Lab 8 Forecasting Part 1

STAT 443

Caden Hewlett

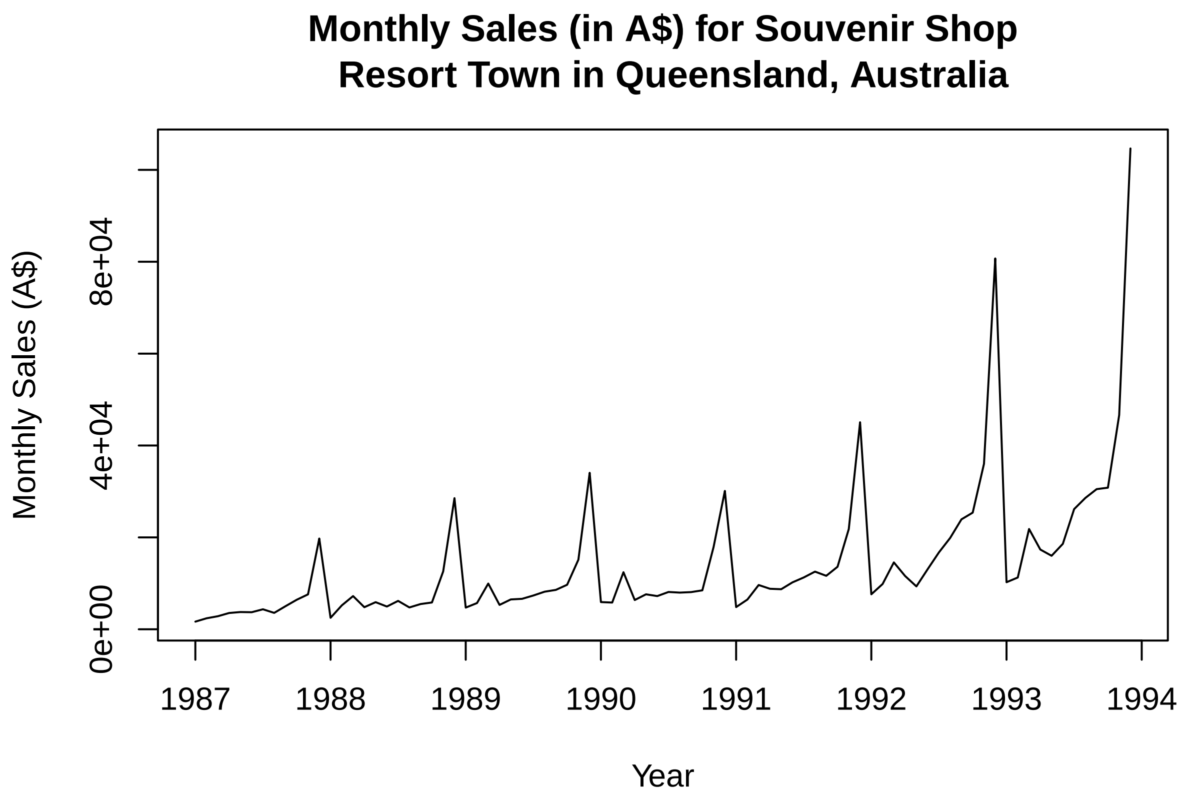
2024-03-08

Download the data file souvenir.txt. It contains monthly sales (in A$) for a souvenir shop at a beach resort town in Queensland, Australia, for January 1987–December 1993. Import the data into R as a time series object.

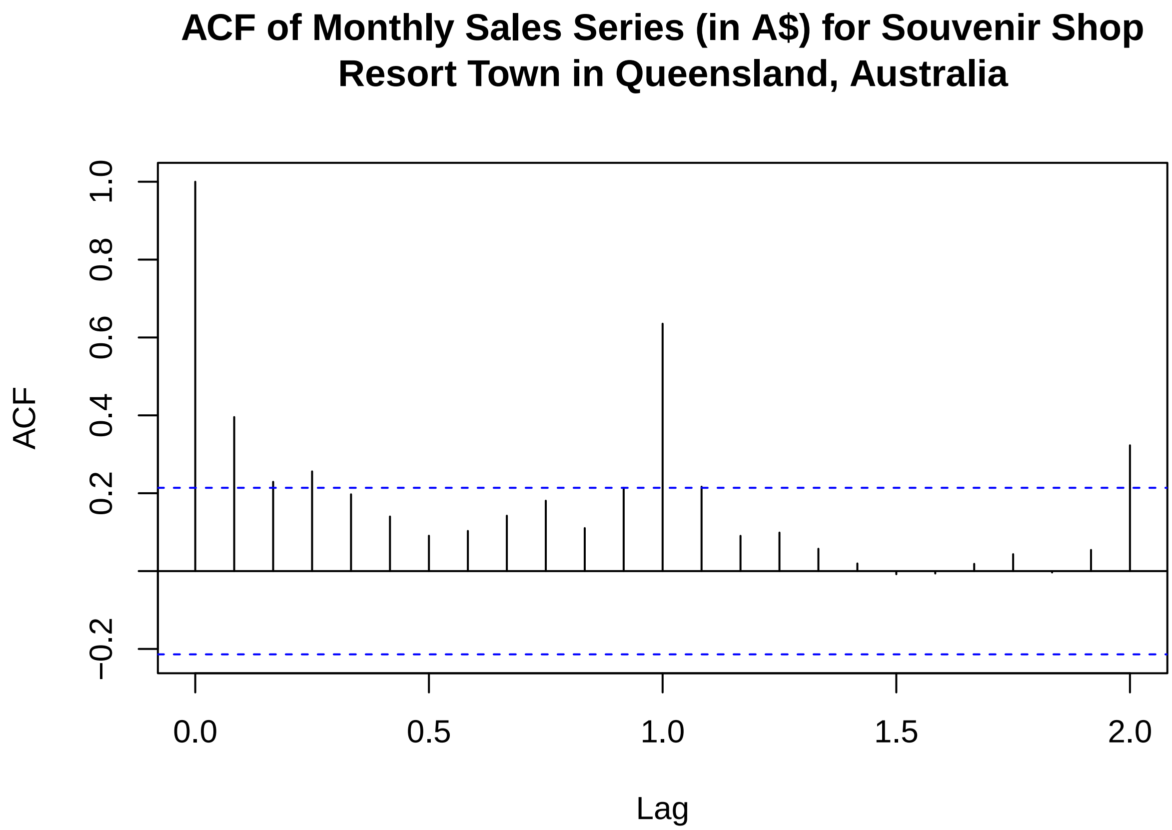
Question 1

Plot the time series and its sample acf and comment on what you see. If you deduce there is a seasonal effect, is it additive or multiplicative? Explain your reasoning

```
df = read.table("souvenir.txt")
souvenir_ts = ts(data = (df),
                 start = c(1987,1),
                 end = c(1993, 12),
                 frequency = 12)
plot(souvenir_ts, xlab = "Year", ylab = "Monthly Sales (A$)",
     main = "Monthly Sales (in A$) for Souvenir Shop
  Resort Town in Queensland, Australia")
```




```
acf(souvenir_ts, lag.max = 24, main = "ACF of Monthly Sales Series (in A$) for Souvenir Shop
  Resort Town in Queensland, Australia")
```



There *is* a seasonal component to this time series (with what initially appears to be $p = 12$ - we set `lag.max = 24`and see two full cycles), as can be seen with the periodicity (peaks and troughs) in the series. Further, because the amplitude of the "sine-curve" like component is increasing in magnitude over time (i.e. there isn't a constant up and down,) this is indicative of multiplicative seasonal component.

Question 2

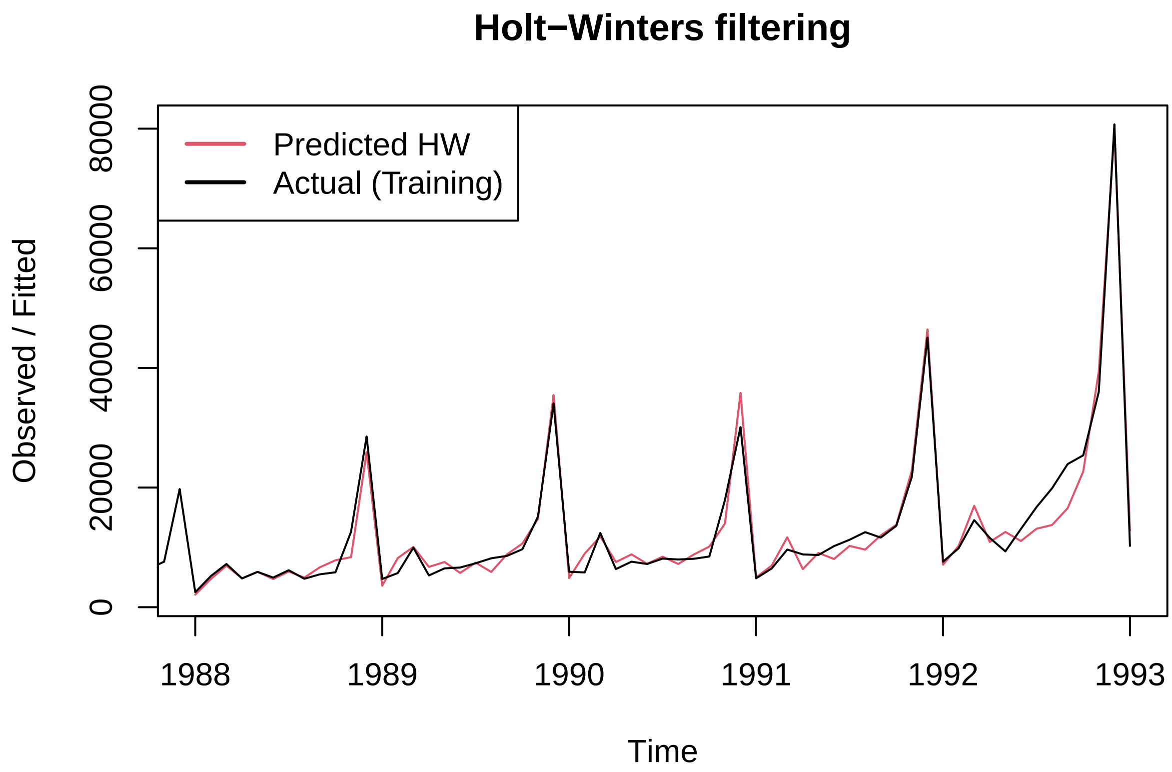
Fit a prediction model based on the training data using the R function `HoltWinters()`. Set the options according to what you decided above. Provide the parameter values for your smoothing model. Plot the data along with the fitted values by applying the `plot()` function on the HoltWinters object

First, we split the data into train and test.

```
train = window(souvenir_ts, start = c(1987, 1), end = c(1993, 1))
test = window(souvenir_ts, start = c(1993, 2), end = c(1993, 12))
```

Now, we fit a Holt Winters model.

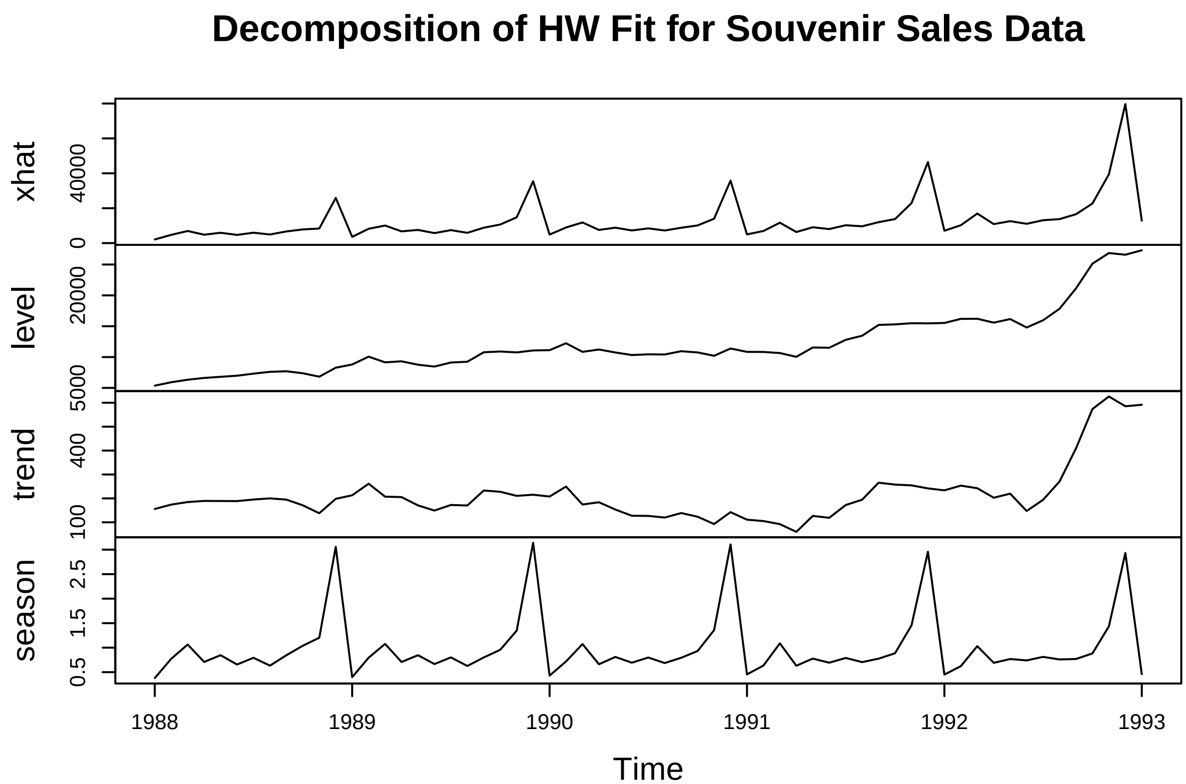
```
fit = HoltWinters(train, seasonal = "multiplicative")
plot(fit)
legend(x = "topleft",
       legend = c("Predicted HW", "Actual (Training)"),
       lty = c(1, 1),
       col = c(2, 1),
       lwd = 2)
```



Question 3

Plot the estimates of level L_t expected change per unit time of the trend component T_t and seasonal effect It over your training period. You can do this by applying `fitted()` function on the `HoltWinters` object and then plotting the output.

```
plot(fitted(fit), main = "Decomposition of HW Fit for Souvenir Sales Data")
```



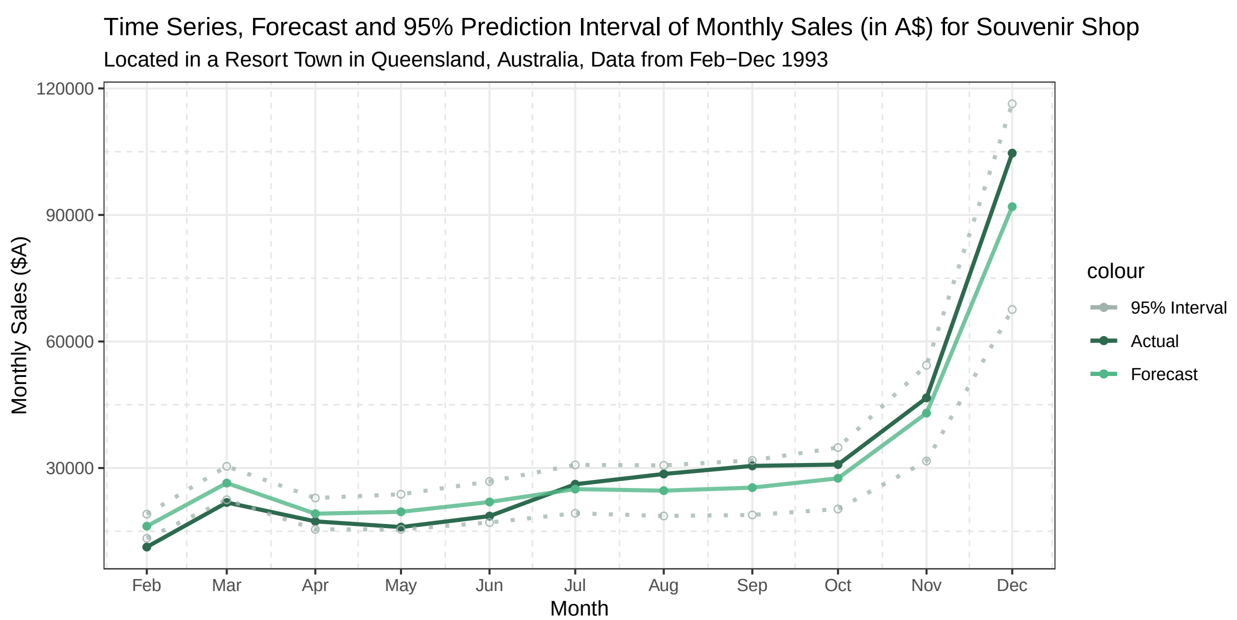
Question 4

ow use the prediction model from above to predict monthly sales from February to December of 1993 via the predict function. Plot the predicted values along with 95% prediction intervals and the actual data from the test set on the same plot. Make sure to use different line types (option lty) and line colours (option col) to distinguish different lines, and remember to include a legend. Use options `type="b"` and `pch=19` to display points and connecting lines for point forecasts and observations. Comment on the accuracy of forecasts

```
preds = as.data.frame(predict(fit, newdata = test, n.ahead = 11,
                              prediction.interval = TRUE, level = 0.95))
forecast_df <- data.frame(
  Time = as.Date(time(test)),
  Observed = as.numeric(test),
  Forecast = as.numeric(preds$fit),
  Lower95 = as.numeric(preds$lwr),
  Upper95 = as.numeric(preds$upr)
)
```

```
g<-ggplot(data = forecast_df, aes(x = Time)) +
  geom_line(aes(y = Observed, color = "Actual"), linewidth = 1, na.rm = TRUE) +
  geom_line(aes(y = Forecast, color = "Forecast"), linewidth = 1, alpha = 0.8) +
  geom_line(aes(y = Lower95, color = "95% Interval"),  alpha = 0.75, linewidth = 1,
            lty = 'dotted') +
  geom_line(aes(y = Upper95, color = "95% Interval"), alpha = 0.75, linewidth = 1,
```

```
lty = "dotted") +
  geom_point(aes(y = Observed, color = "Actual"), pch = 19, na.rm = TRUE) +
  geom_point(aes(y = Forecast, color = "Forecast"), pch = 19, na.rm = TRUE) +
  geom_point(aes(y = Lower95, color = "95% Interval"), alpha = 0.85,
             pch = 21, na.rm = TRUE) +
  geom_point(aes(y = Upper95, color = "95% Interval"), alpha = 0.85,
             pch = 21, na.rm = TRUE) +
  scale_color_manual(
    values = c("Actual" = "#2d6a4f",
               "Forecast" = "#52b788",
               "95% Interval" = "#a1b3ac")) +
      scale_x_date(date_breaks = "1 month",  date_labels = "%b") +
  labs(
    title = "Time Series, Forecast and 95% Prediction Interval of Monthly Sales (in A$) for Souvenir Sh
    subtitle = "Located in a Resort Town in Queensland, Australia, Data from Feb-Dec 1993",
    x = "Month",
    y = "Monthly Sales ($A)"
  ) +
  theme_bw() +
  theme(panel.grid.minor =  element_line(color = "grey90",
                                         size = 0.35,
                                         linetype = "dashed"))
g
```



The forecast accuracy seems to be very good, with the observed values being relatively near the forecasted values and wholly within the prediction interval.

Question 5

Report the forecast values for February, March and April of 1993.

The table below is in A$

```r
blargh = (t(forecast_df[1:3, 2:3]))
colnames(blargh) = c("February",
                     "March", "April")
kable(blargh)
```

	February	March	April
Observed	11266.88	21826.84	17357.33
Forecast	16196.15	26440.10	19170.49

Question 6

It seems to me like a logarithmic transformation could theoretically be considered for this model. If we have a multiplicative model (using the standard nomenclature of the course) given by $X_t = m_t s_t Z_t$, then the logarithm is an additive model, i.e. $\log(X_t) = \log(m_t) + \log(s_t) + \log(Z_t)$ which is additive with respect to $\log(X_t)$. A decomposition of this fashion would allow us to easily fit a OLS model to any potential trend in the model (if desired) and it would make visualization easier.