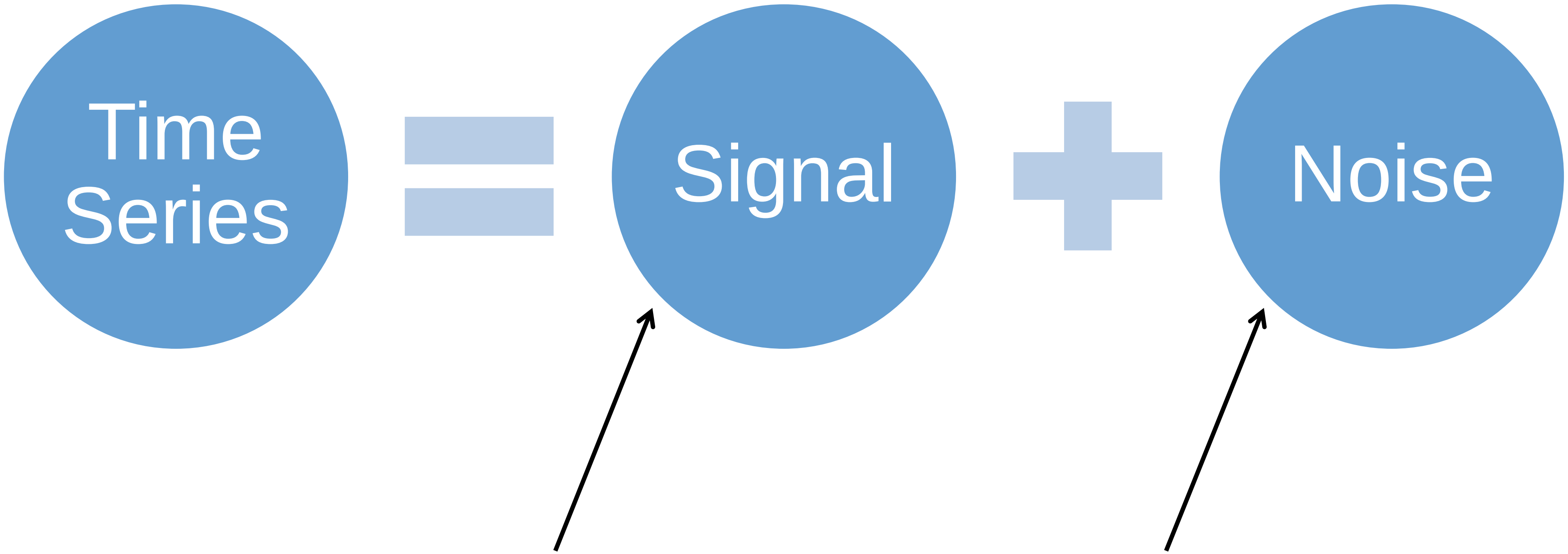
SEASONALITY MODELS

Dr. Aric LaBarr

Institute for Advanced Analytics

QUICK REVIEW

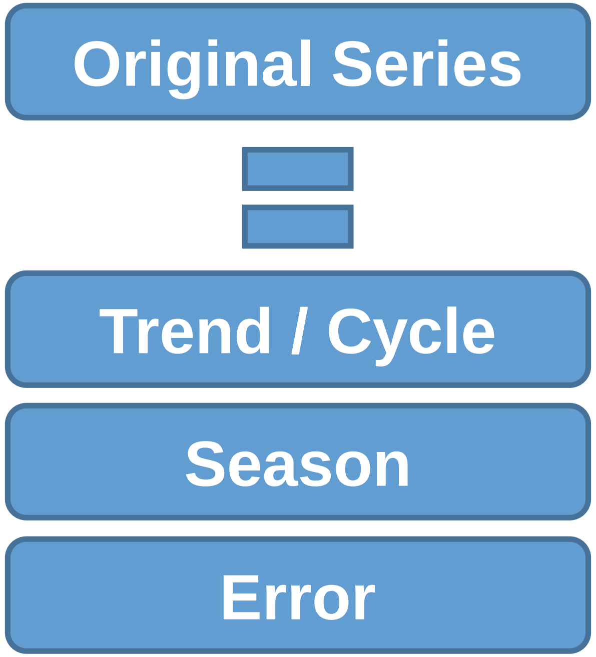
Time Series Data

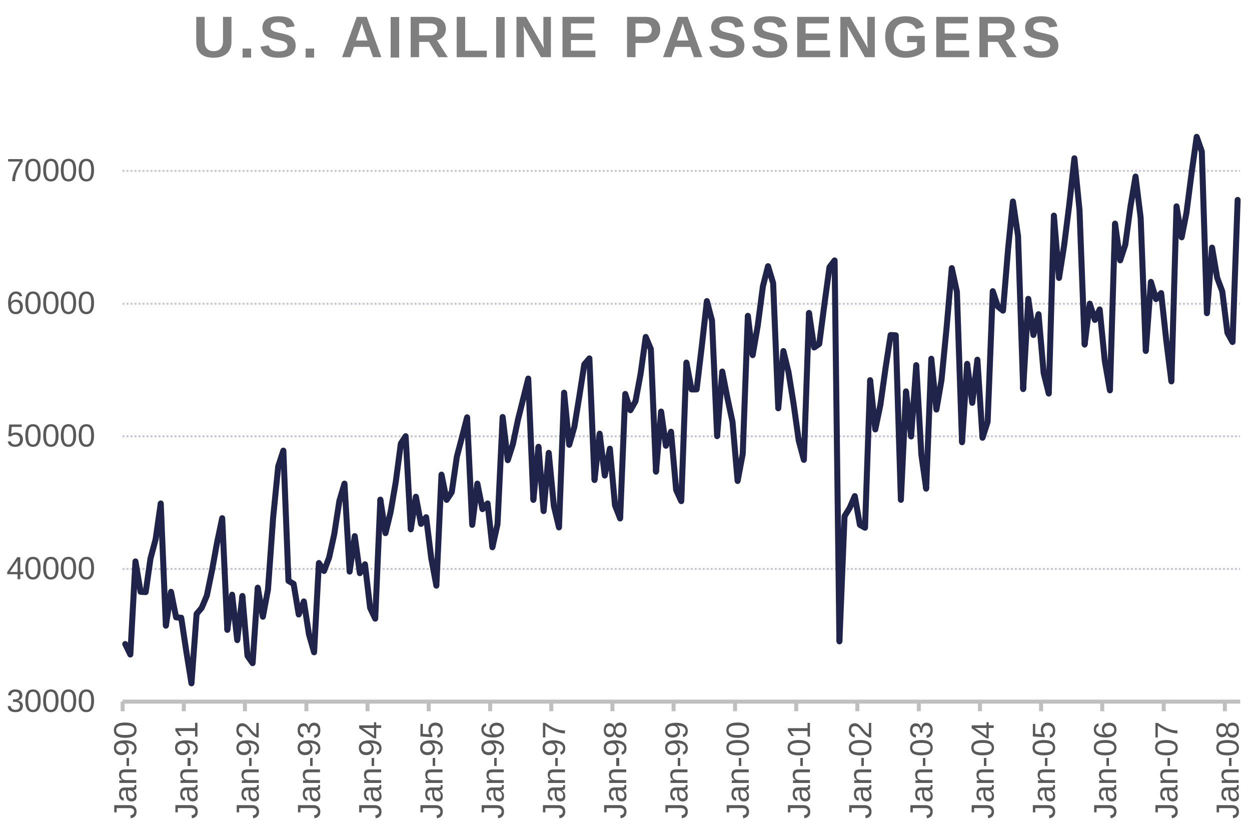


Forecasts extrapolate signal portion of model.

Confidence intervals account for uncertainty.

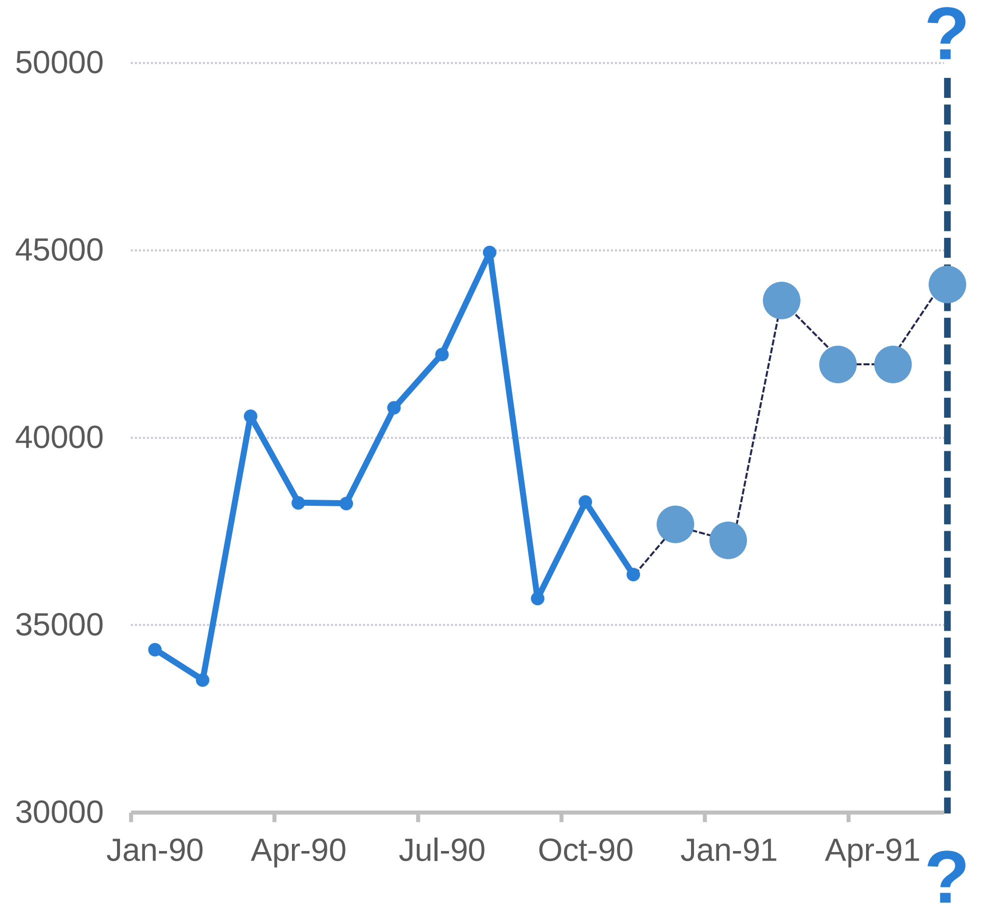
Time Series Data





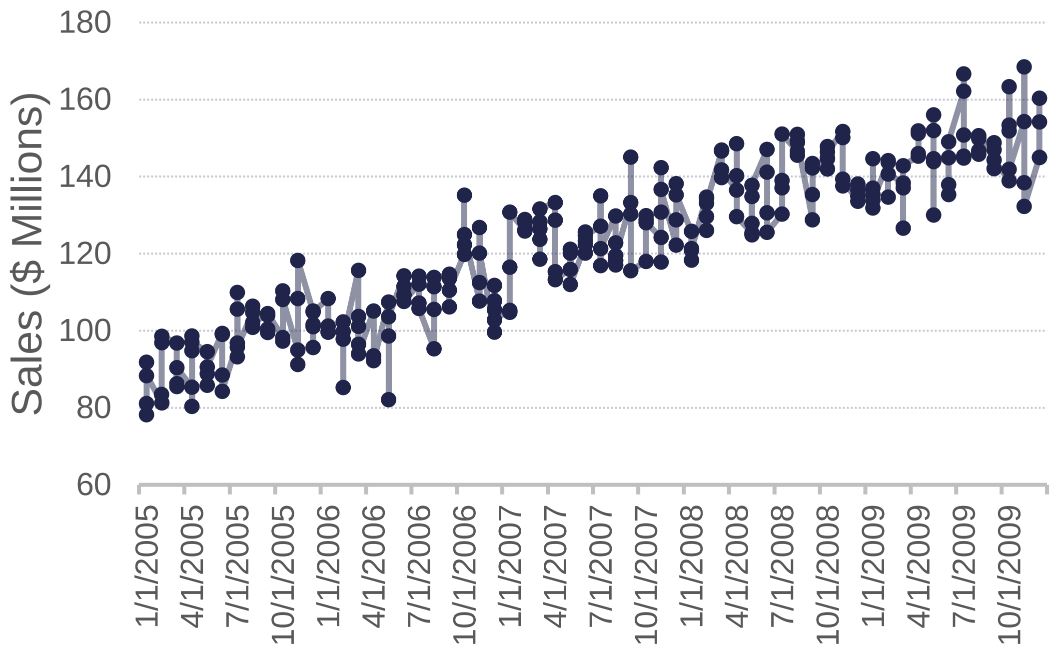
Exponential Smoothing Models

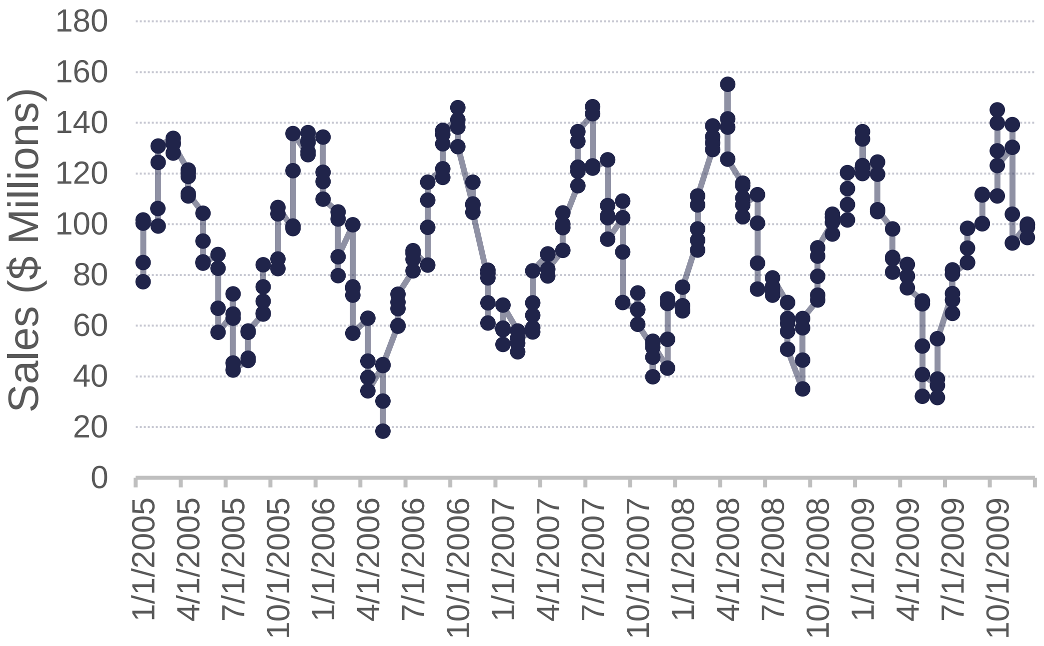
Original Series	$\hat{Y}_{t+k} = L_t + kT_t + S_{t-p+k}$
=	=
Level	$L_t = \theta Y_t + (1-\theta)L_{t-1}$
Trend	$T_t = \gamma(L_t - L_{t-1}) + (1-\gamma)T_{t-1}$
Season	$S_t = \delta(Y_t - L_t) + (1-\delta)S_{t-p}$



Stationarity

- Need consistency of mean and variance.
- What about changes in mean – trending, seasonality? **NOT** stationary.

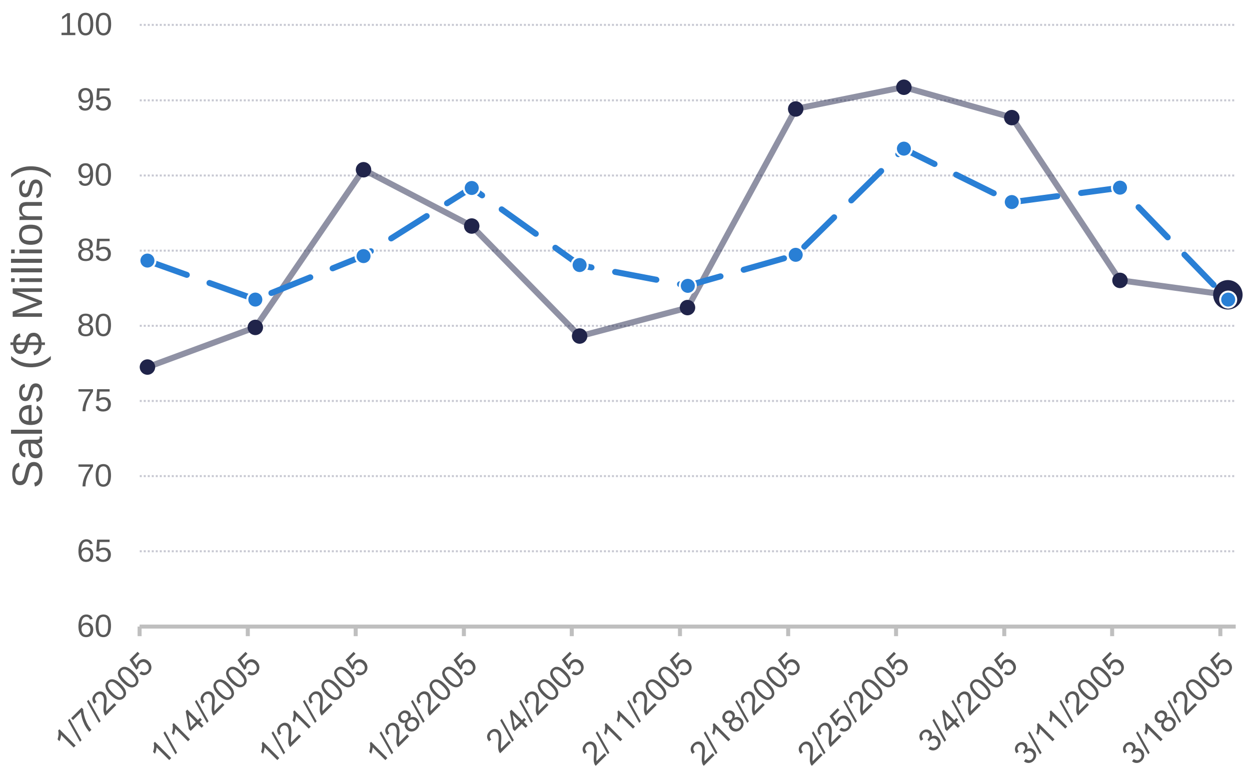




ARIMA Models

- AR – forecast a series based solely on the past values in the series – called **lags.**

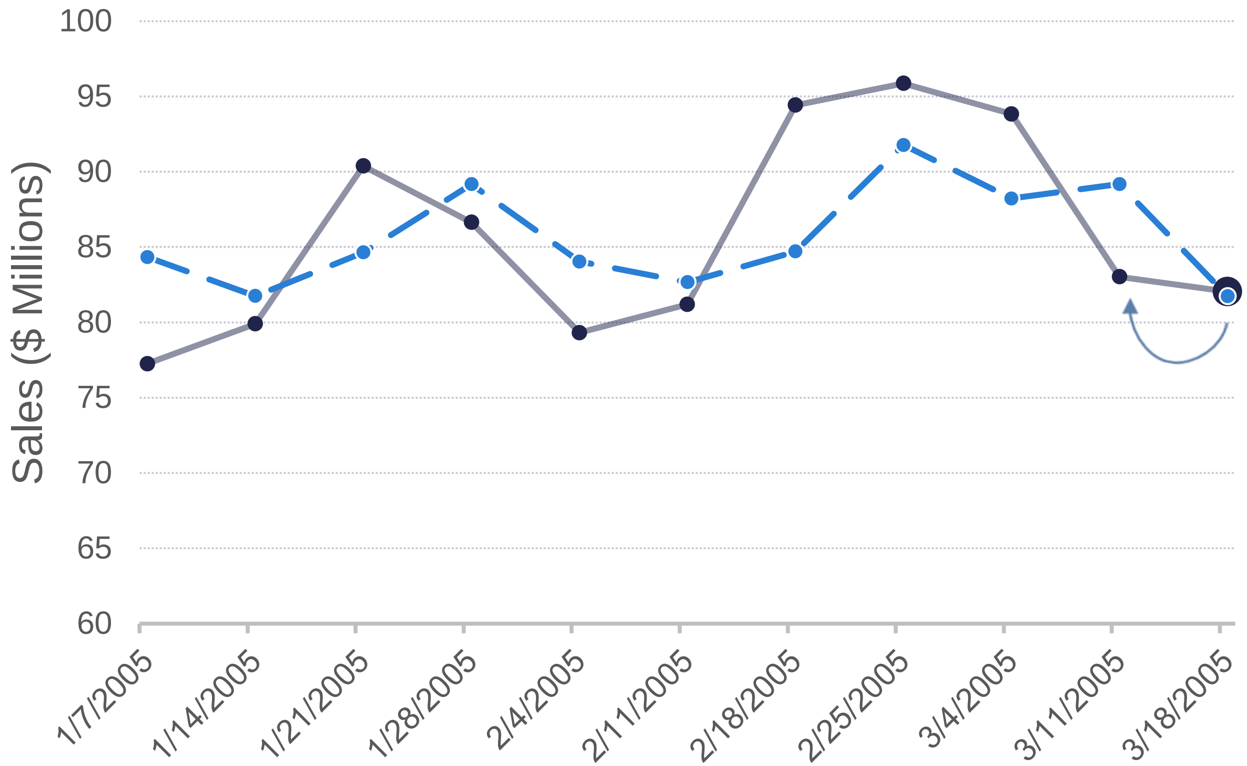
- MA – forecast a series based solely on the past errors in the series – called **error lags**.



ARIMA Models

- AR – forecast a series based solely on the past values in the series – called **lags.**

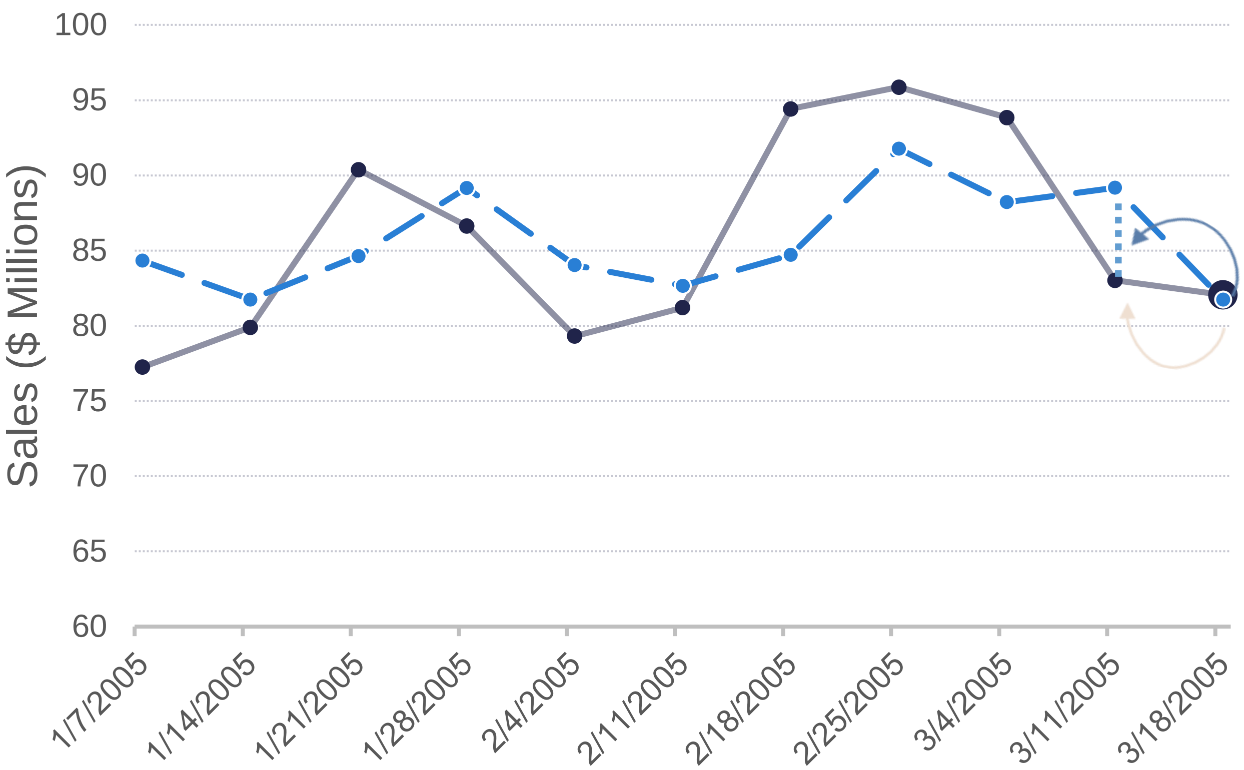
- MA – forecast a series based solely on the past errors in the series – called **error lags.**



ARIMA Models

- AR – forecast a series based solely on the past values in the series – called **lags.**

- MA – forecast a series based solely on the past errors in the series – called **error lags**.



ARIMA Models

- ARIMA Models are typically written as the following:

$$\text{ARIMA}(p, d, q)$$

# of AR Terms

# of MA Terms

# of First Differences

U.S. Airlines Passengers 1990 – 2007

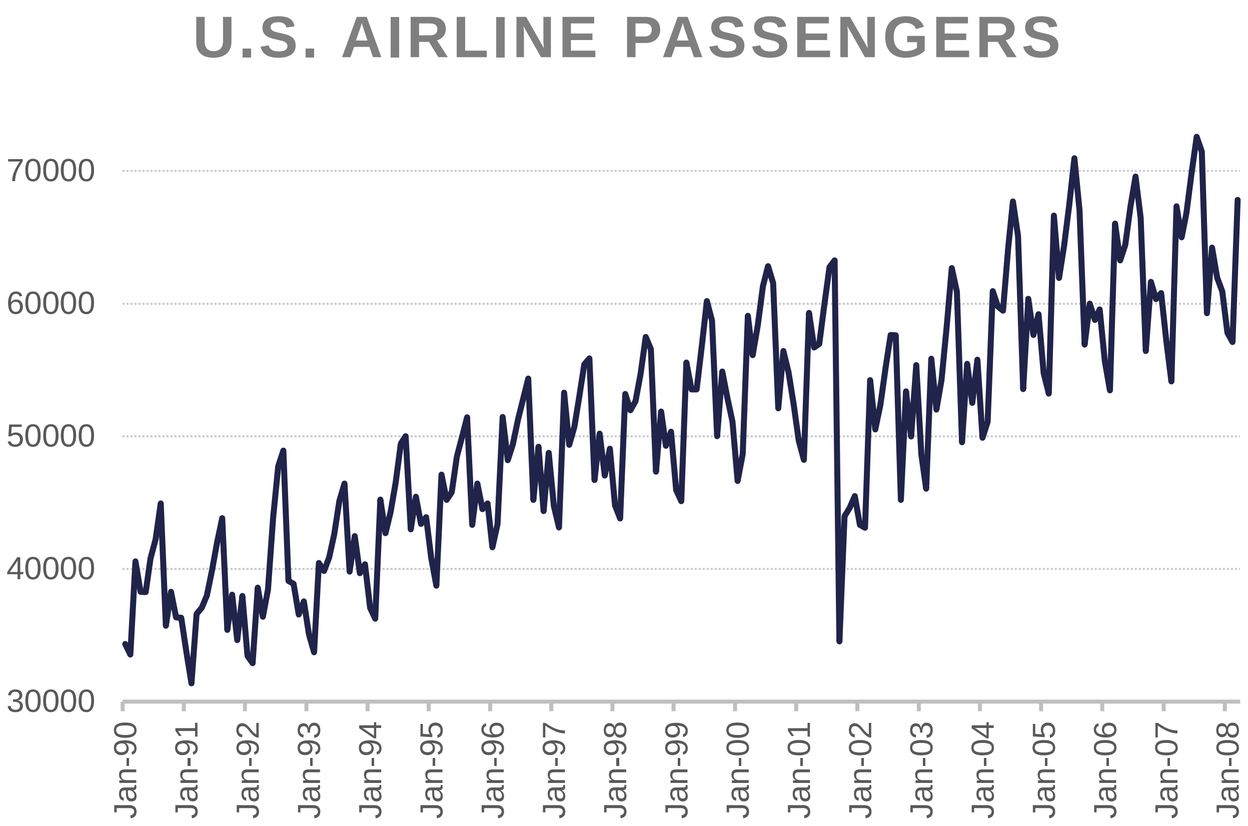
Original Series

=

Trend / Cycle

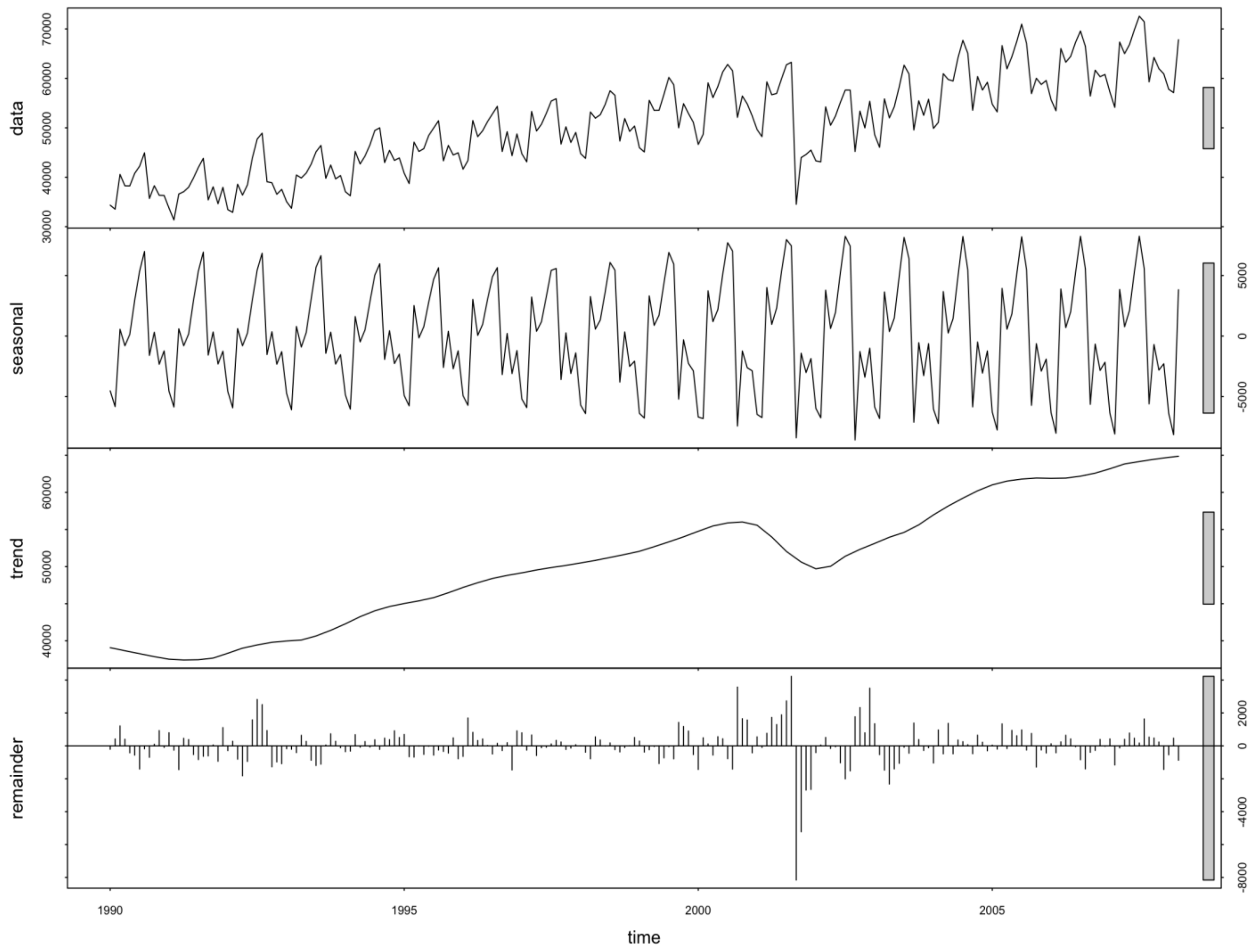
Season

Error



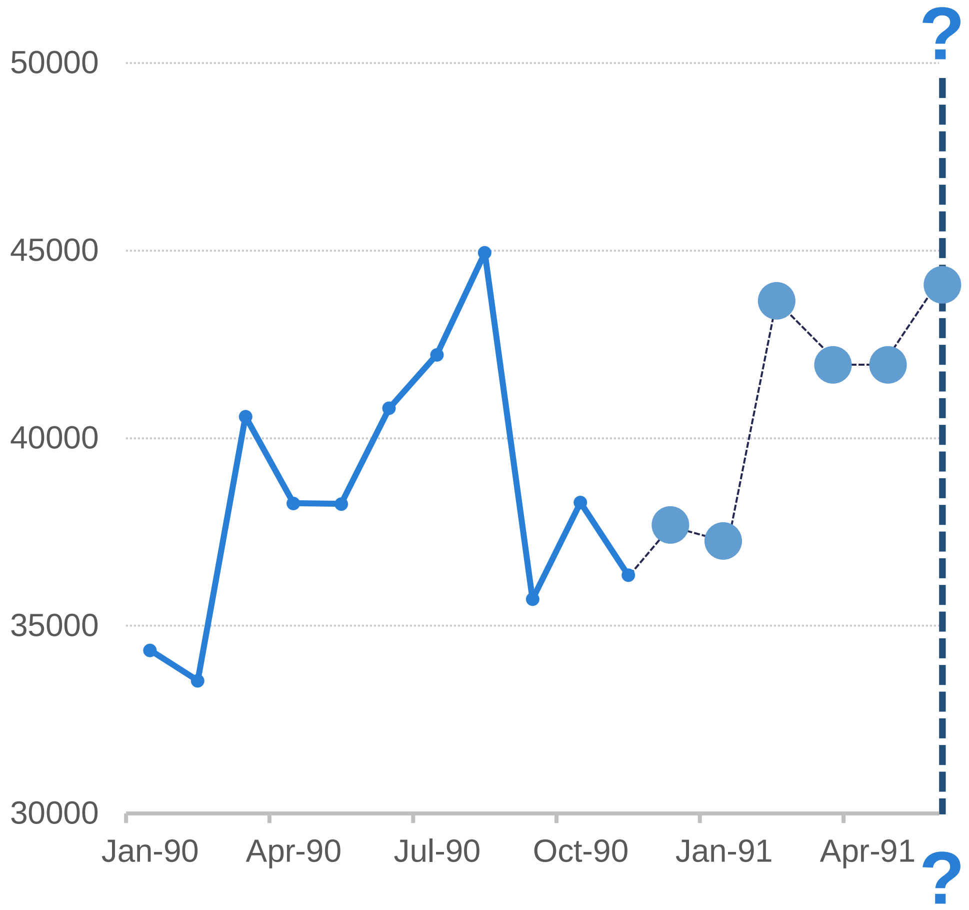
Split into Training and Validation

```
training <- subset(Passenger, end =
length(Passenger)-12)

test <- subset(Passenger, start =
length(Passenger)-11)


decomp_stl <- stl(training,
                  s.window = 7)

plot(decomp_stl)
```



Exponential Smoothing Models

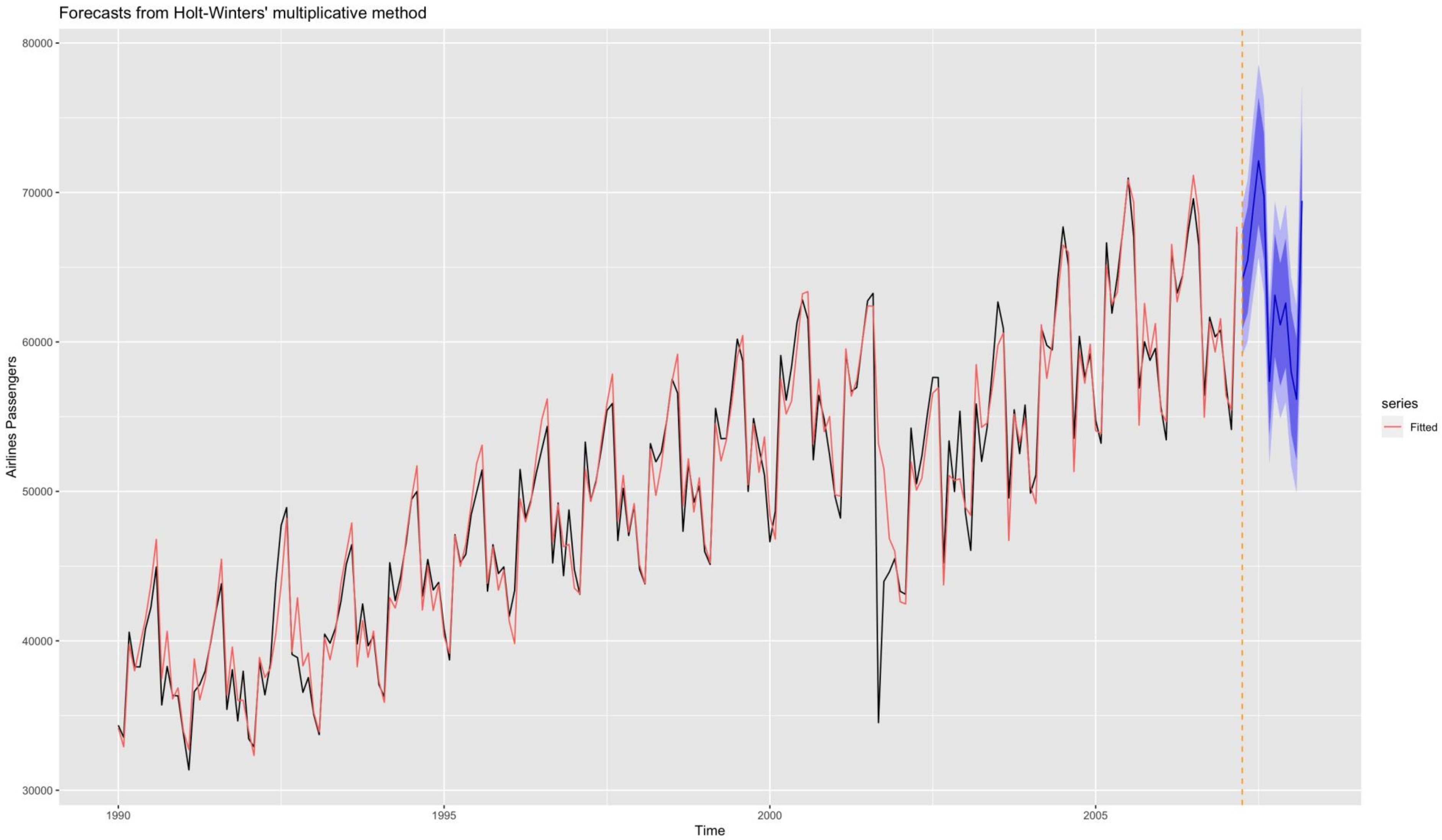
Original Series	$\hat{Y}_{t+k} = L_t + kT_t + S_{t-p+k}$
=	=
Level	$L_t = \theta Y_t + (1-\theta)L_{t-1}$
Trend	$T_t = \gamma(L_t - L_{t-1}) + (1-\gamma)T_{t-1}$
Season	$S_t = \delta(Y_t - L_t) + (1-\delta)S_{t-p}$



Exponential Smoothing Models

```
HWES.USAir.train <- hw(training, seasonal = "multiplicative", initial='optimal',h=12)

autoplot(HWES.USAir.train) +
  autolayer(fitted(HWES.USAir.train), series="Fitted") +
  ylab("Airlines Passengers") +
  geom_vline(xintercept = 2007.25,color="orange",linetype="dashed")

HW.error <- test - HWES.USAir.train$mean

HW.MAE <- mean(abs(HW.error))
HW.MAPE <- mean(abs(HW.error)/abs(test))*100
```

Exponential Smoothing Models



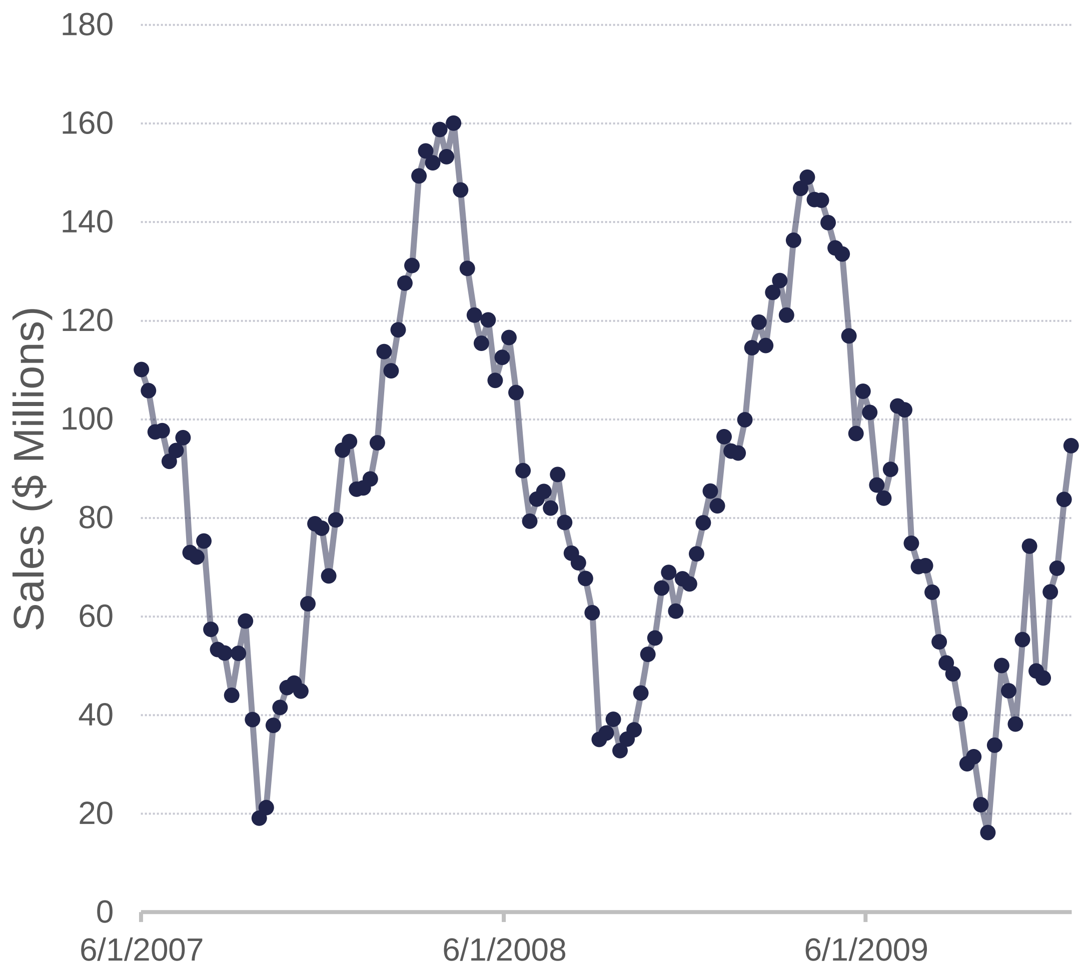
Model Evaluation on Test Data

Model	MAE	MAPE
HW Exponential Smoothing	1134.58	1.76%

SEASONALITY

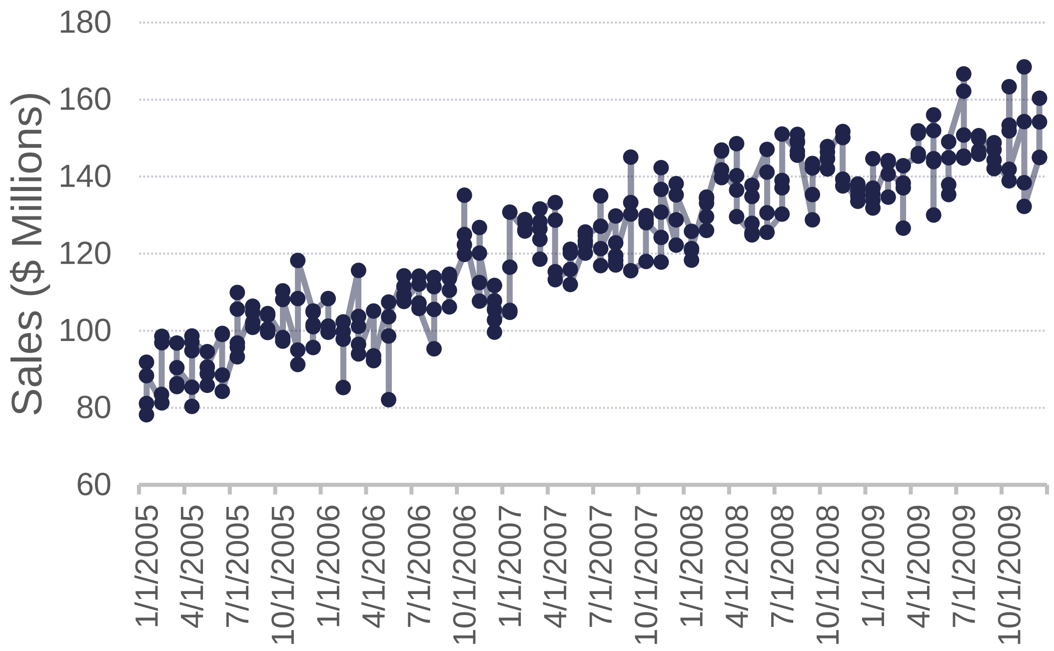
Seasonality

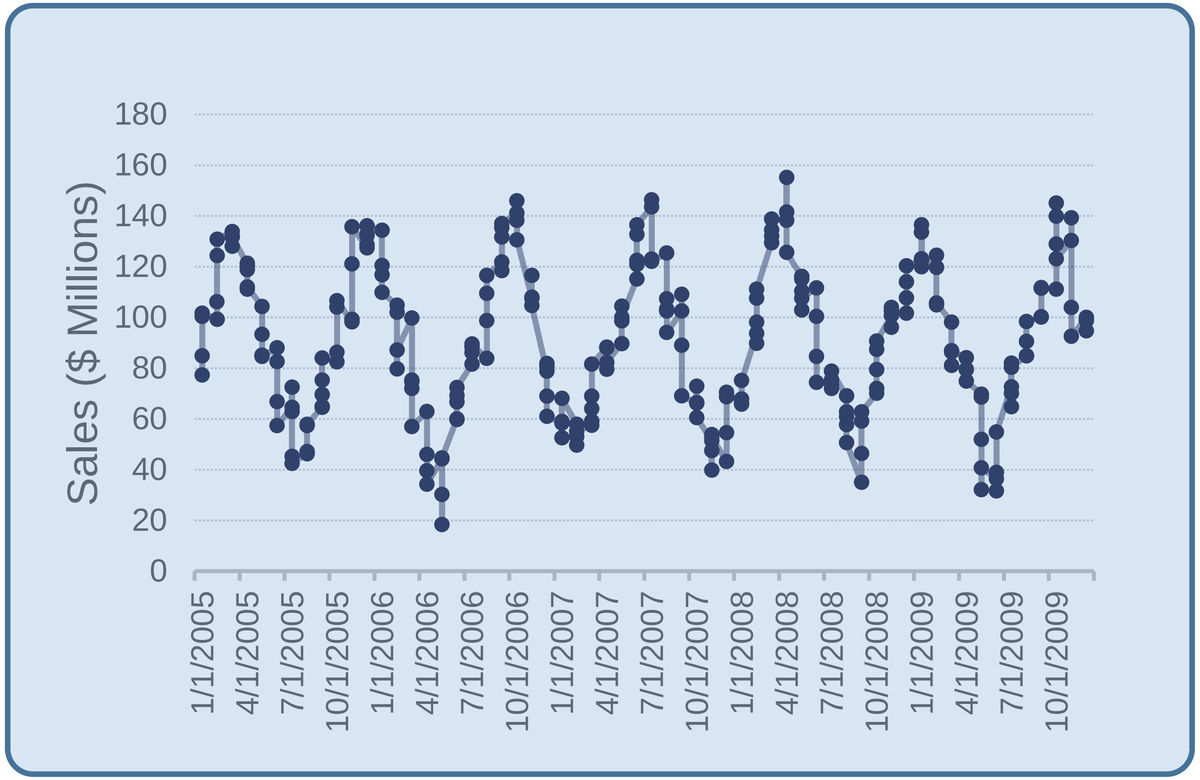
- Seasonality is the component of time series that represents the effects of seasonal variation.
- Component that describes repetitive behavior known as seasonal periods.
  - Seasonal period = S
  - Seasonal factors repeat every S units of time.



Seasonality and Stationarity

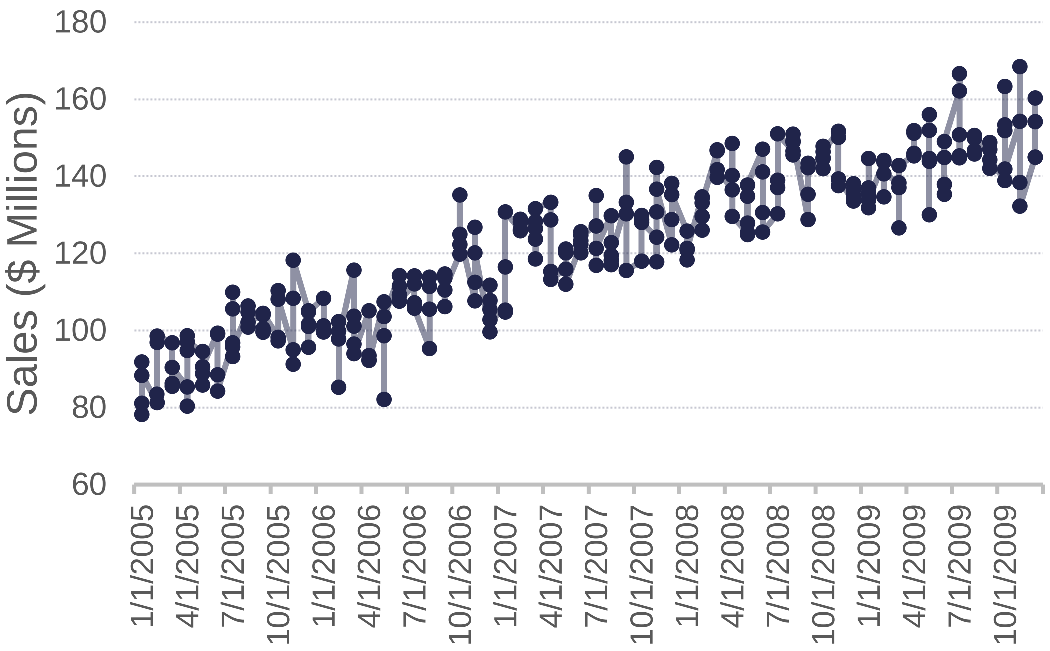
- Need consistency of mean and variance.
- What about changes in mean – trending, seasonality? **NOT** stationary.

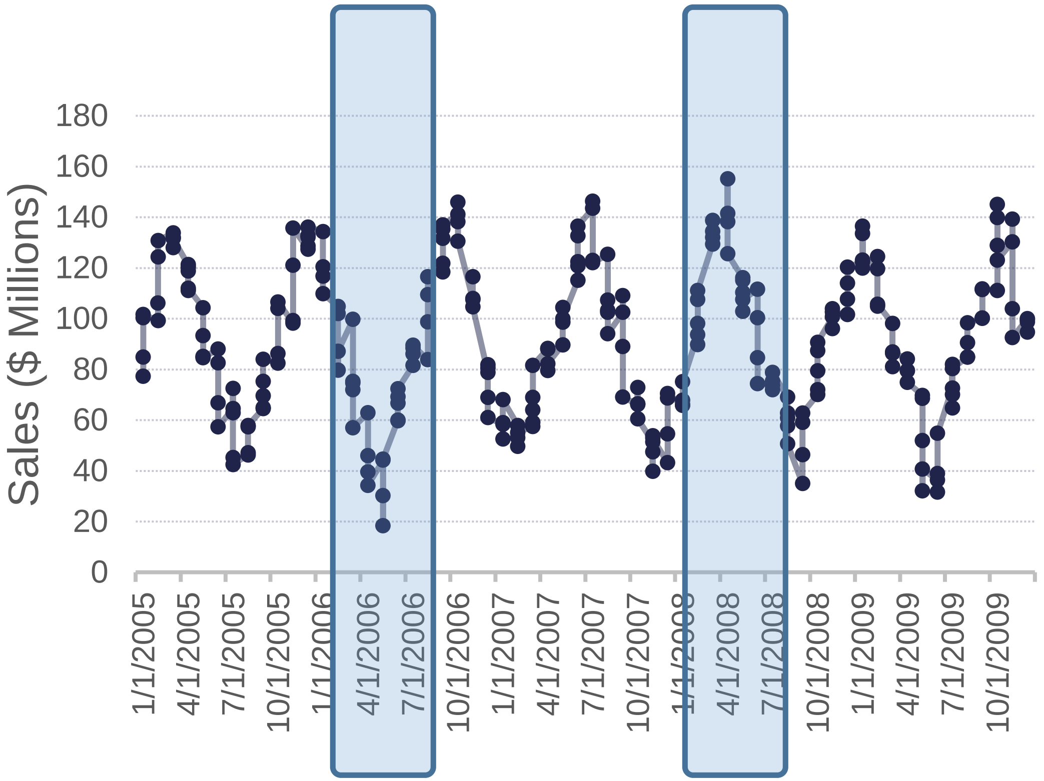




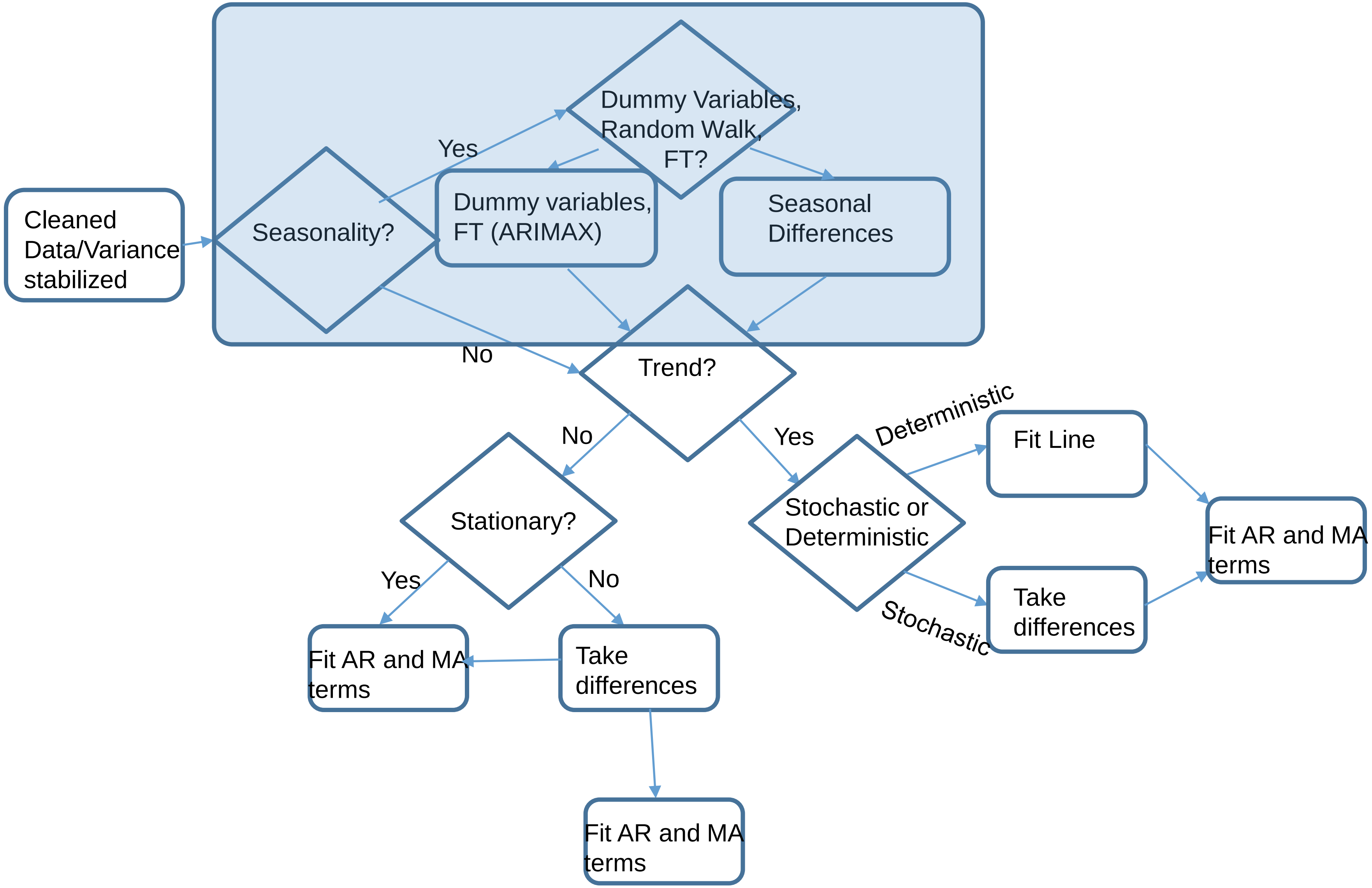
Seasonality and Stationarity

- Need consistency of mean and variance.
- What about changes in mean – trending, seasonality? **NOT** stationary.





ARIMA Framework



Seasonal ARIMA Models

- Similar to trend, seasonality can be solved with a deterministic solution or a stochastic solution.
  - **Deterministic** – Seasonal dummy variables, Fourier transforms, predictor variables
  - **Stochastic** – Seasonal differences

- Once data is made stationary, we can model with traditional ARIMA approaches.

Seasonal Unit-Root Testing

- Similar to trend, we can perform statistical tests (Canova-Hansen test) for evaluating whether a unit root exists for seasonal data.
- Hypotheses:

$$H_0\text{: Deteministic Seasonality (Differencing not going to help)}$$

$$H_a\text{: Stochastic Seasonality (Differencing needed)}$$

Seasonal Unit-Root Testing

```
training %>% nsdiffs()

## [1] 1
```

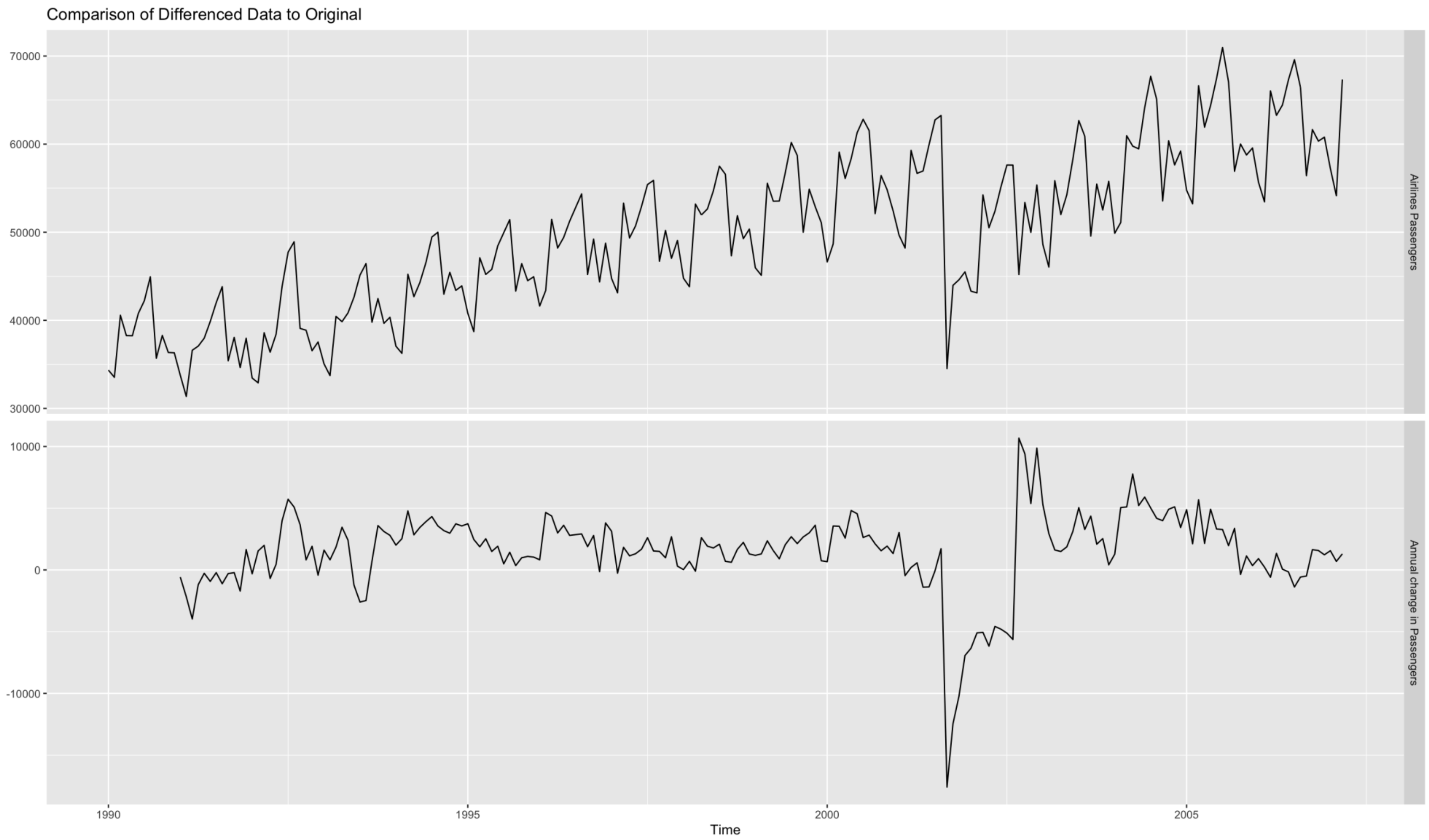
Should take one **seasonal** difference

Seasonal Unit-Root Testing

```
training %>% nsdiffs()
## [1] 1
```

Not really good formal tests for seasons beyond 24 (Dickey-Fuller ends at season of 12)

Differenced Data



Unit-Root Testing

```
training %>% diff(lag = 12) %>% ndiffs()
## [1] 0
```

Should take 0 **regular** differences AFTER taking the seasonal difference

DETERMINISTIC SOLUTIONS

Which Deterministic Solution?

- Similar to trend, seasonality can be solved with a deterministic solution or a stochastic solution.
  - **Deterministic** – Seasonal dummy variables, Fourier transforms, predictor variables
  - **Stochastic** – Seasonal differences

- Once data is made stationary (model away the seasonality), we can model with traditional ARIMA approaches.

Seasonal Dummy Variables

- For a time series with *S* periods within a season, there will be *S-1* dummy variables, one for each period (and one accounted for with the intercept).
- If there is a constant term (intercept), only *S-1* dummy variables are required.
- Monthly Data:
  - One dummy variable for each month (S = 12)
- Weekly Data:
  - One dummy variable for each day of week (S = 7)
- Hourly Data:
  - One dummy variable for each hour (S = 24)

Seasonal Dummy Variables

- Example model with intercept:

$$Y_t = \beta_0 + \beta_1 JAN + \beta_2 FEB + \cdots + \beta_{11} NOV + e_t$$

$$\beta_0 + \beta_M = \text{effect of M}^{\text{th}} \text{ month}$$

$$\beta_0 = \text{effect of December}$$

Seasonal Dummy Variables

```
Month <- rep(0, length(training))
Month <- Month + 1:12

M <- factor(Month)
M <- relevel(M, ref="12")
```

Seasonal Dummy Variables

```
Season.Lin <- lm(training ~ M)
summary(Season.Lin)
```

```
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  48761.5     2002.3  24.353   <2e-16 ***
## M1           -4461.9     2792.1  -1.598   0.1116
## M2           -5433.9     2792.1  -1.946   0.0531 .
## M3            4099.8     2792.1   1.468   0.1436
## M4             814.9     2831.7   0.288   0.7738
## M5            1951.8     2831.7   0.689   0.4915
## M6            4844.6     2831.7   1.711   0.0887 .
## M7            7504.9     2831.7   2.650   0.0087 **
## M8            7297.4     2831.7   2.577   0.0107 *
## M9           -3242.5     2831.7  -1.145   0.2536
## M10           1064.1     2831.7   0.376   0.7075
## M11          -1268.2     2831.7  -0.448   0.6548
## ---
```

Seasonal Dummy Variables

```
M.Matrix <- model.matrix(~M)

Trend <- 1:length(training)

SD.ARIMA <- auto.arima(training, xreg = M.Matrix[,2:12], method="ML", seasonal = FALSE)
summary(SD.ARIMA)
```

Seasonal Dummy Variables

```
## Series: training
## Regression with ARIMA(1,1,1) errors
##
## Coefficients:
##          ar1      ma1     drift          M1          M2          M3          M4
##       0.4292  -0.7971  120.7148  -3947.9347  -5040.318  4373.0410  1776.4821
## s.e.  0.1142   0.0773   47.0832    485.1706    583.269   625.8909   653.3822
##              M5         M6          M7          M8          M9         M10
##       2774.5010  5539.126  8075.0713  7745.7280  -2913.554  1275.9094
## s.e.   664.7169   667.948   665.0425   655.1154    633.936   589.7366
##              M11
##       -1168.1437
## s.e.    486.9715
##
## sigma^2 estimated as 3751114:  log likelihood=-1844.41
## AIC=3718.82   AICc=3721.34   BIC=3768.74
##
## Training set error measures:
##                      ME     RMSE      MAE        MPE     MAPE      MASE
## Training set -13.60989 1865.287 1119.089 -0.1822502 2.399032 0.4156601
##                       ACF1
## Training set -0.002860136
```

Advantages and Disadvantages

Advantages

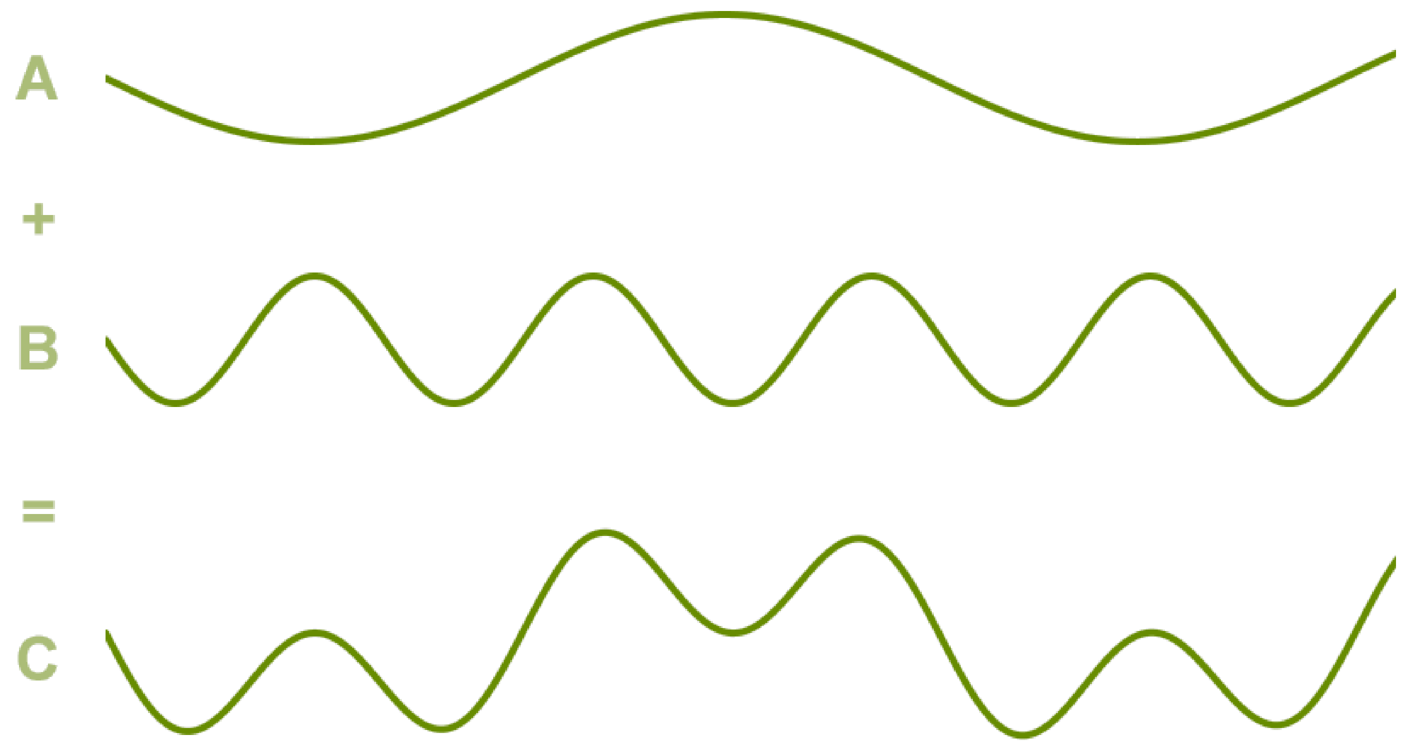
- Interpretation still holds.
  - Can easily measure and interpret effects from different parts of the season.
- Straight forward to implement.

Disadvantages

- Especially long or complex seasons are hard to deal with.
  - More than 24 periods in a season (365 days in year for example) is burdensome.
  - Some seasons are complex (365.25 days in a year, 52.17 weeks in a year, etc.).
- Seasonal effects remain constant.

Fourier Transforms (Harmonic Regression)

- Fourier showed that series of sine and cosine terms of the right frequencies approximate periodic series.



Fourier Transforms (Harmonic Regression)

- Add Fourier variables to a regression model predicting the target to remove the seasonal pattern.

$$X_{1,t} = \sin\left(\frac{2\pi t}{S}\right) \qquad X_{3,t} = \sin\left(2 \times \frac{2\pi t}{S}\right) \qquad X_{5,t} = \sin\left(3 \times \frac{2\pi t}{S}\right) \qquad \cdots$$

$$X_{2,t} = \cos\left(\frac{2\pi t}{S}\right) \qquad X_{4,t} = \cos\left(2 \times \frac{2\pi t}{S}\right) \qquad X_{6,t} = \cos\left(3 \times \frac{2\pi t}{S}\right) \qquad \cdots$$

$$Y_t = \beta_0 + \beta_1 X_{1,t} + \beta_2 X_{2,t} + \beta_3 X_{3,t} + \beta_4 X_{4,t} + \cdots + e_t$$

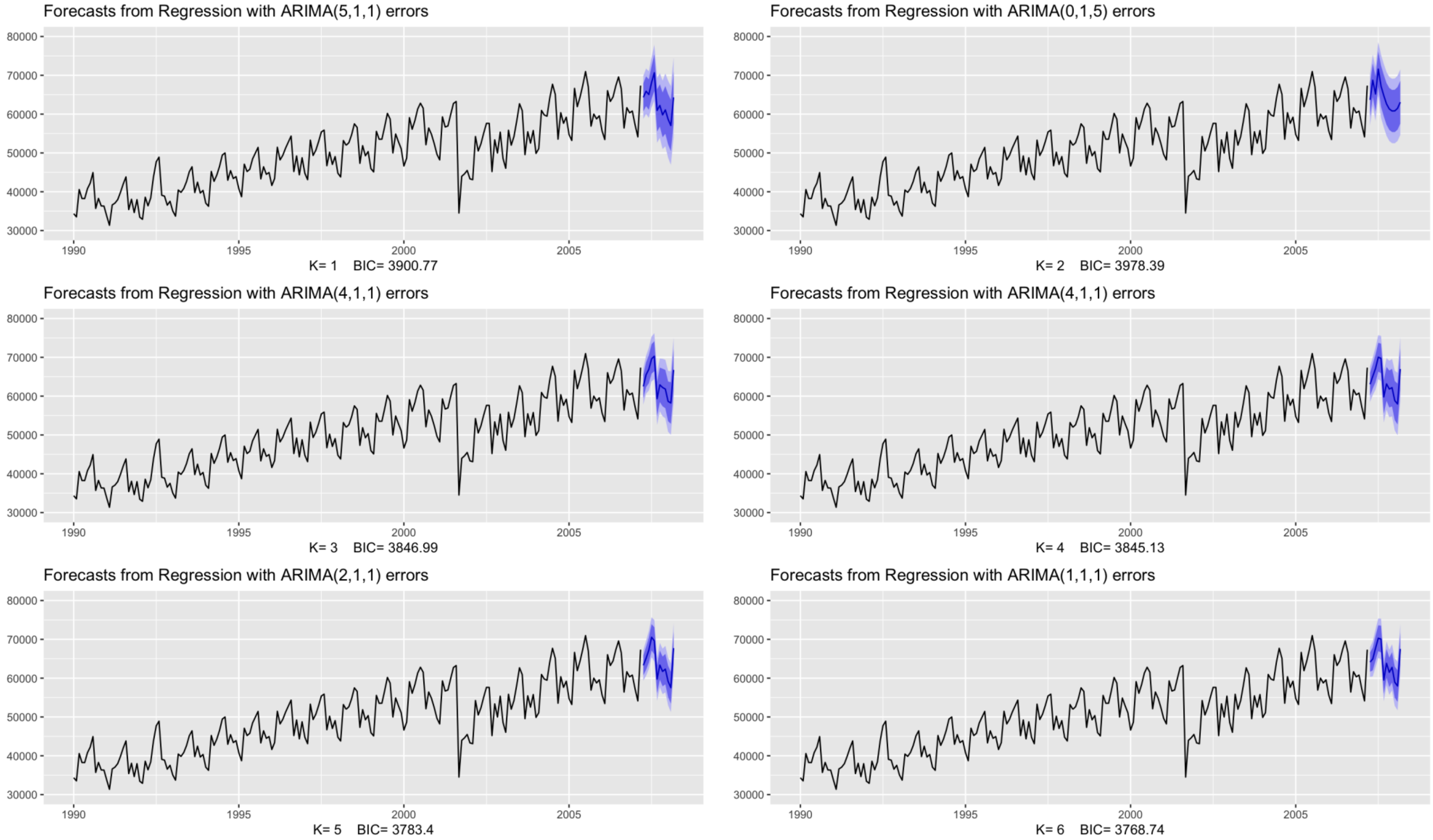
Fourier Transforms (Harmonic Regression)

- Add Fourier variables to a regression model predicting the target to remove the seasonal pattern.
- If you add the same number of Fourier variables as you have seasonal dummy variables, you will get the same predictions.
- However, typically do not need all the Fourier variables → especially with large values of S.

Fourier Transforms (Harmonic Regression)

```
plots <- list()
for (i in seq(6)) {
  fit <- auto.arima(training, xreg = fourier(training, K = i),
                    seasonal = FALSE, lambda = NULL)

  plots[[i]] <- autoplot(forecast::forecast(fit,
                         xreg = fourier(training, K=i, h=12))) +
    xlab(paste("K=",i,"   BIC=",round(fit[["bic"]],2))) +
    ylab("") + ylim(30000,80000)
}

gridExtra::grid.arrange(
  plots[[1]],plots[[2]],plots[[3]],
  plots[[4]],plots[[5]],plots[[6]], nrow=3)
```

Fourier Transforms (Harmonic Regression)



Fourier Transforms (Harmonic Regression)

```
F.ARIMA <- auto.arima(training, xreg = fourier(training, K = 6), seasonal = FALSE)
```

```
## Series: training
## Regression with ARIMA(1,1,1) errors
##
## Coefficients:
##          ar1      ma1     drift       S1-12       C1-12     S2-12     C2-12
##       0.4289  -0.7970  120.6974  -1232.3084  -4334.8049  313.9197  677.9491
## s.e.  0.1142   0.0773   47.1030    270.2688    270.3805  194.1284  193.6296
##           S3-12      C3-12     S4-12     C4-12      S5-12     C5-12     C6-12
##       -2561.1131  1291.3900  413.8895  208.6503  2314.3804  274.0798  341.8763
## s.e.    152.7962   153.1557  130.4249  130.6610   118.9888  119.5082   81.8796
##
## sigma^2 estimated as 3751119:  log likelihood=-1844.41
## AIC=3718.82   AICc=3721.34   BIC=3768.74
##
## Training set error measures:
##                     ME     RMSE      MAE        MPE     MAPE      MASE
## Training set -13.55273 1865.288 1119.117 -0.1820955 2.399079 0.4156704
##                      ACF1
## Training set -0.002801581
```

Advantages and Disadvantages

Advantages

- Can handle long and complex seasonality.
  - If multiple seasons, just add more Fourier variables to account for them.

Disadvantages

- Trial and error for “right” amount of Fourier variables to use.
- No interpretable value.
- Effect of season remains constant.

Predictor Variables for Seasonality

- Last common approach to accounting for seasonality in data is to use other predictor variables that have matching season.
- Modeling these variables against the target might remove the seasonality.
- Example: Weather data and energy data
  - Hourly temperature correlates with hourly energy usage in the summer months (high heat → high energy usage)
  - Have same 24 hour cycle

Advantages and Disadvantages

Advantages

- Can handle long and complex seasonality.
  - If multiple seasons, just add more variables to account for them.
- Interpretation still holds.
  - Can easily measure and interpret effects from these variables.

Disadvantages

- Trial and error for “right” variables to use.
- Might not have predictor variables to use in this context.

What Next?

- After removing the seasonality through deterministic approaches, the remaining error term (residuals) are modeled with Seasonal ARIMA models.

$$Y_t = \beta_0 + \beta_1 X_{1,t} + \beta_2 X_{2,t} + \beta_3 X_{3,t} + \beta_4 X_{4,t} + \cdots + e_t$$

Seasonal ARIMA here!

- Still might need seasonal effects even though season is removed.

?

STOCHASTIC SOLUTION (DIFFERENCING)

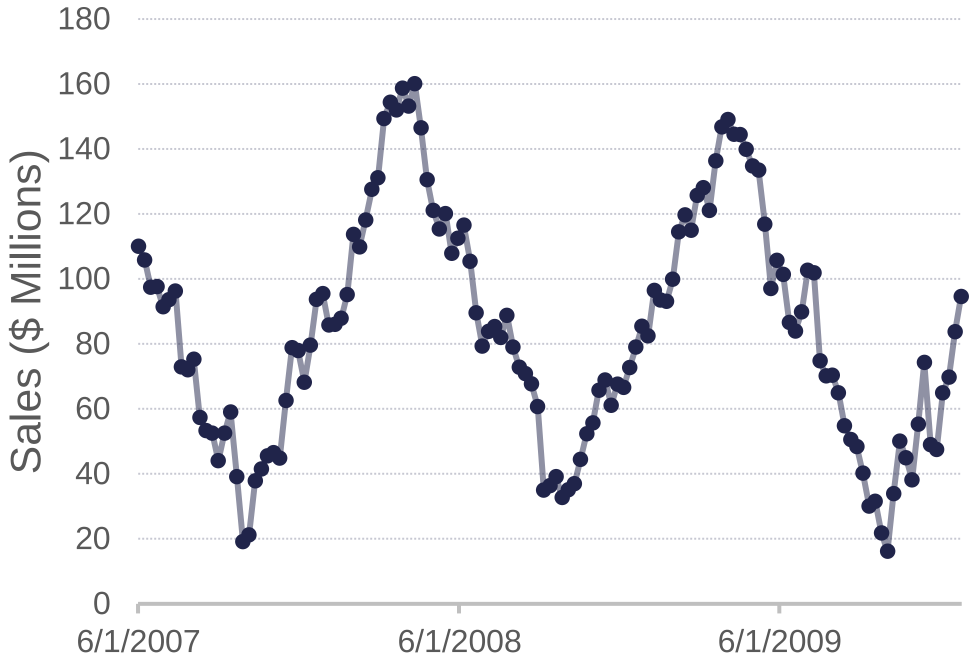
Stochastic Solution

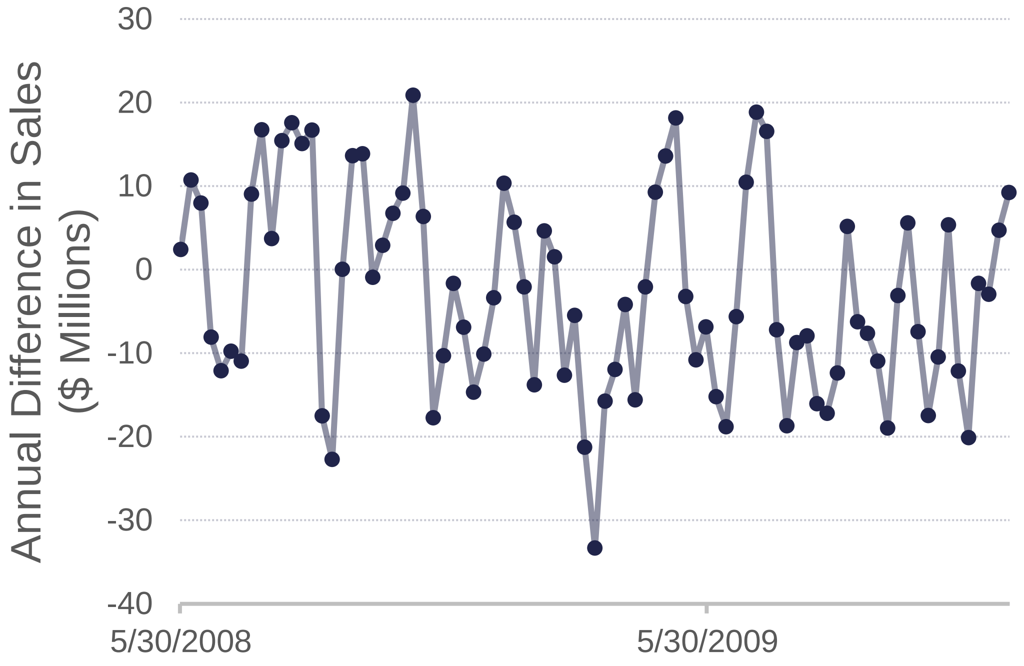
- Similar to trend, seasonality can be solved with a deterministic solution or a stochastic solution.
    - **Deterministic** – Seasonal dummy variables, Fourier transforms, predictor variables
    - **Stochastic** – Seasonal differences

- Once data is made stationary (model away the seasonality), we can model with traditional ARIMA approaches.

Seasonal Differencing

- Differencing on season → look at difference between current point and the same point in the previous season: $Y_t - Y_{t-S}$





What Next?

- After removing the seasonality through stochastic approaches, the remaining differences are modeled with Seasonal ARIMA models.

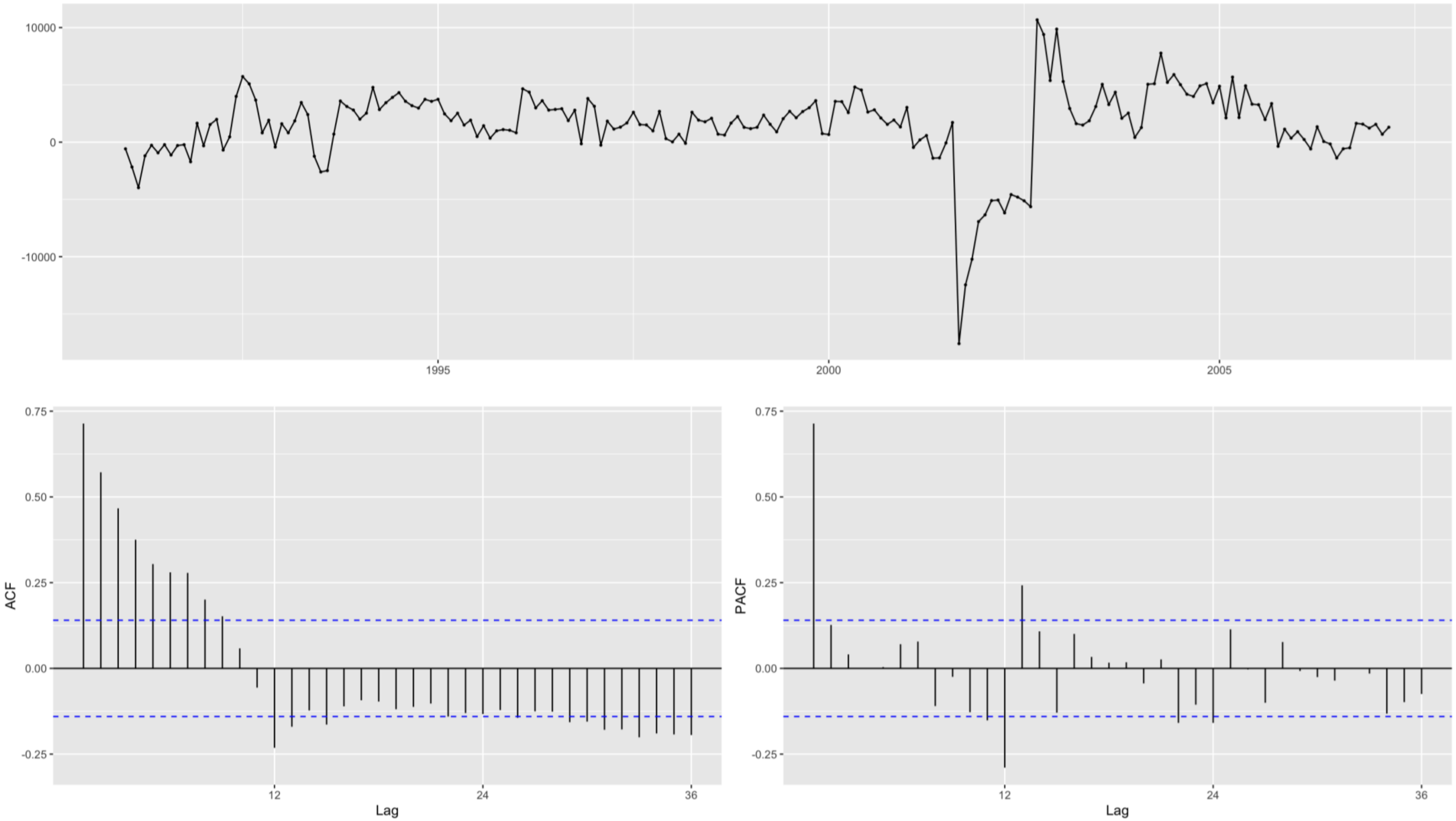
$$Y_t - Y_{t-s} = e_t$$

Seasonal ARIMA here!

- Still might need seasonal effects even though season is removed.

Seasonal Differencing

```
training %>% diff(lag = 12) %>% ggtsdisplay()
```



Limitations of Differencing

- Hard to evaluate stochastic effects for long and complex seasons.
- Most statistical tests for stochastic vs. deterministic can not handle past 12 or 24 periods in a season.
- Long/complex seasons → Best to just approach with deterministic solutions

SEASONAL ARIMA

More Complex ARIMA

- When extending to the Seasonal ARIMA framework, we add another set of terms – *P*, *D*, *Q*, and *S*.

$$\text{ARIMA}(p, d, q)(P, D, Q)_S$$

# of AR Terms

# of MA Terms

# of First Differences

More Complex ARIMA

- When extending to the Seasonal ARIMA framework, we add another set of terms – *P*, *D*, *Q*, and *S*.

$$\mathrm{ARIMA}(p,d,q)(P,D,Q)_S$$

Length of Season

# of Seasonal AR Terms

# of Seasonal Differences

# of Seasonal MA Terms

Seasonal ARIMA

- **Seasonal** ARIMA models are typically written as the following:

$$\mathrm{ARIMA}(1,0,1)(2,1,0)_{12}$$

Seasonal ARIMA

- **Seasonal** ARIMA models are typically written as the following:

$$\text{ARIMA}(1,0,1)(2,1,0)_{12}$$

$$Y_t - Y_{t-12} = W_t$$

Seasonal ARIMA

- **Seasonal** ARIMA models are typically written as the following:

$$\text{ARIMA}(1,0,1)(2,1,0)_{12}$$

$$Y_t - Y_{t-12} = W_t$$

Seasonal ARIMA

- **Seasonal** ARIMA models are typically written as the following:

$$\text{ARIMA}(1,0,1)(2,1,0)_{12}$$

$$Y_t - Y_{t-12} = W_t$$

$$W_t = \omega + \phi_1 W_{t-1} + \phi_2 W_{t-12} + \phi_3 W_{t-24} + \theta_1 e_{t-1} + e_t$$

Seasonal ARIMA

- **Seasonal** ARIMA models are typically written as the following:

$$\text{ARIMA}(1,0,1)(2,1,0)_{12}$$

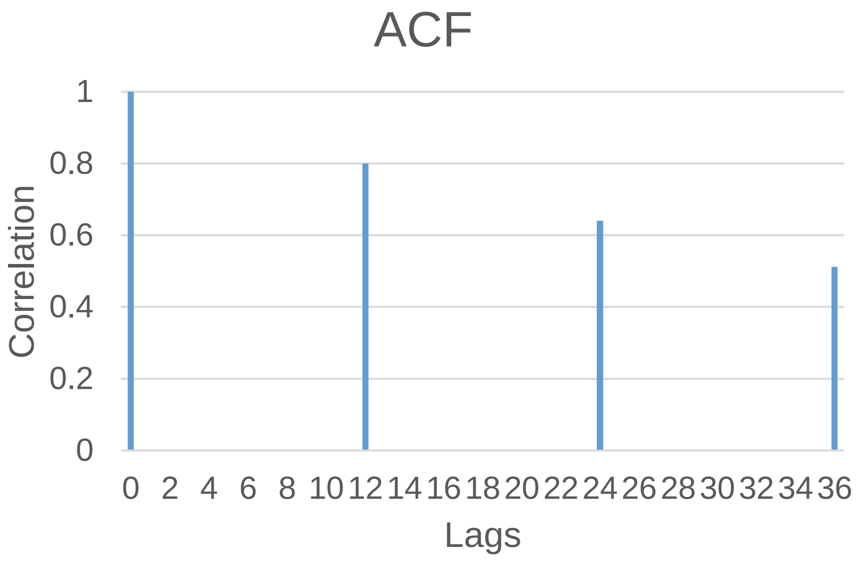
$$Y_t - Y_{t-12} = W_t$$

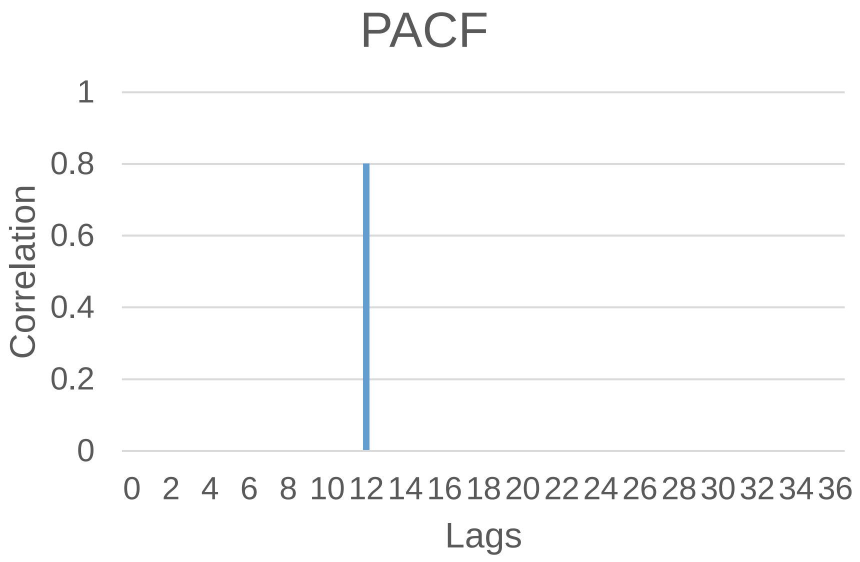
$$W_t = \omega + \phi_1 W_{t-1} + \phi_2 W_{t-12} + \phi_3 W_{t-24} + \theta_1 e_{t-1} + e_t$$

Seasonal ARIMA

- Seasonal ARIMA models have the same structure and approach as typical ARIMA models with AR and MA patterns in the PACF and ACF.
- The pattern is just on the seasonal lag instead of the individual lags.

$$\text{ARIMA}(0,0,0)(1,0,0)_{12}$$

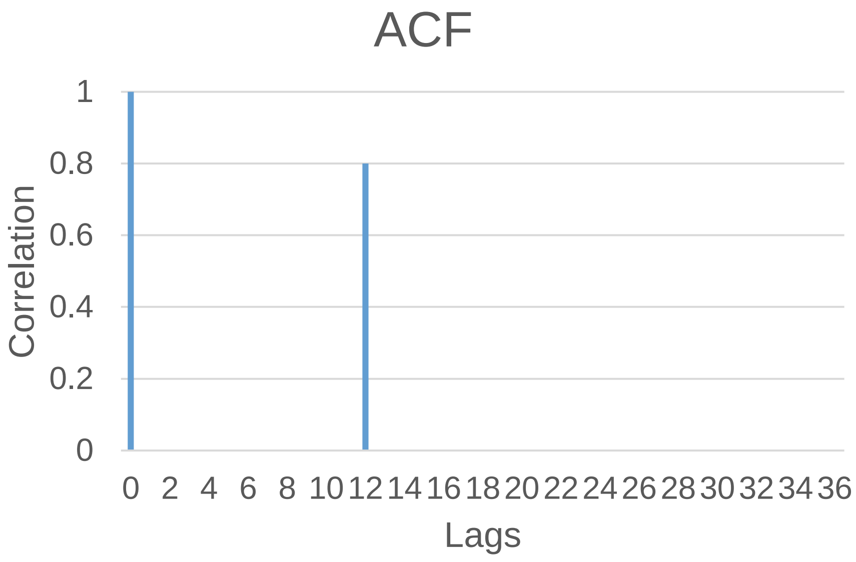


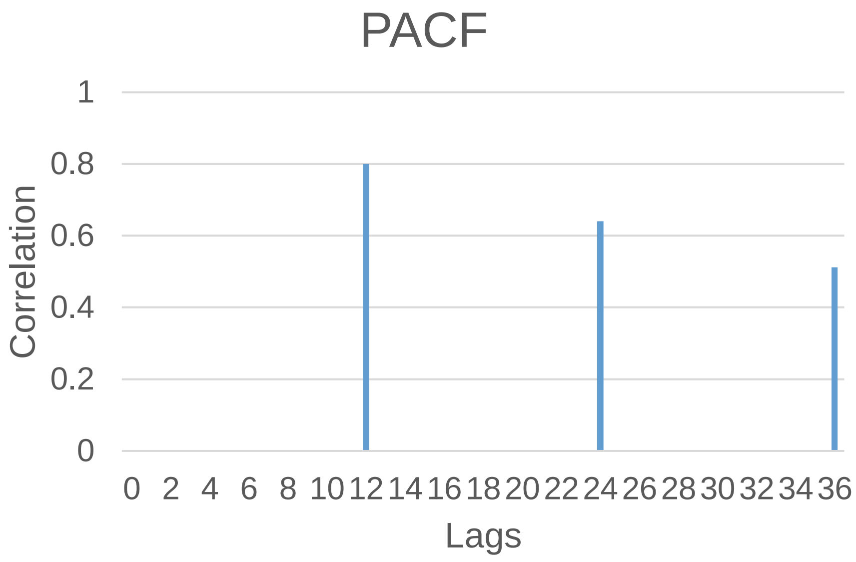


Seasonal ARIMA

- Seasonal ARIMA models have the same structure and approach as typical ARIMA models with AR and MA patterns in the PACF and ACF.
- The pattern is just on the seasonal lag instead of the individual lags.

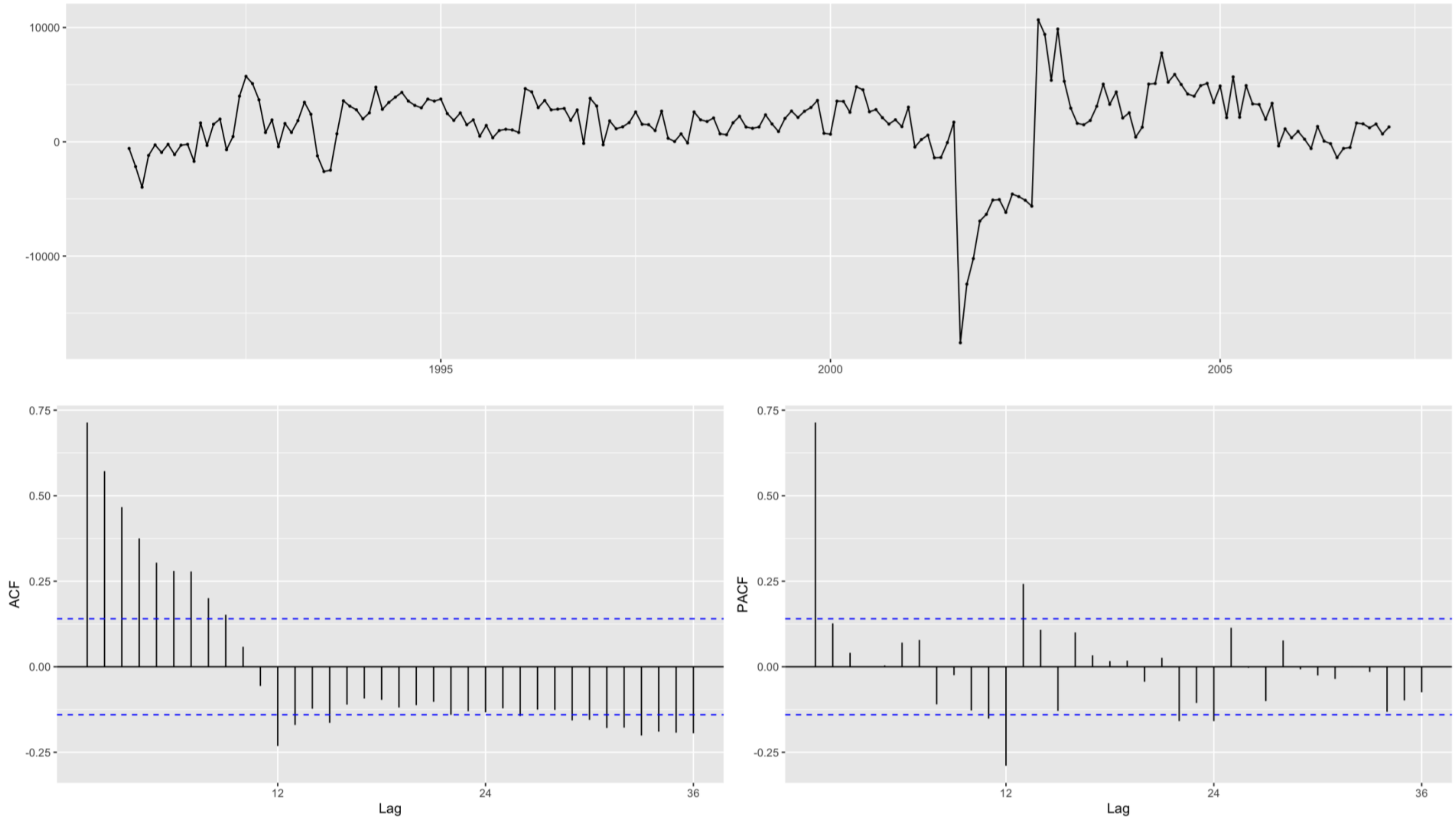
$$\text{ARIMA}(0,0,0)(0,0,1)_{12}$$





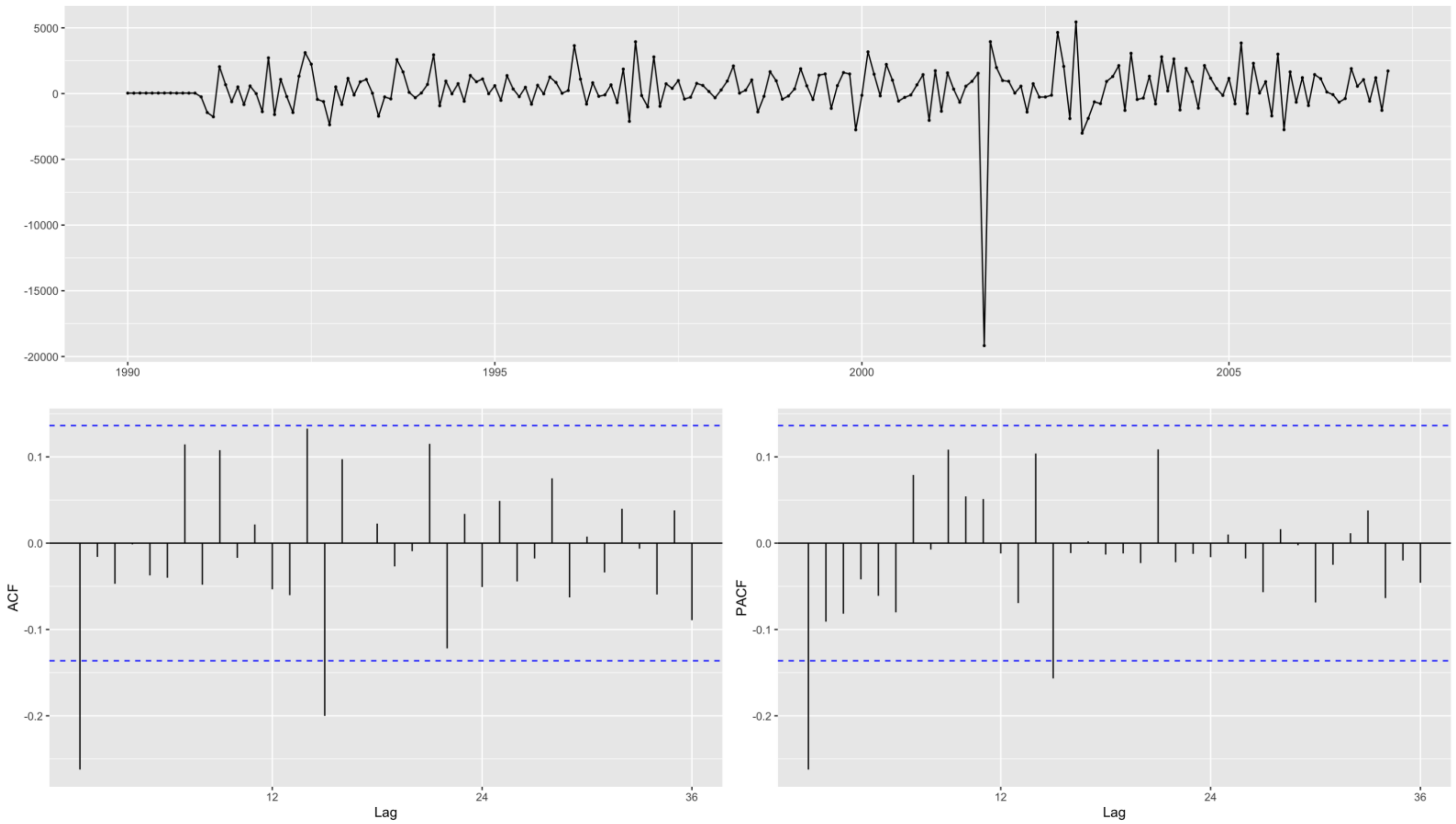
Seasonal Differencing

```
training %>% diff(lag = 12) %>% ggtsdisplay()
```



Seasonal ARIMA

```
training %>%
  Arima(order=c(1,0,0), seasonal=c(1,1,1)) %>%
  residuals() %>% ggtsdisplay()
```



Seasonal ARIMA

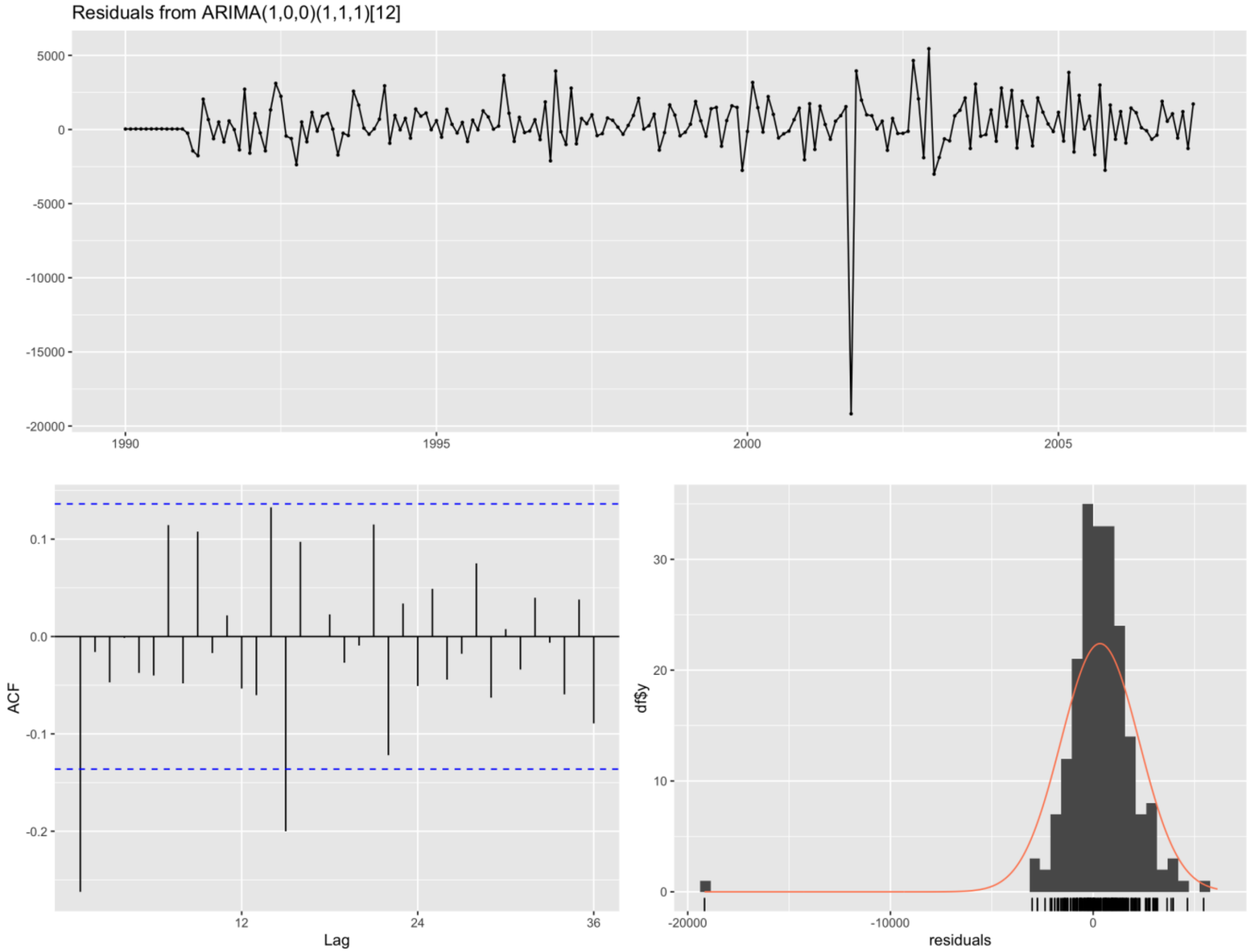
```
S.ARIMA <- Arima(training, order=c(1,0,0), seasonal=c(1,1,1))
summary(S.ARIMA)
```

```
## Series: training
## ARIMA(1,0,0)(1,1,1)[12]
##
## Coefficients:
##          ar1     sar1     sma1
##       0.9056   0.0917   -0.672
## s.e.  0.0364   0.1091    0.093
##
## sigma^2 estimated as 4126436:  log likelihood=-1763.94
## AIC=3535.88   AICc=3536.09   BIC=3548.97
##
## Training set error measures:
##                    ME     RMSE      MAE       MPE     MAPE      MASE       ACF1
## Training set 338.6503 1956.379 1156.221 0.5565257 2.418163 0.4294517 -0.2622466
```

Seasonal ARIMA

```
checkresiduals(S.ARIMA)
```

```
##  Ljung-Box test
##
## data:  Residuals from ARIMA(1,0,0)(1,1,1)[12]
## Q* = 45.934, df = 21, p-value = 0.001304
##
## Model df: 3.   Total lags used: 24
```



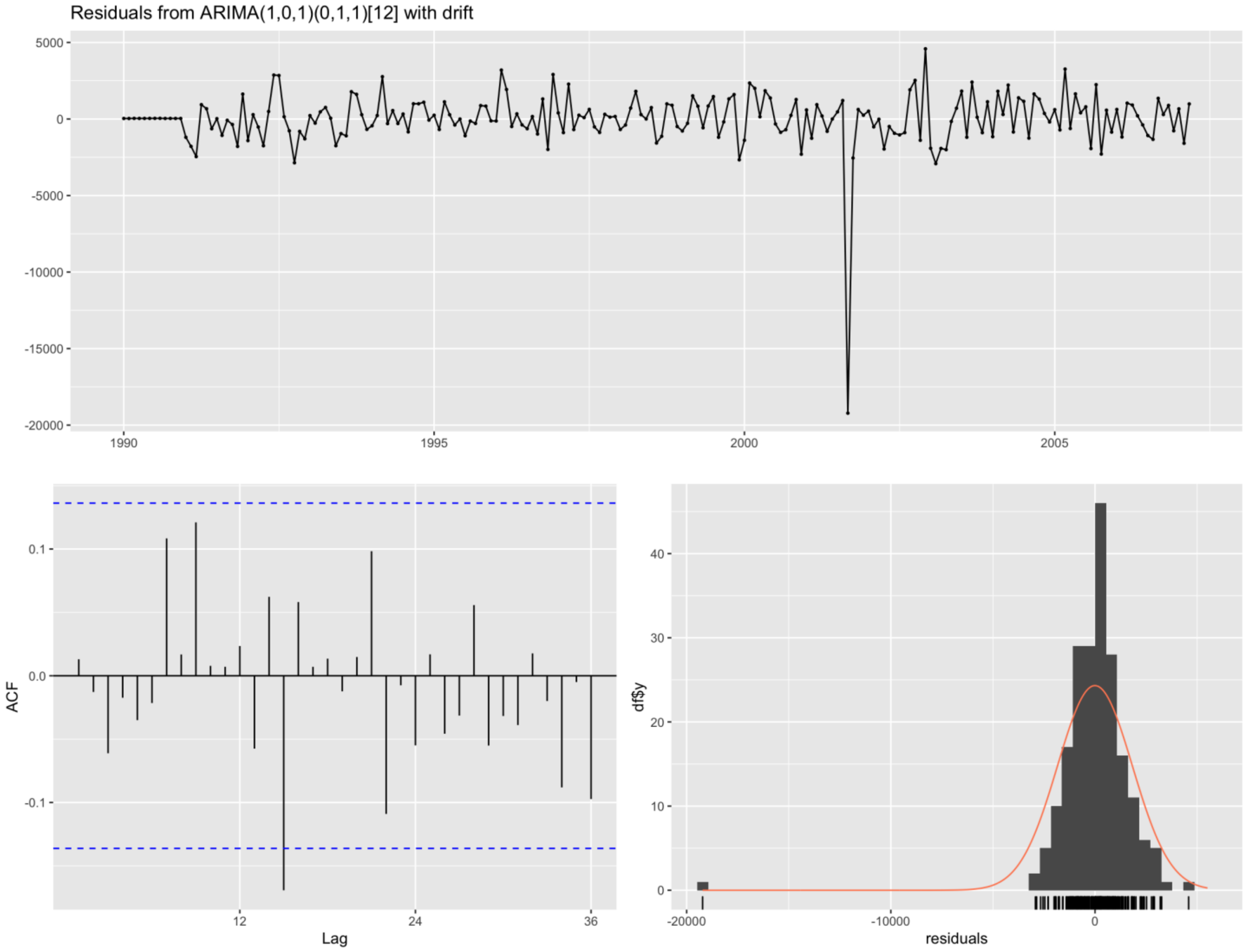
Seasonal ARIMA

```
S.ARIMA <- auto.arima(training, method="ML", seasonal = TRUE)
summary(S.ARIMA)
```

```
## Series: training
## ARIMA(1,0,1)(0,1,1)[12] with drift
##
## Coefficients:
##          ar1      ma1     sma1     drift
##       0.8800  -0.2962  -0.6785  124.9788
## s.e.  0.0454   0.0950   0.0600   23.6330
##
## sigma^2 estimated as 3639517:  log likelihood=-1751.67
## AIC=3513.34   AICc=3513.66   BIC=3529.7
##
## Training set error measures:
##                       ME    RMSE     MAE        MPE     MAPE      MASE       ACF1
## Training set -4.332616 1832.54 1055.07 -0.1745474 2.217472 0.3918815 0.01300462
```

Seasonal ARIMA

```
checkresiduals(S.ARIMA)


##  Ljung-Box test
##
## data:  Residuals from ARIMA(1,0,1)(0,1,1)[12]
## Q* = 21.957, df = 20, p-value = 0.3428
##
## Model df: 4.   Total lags used: 24
```



Multiple Differences

- Models can contain both unit roots and seasonal unit roots.
- After removing the seasonal unit root through differencing to get W_t, ordinary differences can be calculated.

$$W_t = Y_t - Y_{t-12}$$

$$W_t = W_{t-1} + e_t - \beta e_{t-1}$$

$$W_t - W_{t-1} = e_t - \beta e_{t-1}$$

$$(Y_t - Y_{t-12}) - (Y_{t-1} - Y_{t-13}) = e_t - \beta e_{t-1}$$

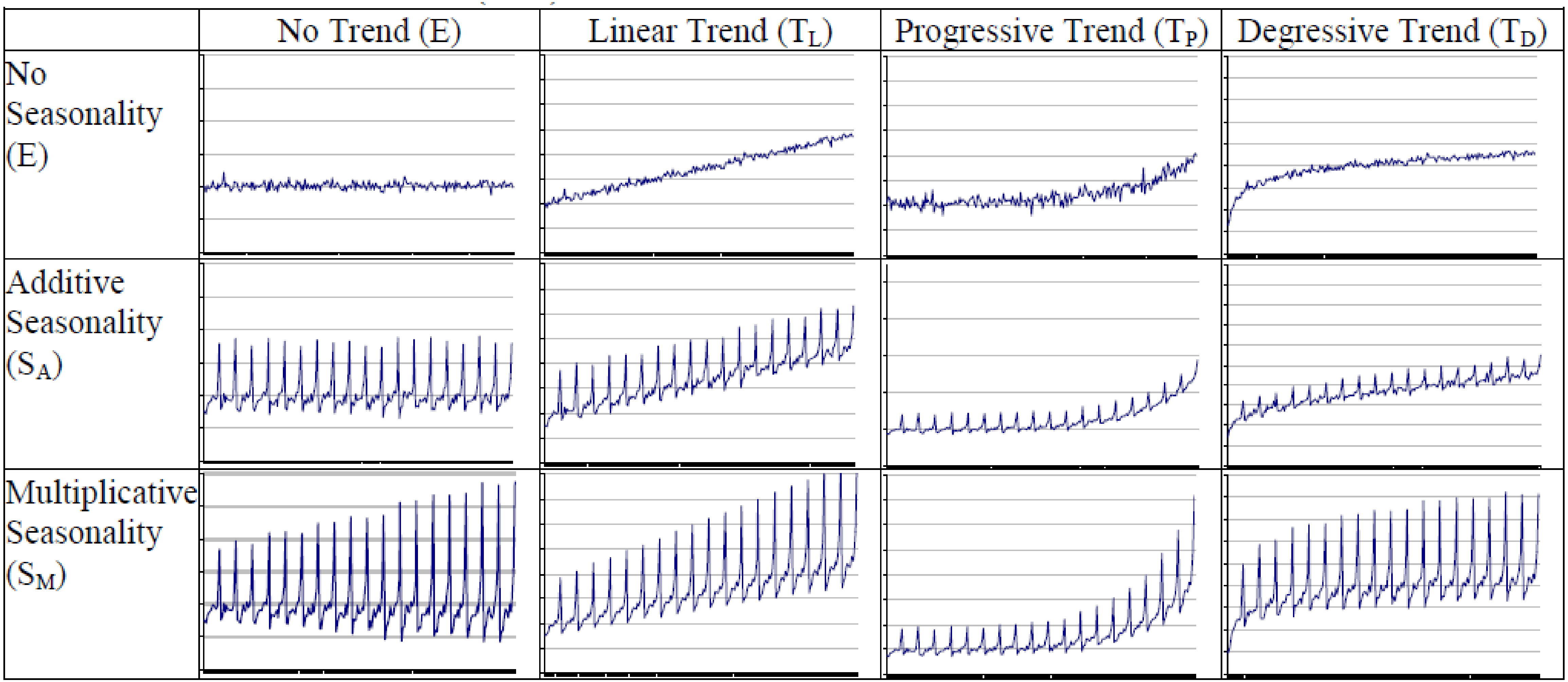
$$Y_t = Y_{t-1} + Y_{t-12} - Y_{t-13} + e_t - \beta e_{t-1}$$

Limitations of Differencing

- Hard to evaluate stochastic effects for long and complex seasons.
- Most statistical tests for stochastic vs. deterministic can not handle past 12 or 24 periods in a season.
- Long/complex seasons → Best to just approach with deterministic solutions

MULTIPLICATIVE VS. ADDITIVE

Multiplicative vs. Additive



Backshift Operator – B

- The backshift operator is the mathematical operator to convert observations to their lags.
  - $\mathrm{B}(Y_t) = Y_{t-1}$

- This can be extended to any number of lags.
  - $\mathrm{B}^2(Y_t) = \mathrm{B}(Y_{t-1}) = Y_{t-2}$

Structures to Seasons

Additive

$$(1 - \alpha_1 \mathrm{B} - \alpha_2 \mathrm{B}^{12})Y_t = e_t$$

Multiplicative

$$(1 - \alpha_1 \mathrm{B})(1 - \alpha_2 \mathrm{B}^{12})Y_t = e_t$$

Structures to Seasons

Additive

$$(1 - \alpha_1 \mathrm{B} - \alpha_2 \mathrm{B}^{12}) Y_t = e_t$$

$$Y_t - \alpha_1 \mathrm{B}(\mathrm{Y_t}) - \alpha_2 \mathrm{B}^{12}(Y_t) = e_t$$

Multiplicative

$$(1 - \alpha_1 \mathrm{B})(1 - \alpha_2 \mathrm{B}^{12}) Y_t = e_t$$

Structures to Seasons

Additive

$$(1 - \alpha_1 \mathrm{B} - \alpha_2 \mathrm{B}^{12})Y_t = e_t$$

$$Y_t - \alpha_1 \mathrm{B}(\mathrm{Y_t}) - \alpha_2 \mathrm{B}^{12}(Y_t) = e_t$$

$$Y_t - \alpha_1 Y_{t-1} - \alpha_2 Y_{t-12} = e_t$$

Multiplicative

$$(1 - \alpha_1 \mathrm{B})(1 - \alpha_2 \mathrm{B}^{12})Y_t = e_t$$

Structures to Seasons

Additive

$$(1-\alpha_1 \mathrm{B}-\alpha_2 \mathrm{B}^{12})Y_t = e_t$$

$$Y_t - \alpha_1 \mathrm{B}(\mathrm{Y_t}) - \alpha_2 \mathrm{B}^{12}(Y_t) = e_t$$

$$Y_t - \alpha_1 Y_{t-1} - \alpha_2 Y_{t-12} = e_t$$

$$Y_t = \alpha_1 Y_{t-1} + \alpha_2 Y_{t-12} + e_t$$

Multiplicative

$$(1-\alpha_1 \mathrm{B})(1-\alpha_2 \mathrm{B}^{12})Y_t = e_t$$

Structures to Seasons

Additive

$$(1 - \alpha_1 \mathrm{B} - \alpha_2 \mathrm{B}^{12}) Y_t = e_t$$

Multiplicative

$$(1 - \alpha_1 \mathrm{B})(1 - \alpha_2 \mathrm{B}^{12}) Y_t = e_t$$

$$\Downarrow$$

$$(1 - \alpha_1 \mathrm{B} - \alpha_2 \mathrm{B}^{12} + \alpha_1 \alpha_2 \mathrm{B}^{13}) Y_t = e_t$$

Structures to Seasons

Additive

$$(1-\alpha_1\mathrm{B}-\alpha_2\mathrm{B}^{12})Y_t=e_t$$

$$(1-\alpha_1\mathrm{B}-\alpha_2\mathrm{B}^{12}-\boxed{\alpha_3}\mathrm{B}^{13})Y_t=e_t$$

Multiplicative

$$(1-\alpha_1\mathrm{B})(1-\alpha_2\mathrm{B}^{12})Y_t=e_t$$

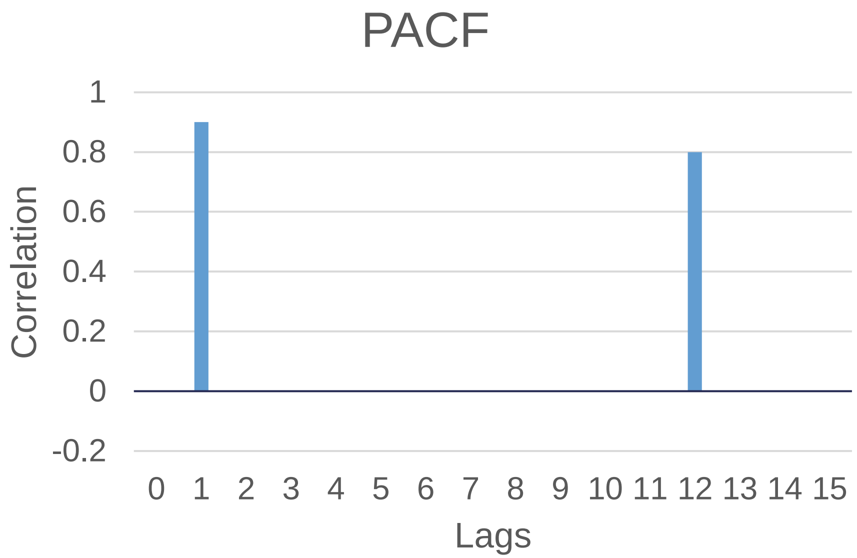
$$(1-\alpha_1\mathrm{B}-\alpha_2\mathrm{B}^{12}+\boxed{\alpha_1\alpha_2}\mathrm{B}^{13})Y_t=e_t$$

?

Structures to Seasons

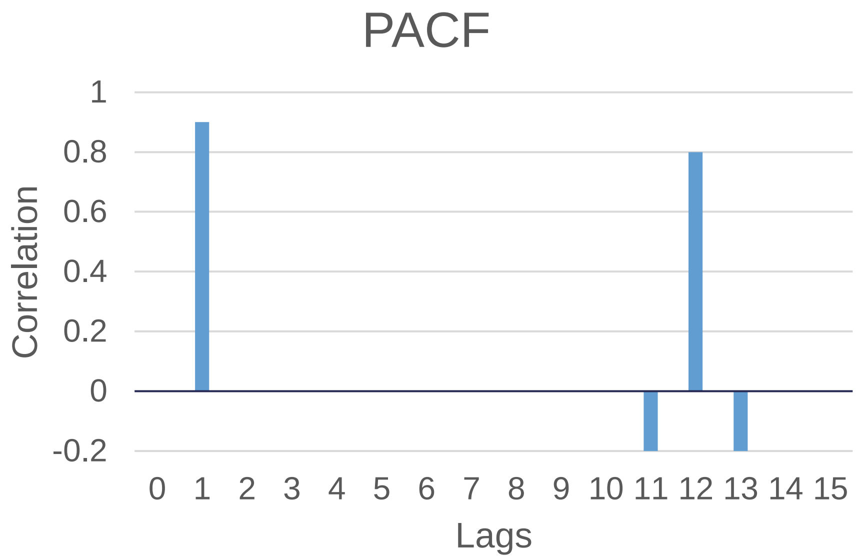
Additive

$$(1 - \alpha_1 \mathrm{B} - \alpha_2 \mathrm{B}^{12}) Y_t = e_t$$



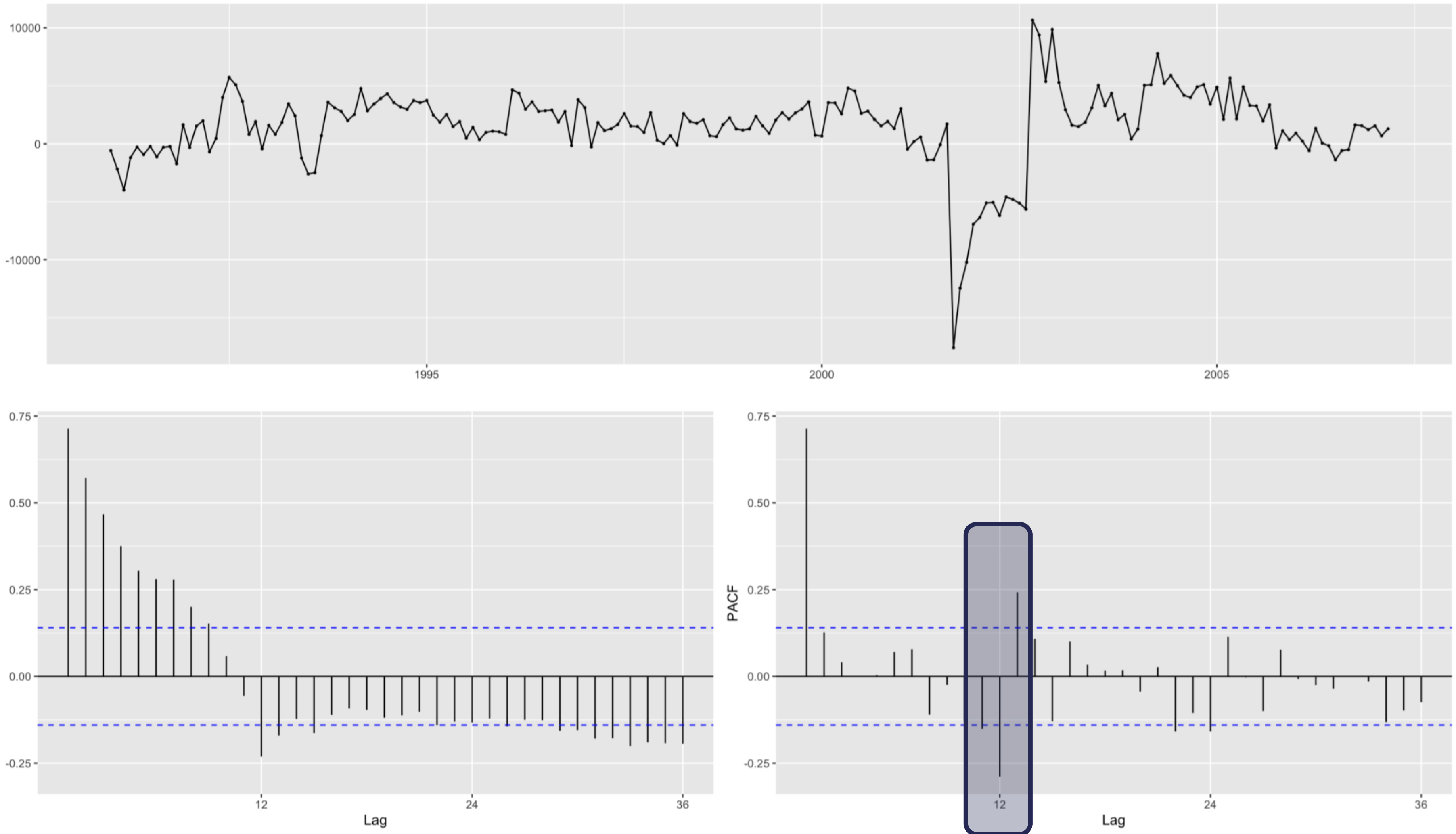
Multiplicative

$$(1 - \alpha_1 \mathrm{B})(1 - \alpha_2 \mathrm{B}^{12}) Y_t = e_t$$



Seasonal Differencing

```
training %>% diff(lag = 12) %>% ggtsdisplay()
```



Structures to Seasons

Additive

```
S.ARIMA <- Arima(training, order=c(1,0,13),
                 seasonal=c(0,1,0),
                 fixed=c(NA,NA,0,0,0,0,0,0,0,0,0,0,NA,NA),
                 method="ML",)
summary(S.ARIMA)
```

```
## Coefficients:
##          ar1      ma1  ma2  ma3  ma4  ma5  ma6  ma7  ma8
##       0.9679  -0.3698    0    0    0    0    0    0    0
## s.e.  0.0237   0.0880    0    0    0    0    0    0    0
##       ma9   ma10   ma11     ma12    ma13
##         0      0      0  -0.6612  0.2490
## s.e.    0      0      0   0.0626  0.0766
```

Multiplicative

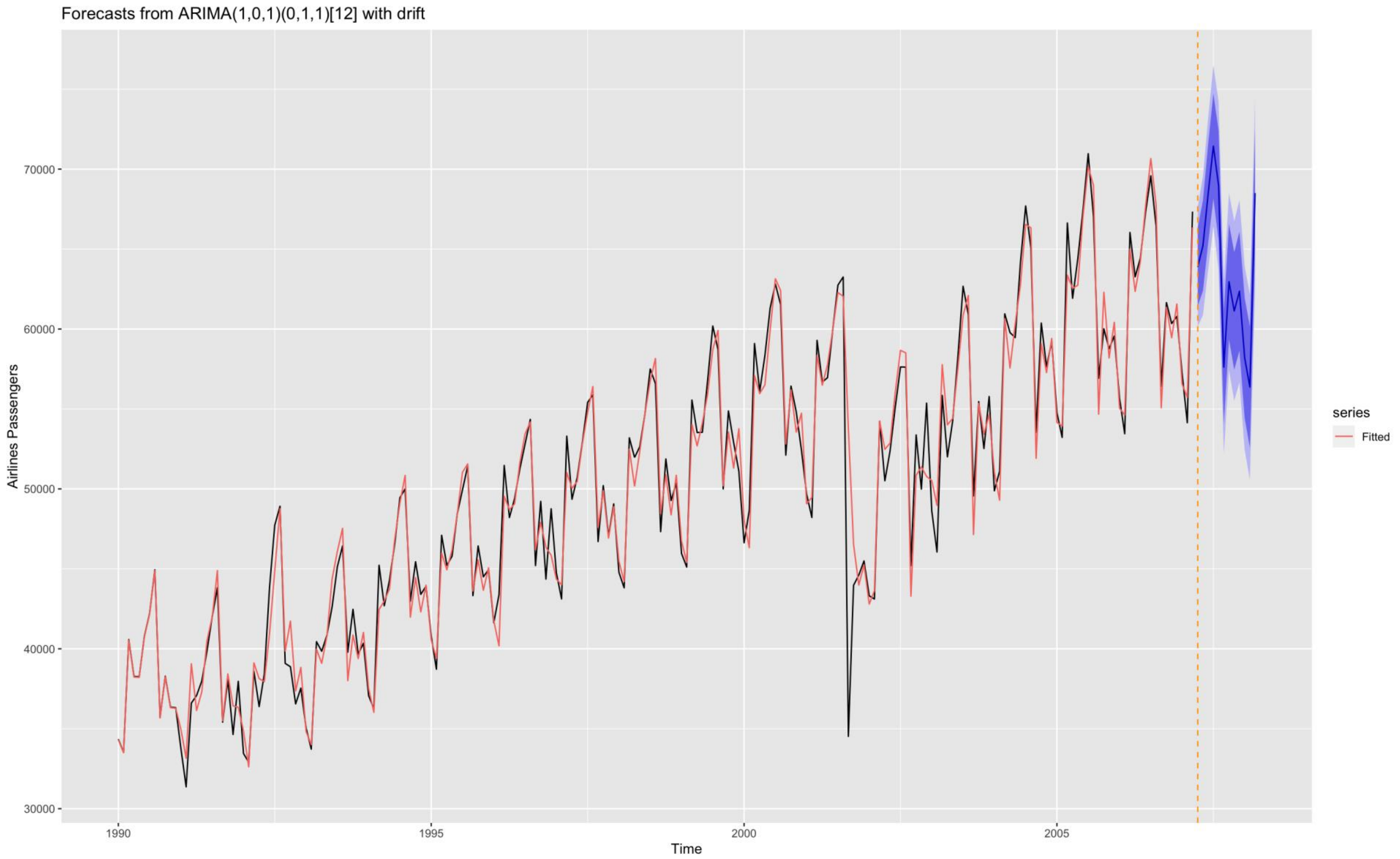
```
S.ARIMA <- auto.arima(training, method="ML",
                      seasonal = TRUE)
summary(S.ARIMA)
```

```
## Coefficients:
##          ar1      ma1     sma1     drift
##       0.8800  -0.2962  -0.6785  124.9788
## s.e.  0.0454   0.0950   0.0600   23.6330
```

Seasonal ARIMA Models

```
S.ARIMA <- auto.arima(training, method="ML", seasonal = TRUE)

autoplot(forecast::forecast(S.ARIMA, h = 12)) +
  autolayer(fitted(S.ARIMA), series="Fitted") +
  ylab("Airlines Passengers") +
  geom_vline(xintercept = 2007.25,color="orange",linetype="dashed")

S.ARIMA.error <- test - forecast::forecast(S.ARIMA, h = 12)$mean

S.ARIMA.MAE <- mean(abs(S.ARIMA.error))
S.ARIMA.MAPE <- mean(abs(S.ARIMA.error)/abs(test))*100
```

Seasonal ARIMA Models



Model Evaluation on Test Data

Model	MAE	MAPE
HW Exponential Smoothing	1134.58	1.76%
Seasonal ARIMA	1229.21	1.89%

?