ISA 444: Business Forecasting

27: Combining Regression with ARIMA

Fadel M. Megahed, PhD

Endres Associate Professor
Farmer School of Business
Miami University

@FadelMegahed
fmegahed
fmegahed@miamioh.edu
? Automated Scheduler for Office Hours

Spring 2023

Quick Refresher from Last Class

- ✅ Use a simple linear regression model for trend adjustment (time-series data).
- ✅ Interpret regression diagnostic plots.
- ✅ Create prediction intervals for individual values of the response variable.
- ✅ Use regression to **account (?)** for seasonality in a time series.

Interpreting the Model from Last Class

10:00

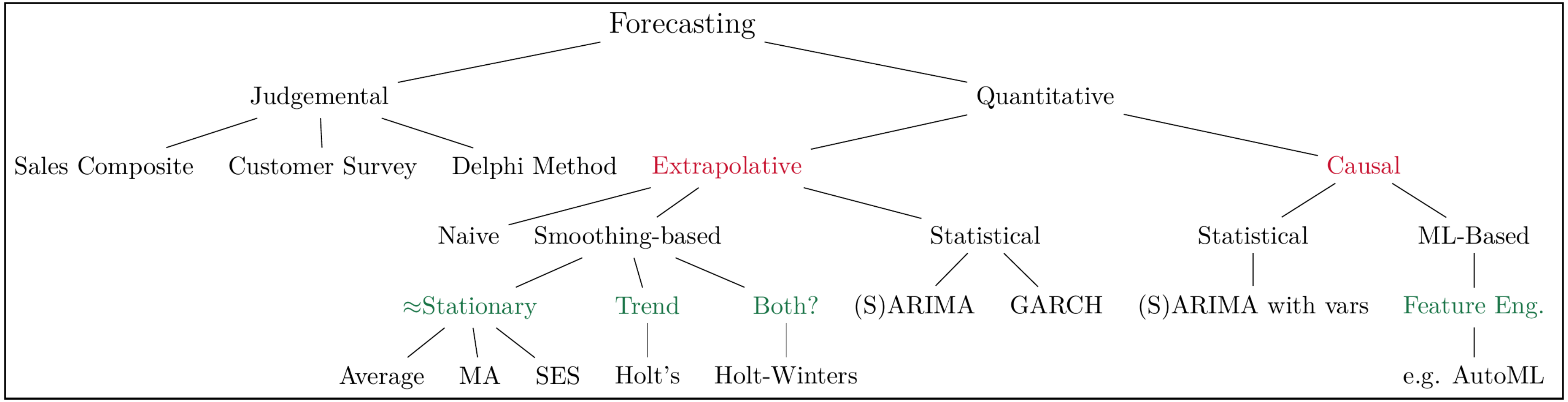
```r
log_jj = log(astsa::jj)

# extracting time and quarter
t = time(log_jj)
q = cycle(log_jj) |> factor()

model3 = lm(log_jj ~ t + q)

summary(model3)
```

From the output, what are the:

- Regression equation: ...
- Intercept interpretation: ...
- The predicted value for logged EPS at 1964 Q1: ...
- Interpretation of 0.02812 coefficient for Q2: ...
- The baseline for Q3 coefficient: ...

```
##
## Call:
## lm(formula = log_jj ~ t + q)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.29318 -0.09062 -0.01180  0.08460  0.27644
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -3.283e+02  4.451e+00 -73.761  < 2e-16 ***
## t            1.672e-01  2.259e-03  73.999  < 2e-16 ***
## q2           2.812e-02  3.870e-02   0.727   0.4695
## q3           9.823e-02  3.871e-02   2.538   0.0131 *
## q4          -1.705e-01  3.873e-02  -4.403 3.31e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1
##
## Residual standard error: 0.1254 on 79 degrees of freedom
## Multiple R-squared:  0.9859,    Adjusted R-squared:  0.9
## F-statistic:  1379 on 4 and 79 DF,  p-value: < 2.2e-16
```

Overview of Univariate Forecasting Methods



A 10,000 foot view of forecasting techniques

Notes: My (incomplete) classification of **univariate** forecasting techniques, i.e., they exclude popular approaches used in multivariate time series forecasting.

Learning Objectives for Today's Class

- Combine regression with ARIMA models to model a time series with autocorrelated errors.
- Use the `xreg` argument to combine ARIMA models with regression predictors.

Combining Regression with ARIMA Models

Preface

- We have learned to fit ARIMA models to predict a series from itself, **removing the autocorrelation from a series**.
    - These methods are useful, but don't allow us to combine **outside** information to boost our forecast.
- Now we will discuss combining regression (**outside information**) and ARIMA models (**inside information**) to forecast a time series.
- A multiple regression model takes on the form:

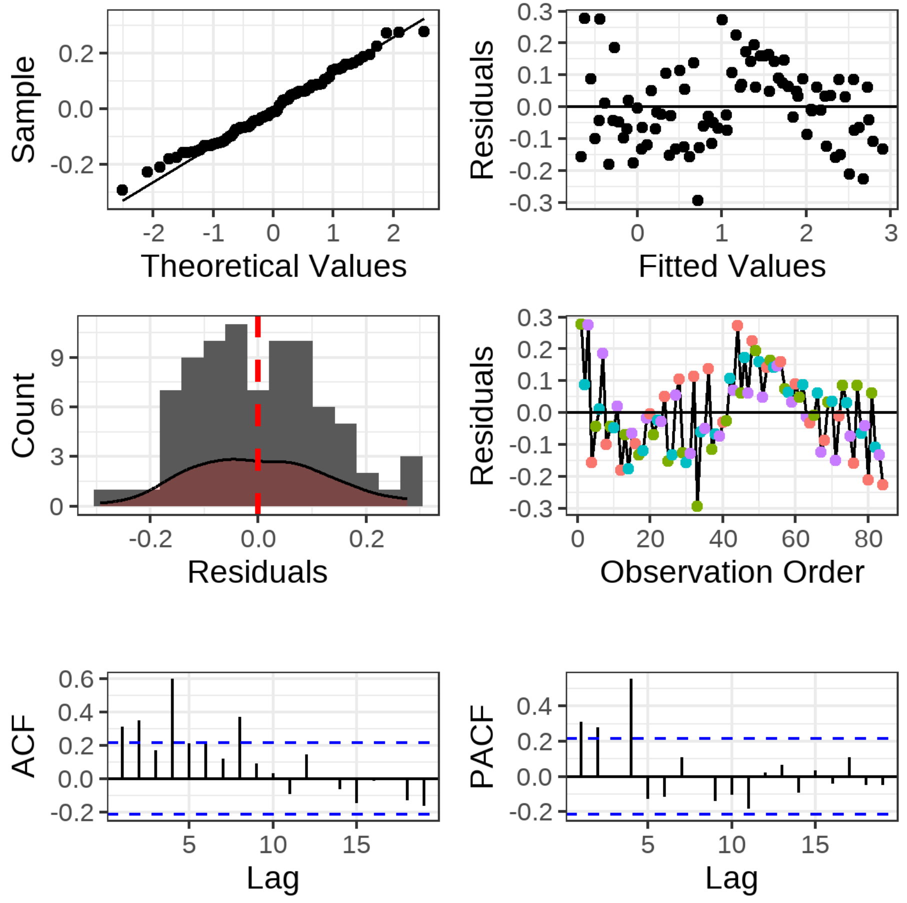
$$Y_t = \beta_0 + \beta_1 x_{t,1} + \beta_2 x_{t,2} + \cdots + \beta_q x_{t,q} + \epsilon_t,$$

where we typically assume that ϵ_t is independent, identically distributed white noise (normally distributed).

What we Learned from the J&J Example

```r
log_jj = log(astsa::jj)

t = time(log_jj)
q = cycle(log_jj) |> factor()

model3 = lm(log_jj ~ t + q)

resplot(
  res = model3$residuals,
  fit = model3$fitted.values,
  freq= 4)
```

- **Based on the J&J Example**, when our dependent and independent variables are observed over time, ϵ_t **is often correlated over time**.
- *In such cases, the assumptions of iid residuals are not met and the fitted regression models should not be used.*



One Possible Solution

We will restate our model as follows: $Y_t = \beta_0 + \beta_1 x_{t,1} + \beta_2 x_{t,2} + \cdots + \beta_q x_{t,q} + \eta_t$,

where η_t **follows an ARIMA model.** When we model η_t, there will be errors from this model, denoted as ϵ_t. Thus, we have the errors from the regression, η_t, and the errors from the ARIMA model, denoted as ϵ_t. Only the errors from the ARIMA, η_t are iid white noise.

If we are using an AR(1) model for the residuals, in this case our model would look as follows

$$Y_t = \beta_0 + \beta_1 x_{t,1} + \beta_2 x_{t,2} + \cdots + \beta_q x_{t,q} + \eta_t,$$

where $\eta_t = \delta + \phi_1 \eta_{t-1} + \epsilon_t$ **follows an AR(1) model.**

If we were to use an ARMA(1,1) model for the residuals, in this case our model would look as follows

$$Y_t = \beta_0 + \beta_1 x_{t,1} + \beta_2 x_{t,2} + \cdots + \beta_q x_{t,q} + \eta_t,$$

where $\eta_t = \delta + \phi_1 \eta_{t-1} + \epsilon_t - \theta_1 \epsilon_{t-1}$ **follows an ARMA(1,1) model.**

R Implementation

We will use the “uschange” dataset from the `fpp2` package to forecast changes in personal consumption expenditures based on personal disposable income from 1970 to 2016.

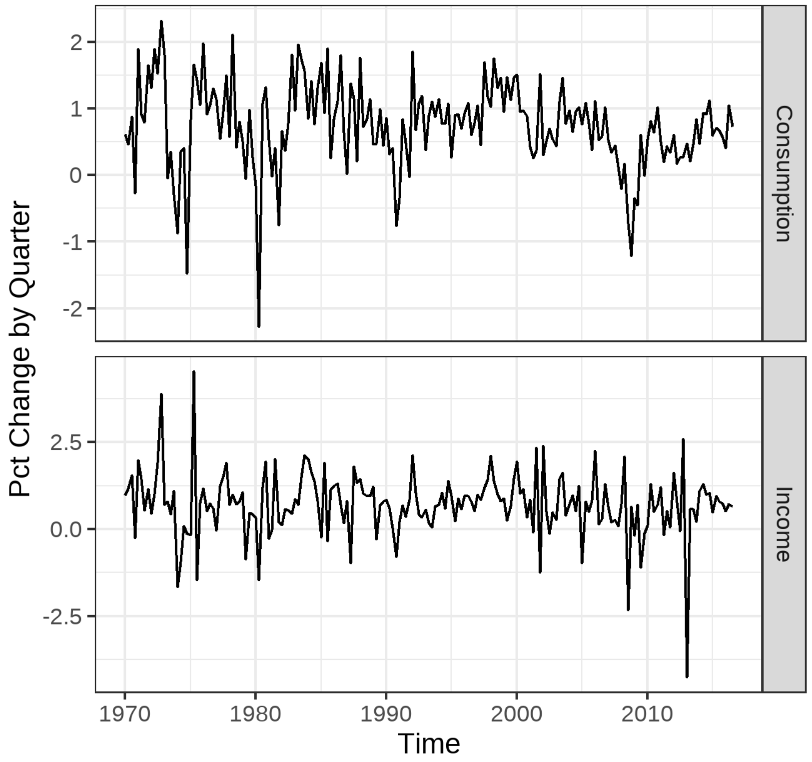
Process:

(1) Start by plotting the quarterly changes in US consumption and personal income
(2) Fit a regression with Y = change in consumption and X = Change in personal income, with autocorrelated errors -- using the `auto.arima()`. The **new part** is that we would be using the argument `xreg` for the predictor/explanatory variables.
(3) Check the residual plots to ensure the assumptions of the model are met.

The Example

```
uschange = fpp2::uschange

class(uschange)

forecast::autoplot(
  uschange[, c('Consumption', 'Income')], facets =
) +
  ggplot2::labs(y = "Pct Change by Quarter") +
  ggplot2::theme_bw()
```

```
## [1] "mts"      "ts"       "matrix"
```



The Example

```r
uschange = fpp2::uschange

p = # store plot into an object so it won't print
  forecast::autoplot(
  uschange[, c('Consumption', 'Income')], facets = 
  ) +
  ggplot2::labs(y = "Pct Change by Quarter") +
  ggplot2::theme_bw()

model1 =
  forecast::auto.arima(
    uschange[,'Consumption'],
    xreg = uschange[, 'Income']
    )

summary(model1)
```

```
## Series: uschange[, "Consumption"]
## Regression with ARIMA(1,0,2) errors
##
## Coefficients:
##          ar1      ma1     ma2  intercept    xreg
##       0.6922  -0.5758  0.1984     0.5990  0.2028
## s.e.  0.1159   0.1301  0.0756     0.0884  0.0461
##
## sigma^2 = 0.3219:  log likelihood = -156.95
## AIC=325.91   AICc=326.37   BIC=345.29
##
## Training set error measures:
##                       ME      RMSE       MAE
## Training set 0.001714366 0.5597088 0.4209056 27.
##                     ACF1
## Training set 0.006299231
```

From the above example, our regression equation would look like this:

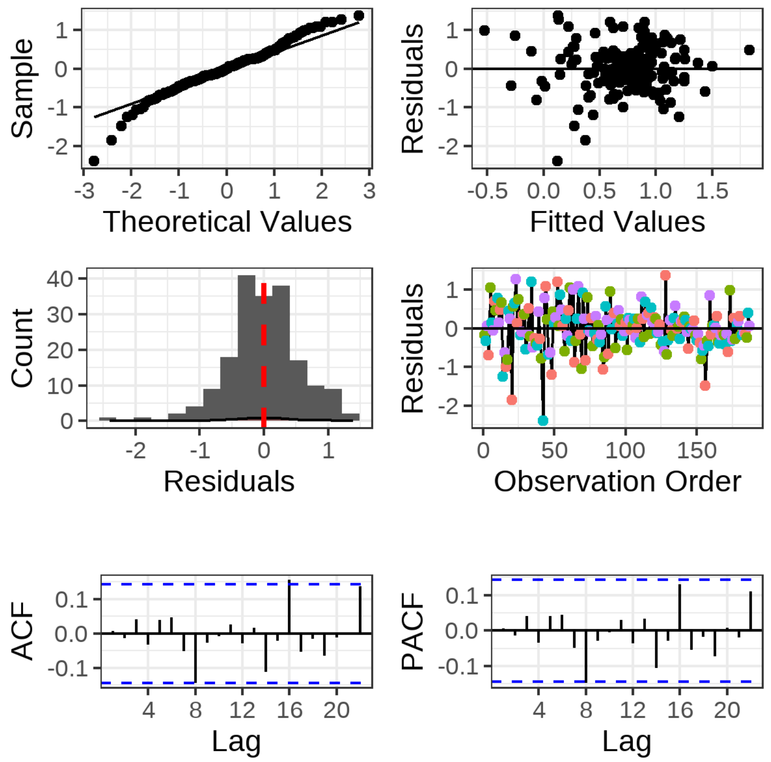
$$Consumption_t = 0.5990 + 0.2028 Income_t + \eta_t,$$

where $\eta_t = \delta + \phi_1 \eta_{t-1} + \epsilon_t + \theta_1 \epsilon_{t-1} + \theta_2 \epsilon_{t-2}$ **follows an ARMA(1,2) model.**

The Example

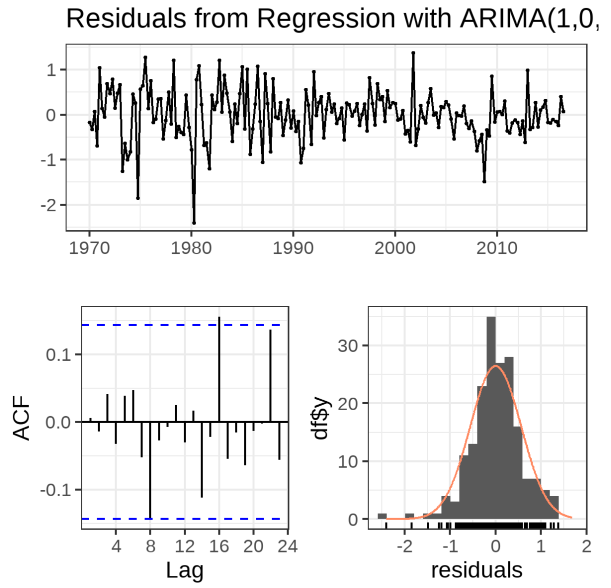
Checking Residuals: Approach 1

```
resplot(res = model1$residuals, fit = model1$fitted, freq =
```



Checking Residuals: Approach 2

```
forecast::checkresiduals(model1)
```




```
##
##      Ljung-Box test
##
## data:  Residuals from Regression with ARIMA(1,0,2) error
## Q* = 5.8916, df = 5, p-value = 0.3169
##
## Model df: 3.   Total lags used: 8
```

Class Activity

10:00

- Compare the results from the class example, with the following four models:
  - `forecast::auto.arima()` using only the Y series. Name this as `model2`.
  - `forecast::auto.arima()` using two explanatory variables ("Income" and "Savings"). Name this as `model3`.
  - `lm()` using both income and savings. Name this as `model4`.
  - `lm()` using income only. Name this as `model5`.
- Which models are suitable? (i.e., the assumptions about the model residuals are met).
- Among the suitable models, pick the best model using the BIC. R function: `BIC()` from base R.
- Predict 1-4 quarters ahead using `model1` (irrespective of whether it is the best model)

Recap

Summary of Main Points

By now, you should be able to do the following:

- Combine regression with ARIMA models to model a time series with autocorrelated errors.
- Use the `xreg` argument to combine ARIMA models with regression predictors.

Things to Do to Prepare for Next Class

- Go through the slides, examples and make sure you have a good understanding of what we have covered.
- Read Chapter 7 in our reference book Principles of Business Forecasting.