US Consumption Forecast

Jiyoon Moon

Last compiled on: 11 June, 2025

Abstract

This report analyzes quarterly US consumption data using multiple time series models. Both ARIMA-based and regression-based models are evaluated to understand the drivers of consumption and to forecast its behavior in the near future. External variables such as income, production, and unemployment are explored to improve forecast accuracy.

1. Time Series Analysis and EDA

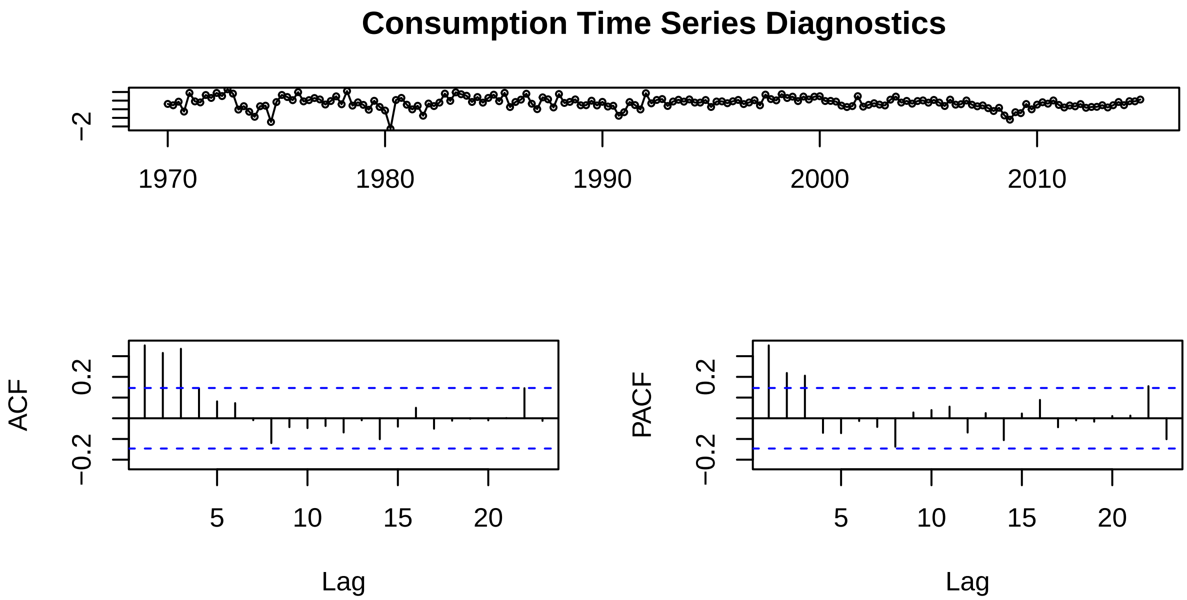


Figure 1: US Consumption Time Series (1970–2014)

The top panel of Figure 1 shows the time series of US Consumption from 1970 to 2014. The series shows no clear seasonality but fluctuates moderately over time, with alternating periods of mild increases and decreases, a sharp dip around 2008–2009 likely due to the Global Financial Crisis, and an overall appearance of stationarity with minor changes in trend and volatility. The ACF shows significant autocorrelation at lags 1 to 3 and then drops off sharply at lag 4, indicating a potential MA(3) structure, while the PACF gradually decreases without a clear cut-off, which is also characteristic of a moving average (MA) process rather than an autoregressive one.

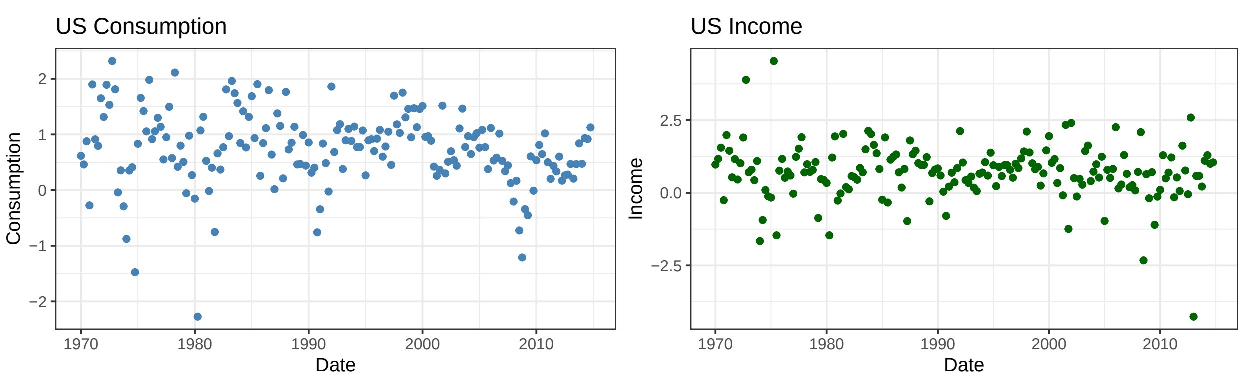


Figure 2: Quarterly US Consumption and Income Over Time

Figure 2 shows the quarterly evolution of US consumption and income. Consumption remains relatively stable over time, with no clear upward or downward trend. It fluctuates around a consistent level, with noticeable drops around 2008 likely due to the global financial crisis. Income also shows mild variation across quarters, without a strong long-term increase.

2. Model

Model 1) ARIMA(0,0,3)

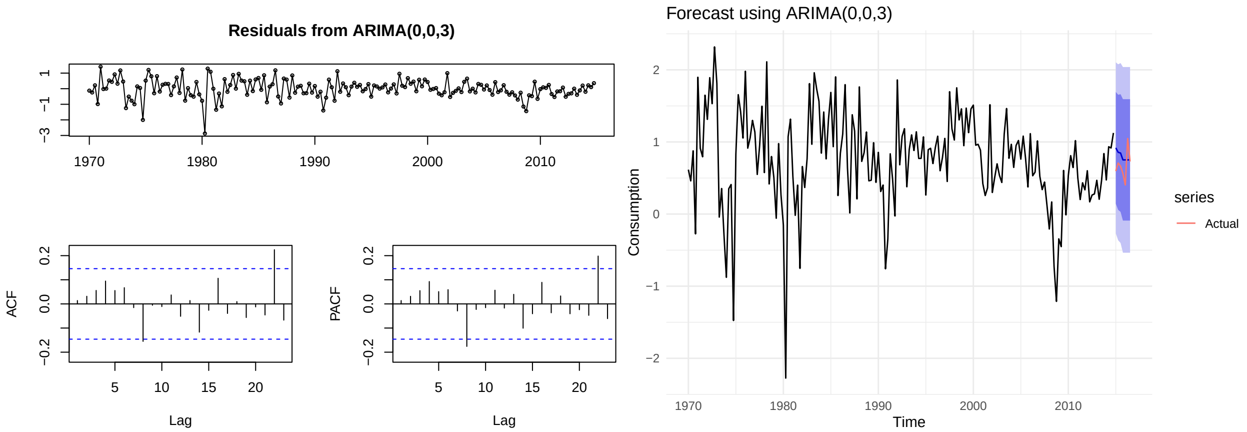


Figure 3: Residuals and Forecast - ARIMA(0,0,3)

The ACF plot shows a sharp drop after lag 3, while the PACF declines gradually across lags 1 to 3 with no significant spikes beyond, suggesting an MA(3) structure. Based on this, the ARIMA(0,0,3) model was selected. The residuals resemble white noise, indicating a good fit, and the forecast closely matches the actual test values within appropriate prediction intervals. This model is simple, interpretable, and reliable for short-term forecasting, with a training RMSE of about 0.60 and a test RMSE of 0.24.

Model 2) ARIMA(3,0,0)(2,0,0)[4]

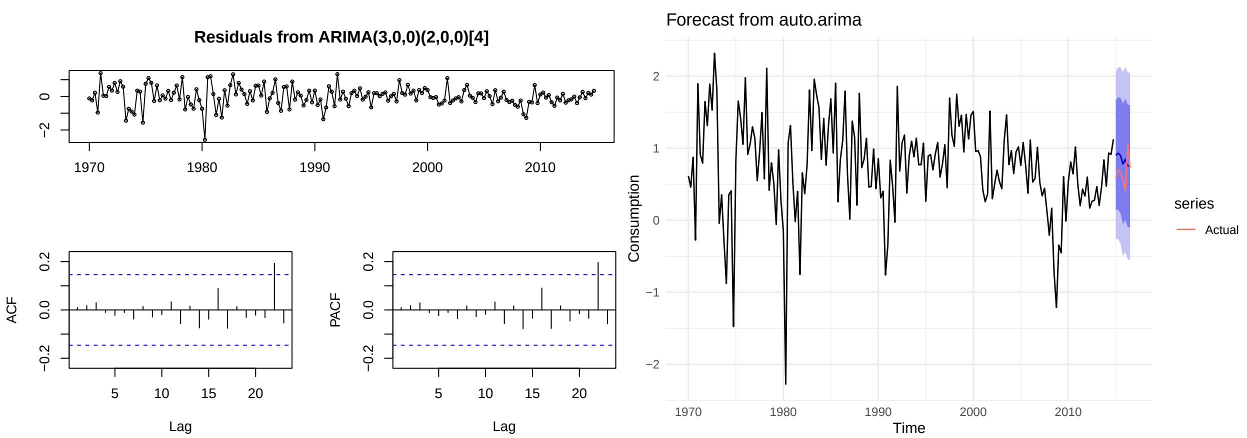


Figure 4: Residuals and Forecast - ARIMA(3,0,0)(2,0,0)[4]

The model includes both non-seasonal AR(3) and seasonal AR(2) terms which were selected based on significant PACF spikes at lags 1–3 and the presence of quarterly seasonality. It performs well, with residuals resembling white noise and delivering high forecast accuracy. The ARIMA(3,0,0)(2,0,0)[4] model effectively captures both seasonal and autoregressive dynamics and is suitable for short-term prediction with a test RMSE of approximately 0.27.

Model 3) Regression with ARIMA(1,1,1)(0,0,1)[4] Errors

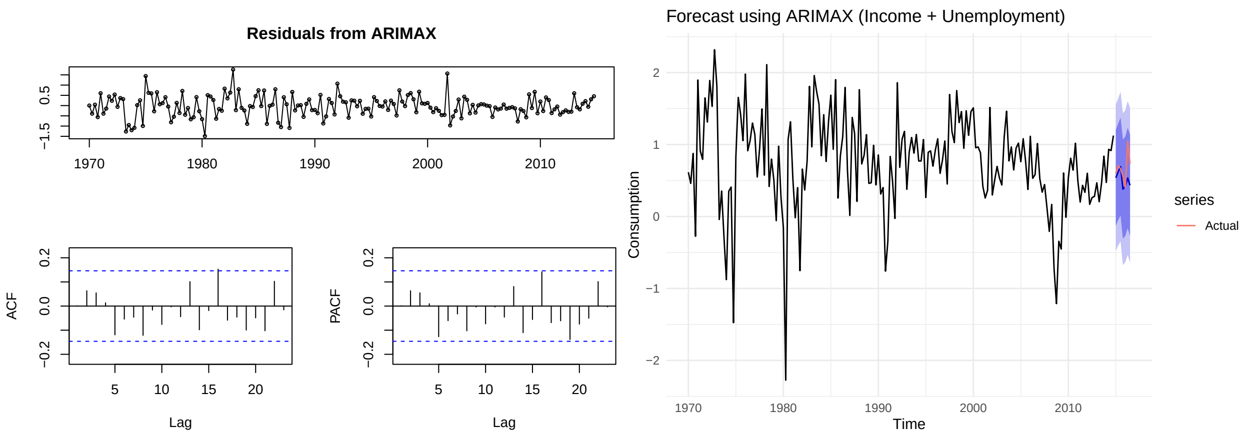


Figure 5: Residuals and Forecast - ARIMAX (Income + Unemployment)

The dynamic regression model uses Income and Unemployment as regressors while implementing an ARIMA(1,1,1)(0,0,1)[4] structure for error modeling which automatically detects both explanatory and seasonal components. The model shows well-behaved residuals and strong forecast performance and the external regressors improve the accuracy. The ARIMAX specification among all models tested provides the best balance of accuracy and interpretability with the lowest test RMSE (0.24) and MAE (0.17) which indicates robust out-of-sample performance.

3. Model Comparison and Conclusion

Table 1: Comparison of three models

Model	Test.RMSE	Test.MAE	Theil.s.U	Ljung.Box.p.value
ARIMA(0,0,3)	0.2401	0.2153	0.6391	0.1297
ARIMA(3,0,0)(2,0,0)[4]	0.2747	0.2482	0.7365	0.8787
ARIMAX(1,1,1)(0,0,1)[4]	0.2374	0.1687	0.8030	0.1703

The ARIMAX(1,1,1)(0,0,1)[4] model produced the lowest RMSE value of 0.237 and the lowest MAE value of 0.169 which indicates its best out-of-sample prediction accuracy. The Theil's U value of this model was slightly higher than others but it still indicates good forecast performance. The ARIMAX(1,1,1)(0,0,1)[4] model with Income and Unemployment variables should be recommended because it provides the best accuracy and economic interpretability. However, there are limitations: only two external variables were included, the models assumed linearity which may overlook complex patterns, structural breaks or shocks were not accounted for, and the static training/testing split could limit generalizability. Future improvements may involve adding more macroeconomic predictors (e.g., interest rate, inflation), exploring non-linear or machine learning approaches, and incorporating structural break detection or rolling forecast evaluations.

[END of the REPORT]

R code

Question 1: Consumption Time Series Analysis

(a) Data Splitting

```r
# Load the uschange dataset from fpp2 package
data("uschange")
```

```r
# Split the dataset into training (1970 Q1 to 2014 Q4) and test (2015 Q1 to 2016 Q3)
train <- window(uschange, end=c(2014,4))
test  <- window(uschange, start=c(2015,1))
```

(b) Exploratory Data Analysis (EDA)

```r
# Summary statistics of training data
summary(train)
```

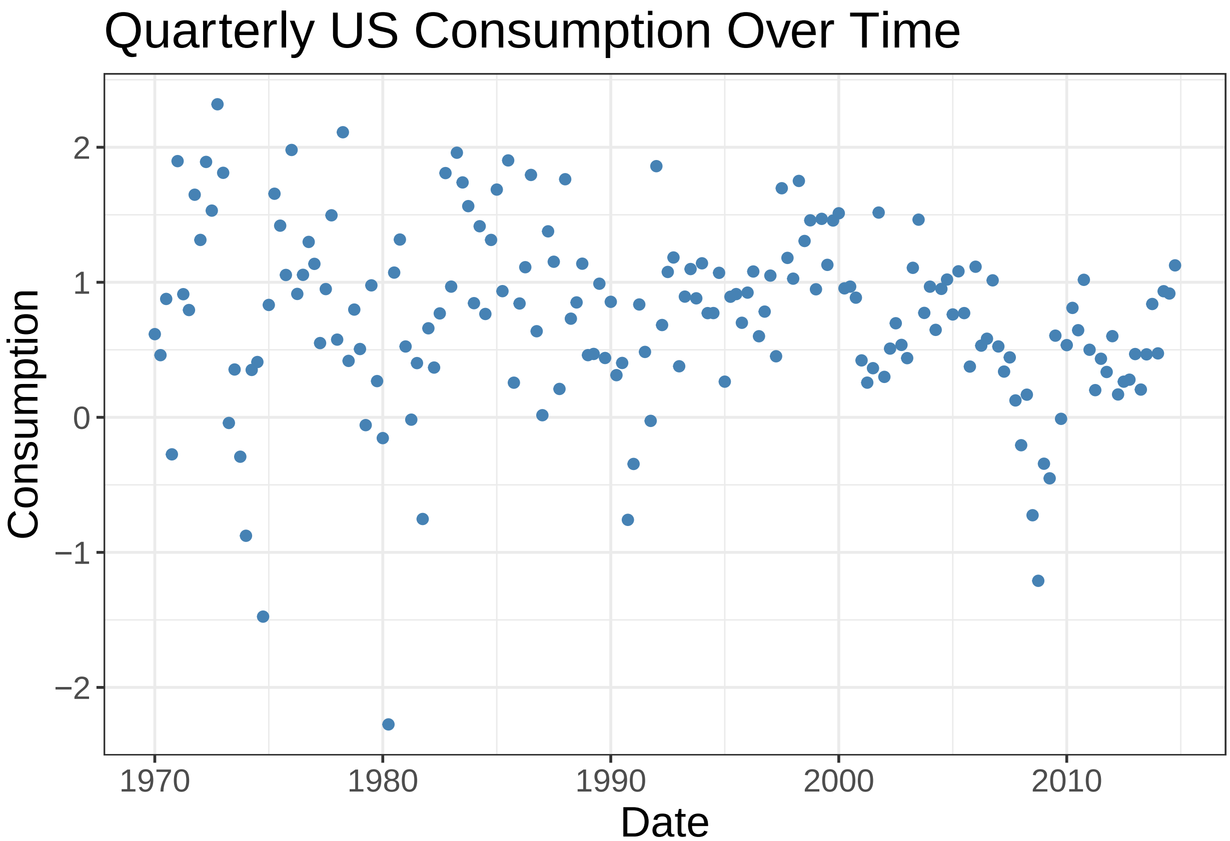
```
##   Consumption          Income          Production         Savings
##  Min.   :-2.2741   Min.   :-4.2652   Min.   :-6.8510   Min.   :-68.788
##  1st Qu.: 0.4159   1st Qu.: 0.2833   1st Qu.: 0.1429   1st Qu.: -4.820
##  Median : 0.7888   Median : 0.7232   Median : 0.6979   Median :  1.133
##  Mean   : 0.7493   Mean   : 0.7185   Mean   : 0.5377   Mean   :  1.215
##  3rd Qu.: 1.1083   3rd Qu.: 1.1727   3rd Qu.: 1.3420   3rd Qu.:  7.065
##  Max.   : 2.3183   Max.   : 4.5365   Max.   : 4.1496   Max.   : 50.758
##   Unemployment
##  Min.   :-0.90000
##  1st Qu.:-0.20000
##  Median : 0.00000
##  Mean   : 0.01167
##  3rd Qu.: 0.10000
##  Max.   : 1.40000
```

```r
# Check for missing values
colSums(is.na(train))
```

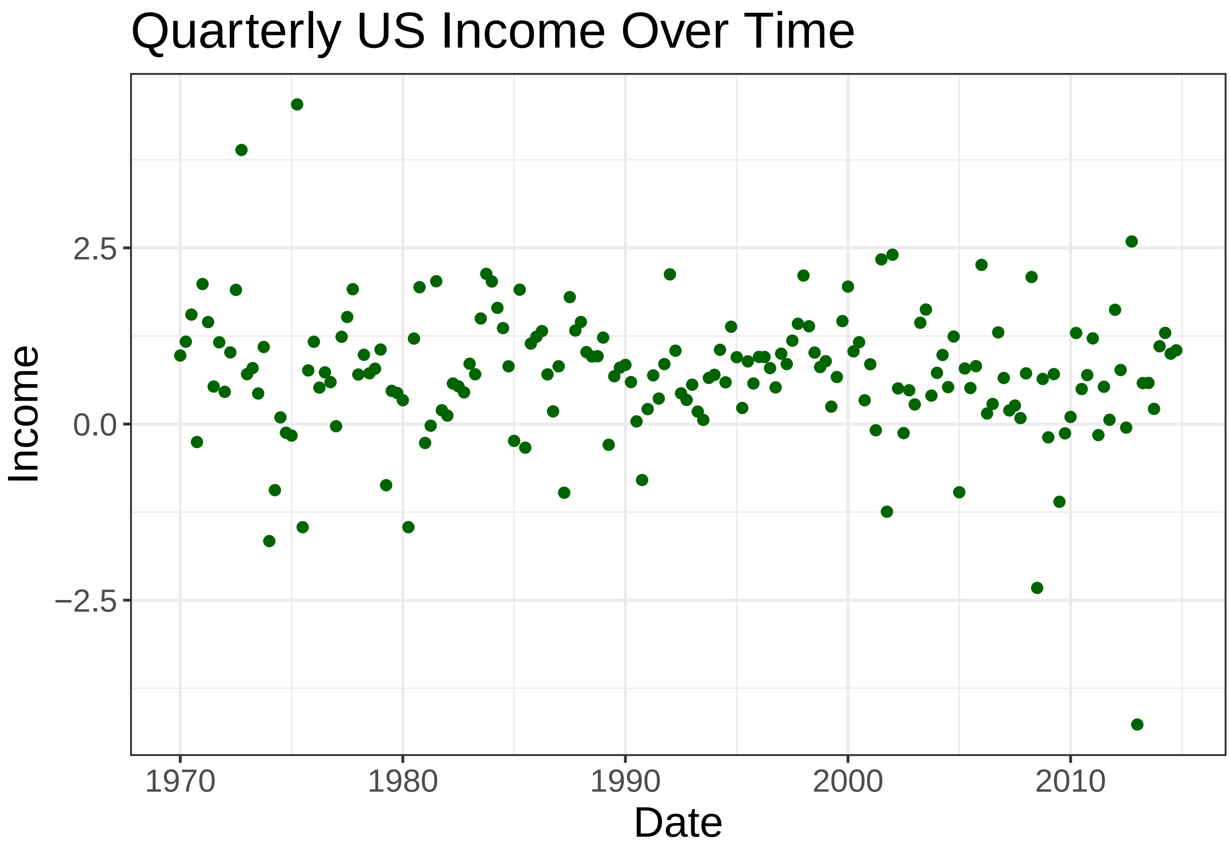
```
##  Consumption       Income   Production      Savings Unemployment
##            0            0            0            0            0
```

```r
# Visualizing Consumption over time
train_df <- as.data.frame(train)  # Convert ts to data frame for ggplot
train_df$Date <- time(train)      # Add time column for plotting

ggplot(train_df, aes(x = Date, y = Consumption)) +
  geom_point(color = "steelblue") +
  labs(title = "Quarterly US Consumption Over Time") +
```

```
  theme_bw() +
  theme(axis.text.x = element_text(size = 12),
        axis.text.y = element_text(size = 12),
        title = element_text(size = 16))
```




```
# Visualizing average Income over time
# Although ts is evenly spaced, we emulate the behavior of previous project EDA

ggplot(train_df, aes(x = Date, y = Income)) +
  geom_point(color = "darkgreen") +
  labs(title = "Quarterly US Income Over Time") +
  theme_bw() +
  theme(axis.text.x = element_text(size = 12),
        axis.text.y = element_text(size = 12),
        title = element_text(size = 16))
```

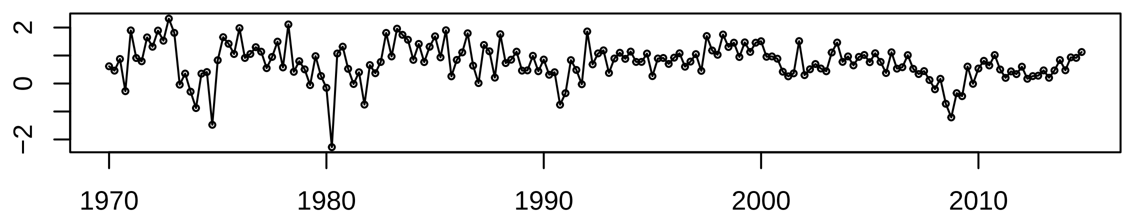


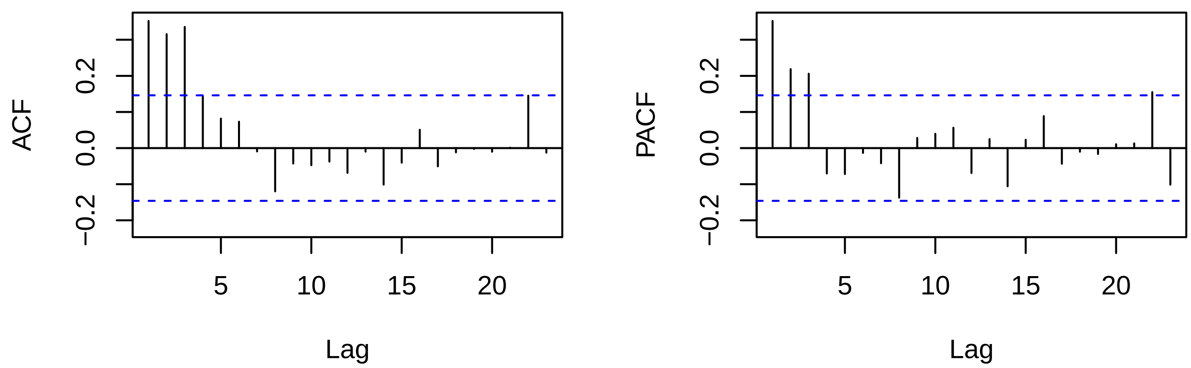

```r
# Calculate correlation matrix to assess variable relationships
cor(train)
```

```
##              Consumption     Income  Production     Savings Unemployment
## Consumption    1.0000000  0.3989102  0.54879015 -0.23931232   -0.5439411
## Income         0.3989102  1.0000000  0.27944720  0.71379623   -0.2312733
## Production     0.5487901  0.2794472  1.00000000 -0.06811485   -0.7989555
## Savings       -0.2393123  0.7137962 -0.06811485  1.00000000    0.1109808
## Unemployment  -0.5439411 -0.2312733 -0.79895549  0.11098078    1.0000000
```

```r
# ACF and PACF plots to assess autocorrelation and potential ARIMA components
tsdisplay(train[, "Consumption"], main = "Consumption Time Series Diagnostics")
```

Consumption Time Series Diagnostics





(c) Model Fitting and Forecasting

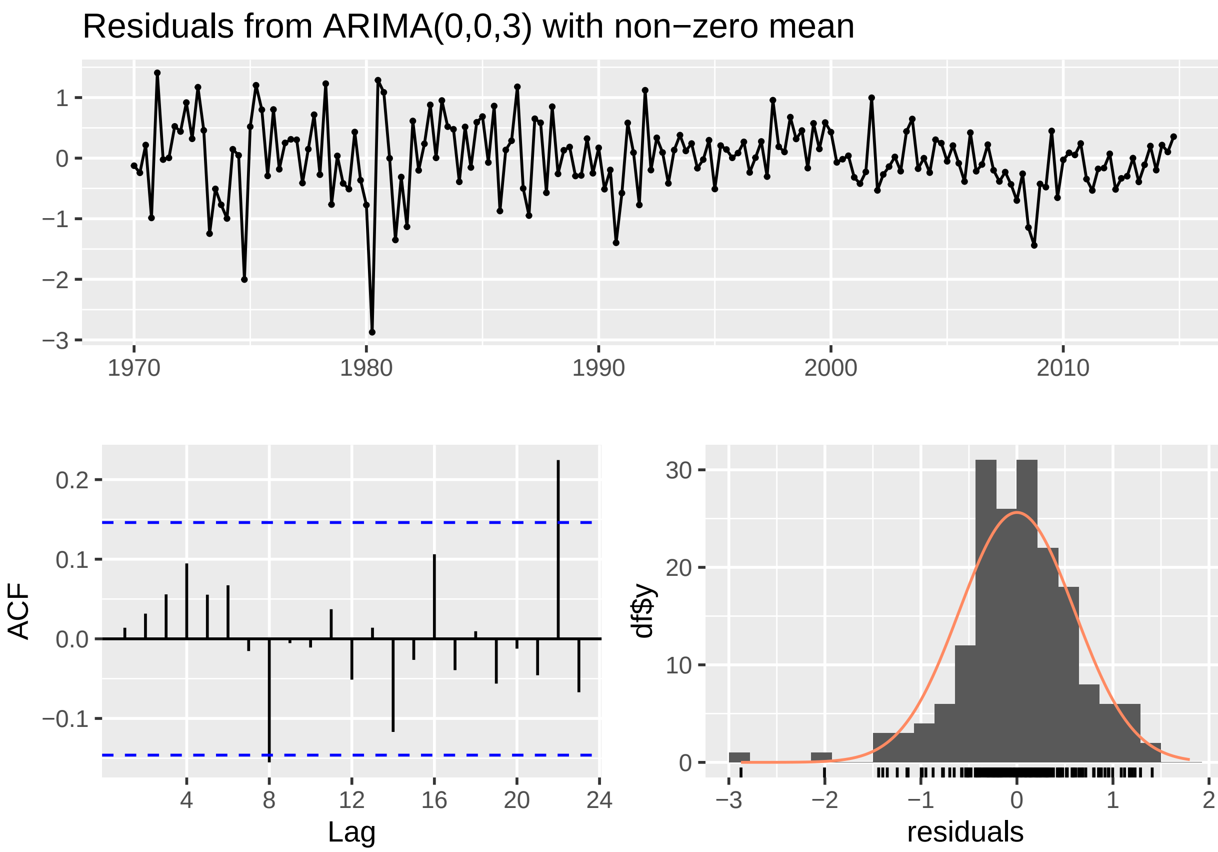
1) Model Selection: ARIMA(0,0,3)

```
# Fit ARIMA(0,0,3) model (pure MA(3) model)
fit_ma3 <- Arima(train[,"Consumption"], order = c(0, 0, 3))

# Display model summary: coefficients, AIC, etc.
summary(fit_ma3)
```

```
## Series: train[, "Consumption"]
## ARIMA(0,0,3) with non-zero mean
##
## Coefficients:
##          ma1     ma2     ma3    mean
##       0.2435  0.2183  0.2679  0.7517
## s.e.  0.0733  0.0734  0.0648  0.0768
##
## sigma^2 = 0.366:  log likelihood = -163.07
## AIC=336.15   AICc=336.49   BIC=352.11
##
## Training set error measures:
##                            ME      RMSE        MAE       MPE      MAPE       MASE
```

```
## Training set -5.533779e-05 0.5981964 0.4416385 83.09945 209.9472 0.6756687
##                      ACF1
## Training set 0.01392546
```

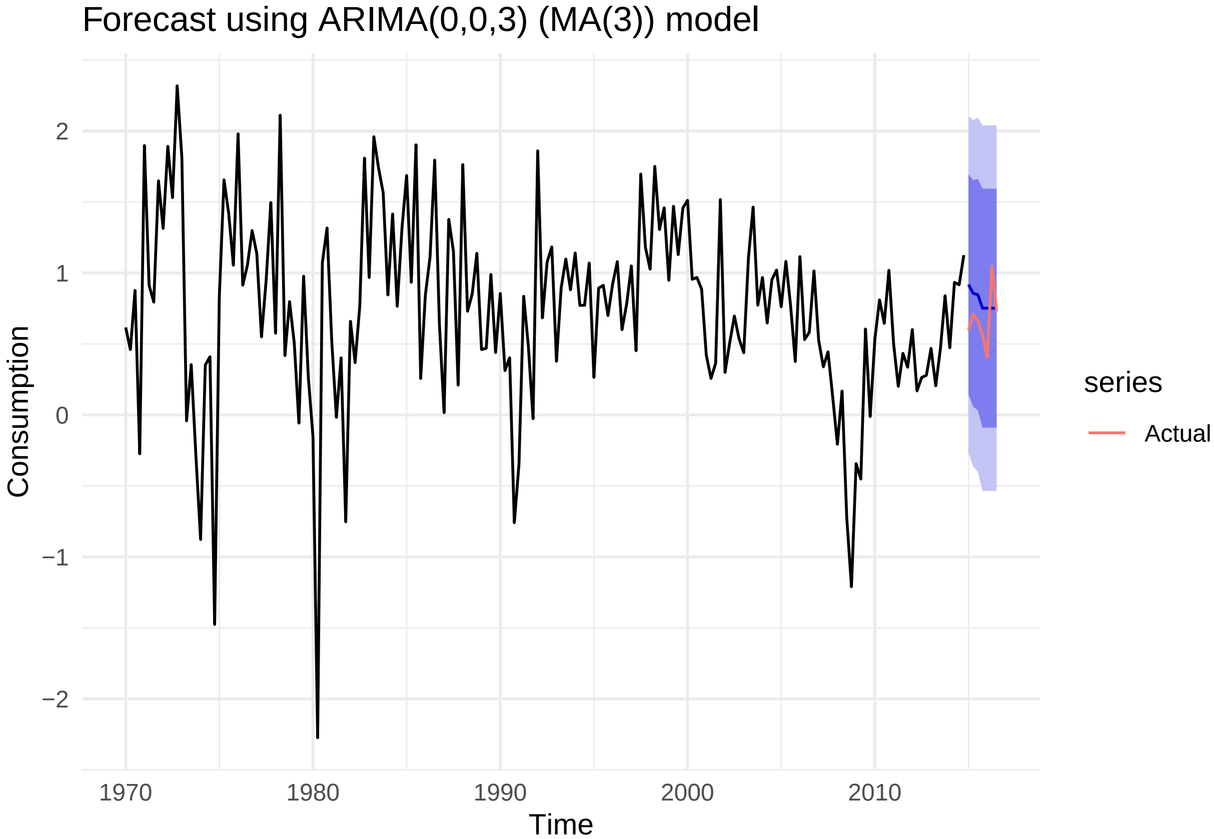
```r
# Check if residuals are white noise (no autocorrelation)
checkresiduals(fit_ma3)
```




```
##
##  Ljung-Box test
##
## data:  Residuals from ARIMA(0,0,3) with non-zero mean
## Q* = 8.5225, df = 5, p-value = 0.1297
##
## Model df: 3.   Total lags used: 8
```

```r
# Forecast 7 steps ahead (to match test set)
fc_ma3 <- forecast(fit_ma3, h = 7)

# Plot forecast with actual test values
autoplot(fc_ma3) +
  autolayer(test[,"Consumption"], series = "Actual") +
  labs(title = "Forecast using ARIMA(0,0,3) (MA(3)) model",
       y = "Consumption", x = "Time") +
```

```r
  theme_minimal()
```




```r
# Compare forecast to test set
accuracy(fc_ma3, test[,"Consumption"])
```

```
##                         ME      RMSE       MAE       MPE      MAPE      MASE
## Training set -5.533779e-05 0.5981964 0.4416385  83.09945 209.94724 0.6756687
## Test set     -1.307562e-01 0.2400561 0.2153278 -28.09347  36.16553 0.3294329
##                     ACF1 Theil's U
## Training set  0.01392546        NA
## Test set     -0.09102033 0.6391078
```

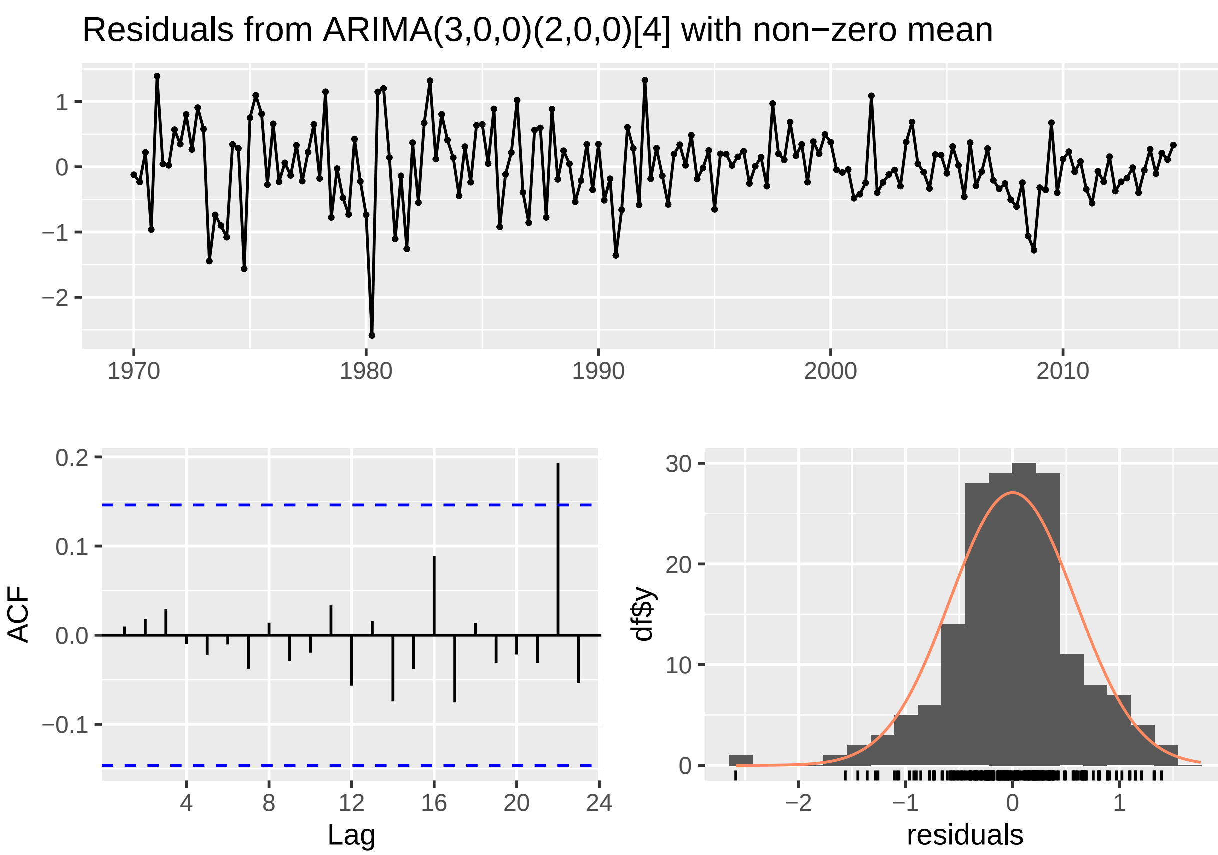
2) Model Selection: ARIMA(3,0,0)(2,0,0)[4]

```r
# Fit an ARIMA model automatically selected by AICc
fit_auto <- auto.arima(train[,"Consumption"])

# Display summary of the automatically selected model
summary(fit_auto)
```

```
## Series: train[, "Consumption"]
## ARIMA(3,0,0)(2,0,0)[4] with non-zero mean
##
```

```
## Coefficients:
##          ar1     ar2     ar3     sar1     sar2    mean
##       0.2271  0.1777  0.2200  -0.0334  -0.1803  0.7522
## s.e.  0.0737  0.0736  0.0724   0.0772   0.0744  0.0946
##
## sigma^2 = 0.353:  log likelihood = -158.94
## AIC=331.88   AICc=332.53   BIC=354.23
##
## Training set error measures:
##                        ME      RMSE       MAE      MPE     MAPE     MASE
## Training set 0.0002895183 0.5841338 0.4391314 65.53638 188.5312 0.671833
##                     ACF1
## Training set 0.009685967
```

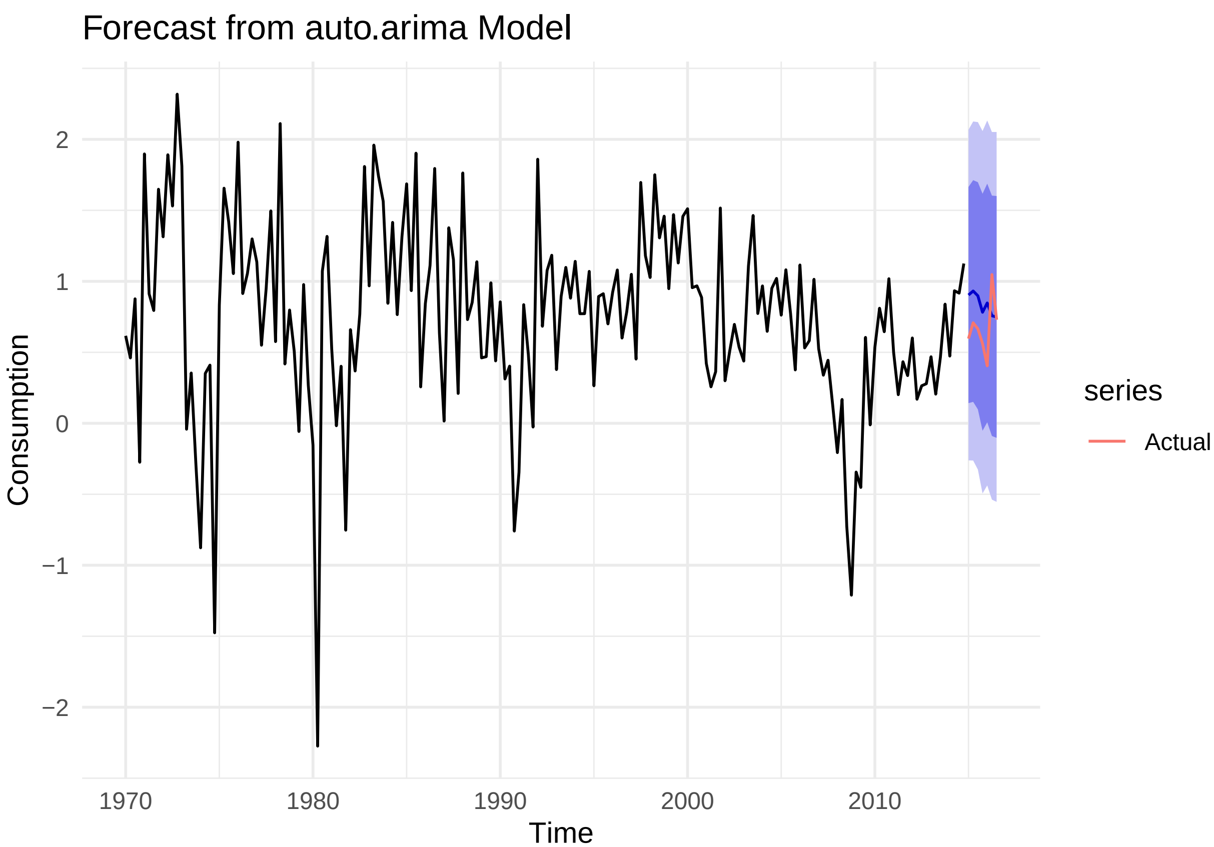
```r
# Residual diagnostics: check whether residuals are uncorrelated and normally distribu
checkresiduals(fit_auto)
```




```
##
##  Ljung-Box test
##
## data:  Residuals from ARIMA(3,0,0)(2,0,0)[4] with non-zero mean
## Q* = 0.67677, df = 3, p-value = 0.8787
##
```

```
## Model df: 5.   Total lags used: 8
```

```r
# Forecast 7 steps ahead using the auto.arima model
fc_auto <- forecast(fit_auto, h = 7)

# Plot forecast and overlay actual test values
autoplot(fc_auto) +
  autolayer(test[,"Consumption"], series = "Actual") +
  labs(title = "Forecast from auto.arima Model",
       y = "Consumption", x = "Time") +
  theme_minimal()
```




```r
# Evaluate forecast accuracy against the test set
accuracy(fc_auto, test[,"Consumption"])
```

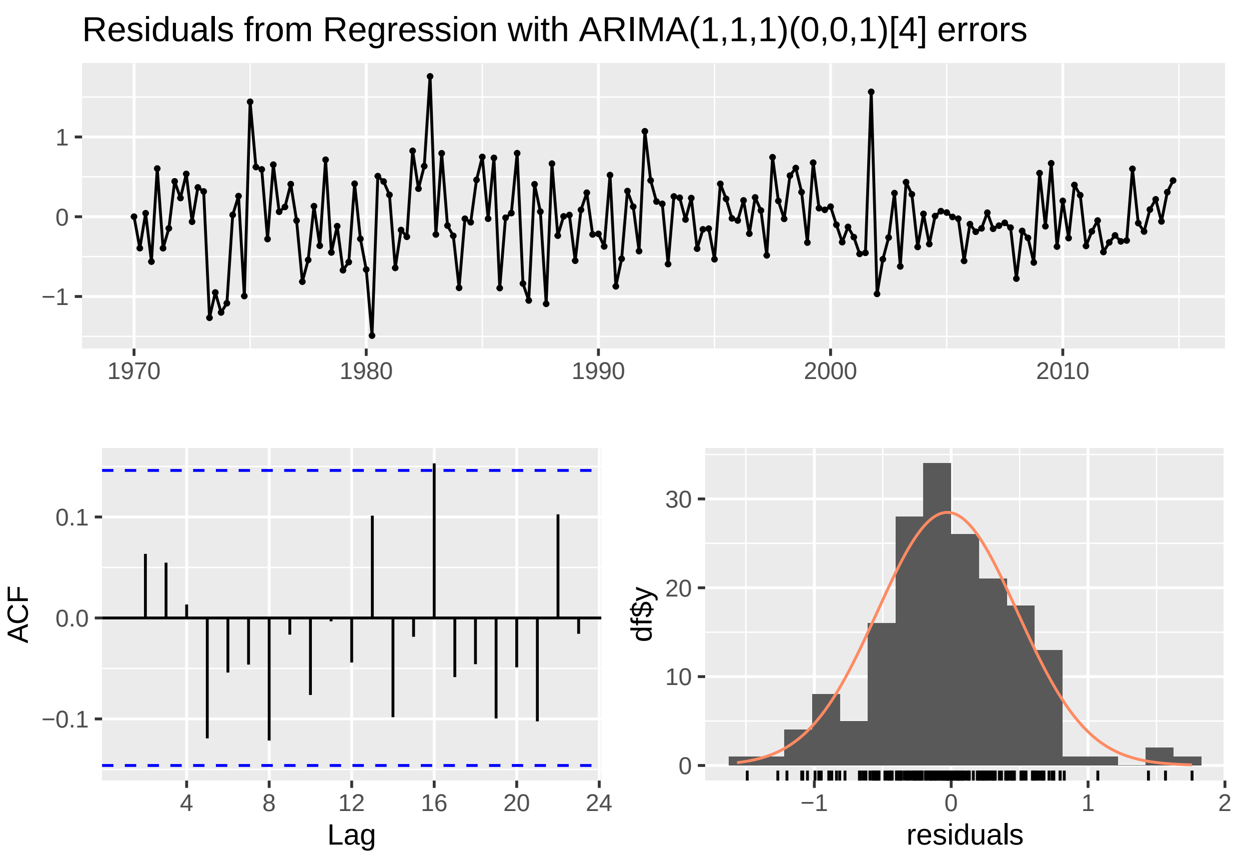
```
##                         ME      RMSE       MAE       MPE      MAPE      MASE
## Training set  0.0002895183 0.5841338 0.4391314  65.53638 188.53118 0.6718330
## Test set     -0.1651797648 0.2746917 0.2481962 -34.54071  42.46434 0.3797188
##                      ACF1 Theil's U
## Training set  0.009685967        NA
## Test set     -0.085512024 0.7365495
```

3) Model Selection: Regression with ARIMA(1,1,1)(0,0,1)[4] Errors

```r
# Fit ARIMA model with external regressors: Income and Unemployment
# This is a dynamic regression model with ARIMA errors
xreg_train <- as.matrix(train[, c("Income", "Unemployment")])
xreg_test <- as.matrix(test[, c("Income", "Unemployment")])

fit_arimax <- auto.arima(train[,"Consumption"], xreg = xreg_train)

# Show model summary
summary(fit_arimax)
```

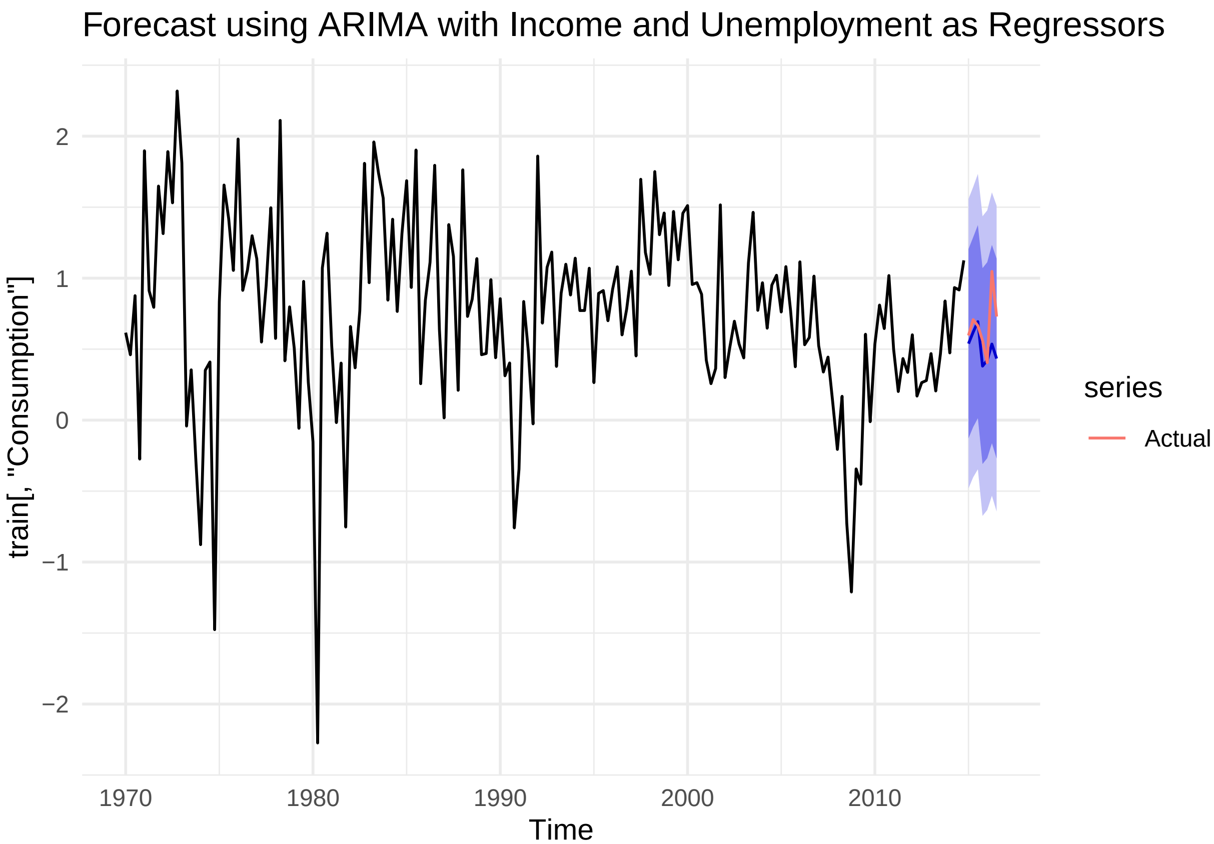
```
## Series: train[, "Consumption"]
## Regression with ARIMA(1,1,1)(0,0,1)[4] errors
##
## Coefficients:
##           ar1      ma1     sma1  Income  Unemployment
##       -0.1261  -0.8024  -0.1998  0.1592       -0.9085
## s.e.   0.1197   0.1041   0.0994  0.0440        0.1095
##
## sigma^2 = 0.2706:  log likelihood = -135.27
## AIC=282.53   AICc=283.02   BIC=301.66
##
## Training set error measures:
##                       ME      RMSE     MAE      MPE     MAPE      MASE
## Training set -0.02826937 0.5114767 0.39035 29.05286 169.1049 0.5972017
##                      ACF1
## Training set 0.0009526163
```

```r
# Residual diagnostics
checkresiduals(fit_arimax)
```




```
##
##  Ljung-Box test
##
## data:  Residuals from Regression with ARIMA(1,1,1)(0,0,1)[4] errors
## Q* = 7.7544, df = 5, p-value = 0.1703
##
## Model df: 3.   Total lags used: 8
```

```r
# Forecast next 7 quarters using test xreg data
fc_arimax <- forecast(fit_arimax, xreg = xreg_test, h = 7)

# Plot forecast with actual values
autoplot(fc_arimax) +
  autolayer(test[,"Consumption"], series = "Actual") +
  labs(title = "Forecast using ARIMA with Income and Unemployment as Regressors") +
  theme_minimal()
```




```r
# Forecast accuracy on test set
accuracy(fc_arimax, test[,"Consumption"])
```

```
##                       ME      RMSE       MAE      MPE      MAPE      MASE
## Training set -0.02826937 0.5114767 0.3903500 29.05286 169.10494 0.5972017
## Test set      0.15550625 0.2374365 0.1687326 19.30715  21.77719 0.2581463
##                       ACF1 Theil's U
## Training set  0.0009526163        NA
## Test set     -0.0077286118 0.8029547
```