differencing term and a moving average term. The ARIMA models are thus a general description of a variety of models that share some properties, but differ greatly. For the purpose of the selection of a seasonal $ARIMA(p,d,q)(P,D,Q)_m$ model, where m denotes the seasonal lag, we will conduct an automatic selection of an optimal ARIMA model on the AICc criteria. The stats package allows for an exhaustive search for an optimal ARIMA model based on our selected training data.

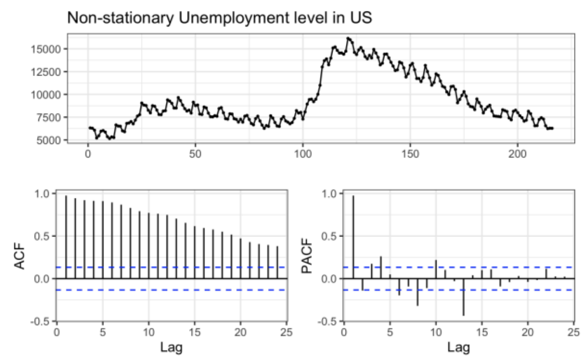


Figure 15: Time series, ACF and PACF plots of the unemployment time series

Plotting the autocorrelated lags of the unemployment level in the US, figure 17, reveals a clear problem of non-stationarity. An accepted method of dealing with non-stationarity in the mean of the time series is differencing. We conduct a KPSS to the original unemployment level which outputs a p-value of 1% indicating that the time series is not white noise (Kwiatkowski et al, 1992). Observing the ACF[7] and PACF[8] plot, we can see that there appears to be a seasonal pattern besides the autocorrelation problem. Hence, we first perform a

[7] ACF: Autocorrelation Function shows correlations between observations

[8] PACF: Partial Autocorrelation Function shows correlations between observations while adjusting for other lags

seasonal differencing, and investigate if there is a need for further differencing. The seasonal differenced in figure 18 seems to display less of the seasonal pattern, but the ACF plot continues to have a problem of autocorrelation.

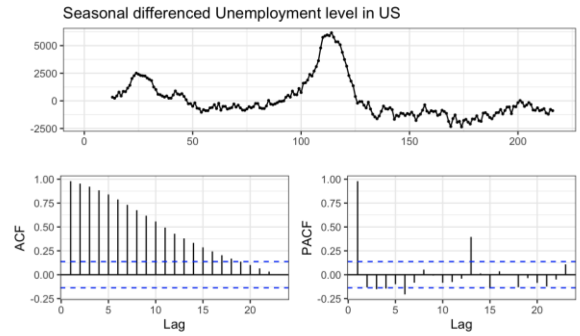


Figure 18: Seasonally differenced unemployment series

A new KPSS test on the seasonally differenced data states, again, problems with stationarity with 1% p-value. The next step is to conduct a first order differencing on the seasonal differenced unemployment level, which now yields a KPSS test with p-value of 10%, which gives evidence for stationarity. This plot in figure 19, lays now the foundation for choosing the autoregressive (AR) and moving averages (MA) components of the seasonal ARIMA model.

The autocorrelation plot in figure 19 shows clear evidence of seasonal correlation at lag 12 and 24 in the PACF plot, which can be treated as moving average of order one with lag in the seasonal part of 12 months. In the non-seasonal part there is a small indication of pattern lag five months behind in the ACF plot. This can be adjusted for with an autoregressive part of order 5 in the non-seasonal part of the seasonal ARIMA model. The optimal ARIMA model confirms these notions, as the AIC_c an optimized model is composed of an AR(5) term on the non-seasonal component and an MA(1) term on the seasonal component. As we have reasoned, it is also differenced at first order and seasonal levels. The forecast of this appropriate **ARIMA(5,1,0)(0,1,1)12** is provided in figure 20 and the accuracy has a **MASE** of **0.117**, which is even better than the formed decomposition forecast in section 4.

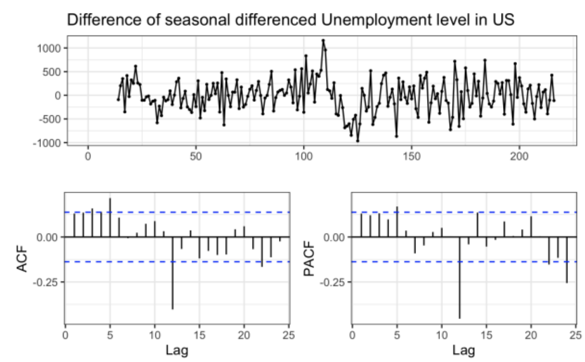


Figure 19: First order and seasonally differenced unemployment series

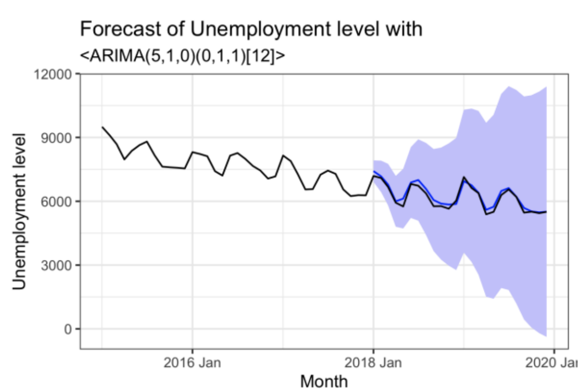


Figure 20: Forecast plot of optimal ARIMA model

The residual plot, found in appendix 14.3, is not indicating a problem of heteroscedasticity and the autocorrelation plot has no sign of any significant lags. Furthermore, the Ljung-box test does not reject the null hypothesis of white noise. The histogram of residual frequencies is approximately normally distributed with no sign of tails, which leads to a reliable prediction interval.

ARIMA(5,1,0)(0,1,1)12 forecast

Model	RMSE	MASE	MAE	MAPE	RMSSE
ARIMA_optimal	175.73	0.12	145.73	2.39	0.1

Figure 21: Performance metric table of optimal ARIMA model

7. Multivariate forecast

We proceed to expand our forecasts using a multivariate VAR model, composed of two additional variables, namely the Consumer Price Index data (CPI) in the US and the export of the US. These variables are retrieved from the OECD database. A multivariate Vector Autoregressive model (VAR) is a series of variables each composed of Autoregressive parts of its own lag and that of the other variables (Pennsylvania State, 2021). The CPI is defined as the annual change in price in a basket of goods, and it is common to exclude food and energy consumption from the CPI data, as these products are less sensitive to price changes. It could be a reasonable assumption that price changes interacts with unemployment as described by the Phillips-Curve, Phillips (1958). As many jobs are tied to exporting goods and services to foreign markets outside the United States, export data might improve our forecast as an added predictor. Lag diagnostics and a time series plot of CPI and export data is provided in figure 22 and 23.

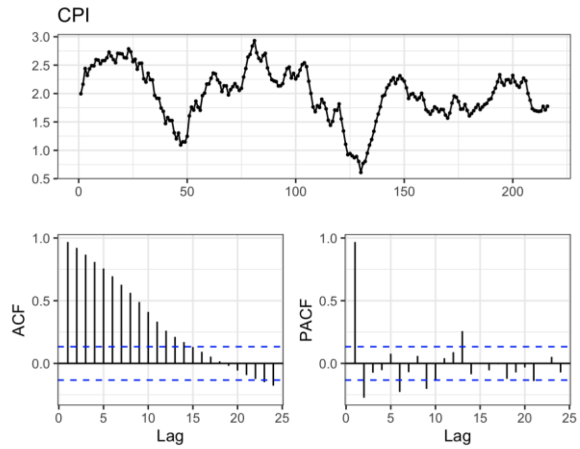


Figure 22: CPI time series, ACF and PACF plot

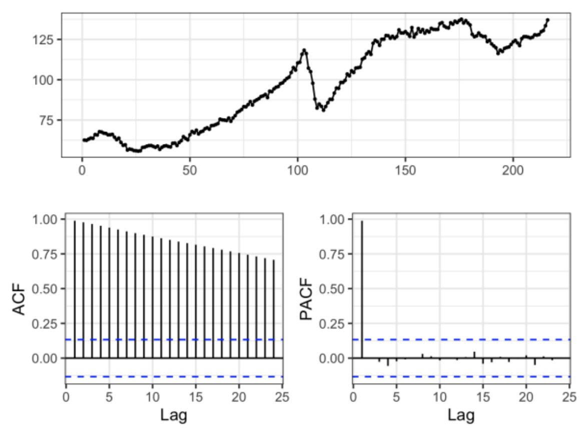


Figure 23: Export time series, ACF and PACF plot

We conduct KPSS unit root tests and confirm what the ACF plots for CPI and Exports data clearly shows. The null hypothesis of stationarity is rejected with p-value of 1%, 1.25% and 1% for Exports, CPI and the unemployment time series, respectively. First order differencing is conducted to cope with the stationarity issue, and we confirm that no further differencing is required by running another KPSS unit root test. We construct two VAR models: BIC and AIC_c optimized models with VAR terms of 3 and 5, respectively. We confirm that our models is stationary by conducting an augmented Dickey Fuller, in which we reject the null-hypothesis of non-stationarity with a p-value of 2.7% and 1% for the AIC_c and BIC optimized models, respectively. Figure 24displays the result of these VAR models, both subject to first order differencing. A portmanteau test for autocorrelation reveals that there are significant autocorrelation issues. Lütkepohl (2005) states that autocorrelation may not be an issue in forecasting as long as the model's performance is good. Hence, we will primarily try to deal with the non-stationarity.

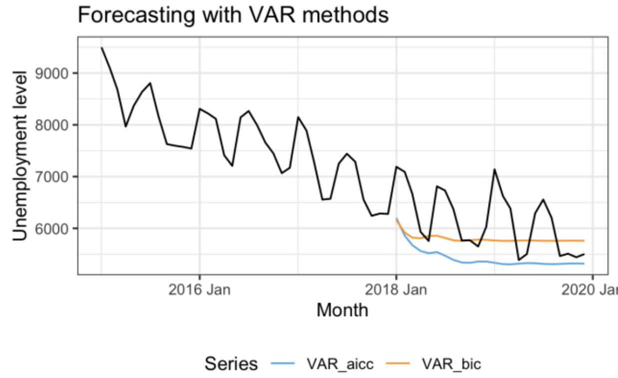


Figure 24: Forecast of BIC and AICc optimized VASR models

While the non-stationarity issue is resolved by taking the first order difference of our variables, we may lose valuable information contained within the variables. Our models will look more like a simple naïve forecast, and underfit the forecast. Instead of performing differencing, a non-stationary time series can be made stationary through cointegrated VECM model. Some variables have long term forces that draw them to a common mean, such as interest rates and investments. If such long term relationships can be proven, then we say that our variables are cointegrated, as described in Granger (1987, p. 2).

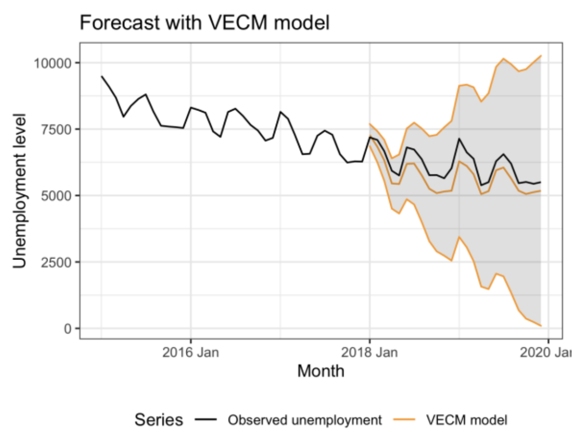


Figure 25: Forecast of VECM model

We perform a Johansen trace test for cointegrated relationships between our three variables, unemployment, CPI and Export. We find our cointegration level by choosing the level for which the null hypothesis is not rejected, in our case r = 1. In figure 25 the reader will find a plot with a forecast period of 24 months.

Based on the cointegrated levels, we construct a VECM model using a var lag order of 10 by passing our cointegration relationship value to the vec2var function from the vars package. This model solves the stationarity issue through cointegration, as we confirm with an augmented Dickey

Fuller test with p-value of 1.1%. It has a similar issue with autocorrelation, but it seems to outperform our differenced VAR models, with **MASE of 0.37**, in terms of all performance metrics as seen in the table provided in figure 26.

Multivariate VAR models

Model	Type	RMSE	MAE	MAPE	MASE	RMSSE
Multivariate VECM model VAR(10)	Test	498.57	465.35	7.58	0.37	0.29
Multivariate VAR model BIC optimized VAR(3)	Test	634.71	555.22	8.88	0.45	0.36
Multivariate VAR model AICc optimized VAR(5)	Test	647.06	566.90	9.05	0.46	0.37

Figure 26: Performance metrics of VAR and VECM models

8. Dynamic forecast

According to Hyndman and Athanasopoulos (2021) there is a need to either forecast each of the predictors or make an assumed future value if these predictors are unknown. At first we are fitting ARIMA, ETS, random walk with drift, naïve, seasonal naïve and mean models and comparing the different models on MASE, where the ARIMA is clearly the best model fitted on the training set. The fitted ARIMA model on CPI and Export results in ARIMA(2,1,3)(0,0,1)[12] and ARIMA(2,1,2)(0,0,2)[12] w/ drift, respectively. One of the arguments for using the resulting ARIMA models is that these take the non-stationarity problem into account by differencing on the non-seasonal component with first order differencing for both CPI and Export. This is due to the fact that the unit root test of KPSS under the null hypothesis of stationarity is rejected under 5%-significance level, p-value of 1.26% and 1% for CPI and Export, respectively. After a first order differencing of the predictors, with p-values of 10%, indicates evidence for stationarity by the unit root test of KPSS.

Model fitting of predictor: CPI

Model	Type	RMSE	MAE	MPE	MAPE	MASE	RMSSE
arima	Training	0.09	0.07	-0.47	4.08	0.16	0.16
ets	Training	0.12	0.09	-0.29	5.16	0.20	0.20
naive	Training	0.12	0.09	-0.29	5.18	0.21	0.20
drift	Training	0.12	0.09	-0.24	5.18	0.21	0.20
mean	Training	0.46	0.37	-7.88	22.94	0.82	0.82
snaive	Training	0.56	0.45	-7.74	26.95	1.00	1.00

Figure 27: Fit of models trained on CPI data

Model fitting of predictor: Export

arima	Training	2.04	1.48	-0.04	1.55	0.16	0.18
ets	Training	2.12	1.56	0.17	1.63	0.17	0.19
drift	Training	2.20	1.59	-0.05	1.66	0.18	0.20
naive	Training	2.23	1.62	0.34	1.69	0.18	0.20
snaive	Training	11.16	9.07	3.33	9.49	1.00	1.00
mean	Training	27.79	25.26	-9.50	29.44	2.79	2.49

Figure 28: Fit of models trained on exports data.

With the non-stationarity problem from the ARIMA section in mind, the dynamic forecasting model can be fitted with an ARIMA model using CPI and Export as additional predictors. By minimizing AICc the optimal model appears to be a *LM w/ ARIMA(1,1,4)(1,0,0)[12] errors*. The forecast

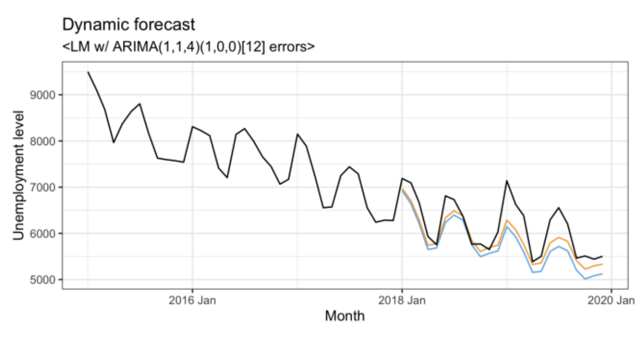


Figure 29: Plot of dynamic forecast model

plot of this model with forecasted predictors using ARIMA is provided in figure 29, where blue series is ARIMA forecast using ARIMA forecasted predictors and orange series is ARIMA forecast using naïve forecasted predictors.

In order to evaluate the plausibility of the ARIMA forecasting using ARIMA forecasted predictors we opt to include an ARIMA forecast including naïve forecasted predictors. The accuracy of the optimal forecasted predictors appears to have less successful accuracy with MASE of 0.33 compared to the naïve with MASE equals 0.23. Hence, in our case of dynamic forecasting there is no need to perform more advanced forecasting of the predictors in order to get a good dynamic forecasting model, one can simply take the last observed CPI and Export value in order to forecast the unemployment level. The residuals plots, which can be seen in appendix 14.4, have zero mean and there is no sign of autocorrelations, indicating white noise.

9.Forecasting using deterministic terms

Up until this point, we have constructed forecasting models with solely stochastic terms. Stochastic terms methods have in common that although they have an expected mean, the variance of the term may change across time series.

Accuracy Dynamic Forecast

Model	RMSE	MAE	MPE	MAPE	MASE	RMSSE
ARIMA forecast with NAIVE forecasted predictors	362.29	286.91	4.25	4.48	0.23	0.21
ARIMA forecast with ARIMA forecasted predictors	483.63	413.87	6.59	6.59	0.33	0.28

Figure 30: Performance metrics of dynamic forecasts

A deterministic term is predetermined, that is, any given output can be calculated from the models' equations. A property of a deterministic trend is that its variance will not change over time, whereas a stochastic trend may evolve over time in a stationary or non-stationary manner. The prediction interval for stochastic models takes the future uncertainty into account by allowing for a wider prediction interval for long term forecasting, which is not the case for deterministic terms (Hyndman & Athanasopoulos, 2021). For the remainder of this section, we will compare two forecasting methods with deterministic terms: a model with deterministic trend term with ARMA processed error terms and a model with Fourier terms and ARIMA processed error terms.

9.1 Deterministic trend forecast

In order to forecast a model with deterministic trend term one needs an intercept, a trend term and ARMA error by fixing the integrated part to zero. By minimizing AICc the optimal deterministic trend model appears to be a linear model with ARMA-error where AR is of second order, MA of first order and seasonal lagged AR of first order with lag of 12 months. The coefficients are provided in figure 31.

Deterministic trend

term	estimate
ar1	1.91
ar2	-0.92
ma1	-0.84
sar1	0.83
trend()	3.88
intercept	7904.72

Figure 31: Coefficients of ARIMA model with linear trend

The model does indeed follow some seasonal pattern, but does, on the other hand, deviate from the test set. The error term in the deterministic trend model is treated with an expectation of 0 and variance of 10316, NID(0, 103166). In addition, there is no sign of residual autocorrelation, which can be found in appendix 14,5, as such it resembles white noise.

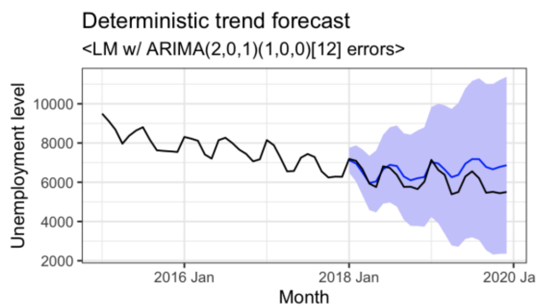


Figure 32: Plot of forecast of ARIMA model with linear trend

9.2 Fourier terms forecast

Forecasting with fourier terms is assuming fixed seasonality, which means that the patterns are not changing over time (Hyndman, 2010). This could imply some disadvantages if the seasons vary drastically from year to year, but as we have seen earlier, the seasonality seems to be similar/fixed for the unemployment level in the US. The error is handled the same way as an ARMA model. The variable K in this fourier model is simply the maximum order of fourier terms, sine and cosine pairs, where a small number for K returns a smoother seasonal pattern. We have modelled K from 1 to 6 and presented the forecast in the figure 33.

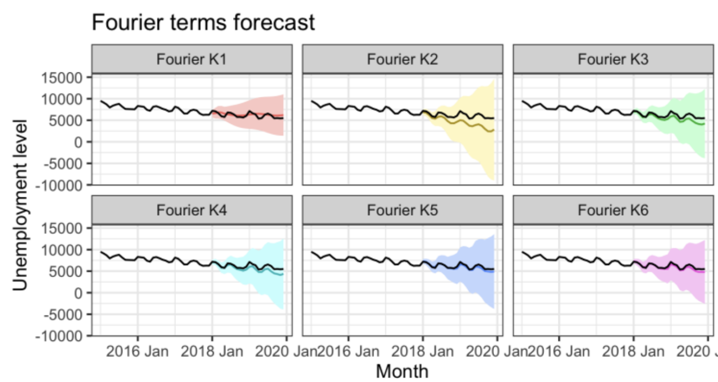


Figure 33: Facet plot of different ARIMA models with Fourier terms

Minimizing the Fourier models using AIC_c the best model on the training set is Fourier with maximum terms of 6, but measuring with MASE on the test set, we find that the best fitted forecast on the test set is Fourier with maximum terms of 5. This model, *LM w/ ARIMA(0,2,4) errors*, is a linear model with deterministically Fourier terms where the error is characterized with second order differencing and fourth order MA on the non-seasonal component. The error is treated as normally and independently distributed with mean zero and variance of 167363. The coefficients are shown in table 34. The residual plots for this model can be found in appendix 14.6.

Fourier terms

term	estimate
ma1	-1.01
ma2	0.34
ma3	-0.33
ma4	0.21
fourier(K = 5)C1_12	33.61
fourier(K = 5)S1_12	203.18
fourier(K = 5)C2_12	496.25
fourier(K = 5)S2_12	198.02
fourier(K = 5)C3_12	-58.89
fourier(K = 5)S3_12	207.57
fourier(K = 5)C4_12	57.72
fourier(K = 5)S4_12	-7.77
fourier(K = 5)C5_12	130.88
fourier(K = 5)S5_12	3.45

Figure 34: Coefficients of ARIMA(0,2,4) with K = 5 Fourier terms

The table provided in figure 35 indicates that a forecast with a deterministic Fourier trend with maximum terms of 5 has 0.22 (0.43 - **0.21**) lower **MASE**, which yields better accuracy compared to the model with deterministic trend term forecast.

Deterministic forecasting methods

Model	RMSE	MAE	MPE	MAPE	MASE	RMSSE
Fourier K5	337.321	260.478	3.694	4.353	0.209	0.193
Fourier K6	377.973	297.166	4.346	4.972	0.239	0.216
Fourier K1	561.586	479.863	-3.797	8.046	0.386	0.321
Deterministic trend	686.575	539.601	-8.569	9.314	0.434	0.392
Fourier K4	739.138	618.113	10.231	10.345	0.497	0.423
Fourier K3	835.187	709.525	11.784	11.863	0.571	0.477
Fourier K2	1795.227	1565.759	26.233	26.233	1.259	1.026

Figure 35: Performance metrics table of models using deterministic trends

10. Combinational forecast

The combinational forecasting method involves combining several forecasts on a predefined criterion and has been shown to increase the accuracy of forecasts according to Cleven (1989). This type of forecasting is said to increase the forecasting accuracy according to Clive Granger and John Bates research (1969). The most primitive combination method is by performing averaging, where the forecasts are equally weighted. This has also been stated by Robert Clemen (1989) that this, in many cases, improves the performance dramatically. We are going to perform the simple method and compare it to the MSE- and AIC-weighted combinational forecasting methods. For the simplicity in this case, the combination is conducted on three well performing models: the ARIMA optimal model (ARIMA(5,1,0)(0,1,1)12), ETS optimal model (A, Ad, A) and the ARIMA dynamic optimal model with naïve predicted predictors CPI and Export (ARIMA(1,1,4)(1,0,0)12).

The formula for mean squared error (MSE) weights' is provided in figure 36 where i is indicating a model, k equals 1 and the sum of weights is 1. R is the number of models. The AIC weight for

$$w_i = \frac{1/MSE_i^k}{\sum_{r=1}^{R} 1/MSE_r^k}, \quad where \quad \sum w_i = 1$$

Figure 36: MSE weighting formula (Lyhagen 2021)

each model i is expressed by taking the exponential of -0.5 multiplied by subtracting the minimum AIC of all the models by AIC for each model, then divided by the sum of these exponential expressions for all the models. The interpretation of these weights is the model i’s likelihood of being the best model. The sum of weights is one.

One can observe from the plot provided in figure 38 that both mean weighted and MSE weighted forecasts performance seemingly similar compared to the AIC weighted forecast. The table in figure 39 of combined forecast accuracy on the MASE key performance indicator shows that there is slightly little difference between the averaging method versus the more advanced MSE weighted, only 0.003 in difference.

$$w_i = \frac{exp(-0.5\Delta_i)}{\sum_{r=1}^{R} exp(-0.5\Delta_r)}, \quad where \quad \Delta_i = AIC_i - AIC_{min}, \quad \sum w_i = 1$$

Figure 37: AIC weighting formula (Lyhagen 2021)

In our findings, we can state that the averaging method is a simple and well performing combination method. The weakness of this experiment is that it is only compared to a few advanced methods, while there exist other methods that may perform even better or worse. For the purpose of this report, the averaging method is the best performing method with the clearly lowest **MASE of 0.093** compared to the formed decomposition forecast, ETS, ARIMA, dynamic and deterministic forecasting methods.

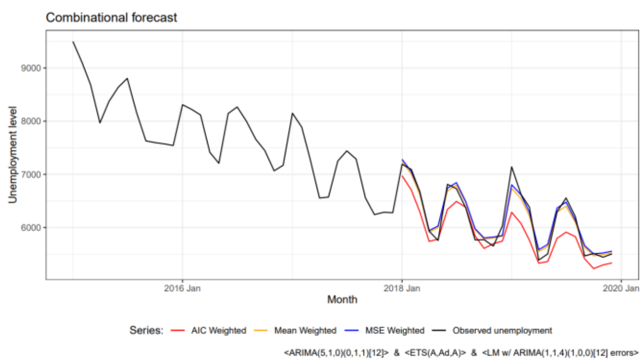


Figure 16: Combinational forecast plot

Combined forecast accuracy

Model	RMSE	MASE	MAE	MAPE	RMSSE
MSE weighted	141.387	0.090	112.068	1.363	0.081
Mean weighted	145.800	0.093	115.910	1.395	0.083
AIC weighted	362.294	0.231	286.909	3.359	0.207

Figure 39: Performance metrics table

11. Monte-Carlo simulation

A Monte Carlo simulation is a simulation process whose name is derived from gambling in the Monaco casinos of Monte Carlo. In each iteration we introduce randomness by drawing on a new series of generated data, also called a sample path. The optimal ARIMA model, as described in section 6, is used to generate a new series of data on which we will test an univariate ARIMA model and a multivariate VAR model. The arima.sim() function from the stats package, called upon in the generate_y function of our R program generates a new time series y_t. An example series generated by our function is provided in figure 40. Based on this series, we will create an additional new variable which will be added as an independent variable in our multivariate VAR model. The series x_t , is defined as:

$$x_t = 0.5x_{t-1} + 0.5y_{t-1} + \epsilon_t$$

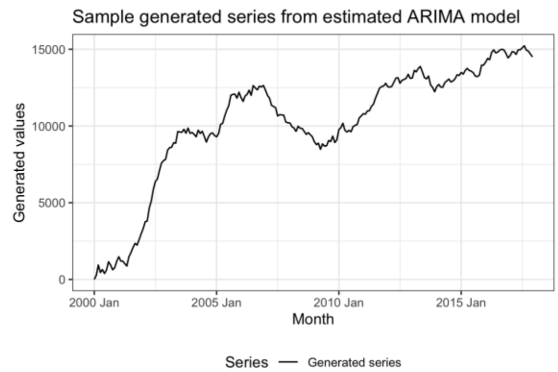


Figure 40: Example of series generated by generate_y

We assume that the error term follows a normal distribution, with a mean of zero and a standard deviation of each y series. Our goal is to determine the impact of this added series x, by comparing a univariate ARIMA model containing only y and a multivariate VAR model containing both y and x. We want to isolate the effect of the added independent variable x_t, so we make sure that our VAR and ARIMA model uses the same AR(1) terms. Our comparison is thus made between a differenced VAR(1) with two variables, and a differenced AR(1) with one variable.

By the law of large numbers, we should expect that if we run the simulation sufficiently many times the average of our evaluation metrics will converge to some theoretical mean. We will utilize this fact by computing several performance metrics and averaging them across a large number of simulation runs. The simulation is done 4 times at different sample sizes, each with a 1000 sample paths. In each sample path a training and test set is made in 80/20 proportions. The reader can consult the appendix to find the code which performs this task. In figure 41, the reader will find a set of forecast evaluation metrics comparing the multivariate VAR model against a univariate ARIMA model.

Monte Carlo simulations: 1000 sample paths

	RMSE	MASE	MAE	MAPE	RMSSE	Sample length
VAR multivariate	271.906	0.785	223.696	242.876	0.772	50
ARIMA yt	269.089	0.777	221.395	105.821	0.764	50
VAR multivariate1	274.914	0.773	224.119	303.056	0.761	100
ARIMA yt1	270.346	0.759	219.999	105.583	0.748	100
VAR multivariate2	272.682	0.754	220.994	204.851	0.744	150
ARIMA yt2	270.186	0.746	218.676	107.116	0.737	150
VAR multivariate3	275.997	0.768	223.015	189.720	0.760	200
ARIMA yt3	272.646	0.758	219.960	101.186	0.751	200

Figure 41: Results of 4 sample lengths, each with 1000 sample paths

We find that the addition of the independent variable x_t slightly hinders forecast accuracy as measured by our evaluation metrics. In seeking to improve forecasts, we might include another relevant variable that sheds information on the dependent variable we want to forecast. If this newly added variable contains causal information, it might improve our forecast. Our added x_t variable, is simply a product of previous values of y_t and a normally distributed stochastic error term. As such it does not add any new information, and we find this type of reasoning to fit well with our results.

12. Comparison of type-optimal forecasting methods

In this report we have forecasted unemployment level using a wide range of forecasting techniques. We define type optimal as the best model we test in this report for each type of model, for instance, only the best of our tested dynamic regression models is included. As the reader can see in figure 42 a combination of forecasts using the MSE weighted method appears to be the best.

Comparison of all optimal models discussed

Model	RMSE	MASE	MAE	MAPE	RMSSE
MSE weighted	141.387	0.090	112.068	1.363	0.081
ARIMA_optimal	175.727	0.117	145.726	2.392	0.100
ETS decompositon	179.791	0.125	155.516	2.604	0.103
ETS_optimal	195.408	0.138	171.075	2.886	0.112
Fourier K5	337.175	0.209	260.318	4.350	0.193
ARIMA_dynamic	362.294	0.231	286.909	4.478	0.207
Multivariate VECM model VAR(10)	498.568	0.374	465.346	7.576	0.285

Figure 42: Performance metrics of type optimal forecasting methods

It is noteworthy that a simple combination of well-performing methods may give an overall improvement in performance. Especially considering that the simple mean of the methods still managed to perform better than each of the isolated forecasting methods which it is composed of. It is interesting to note that our multivariate models perform in average significantly worse than our optimal univariate models. We note that the cointegrated VECM model performs the worst of all optimal models. As it is a multivariate model that depends on the cointegration levels between the variables, it would be interesting to change its variables to note its effect. A multivariate dynamic regression model performs better, but still worse than the best of the univariate models. This may suggest that the addition of independent variables in a forecasting model should be done with caution.

Due to the relatively constant seasonality in the unemployment level, we did find that the deterministic term forecast, especially Fourier, does perform well as the seasonality seems to be harmonic, although the seasonal length is as short as 12 months. Furthermore, we also find that the averaged weighted combination forecast is a simple method that performs very well, compared to other more advanced weighting methods, and performs better than the single optimal forecasting models standing alone. Regardless of which method is used, the overall forecast output performs well in terms of our chosen performance metrics. We should note that we have statically divided the training and test set, and chosen our best model based entirely on a single train set. The weakness of this approach is that it may give a false sense of security, especially considering the cyclical nature of unemployment. If our test set represents a high growth economic environment with a low and stable unemployment level, our forecast may underestimate during a following recession marked by high and unstable levels of unemployment.

13. Conclusion

As we have seen, the unemployment level in the US has a relatively constant seasonal variation. Our use of the X11 decomposition method gives a close resemblance of the decomposition made by the U.S Labor Statistics Bureau, and we have shown that a forecast made by the seasonal and seasonally adjusted combined added together yields a better forecast than a forecast made on the time series as a whole. We compare and construct optimal ETS and ARIMA models, since these models are designed for a short seasonal period such as 12 months, based on the fable package. Then use the optimal ARIMA model as the starting point of a multivariate dynamic regression model. Our two multivariate models, the dynamic ARIMA model with naïve forecasted predictors and the VECM model, does not outperform the best univariate models. The forecasting models using deterministic trends perform well overall, although the stability of the models may suffer in the long term. As is consistent with findings in the forecasting literature, a combination of forecasts improves accuracy, and it is through a MSE weighted combinational forecasting we achieve our highest performing model.

We performed a Monte Carlo simulation of a univariate ARIMA model and a multivariate VAR model and find that the inclusion of a new irrelevant variable slightly worsens forecast performance. Our impression, after this report, is that a multivariate model should be considered carefully, and additional independent variables should only be added if they contribute significant information to the model.

14. Literature

Bates, J. M., & Granger, C. W. J. (n.d.). The combination of forecasts. Operational Research Quarterly, 20(4), 451–468. *Journal of the Operational Research Society*, *20*. https://doi.org/10.1057/jors.1969.103

Cleveland, R. B., Cleveland, W. S., McRae, J. E., & Terpenning, I. (1990). STL: A Seasonal-Trend Decomposition Procedure Based on Loess. *Journal of Official Statistics*, *6*. http://bit.ly/stl1990

Gardner, E. S., & McKenzie, ED. (1985). Forecasting trends in the time series. *Management Science*, *31*. https://www.jstor.org/stable/pdf/2631713.pdf?refreqid=excelsior%3Af21ae290f5b0266a9b81f01675adc908

Granger, C. W. J., & Engle, R. F. (1987). Co-Integration and Error Correction: Representation, Estimation, and Testing. *Econometrica*, *55*. https://doi.org/10.2307/1913236

Hyndman, R. J. (2010, September 29). *Forecasting with long seasonal periods*. https://robjhyndman.com/hyndsight/longseasonality/

Hyndman, R. J., & Athanasopoulos, G. (2018). *Forecasting: Principles and Practice* (2nd edition). https://otexts.com/fpp2/

Hyndman, R. J., & Athanasopoulos, G. (2021). *Forecasting: Principles and Practice* (3rd edition). https://otexts.com/fpp3/

Hyndman, R. J., & Koelher, A. B. (2005). *Another look at measures of forecast accuracy*. https://robjhyndman.com/papers/mase.pdf

Kuha, J. (2004). AIC and BIC Comparisons of Assumptions and Performance. *Sociological Methods & Research*, *33*. https://doi.org/10.1177/0049124103262065

Kwiatkowski, D., Phillips, P. C. B., Schmidt, P., & Shin, Y. (1991). Testing the null hypothesis of stationarity against the alternative of a unit root. *Journal of Econometrics*, *54*. https://debis.deu.edu.tr/userweb/onder.hanedar/dosyalar/kpss.pdf

Lütkepohl, H. (2005). *New Introduction to Multiple Time Series Analysis*. Springer.

Lyhagen, Johan. (n.d.). *Forecast combination*. https://nhh.instructure.com/courses/1287/files/folder/Lecture%20Notes?preview=211785

Philips, A. W. (1958). The Relation Between Unemployment and the Rate of Change of Money Wage Rates in the United Kingdom, 1861–1957. *Wiley*, *25*. https://doi.org/10.1111/j.1468-0335.1958.tb00003.x

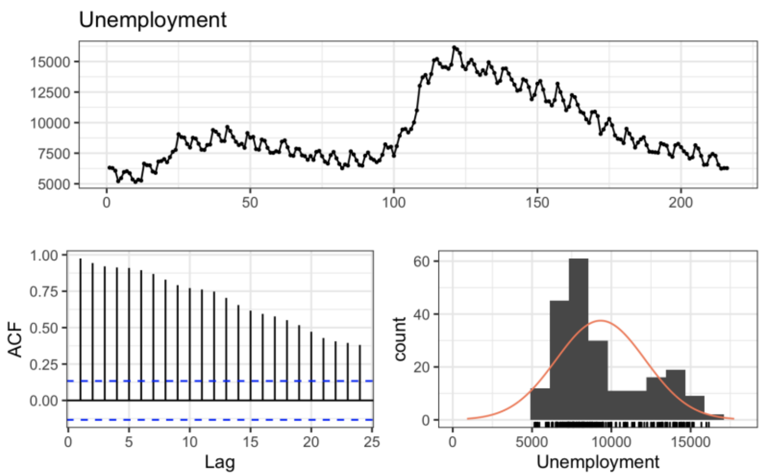
Roberts, S. A. (1982). A General Class of Holt-Winters Type Forecasting Models. *Management Science*, *28*. https://www.jstor.org/stable/2631264

Tiller, R. B., & Evans, T. D. (2018, December 13). *Seasonal Adjustment Methodology for National Labor Force Statistics from the Current Population Survey (CPS)*.

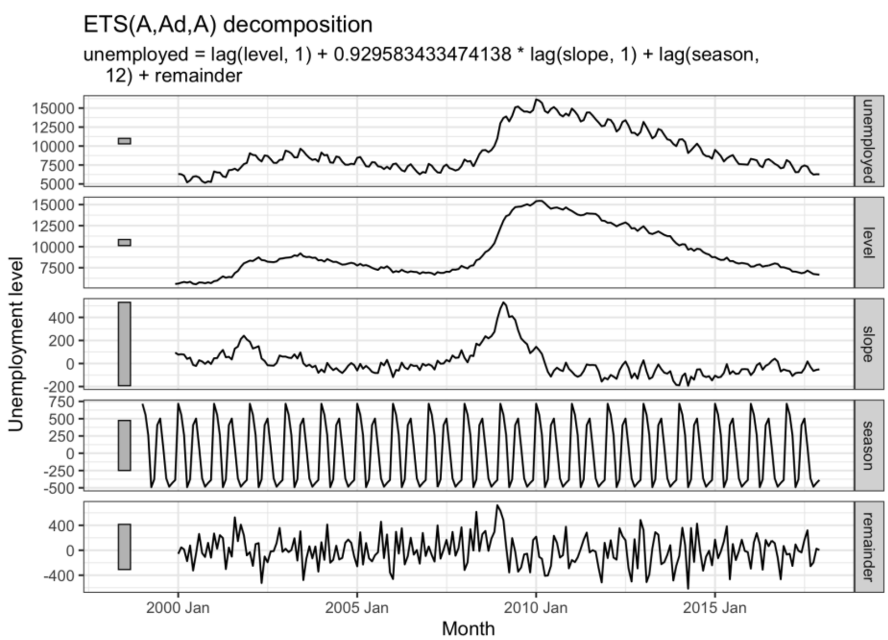
Vector Autoregressive models VAR(p) models. (n.d.). Pennsylvania State University. https://online.stat.psu.edu/stat510/lesson/11/11.2

14. Appendix

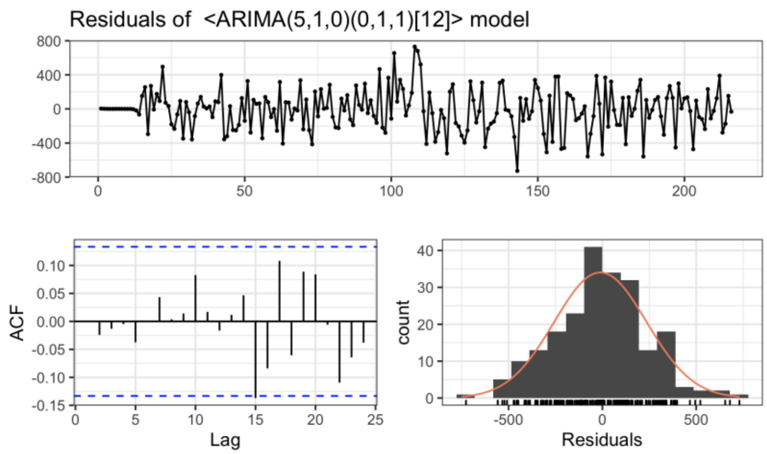
14.1



14.2

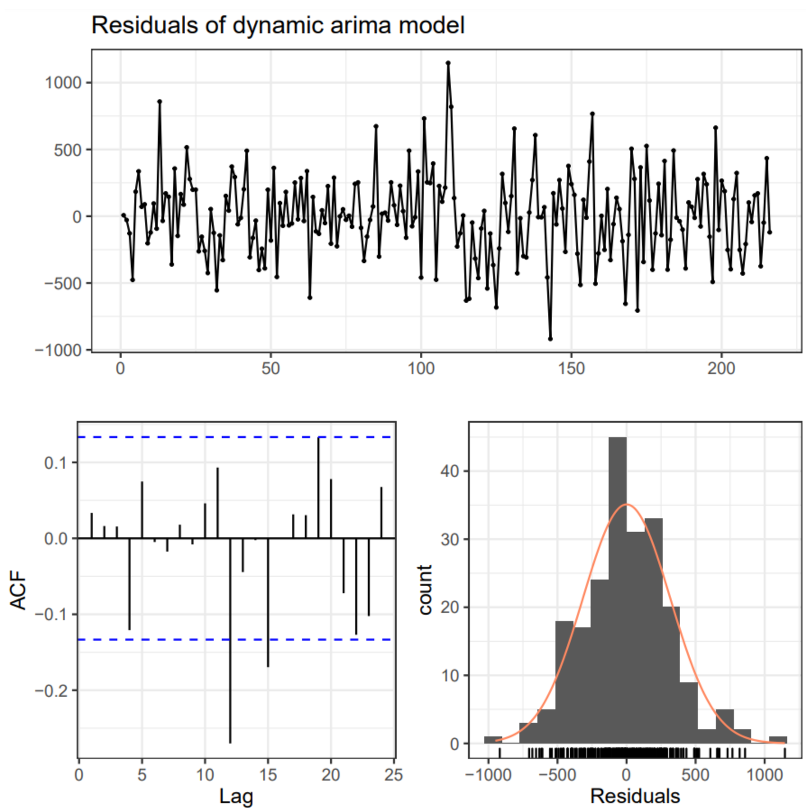


14.3



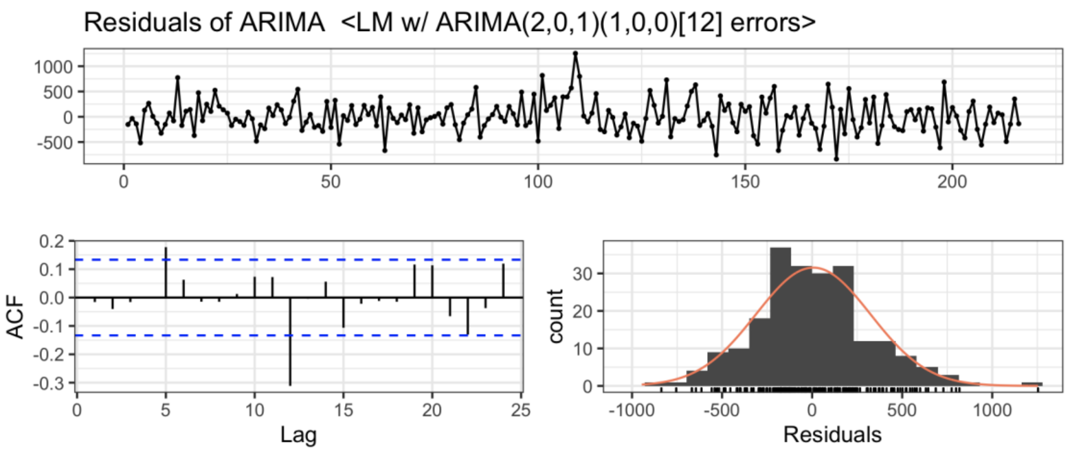
Residuals of <ARIMA(5,1,0)(0,1,1)[12]> model
ACF
Lag
count
Residuals

14.4

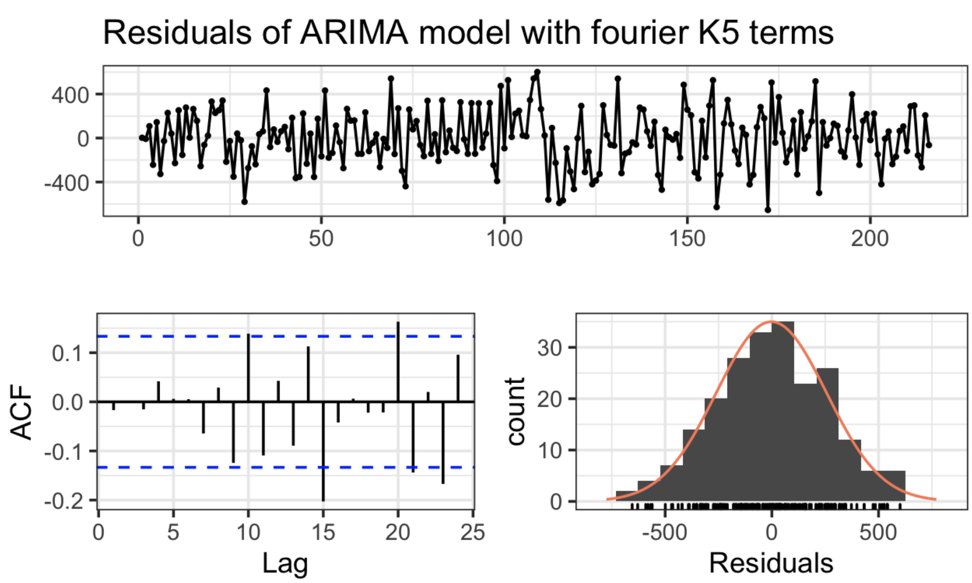


Residuals of dynamic arima model
ACF
Lag
count
Residuals

14.5


Residuals of ARIMA <LM w/ ARIMA(2,0,1)(1,0,0)[12] errors>
ACF
Lag
count
Residuals

14.6


Residuals of ARIMA model with fourier K5 terms
ACF
Lag
count
Residuals

14.7

$$y_t = l_{t-1} + \phi b_{t-1} + s_{t-m} + \epsilon_t$$
$$l_t = l_{t-1} + \phi b_{t-1} + \alpha \epsilon_t$$
$$b_t = \phi b_{t-1} + \beta \epsilon_t$$
$$s_t = s_{t-m} + \gamma \epsilon_t$$

14.8

R- code: ARIMA data simulation and comparison of multivariate and univariate models. The remainder of the R- code can be found in the included .r file.

```
# Generating y_t
generate_y <- function(fit, n) {
  #' Function that passes the standard deviation of the residuals of our optim-
  #' al ARIMA model
  #' Automatically finds and passes ar and ma terms to the arima.sim (stats pa-
  #' ckage), and returns the generated series
  sigma <- sd(residuals(fit)$.resid)
  ar_terms <- (fit %>%
                 coefficients %>%
                 filter(str_detect(term,"ar")))$estimate %>%
    c(.) # AR terms and their coefficients
  sma_terms <- (fit %>% coefficients
                %>% filter(str_detect(term, "sma")))$estimate %>%
    c(.)
  arima_sim_model <- list(order = c(5, 1, 0),
                          ar = ar_terms,
                          sma = sma_terms)
  y <- arima.sim(n = n,
                 arima_sim_model,
                 sd = sigma)
  return(y)
}



# Simulation function
simulate <- function(fit, R, train_length , h ) {
    #' Function that generates a new series x based on an arima simulation retu-
    #' rned by generate_y.
    #' Compares two models, and populates which contains a series of forecast e-
```

```r
#' valuation metrics.
#' Returns the populated matrix.
res <- matrix(0,2,5)
colnames(res) <- c("RMSE", "MASE", "MAE", "MAPE",  "RMSSE")
rownames(res) <- c("VAR multivariate", "ARIMA yt")
for(i in 1:R){
    y <- diff(generate_y(fit, train_length+h))
    y_e <- y[1:train_length]
    y_t <- y[(train_length+1):(train_length+h)]
    x <- c()
    x[1] <- y[1]
    for (j in 2:(train_length+h)) {
        x[j] <- 0.5*y[j-1] + 0.5*x[j-1] + rnorm(n = 1,
                                                mean = 0,
                                                sd = sd(y))
    }
    x_e <- x[1:train_length]
    x_t <- x[(train_length+1):(train_length+h)]
    data_x_y = data.frame(date = (1:train_length),
                          x_e = x_e,
                          y_e = y_e)  %>%
      as_tsibble(index = date)
    var_multi  <- vars:: VAR(data_x_y[,2:3],
                             p  = 1,
                             type = "const")                     # VAR(1) model
    arima_uni <- data_x_y  %>%
      model(Arima = ARIMA(y_e ~ 0 + pdq(1,0,0) + PDQ(0,0,0)))    # ARIMA pdq(1,0,0) model
    var_resids <-   y_t -  predict(var_multi, n.ahead = h)$fcst$y_e[,1]
    arima_resids <- y_t -  (arima_uni %>% forecast(h = h))$.mean

    res[1,1] <- res[1,1] + RMSE(var_resids)/R
    res[2,1] <- res[2,1] + RMSE(arima_resids)/R

    res[1,2] <- res[1,2] + MASE(.resid = var_resids,
                                .train = y_e,
                                .period = 12)/R
    res[2,2] <- res[2,2] + MASE(.resid = arima_resids,
                                .train = y_e,
                                .period = 12)/R

    res[1,3] <- res[1,3] + MAE(.resid = var_resids)/R
```

```
        res[2,3] <- res[2,3] + MAE(.resid = arima_resids)/R

        res[1,4] <- res[1,4] + fabletools::MAPE(.resid = var_resids,
                                                 .actual = y_t,
                                                 .period = 12)/R
        res[2,4] <- res[2,4] + fabletools::MAPE(.resid = arima_resids,
                                                 .actual = y_t,
                                                 .period = 12)/R


        res[1,5] <- res[1,5] + RMSSE(.resid = var_resids,
                                      .train = y_e,
                                      .period = 12)/R
        res[2,5] <- res[2,5] + RMSSE(.resid = arima_resids,
                                      .train = y_e,
                                      .period = 12)/R
    }
    return(res)
}


# Wrapper function
wrapperSim <- function(R, sample_size, test_ratio) {
        #' Wrapper function that splits the unemployment series into
        #' test and training lengths based on an input sample length and
        #' test ratio of the overall series length.
        #' Passes this as parameters to the simulate function
        cl <- parallel::makeCluster(parallel::detectCores())        ### Make clusters
        doParallel::registerDoParallel(cl)
        train_length <- floor(sample_size * (1 - test_ratio))
        h <- ceiling(sample_size * test_ratio)
        print(paste("Training length: ", train_length))
        print(paste("h : ", h))
        start <- (nrow(unemployment_train_ts) - sample_size)
        sim_res <- simulate(fit_arima_optimal, R, train_length, h) %>%
          as.data.frame() %>%
          mutate("Sample length" = sample_size)
        parallel::stopCluster(cl)

        return(sim_res) }
```