ISA 444: Business Forecasting

24: Time Series Regression

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 Couple of Weeks

✅ Explain how ARIMA models work when compared to ARMA models.

✅ Fit an ARIMA model to a time series, evaluate the residuals of a fitted ARMA model to assess goodness of fit, use the Ljung-Box test for correlation among the residuals of an ARIMA model.

✅ Describe AIC, AICc, and BIC and how they are used to measure model fit.

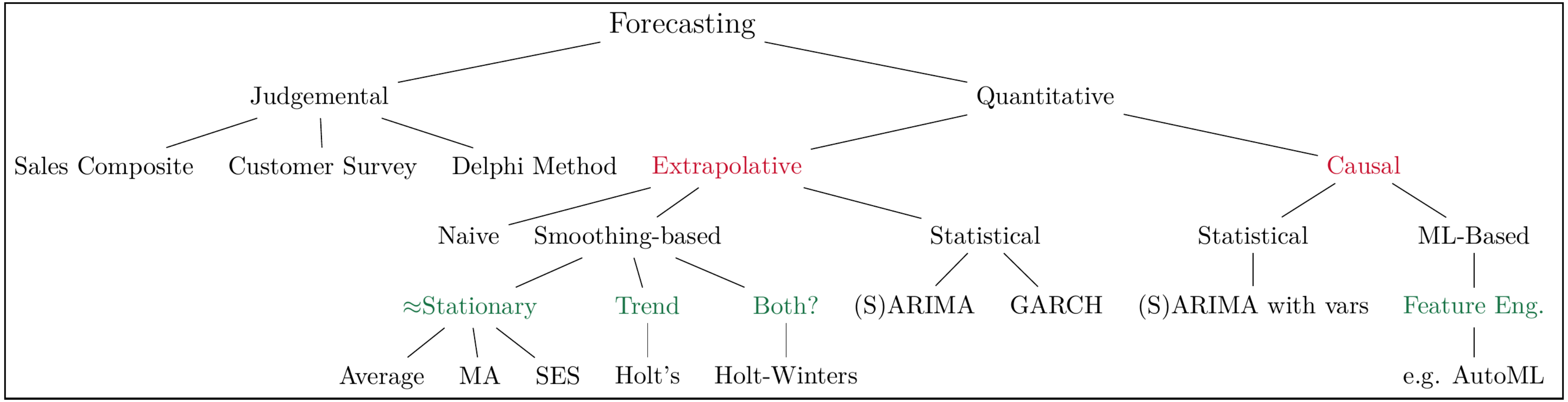
✅ Describe the algorithm used within the `auto.arima()` function to fit an ARIMA model.

✅ Describe the results of the `auto.arima()` function.

✅ Recognize when to fit a seasonal ARIMA model.

✅ Describe a seasonal ARIMA model and explain how it applies to a seasonal time series.

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

- Review Exam 03 and understand any mistakes made.
- Explain the simple and multiple linear regression models and interpret the parameters.
- Interpret the sample linear regression coefficients in the language of the problem.
- Use a simple linear regression model for trend adjustment (time-series data).

Exam 03 Review

Exam 03 Review

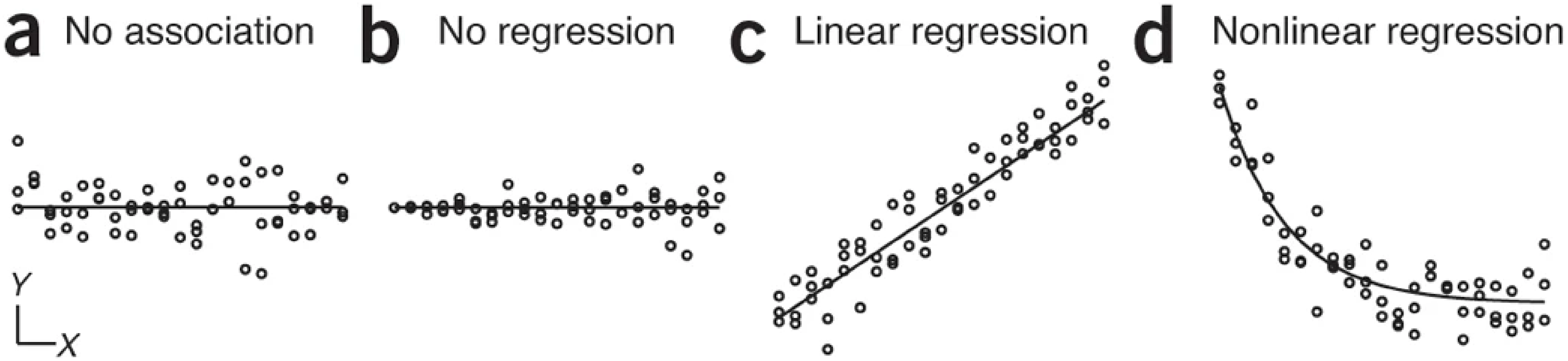
In class, we will go over the exam and talk about each of the questions.

A Review of Simple and Multiple Linear Regression Models

The Simple Linear Regression Model

The simple linear regression model is a population-level model, where we attempt to:

- predict the values of one variable using the values of the other.
- find a 'best line' through the data points.



A variable Y has a regression on variable X if the mean of Y (black line) E(Y|X) varies with X.

Source: Figure is from Altman, N., Krzywinski, M. Simple linear regression. Nature Methods 12, 999–1000 (2015). https://doi.org/10.1038/nmeth.3627.

The Simple Linear Regression Model

To account for time-based values for the response and predictor values, the regression equation can be written as $y_t = \beta_0 + \beta_1 x_{t,1} + \epsilon_t$, where:

- y_t is the **observed value** of our response (a.k.a target, outcome, or dependent) variable at time t.
- β_0 is the expected value (i.e., population mean) of y when $x_{t,1} = 0$. Note that this intercept may not always have a physical meaning.
- $x_{t,1}$ is the **observed value** of our predictor (a.k.a. independent) variable at time t.
- ϵ_t is the random error term. We assume that these $\epsilon_t \sim \mathcal{N}(0, \sigma^2)$.

Multiple Linear Regression with q Predictors

Multiple is used to denote that we have $q \geq 2$ predictors, i.e. $y_t = \beta_0 + \beta_1 x_{t,1} + \beta_2 x_{t,2} + \cdots + \beta_q x_{t,q} + \epsilon_t$, where:

- β_0 is the expected value (i.e., population mean) of y when **all** predictors are set to 0, i.e., $(x_{t,i} = 0, \forall \text{ all } i)$. Note that this often does not have a physical meaning (especially when the number of predictors are large).
- β_i represents the expected change in y_t for one unit increase in predictor i while keeping all other predictors constant.

Using Simple Linear Regression Model for Trend Adjustment (with Time Series Data)

An Example

We will examine an example using the `jj` dataset from the astsa 📦, which captures **Johnson and Johnson's quarterly earnings per share**, 84 quarters (21 years) measured from the first quarter of 1960 to the last quarter of 1980.

```r
if(require(astsa) == F) install.packages('astsa')

jj = astsa::jj

jj
```

```
##           Qtr1      Qtr2      Qtr3      Qtr4
## 1960  0.710000  0.630000  0.850000  0.440000
## 1961  0.610000  0.690000  0.920000  0.550000
## 1962  0.720000  0.770000  0.920000  0.600000
## 1963  0.830000  0.800000  1.000000  0.770000
## 1964  0.920000  1.000000  1.240000  1.000000
## 1965  1.160000  1.300000  1.450000  1.250000
## 1966  1.260000  1.380000  1.860000  1.560000
## 1967  1.530000  1.590000  1.830000  1.860000
## 1968  1.530000  2.070000  2.340000  2.250000
## 1969  2.160000  2.430000  2.700000  2.250000
## 1970  2.790000  3.420000  3.690000  3.600000
## 1971  3.600000  4.320000  4.320000  4.050000
## 1972  4.860000  5.040000  5.040000  4.410000
## 1973  5.580000  5.850000  6.570000  5.310000
## 1974  6.030000  6.390000  6.930000  5.850000
## 1975  6.930000  7.740000  7.830000  6.120000
## 1976  7.740000  8.910000  8.280000  6.840000
## 1977  9.540000 10.260000  9.540000  8.729999
## 1978 11.880000 12.060000 12.150000  8.910000
## 1979 14.040000 12.960000 14.850000  9.990000
## 1980 16.200000 14.670000 16.020000 11.610000
```

An Example

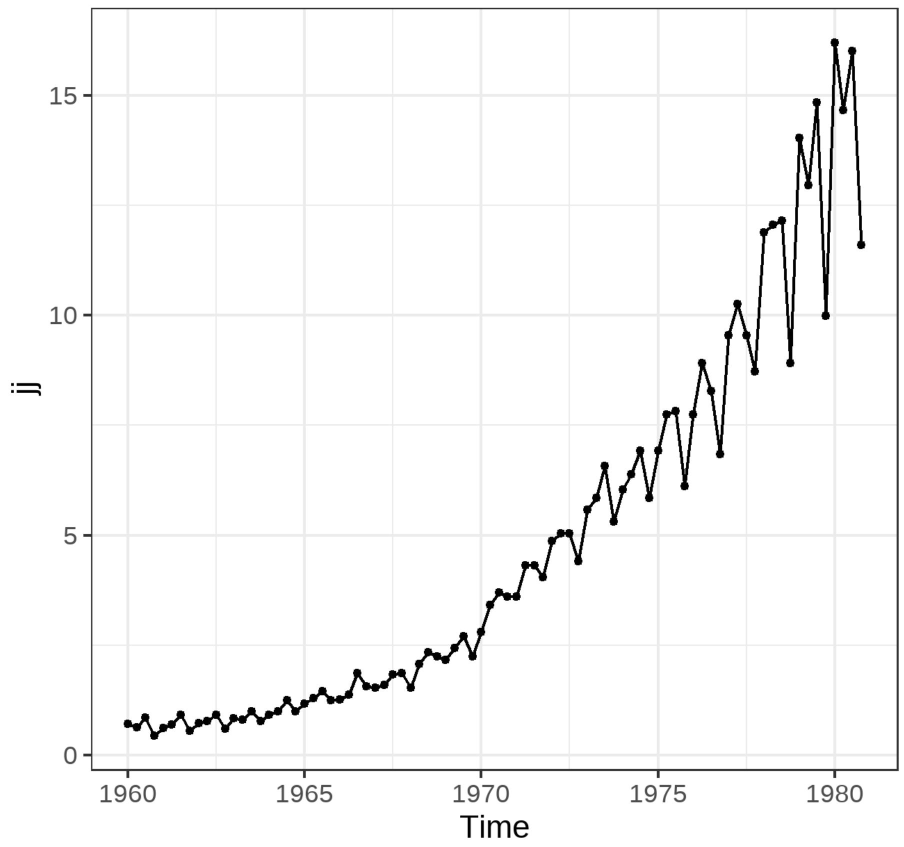
03:00

```r
jj = astsa::jj

# Step 1: Plot the ts data
forecast::autoplot(jj) +
  ggplot2::theme_bw() +
  ggplot2::geom_point(size = 1)
```

What do we learn from this plot?

- Edit me
- ...
- ...



An Example

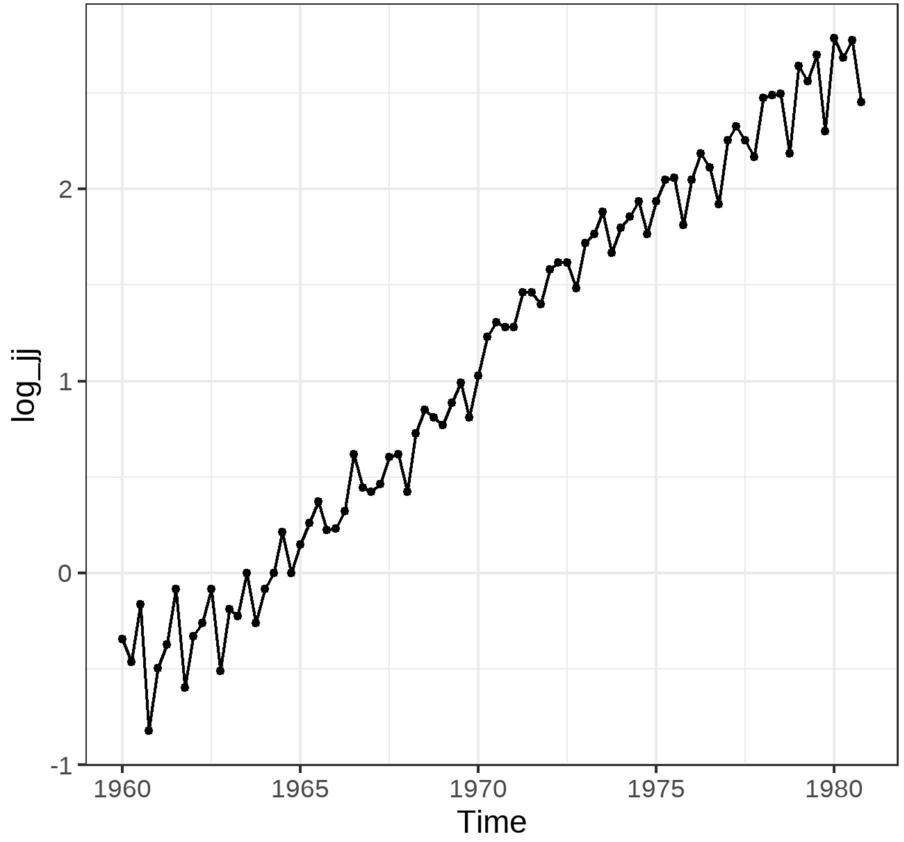
02:00

```
jj = astsa::jj

# Step 1: Plot the ts data (saved to object to not
p = forecast::autoplot(jj) +
  ggplot2::theme_bw() +
  ggplot2::geom_point(size = 1)

# stabilizing the variance
log_jj = log(jj)

# Step 1b: Our updated plot
forecast::autoplot(log_jj) +
  ggplot2::theme_bw() +
  ggplot2::geom_point(size = 1)
```



What do we learn from this plot?

- Edit me
- ...

An Example

```r
jj = astsa::jj

# Step 1: Plot the ts data (saved to object to not
p = forecast::autoplot(jj) +
  ggplot2::theme_bw() +
  ggplot2::geom_point(size = 1)

# stabilizing the variance
log_jj = log(jj)

# Step 1b: Our updated plot
p2 = forecast::autoplot(log_jj) +
  ggplot2::theme_bw() +
  ggplot2::geom_point(size = 1)

# Step 2: Extract time
year = time(log_jj)

# Step 3: Fit the regression model
reg_model = lm(log_jj ~ year)

summary(reg_model)
```

```
##
## Call:
## lm(formula = log_jj ~ year)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.38309 -0.08569  0.00297  0.09984  0.38016
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t
## (Intercept) -3.275e+02  5.623e+00  -58.25   <2e-
## year         1.668e-01  2.854e-03   58.45   <2e-
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05
##
## Residual standard error: 0.1585 on 82 degrees of
## Multiple R-squared:  0.9766,    Adjusted R-squar
## F-statistic:  3416 on 1 and 82 DF,  p-value: < 2
```

An Example

06:00

Based on the previous output, please fill in the following:

- The sample regression eq: $\hat{y} =$ .... + ...(Year)
- Your predicted values for Q3 of 1972: ...
- Residual for Q3 of 1972: ....
- Comments on the intercept term: ....
- Slope interpretation: ....
- R2 interpretation: ....

An Example

How to quickly identify the fitted values and residuals for Q3 of 1972 from the fitted model?

Fitted Values

```r
ts(reg_model$fitted.values, start = c(1960, 1), fre
```

```
##              Qtr1          Qtr2          Qtr3
## 1960 -0.6260763792 -0.5843772017 -0.5426780242 -
## 1961 -0.4592796691 -0.4175804916 -0.3758813141 -
## 1962 -0.2924829590 -0.2507837815 -0.2090846040 -
## 1963 -0.1256862489 -0.0839870714 -0.0422878939 -
## 1964  0.0411104612  0.0828096387  0.1245088162
## 1965  0.2079071712  0.2496063488  0.2913055263
## 1966  0.3747038813  0.4164030589  0.4581022364
## 1967  0.5415005914  0.5831997689  0.6248989465
## 1968  0.7082973015  0.7499964790  0.7916956566
## 1969  0.8750940116  0.9167931891  0.9584923666
## 1970  1.0418907217  1.0835898992  1.1252890767
## 1971  1.2086874318  1.2503866093  1.2920857868
## 1972  1.3754841419  1.4171833194  1.4588824969
## 1973  1.5422808520  1.5839800295  1.6256792070
## 1974  1.7090775620  1.7507767396  1.7924759171
## 1975  1.8758742721  1.9175734497  1.9592726272
## 1976  2.0426709822  2.0843701597  2.1260693373
## 1977  2.2094676923  2.2511668698  2.2928660474
## 1978  2.3762644024  2.4179635799  2.4596627574
```

Residuals

```r
ts(reg_model$residuals, start = c(1960, 1), frequen
```

```
##              Qtr1         Qtr2         Qtr3
## 1960  0.283586070  0.122341742  0.380159095 -0.3
## 1961 -0.035016653  0.046516810  0.292499705 -0.2
## 1962 -0.036021108 -0.010580983  0.125702995 -0.3
## 1963 -0.060643329 -0.139156480  0.042287894 -0.2
## 1964 -0.124492070 -0.082809639  0.090602563 -0.1
## 1965 -0.059487166  0.012757916  0.080258030 -0.1
## 1966 -0.143592160 -0.094319560  0.162474251 -0.0
## 1967 -0.116232856 -0.119465753 -0.020582980 -0.0
## 1968 -0.283029566 -0.022447872  0.058455273 -0.0
## 1969 -0.104985790 -0.028901932  0.034759406 -0.1
## 1970 -0.015849126  0.146050652  0.180337381  0.1
## 1971  0.072246414  0.212868793  0.171169615  0.0
## 1972  0.205554296  0.200222763  0.158523585 -0.0
## 1973  0.176907924  0.182461632  0.256834625  0.0
## 1974  0.087669449  0.103957529  0.143383896 -0.0
## 1975  0.059985541  0.128828238  0.098689883 -0.1
## 1976  0.003730705  0.102804082 -0.012226369 -0.2
## 1977  0.046025793  0.077085970 -0.037372562 -0.1
## 1978  0.098591912  0.071930611  0.037666412 -0.3
```

An Example

A Test of Significance of Model Coefficients:

```r
anova(reg_model)
```

```
## Analysis of Variance Table
##
## Response: log_jj
##           Df Sum Sq Mean Sq F value    Pr(>F)
## year       1 85.872  85.872  3416.5 < 2.2e-16 ***
## Residuals 82  2.061   0.025
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Recap

Summary of Main Points

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

- Explain the simple and multiple linear regression models and interpret the parameters.
- Interpret the sample linear regression coefficients in the language of the problem.
- Use a simple linear regression model for trend adjustment (time-series data).

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.