All Final Models

Group4

July 22, 2016

### Data Prep

##### Seasonally adjusted unemployment

unem = econ.sa$unem\_rate\_sa

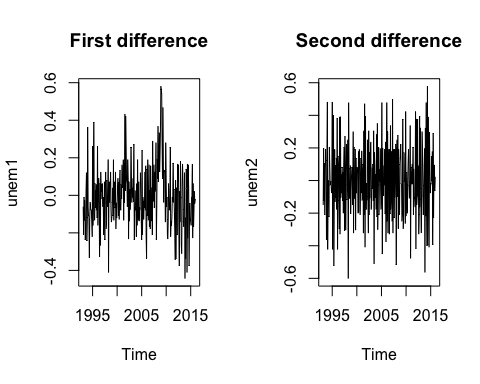
##### Differencing

Everyone seems to agree that 2 differences gets us stationarity... since we specify differencing in the arima model parameters, it feels like we should not be differencing the data beforehand.

unem1 = diff(unem)  
unem2 = diff(unem, differences = 2)

##### Differencing plots

par(mfrow = c(1, 2))  
plot.ts(unem1, main = "First difference")  
plot.ts(unem2, main = "Second difference")



##### ADF of differenced data

adf.test(unem1)

##   
## Augmented Dickey-Fuller Test  
##   
## data: unem1  
## Dickey-Fuller = -3.618, Lag order = 6, p-value = 0.03183  
## alternative hypothesis: stationary

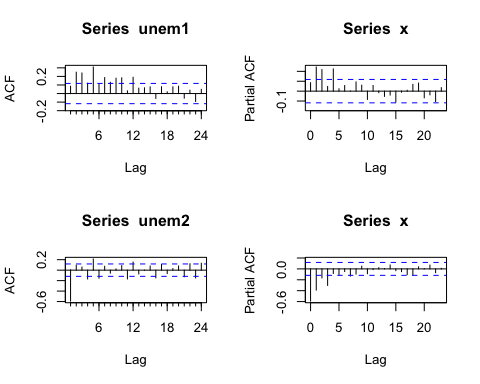
adf.test(unem2)

## Warning in adf.test(unem2): p-value smaller than printed p-value

##   
## Augmented Dickey-Fuller Test  
##   
## data: unem2  
## Dickey-Fuller = -9.3595, Lag order = 6, p-value = 0.01  
## alternative hypothesis: stationary

##### ACF/PACF Plots

par(mfrow = c(2,2))  
Acf(unem1); Pacf(unem1)  
Acf(unem2); Pacf(unem2)



#### ACF/PACF Model Suggestions

##### First Difference

##### -----------------------------

##### Suggested: MA(4)

##### 

##### Second Difference

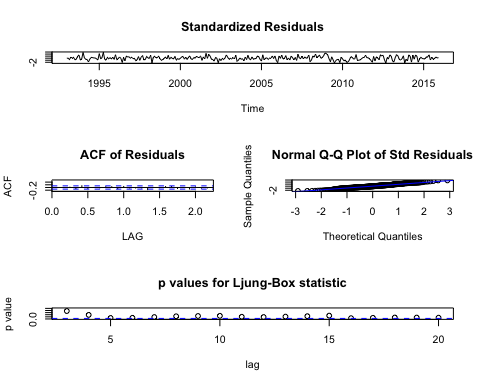
##### -----------------------------

##### Suggested: ARMA(1,3)

##### Putting these back in for the later code

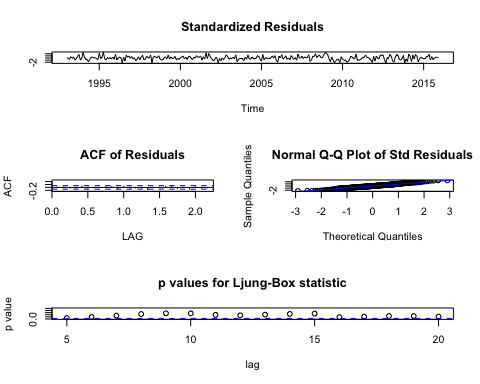
## Models with no regressors  
mdl.1 = sarima(econ.sa$unem\_rate\_sa, p = 1, d = 2, q = 1)

## initial value -1.454782   
## iter 2 value -1.756280  
## iter 3 value -1.798959  
## iter 4 value -1.806922  
## iter 5 value -1.812862  
## iter 6 value -1.814496  
## iter 7 value -1.815632  
## iter 8 value -1.815971  
## iter 9 value -1.815979  
## iter 10 value -1.815979  
## iter 10 value -1.815979  
## iter 10 value -1.815979  
## final value -1.815979   
## converged  
## initial value -1.817240   
## iter 2 value -1.817260  
## iter 3 value -1.817288  
## iter 4 value -1.817288  
## iter 4 value -1.817288  
## iter 4 value -1.817288  
## final value -1.817288   
## converged



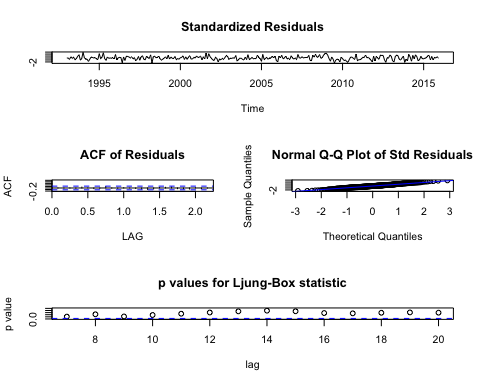
mdl.2 = sarima(econ.sa$unem\_rate\_sa, p = 2, d = 2, q = 2)

## initial value -1.454288   
## iter 2 value -1.692359  
## iter 3 value -1.740417  
## iter 4 value -1.763513  
## iter 5 value -1.795788  
## iter 6 value -1.806460  
## iter 7 value -1.816825  
## iter 8 value -1.817395  
## iter 9 value -1.817657  
## iter 10 value -1.817681  
## iter 11 value -1.817685  
## iter 12 value -1.817701  
## iter 13 value -1.817732  
## iter 14 value -1.817872  
## iter 15 value -1.818549  
## iter 16 value -1.818972  
## iter 17 value -1.818990  
## iter 18 value -1.819126  
## iter 19 value -1.819166  
## iter 20 value -1.819842  
## iter 21 value -1.820786  
## iter 22 value -1.821533  
## iter 23 value -1.822079  
## iter 24 value -1.822345  
## iter 25 value -1.822605  
## iter 26 value -1.823566  
## iter 27 value -1.823792  
## iter 28 value -1.824392  
## iter 29 value -1.824601  
## iter 30 value -1.824638  
## iter 31 value -1.824640  
## iter 32 value -1.824641  
## iter 33 value -1.824642  
## iter 34 value -1.824642  
## iter 35 value -1.824643  
## iter 36 value -1.824643  
## iter 36 value -1.824643  
## iter 36 value -1.824643  
## final value -1.824643   
## converged  
## initial value -1.823660   
## iter 2 value -1.823662  
## iter 3 value -1.823679  
## iter 4 value -1.823680  
## iter 5 value -1.823686  
## iter 6 value -1.823697  
## iter 7 value -1.823699  
## iter 8 value -1.823700  
## iter 9 value -1.823700  
## iter 9 value -1.823700  
## iter 9 value -1.823700  
## final value -1.823700   
## converged



mdl.3 = sarima(econ.sa$unem\_rate\_sa, p = 3, d = 2, q = 3)

## initial value -1.452952   
## iter 2 value -1.673071  
## iter 3 value -1.731027  
## iter 4 value -1.775737  
## iter 5 value -1.807436  
## iter 6 value -1.814339  
## iter 7 value -1.816429  
## iter 8 value -1.817702  
## iter 9 value -1.817895  
## iter 10 value -1.818005  
## iter 11 value -1.818471  
## iter 12 value -1.819672  
## iter 13 value -1.821083  
## iter 14 value -1.821651  
## iter 15 value -1.823209  
## iter 16 value -1.823460  
## iter 17 value -1.823904  
## iter 18 value -1.824379  
## iter 19 value -1.824420  
## iter 20 value -1.824425  
## iter 21 value -1.824441  
## iter 22 value -1.824456  
## iter 23 value -1.824476  
## iter 24 value -1.824491  
## iter 25 value -1.824510  
## iter 26 value -1.824547  
## iter 27 value -1.824659  
## iter 28 value -1.824717  
## iter 29 value -1.824802  
## iter 30 value -1.824859  
## iter 31 value -1.824877  
## iter 32 value -1.824905  
## iter 33 value -1.824940  
## iter 34 value -1.824978  
## iter 35 value -1.825007  
## iter 36 value -1.825040  
## iter 37 value -1.825105  
## iter 38 value -1.825295  
## iter 39 value -1.825892  
## iter 40 value -1.826395  
## iter 41 value -1.826630  
## iter 42 value -1.827159  
## iter 43 value -1.827653  
## iter 44 value -1.828137  
## iter 45 value -1.828359  
## iter 46 value -1.828421  
## iter 47 value -1.828440  
## iter 48 value -1.828472  
## iter 49 value -1.828517  
## iter 50 value -1.828589  
## iter 51 value -1.828750  
## iter 52 value -1.829130  
## iter 53 value -1.829338  
## iter 54 value -1.829424  
## iter 55 value -1.829477  
## iter 56 value -1.829893  
## iter 57 value -1.830005  
## iter 58 value -1.830014  
## iter 59 value -1.830056  
## iter 60 value -1.830131  
## iter 61 value -1.830274  
## iter 62 value -1.830453  
## iter 63 value -1.830591  
## iter 64 value -1.830622  
## iter 65 value -1.830625  
## iter 66 value -1.830625  
## iter 66 value -1.830625  
## final value -1.830625   
## converged  
## initial value -1.834989   
## iter 2 value -1.835059  
## iter 3 value -1.835125  
## iter 4 value -1.835203  
## iter 5 value -1.835349  
## iter 6 value -1.835485  
## iter 7 value -1.835906  
## iter 8 value -1.836444  
## iter 9 value -1.837146  
## iter 10 value -1.837363  
## iter 11 value -1.837472  
## iter 12 value -1.837598  
## iter 13 value -1.837618  
## iter 14 value -1.837657  
## iter 15 value -1.837659  
## iter 16 value -1.837666  
## iter 17 value -1.837667  
## iter 18 value -1.837669  
## iter 19 value -1.837669  
## iter 20 value -1.837670  
## iter 21 value -1.837671  
## iter 22 value -1.837672  
## iter 23 value -1.837675  
## iter 24 value -1.837680  
## iter 25 value -1.837687  
## iter 26 value -1.837691  
## iter 27 value -1.837693  
## iter 27 value -1.837693  
## final value -1.837693   
## converged

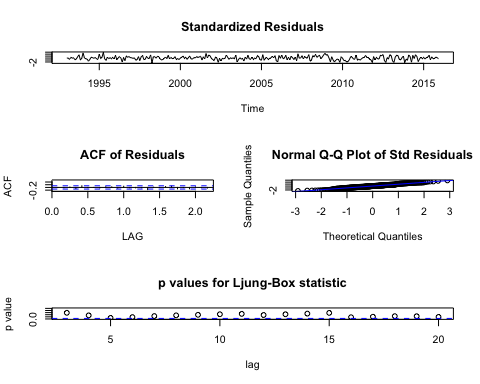


econ.sa.lag = ts.intersect(  
 unem\_rate\_sa = econ.sa$unem\_rate\_sa,  
 industrial\_production\_sa = lag(econ.sa$industrial\_production\_sa, -2),  
 manufacturers\_new\_orders\_sa = lag(econ.sa$manufacturers\_new\_orders\_sa, -4),  
 house\_price\_sa = lag(econ.sa$house\_price\_sa, -5),  
 construction\_spend\_sa = lag(econ.sa$construction\_spend\_sa, -3),  
 retail\_sales\_sa = lag(econ.sa$retail\_sales\_sa, -0),  
 dframe = TRUE  
)

##### Models with Regressors

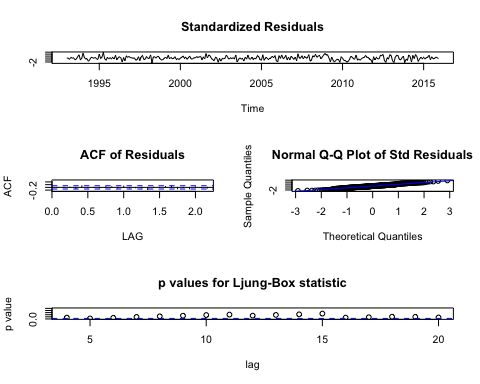
mdl.4 = sarima(econ.sa$unem\_rate\_sa, p = 1, d = 2, q = 1, xreg = econ.sa[, 2:6])

## initial value -1.483456   
## iter 2 value -1.750634  
## iter 3 value -1.755585  
## iter 4 value -1.767723  
## iter 5 value -1.811875  
## iter 6 value -1.830246  
## iter 7 value -1.831185  
## iter 8 value -1.833400  
## iter 9 value -1.833813  
## iter 10 value -1.833991  
## iter 11 value -1.834071  
## iter 12 value -1.834075  
## iter 13 value -1.834081  
## iter 13 value -1.834081  
## iter 13 value -1.834081  
## final value -1.834081   
## converged  
## initial value -1.834132   
## iter 2 value -1.834149  
## iter 3 value -1.834158  
## iter 4 value -1.834182  
## iter 5 value -1.834182  
## iter 6 value -1.834186  
## iter 7 value -1.834187  
## iter 8 value -1.834187  
## iter 8 value -1.834187  
## iter 8 value -1.834187  
## final value -1.834187   
## converged



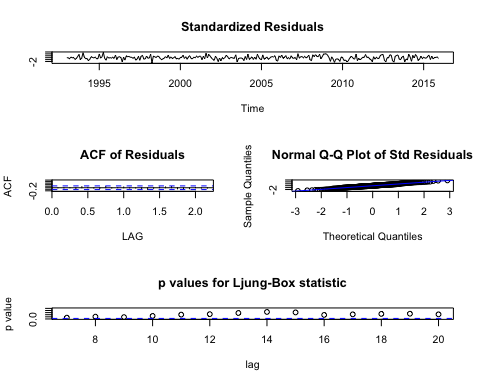
mdl.5 = sarima(econ.sa$unem\_rate\_sa, p = 1, d = 2, q = 2, xreg = econ.sa[, 2:6])

## initial value -1.483456   
## iter 2 value -1.729433  
## iter 3 value -1.739824  
## iter 4 value -1.766036  
## iter 5 value -1.788187  
## iter 6 value -1.811159  
## iter 7 value -1.829278  
## iter 8 value -1.831162  
## iter 9 value -1.832386  
## iter 10 value -1.832834  
## iter 11 value -1.833075  
## iter 12 value -1.833194  
## iter 13 value -1.833290  
## iter 14 value -1.833513  
## iter 15 value -1.833714  
## iter 16 value -1.833895  
## iter 17 value -1.834013  
## iter 18 value -1.834094  
## iter 19 value -1.834096  
## iter 20 value -1.834100  
## iter 21 value -1.834103  
## iter 22 value -1.834123  
## iter 23 value -1.834132  
## iter 24 value -1.834133  
## iter 25 value -1.834133  
## iter 26 value -1.834133  
## iter 27 value -1.834134  
## iter 28 value -1.834134  
## iter 29 value -1.834134  
## iter 30 value -1.834135  
## iter 31 value -1.834135  
## iter 31 value -1.834135  
## iter 31 value -1.834135  
## final value -1.834135   
## converged  
## initial value -1.834354   
## iter 2 value -1.834375  
## iter 3 value -1.834388  
## iter 4 value -1.834391  
## iter 5 value -1.834406  
## iter 6 value -1.834428  
## iter 7 value -1.834438  
## iter 8 value -1.834489  
## iter 9 value -1.834530  
## iter 10 value -1.834572  
## iter 11 value -1.834615  
## iter 12 value -1.834662  
## iter 13 value -1.834688  
## iter 14 value -1.834694  
## iter 15 value -1.834694  
## iter 16 value -1.834694  
## iter 16 value -1.834694  
## iter 16 value -1.834694  
## final value -1.834694   
## converged



mdl.6 = sarima(econ.sa$unem\_rate\_sa, p = 3, d = 2, q = 3, xreg = econ.sa[, 2:6])

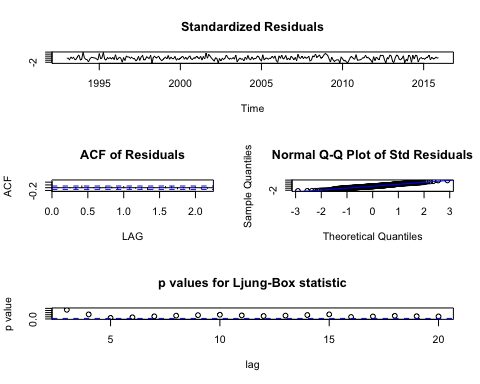
## initial value -1.480210   
## iter 2 value -1.636252  
## iter 3 value -1.684179  
## iter 4 value -1.737597  
## iter 5 value -1.756951  
## iter 6 value -1.809638  
## iter 7 value -1.818770  
## iter 8 value -1.833127  
## iter 9 value -1.833610  
## iter 10 value -1.834468  
## iter 11 value -1.834788  
## iter 12 value -1.835106  
## iter 13 value -1.835215  
## iter 14 value -1.836182  
## iter 15 value -1.836707  
## iter 16 value -1.837723  
## iter 17 value -1.838889  
## iter 18 value -1.839441  
## iter 19 value -1.839696  
## iter 20 value -1.839844  
## iter 21 value -1.839953  
## iter 22 value -1.839965  
## iter 23 value -1.839967  
## iter 24 value -1.839970  
## iter 25 value -1.839970  
## iter 26 value -1.839971  
## iter 27 value -1.839971  
## iter 28 value -1.839974  
## iter 29 value -1.839977  
## iter 30 value -1.840000  
## iter 31 value -1.840104  
## iter 32 value -1.840623  
## iter 33 value -1.840782  
## iter 34 value -1.841228  
## iter 35 value -1.841813  
## iter 36 value -1.842154  
## iter 37 value -1.842471  
## iter 38 value -1.842511  
## iter 39 value -1.842637  
## iter 40 value -1.843125  
## iter 41 value -1.843142  
## iter 42 value -1.843231  
## iter 43 value -1.843290  
## iter 44 value -1.843354  
## iter 45 value -1.843361  
## iter 46 value -1.843440  
## iter 47 value -1.843497  
## iter 48 value -1.843526  
## iter 49 value -1.843557  
## iter 50 value -1.843579  
## iter 51 value -1.843621  
## iter 52 value -1.843670  
## iter 53 value -1.843726  
## iter 54 value -1.843757  
## iter 55 value -1.843771  
## iter 56 value -1.843777  
## iter 57 value -1.843779  
## iter 58 value -1.843779  
## iter 58 value -1.843779  
## iter 58 value -1.843779  
## final value -1.843779   
## converged  
## initial value -1.849721   
## iter 2 value -1.849793  
## iter 3 value -1.849871  
## iter 4 value -1.849985  
## iter 5 value -1.849998  
## iter 6 value -1.850103  
## iter 7 value -1.850204  
## iter 8 value -1.850530  
## iter 9 value -1.850714  
## iter 10 value -1.851101  
## iter 11 value -1.851634  
## iter 12 value -1.852044  
## iter 13 value -1.852541  
## iter 14 value -1.852731  
## iter 15 value -1.852967  
## iter 16 value -1.854178  
## iter 17 value -1.854289  
## iter 18 value -1.854430  
## iter 19 value -1.854775  
## iter 20 value -1.854963  
## iter 21 value -1.855033  
## iter 22 value -1.855171  
## iter 23 value -1.855234  
## iter 24 value -1.855248  
## iter 25 value -1.855248  
## iter 26 value -1.855248  
## iter 27 value -1.855248  
## iter 28 value -1.855248  
## iter 29 value -1.855249  
## iter 30 value -1.855249  
## iter 31 value -1.855249  
## iter 32 value -1.855249  
## iter 32 value -1.855249  
## iter 32 value -1.855249  
## final value -1.855249   
## converged



##### Models with lagged Regressors

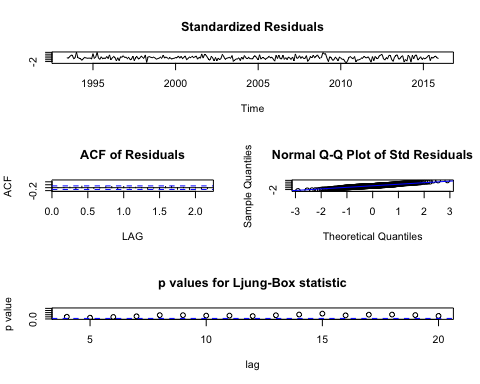
mdl.7 = sarima(econ.sa$unem\_rate\_sa, p = 1, d = 2, q = 1, xreg = econ[, 2:7])

## initial value -1.464512   
## iter 2 value -1.701745  
## iter 3 value -1.728679  
## iter 4 value -1.731572  
## iter 5 value -1.739068  
## iter 6 value -1.774854  
## iter 7 value -1.821381  
## iter 8 value -1.822716  
## iter 9 value -1.822830  
## iter 10 value -1.822919  
## iter 11 value -1.822935  
## iter 12 value -1.822948  
## iter 13 value -1.822950  
## iter 14 value -1.822951  
## iter 15 value -1.822951  
## iter 15 value -1.822951  
## iter 15 value -1.822951  
## final value -1.822951   
## converged  
## initial value -1.824116   
## iter 2 value -1.824125  
## iter 3 value -1.824127  
## iter 4 value -1.824143  
## iter 5 value -1.824162  
## iter 6 value -1.824165  
## iter 7 value -1.824180  
## iter 8 value -1.824181  
## iter 9 value -1.824182  
## iter 9 value -1.824182  
## iter 9 value -1.824182  
## final value -1.824182   
## converged



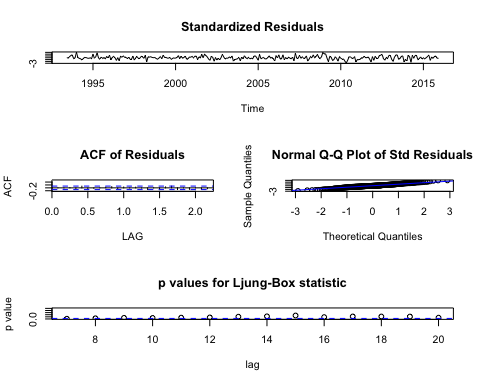
mdl.8 = sarima(econ.sa.lag$unem\_rate\_sa, p = 1, d = 2, q = 2, xreg = econ.sa.lag[, 2:6])

## initial value -1.477884   
## iter 2 value -1.718479  
## iter 3 value -1.733107  
## iter 4 value -1.771112  
## iter 5 value -1.804662  
## iter 6 value -1.841321  
## iter 7 value -1.847109  
## iter 8 value -1.849829  
## iter 9 value -1.854107  
## iter 10 value -1.856502  
## iter 11 value -1.856929  
## iter 12 value -1.857656  
## iter 13 value -1.858419  
## iter 14 value -1.860687  
## iter 15 value -1.863689  
## iter 16 value -1.864318  
## iter 17 value -1.864492  
## iter 18 value -1.864651  
## iter 19 value -1.864654  
## iter 20 value -1.864656  
## iter 21 value -1.864659  
## iter 22 value -1.864663  
## iter 23 value -1.864663  
## iter 24 value -1.864665  
## iter 25 value -1.864667  
## iter 26 value -1.864671  
## iter 27 value -1.864675  
## iter 28 value -1.864678  
## iter 29 value -1.864678  
## iter 30 value -1.864678  
## iter 30 value -1.864679  
## iter 30 value -1.864679  
## final value -1.864679   
## converged  
## initial value -1.858401   
## iter 2 value -1.858882  
## iter 3 value -1.858913  
## iter 4 value -1.859069  
## iter 5 value -1.859074  
## iter 6 value -1.859244  
## iter 7 value -1.859486  
## iter 8 value -1.860030  
## iter 9 value -1.860834  
## iter 10 value -1.861624  
## iter 11 value -1.862374  
## iter 12 value -1.862574  
## iter 13 value -1.862607  
## iter 14 value -1.862614  
## iter 15 value -1.862617  
## iter 16 value -1.862619  
## iter 17 value -1.862619  
## iter 17 value -1.862619  
## iter 17 value -1.862619  
## final value -1.862619   
## converged



mdl.9 = sarima(econ.sa.lag$unem\_rate\_sa, p = 3, d = 2, q = 3, xreg = econ.sa.lag[, 2:6])

## initial value -1.475425   
## iter 2 value -1.691914  
## iter 3 value -1.701581  
## iter 4 value -1.777502  
## iter 5 value -1.801008  
## iter 6 value -1.848809  
## iter 7 value -1.855659  
## iter 8 value -1.862321  
## iter 9 value -1.867637  
## iter 10 value -1.869263  
## iter 11 value -1.869474  
## iter 12 value -1.869681  
## iter 13 value -1.869931  
## iter 14 value -1.870450  
## iter 15 value -1.871500  
## iter 16 value -1.873022  
## iter 17 value -1.874367  
## iter 18 value -1.875803  
## iter 19 value -1.876302  
## iter 20 value -1.876824  
## iter 21 value -1.877050  
## iter 22 value -1.877228  
## iter 23 value -1.879369  
## iter 24 value -1.879486  
## iter 25 value -1.879788  
## iter 26 value -1.879961  
## iter 27 value -1.880338  
## iter 28 value -1.880478  
## iter 29 value -1.881114  
## iter 30 value -1.882095  
## iter 31 value -1.883875  
## iter 32 value -1.887092  
## iter 33 value -1.889703  
## iter 34 value -1.893609  
## iter 35 value -1.895626  
## iter 36 value -1.896142  
## iter 37 value -1.896246  
## iter 38 value -1.897647  
## iter 39 value -1.900174  
## iter 40 value -1.901019  
## iter 41 value -1.904478  
## iter 42 value -1.909764  
## iter 43 value -1.911357  
## iter 44 value -1.914332  
## iter 45 value -1.915995  
## iter 46 value -1.916759  
## iter 47 value -1.919043  
## iter 48 value -1.920851  
## iter 49 value -1.921714  
## iter 50 value -1.922119  
## iter 51 value -1.923153  
## iter 52 value -1.923184  
## iter 52 value -1.923184  
## iter 53 value -1.923318  
## iter 54 value -1.923322  
## iter 54 value -1.923322  
## iter 55 value -1.923340  
## iter 56 value -1.923341  
## iter 56 value -1.923341  
## iter 57 value -1.923343  
## iter 57 value -1.923343  
## iter 58 value -1.923343  
## iter 58 value -1.923343  
## iter 58 value -1.923343  
## final value -1.923343   
## converged  
## initial value -1.845927   
## iter 2 value -1.858893  
## iter 3 value -1.862419  
## iter 4 value -1.862630  
## iter 5 value -1.863510  
## iter 6 value -1.863699  
## iter 7 value -1.865508  
## iter 8 value -1.865911  
## iter 9 value -1.866218  
## iter 10 value -1.866252  
## iter 11 value -1.866273  
## iter 12 value -1.866287  
## iter 13 value -1.866321  
## iter 14 value -1.866403  
## iter 15 value -1.866551  
## iter 16 value -1.866794  
## iter 17 value -1.867138  
## iter 18 value -1.868133  
## iter 19 value -1.868307  
## iter 20 value -1.868389  
## iter 21 value -1.868507  
## iter 22 value -1.868540  
## iter 23 value -1.868550  
## iter 24 value -1.868552  
## iter 24 value -1.868552  
## final value -1.868552   
## converged



##### Comparison of various models

## Comparison of various models  
compare = data.frame(  
 Model = c("Mdl.1", "Mdl.2", "Mdl.3", "Mdl.4", "Mdl.5", "Mdl.6", "Mdl.7", "Mdl.8", "Mdl.9"),  
 Order = c("1,2,1", "2,2,2", "3,2,3", "1,2,1", "2,2,2", "3,2,3", "1,2,1", "2,2,2", "3,2,3"),  
 Xregs = c("","","","Y","Y","Y","","",""),  
 Lag.Xregs = c("","","","","","","Y","Y","Y"),  
 AIC = c(AIC(mdl.1$fit), AIC(mdl.2$fit), AIC(mdl.3$fit), AIC(mdl.4$fit), AIC(mdl.5$fit),  
 AIC(mdl.6$fit), AIC(mdl.7$fit), AIC(mdl.8$fit), AIC(mdl.9$fit)),  
 BIC = c(BIC(mdl.1$fit), BIC(mdl.2$fit), BIC(mdl.3$fit), BIC(mdl.4$fit), BIC(mdl.5$fit),  
 BIC(mdl.6$fit), BIC(mdl.7$fit), BIC(mdl.8$fit), BIC(mdl.9$fit)),  
 Best = c("Best BIC", "", "", "", "", "", "Best AIC", "", "")  
)  
  
kable(compare)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Model | Order | Xregs | Lag.Xregs | AIC | BIC | Best |
| Mdl.1 | 1,2,1 |  |  | -212.2957 | -201.4563 | Best BIC |
| Mdl.2 | 2,2,2 |  |  | -211.8094 | -193.7438 |  |
| Mdl.3 | 3,2,3 |  |  | -215.4772 | -190.1853 |  |
| Mdl.4 | 1,2,1 | Y |  | -211.5564 | -182.6514 |  |
| Mdl.5 | 2,2,2 | Y |  | -209.8342 | -177.3160 |  |
| Mdl.6 | 3,2,3 | Y |  | -215.0983 | -171.7408 |  |
| Mdl.7 | 1,2,1 |  | Y | -204.0732 | -171.5551 | Best AIC |
| Mdl.8 | 2,2,2 |  | Y | -220.7001 | -188.3477 |  |
| Mdl.9 | 3,2,3 |  | Y | -217.8920 | -174.7555 |  |

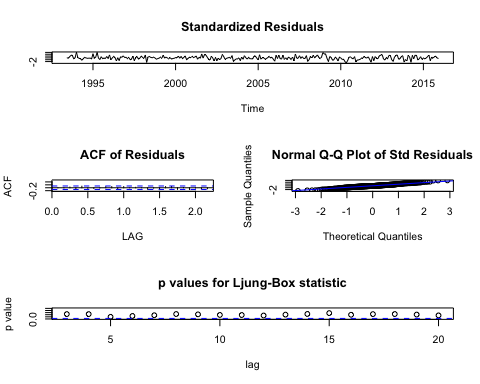
#xtable(compare)

### VAR Model

#### Fit a VAR on non stationary data

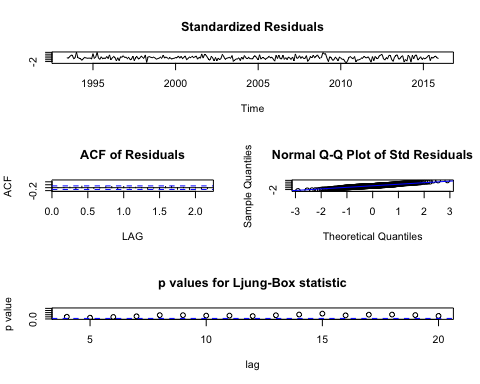
## Fit a VAR on non stationary data  
mdl.var1 = VAR(y = econ.sa[, 1:6], type = "both", p = 1)  
mdl.var2 = VAR(y = econ.sa[, 1:6], type = "both", p = 2)  
mdl.var3 = VAR(y = econ.sa[, 1:6], type = "both", p = 3)  
  
## Residuals of the Unemployment rate series  
# par(mfrow = c(1,3))  
# acf(residuals(mdl.var1)[,1])  
# acf(residuals(mdl.var2)[,1])  
# acf(residuals(mdl.var3)[,1])  
#   
# plot(mdl.var1)  
# plot(mdl.var2)  
# plot(mdl.var3)  
#   
# plot(predict(mdl.var1, n.ahead = 12))  
# plot(predict(mdl.var2, n.ahead = 12))  
# plot(predict(mdl.var3, n.ahead = 12))  
  
## Fit a VAR on non stationary data  
mdl.var4 = VAR(y = econ.sa.lag, type = "both", p = 1)  
mdl.var5 = VAR(y = econ.sa.lag, type = "both", p = 2)  
mdl.var6 = VAR(y = econ.sa.lag, type = "both", p = 3)  
  
## Residuals of the Unemployment rate series  
# par(mfrow = c(1,3))  
# acf(residuals(mdl.var4)[,1])  
# acf(residuals(mdl.var5)[,1])  
# acf(residuals(mdl.var6)[,1])  
#   
# plot(mdl.var4)  
# plot(mdl.var5)  
# plot(mdl.var6)  
#   
# plot(predict(mdl.var4, n.ahead = 12))  
# plot(predict(mdl.var5, n.ahead = 12))  
# plot(predict(mdl.var6, n.ahead = 12))  
  
## Models with lagged Regressors  
mdl.7 = sarima(econ.sa.lag$unem\_rate\_sa, p = 1, d = 2, q = 1, xreg = econ.sa.lag[, 2:6])

## initial value -1.477884   
## iter 2 value -1.738588  
## iter 3 value -1.753791  
## iter 4 value -1.770816  
## iter 5 value -1.826630  
## iter 6 value -1.841338  
## iter 7 value -1.841916  
## iter 8 value -1.854288  
## iter 9 value -1.857758  
## iter 10 value -1.858698  
## iter 11 value -1.859174  
## iter 12 value -1.859190  
## iter 13 value -1.859191  
## iter 14 value -1.859191  
## iter 15 value -1.859191  
## iter 15 value -1.859191  
## iter 15 value -1.859191  
## final value -1.859191   
## converged  
## initial value -1.861031   
## iter 2 value -1.861224  
## iter 3 value -1.861546  
## iter 4 value -1.861611  
## iter 5 value -1.861962  
## iter 6 value -1.862074  
## iter 7 value -1.862148  
## iter 8 value -1.862155  
## iter 9 value -1.862157  
## iter 10 value -1.862158  
## iter 11 value -1.862158  
## iter 11 value -1.862158  
## iter 11 value -1.862158  
## final value -1.862158   
## converged



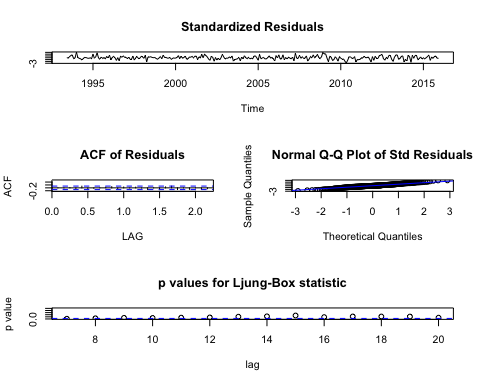
mdl.8 = sarima(econ.sa.lag$unem\_rate\_sa, p = 1, d = 2, q = 2, xreg = econ.sa.lag[, 2:6])

## initial value -1.477884   
## iter 2 value -1.718479  
## iter 3 value -1.733107  
## iter 4 value -1.771112  
## iter 5 value -1.804662  
## iter 6 value -1.841321  
## iter 7 value -1.847109  
## iter 8 value -1.849829  
## iter 9 value -1.854107  
## iter 10 value -1.856502  
## iter 11 value -1.856929  
## iter 12 value -1.857656  
## iter 13 value -1.858419  
## iter 14 value -1.860687  
## iter 15 value -1.863689  
## iter 16 value -1.864318  
## iter 17 value -1.864492  
## iter 18 value -1.864651  
## iter 19 value -1.864654  
## iter 20 value -1.864656  
## iter 21 value -1.864659  
## iter 22 value -1.864663  
## iter 23 value -1.864663  
## iter 24 value -1.864665  
## iter 25 value -1.864667  
## iter 26 value -1.864671  
## iter 27 value -1.864675  
## iter 28 value -1.864678  
## iter 29 value -1.864678  
## iter 30 value -1.864678  
## iter 30 value -1.864679  
## iter 30 value -1.864679  
## final value -1.864679   
## converged  
## initial value -1.858401   
## iter 2 value -1.858882  
## iter 3 value -1.858913  
## iter 4 value -1.859069  
## iter 5 value -1.859074  
## iter 6 value -1.859244  
## iter 7 value -1.859486  
## iter 8 value -1.860030  
## iter 9 value -1.860834  
## iter 10 value -1.861624  
## iter 11 value -1.862374  
## iter 12 value -1.862574  
## iter 13 value -1.862607  
## iter 14 value -1.862614  
## iter 15 value -1.862617  
## iter 16 value -1.862619  
## iter 17 value -1.862619  
## iter 17 value -1.862619  
## iter 17 value -1.862619  
## final value -1.862619   
## converged



mdl.9 = sarima(econ.sa.lag$unem\_rate\_sa, p = 3, d = 2, q = 3, xreg = econ.sa.lag[, 2:6])

## initial value -1.475425   
## iter 2 value -1.691914  
## iter 3 value -1.701581  
## iter 4 value -1.777502  
## iter 5 value -1.801008  
## iter 6 value -1.848809  
## iter 7 value -1.855659  
## iter 8 value -1.862321  
## iter 9 value -1.867637  
## iter 10 value -1.869263  
## iter 11 value -1.869474  
## iter 12 value -1.869681  
## iter 13 value -1.869931  
## iter 14 value -1.870450  
## iter 15 value -1.871500  
## iter 16 value -1.873022  
## iter 17 value -1.874367  
## iter 18 value -1.875803  
## iter 19 value -1.876302  
## iter 20 value -1.876824  
## iter 21 value -1.877050  
## iter 22 value -1.877228  
## iter 23 value -1.879369  
## iter 24 value -1.879486  
## iter 25 value -1.879788  
## iter 26 value -1.879961  
## iter 27 value -1.880338  
## iter 28 value -1.880478  
## iter 29 value -1.881114  
## iter 30 value -1.882095  
## iter 31 value -1.883875  
## iter 32 value -1.887092  
## iter 33 value -1.889703  
## iter 34 value -1.893609  
## iter 35 value -1.895626  
## iter 36 value -1.896142  
## iter 37 value -1.896246  
## iter 38 value -1.897647  
## iter 39 value -1.900174  
## iter 40 value -1.901019  
## iter 41 value -1.904478  
## iter 42 value -1.909764  
## iter 43 value -1.911357  
## iter 44 value -1.914332  
## iter 45 value -1.915995  
## iter 46 value -1.916759  
## iter 47 value -1.919043  
## iter 48 value -1.920851  
## iter 49 value -1.921714  
## iter 50 value -1.922119  
## iter 51 value -1.923153  
## iter 52 value -1.923184  
## iter 52 value -1.923184  
## iter 53 value -1.923318  
## iter 54 value -1.923322  
## iter 54 value -1.923322  
## iter 55 value -1.923340  
## iter 56 value -1.923341  
## iter 56 value -1.923341  
## iter 57 value -1.923343  
## iter 57 value -1.923343  
## iter 58 value -1.923343  
## iter 58 value -1.923343  
## iter 58 value -1.923343  
## final value -1.923343   
## converged  
## initial value -1.845927   
## iter 2 value -1.858893  
## iter 3 value -1.862419  
## iter 4 value -1.862630  
## iter 5 value -1.863510  
## iter 6 value -1.863699  
## iter 7 value -1.865508  
## iter 8 value -1.865911  
## iter 9 value -1.866218  
## iter 10 value -1.866252  
## iter 11 value -1.866273  
## iter 12 value -1.866287  
## iter 13 value -1.866321  
## iter 14 value -1.866403  
## iter 15 value -1.866551  
## iter 16 value -1.866794  
## iter 17 value -1.867138  
## iter 18 value -1.868133  
## iter 19 value -1.868307  
## iter 20 value -1.868389  
## iter 21 value -1.868507  
## iter 22 value -1.868540  
## iter 23 value -1.868550  
## iter 24 value -1.868552  
## iter 24 value -1.868552  
## final value -1.868552   
## converged



#### Comparison of various models

## Comparison of various models  
compare = data.frame(  
 Model = c("Mdl.1", "Mdl.2", "Mdl.3", "Mdl.4", "Mdl.5", "Mdl.6", "Mdl.7", "Mdl.8", "Mdl.9"),  
 Order = c("1,2,1", "2,2,2", "3,2,3", "1,2,1", "2,2,2", "3,2,3", "1,2,1", "2,2,2", "3,2,3"),  
 Xregs = c("","","","Y","Y","Y","","",""),  
 Lag.Xregs = c("","","","","","","Y","Y","Y"),  
 AIC = c(AIC(mdl.1$fit), AIC(mdl.2$fit), AIC(mdl.3$fit), AIC(mdl.4$fit), AIC(mdl.5$fit),  
 AIC(mdl.6$fit), AIC(mdl.7$fit), AIC(mdl.8$fit), AIC(mdl.9$fit)),  
 BIC = c(BIC(mdl.1$fit), BIC(mdl.2$fit), BIC(mdl.3$fit), BIC(mdl.4$fit), BIC(mdl.5$fit),  
 BIC(mdl.6$fit), BIC(mdl.7$fit), BIC(mdl.8$fit), BIC(mdl.9$fit)),  
 Best = c("Best BIC", "", "", "", "", "", "Best AIC", "", "")  
)  
  
kable(compare)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Model | Order | Xregs | Lag.Xregs | AIC | BIC | Best |
| Mdl.1 | 1,2,1 |  |  | -212.2957 | -201.4563 | Best BIC |
| Mdl.2 | 2,2,2 |  |  | -211.8094 | -193.7438 |  |
| Mdl.3 | 3,2,3 |  |  | -215.4772 | -190.1853 |  |
| Mdl.4 | 1,2,1 | Y |  | -211.5564 | -182.6514 |  |
| Mdl.5 | 2,2,2 | Y |  | -209.8342 | -177.3160 |  |
| Mdl.6 | 3,2,3 | Y |  | -215.0983 | -171.7408 |  |
| Mdl.7 | 1,2,1 |  | Y | -222.4520 | -193.6943 | Best AIC |
| Mdl.8 | 2,2,2 |  | Y | -220.7001 | -188.3477 |  |
| Mdl.9 | 3,2,3 |  | Y | -217.8920 | -174.7555 |  |

#xtable(compare)

Best Models

#############################################################  
## Best Models  
  
compare.best = data.frame(  
 Model = c("ARIMA(1,2,1)", "ARIMA(1,2,1)", "VAR(1)", "VAR(2)"),  
 Lag.XRegs = c("", "Y", "", "Y"),  
 AIC = c(-212.29, -222.45, -226.34, -235.43),  
 BIC = c(-201.45, -193.69, -193.79, -181.51),  
 Best = c("Best BIC", "", "", "Best AIC")  
)  
kable(compare.best)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Model | Lag.XRegs | AIC | BIC | Best |
| ARIMA(1,2,1) |  | -212.29 | -201.45 | Best BIC |
| ARIMA(1,2,1) | Y | -222.45 | -193.69 |  |
| VAR(1) |  | -226.34 | -193.79 |  |
| VAR(2) | Y | -235.43 | -181.51 | Best AIC |

#xtable(compare.best)  
#thank you for including the xtable code, it will help with the writeup  
#############################################################  
## Forecasting