

Lab 9 Forecasting Part 2

STAT 443

Caden Hewlett

2024-03-15

Introduction

In this lab you will apply the Box–Jenkins forecasting method using a case study. The dataset LakeHuron (available in R) measures the annual level of Lake Huron from 1875 to 1972, in feet. To load the data, use command `data(LakeHuron)`.

```
data(LakeHuron)
series = LakeHuron
head(series)
```

```
## [1] 580.38 581.86 580.97 580.80 579.79 580.39
```

Note that it is already in a time series format.

Using the window command or otherwise, first subset the Lake Huron data into two separate datasets: training data, which should contain observations up to and including year 1969, and test data containing lake level measurements for 1970-1972. We will use the training dataset to “train” a model and forecast the next three points based on the fitted model.

```
train = window(series, start = c(1875), end = c(1969))
test = window(series, start = c(1970), end = c(1972))
# verify our splitting
length(series) == (length(train) + length(test))
```

```
## [1] TRUE
```

Question 1

The data may show a slightly decreasing trend.

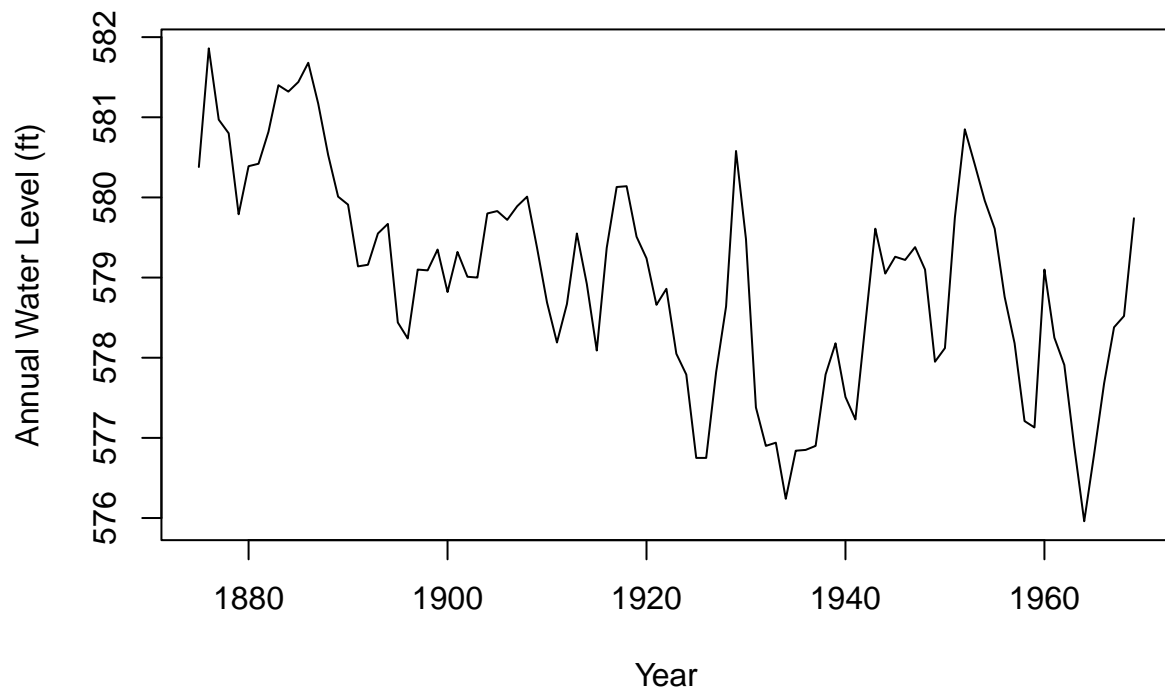
Ignore this possible decreasing trend in what follows.

Plot the training data, its acf and pacf.

Main Plot:

```
plot(train, main = "Annual Water Level of Lake Huron in Feet, 1875-1969",
      ylab = "Annual Water Level (ft)", xlab = "Year")
```

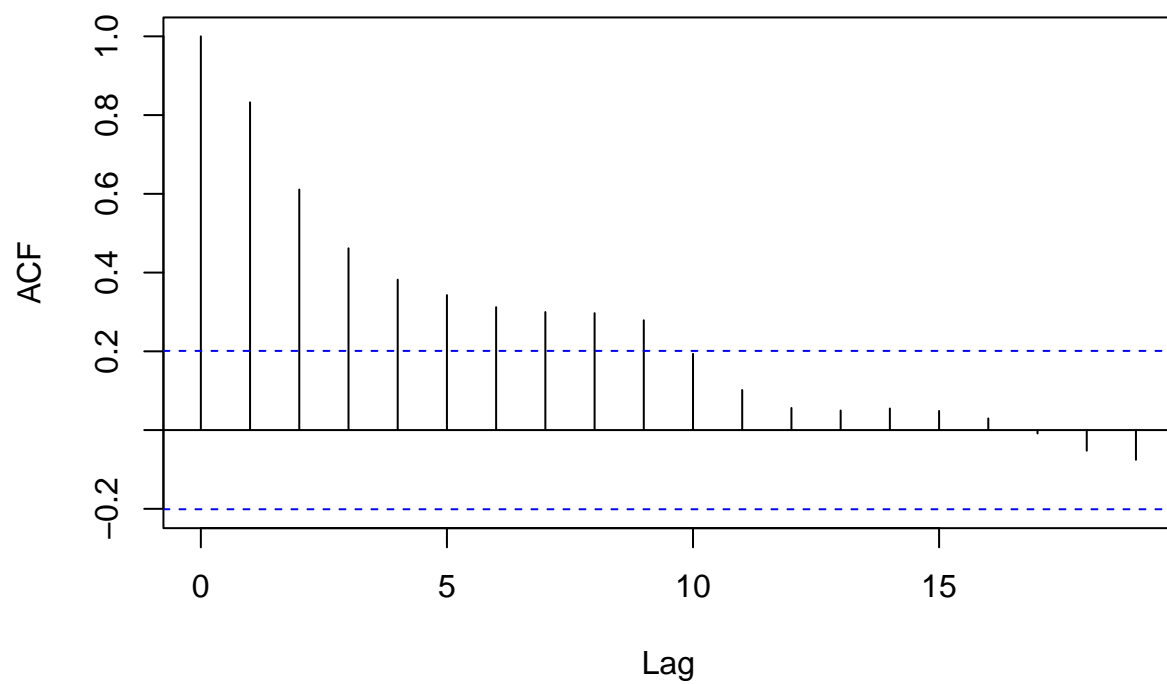
Annual Water Level of Lake Huron in Feet, 1875–1969



ACF:

```
acf(train, main = "ACF of Annual Water Level of Lake Huron in Feet, 1875-1969")
```

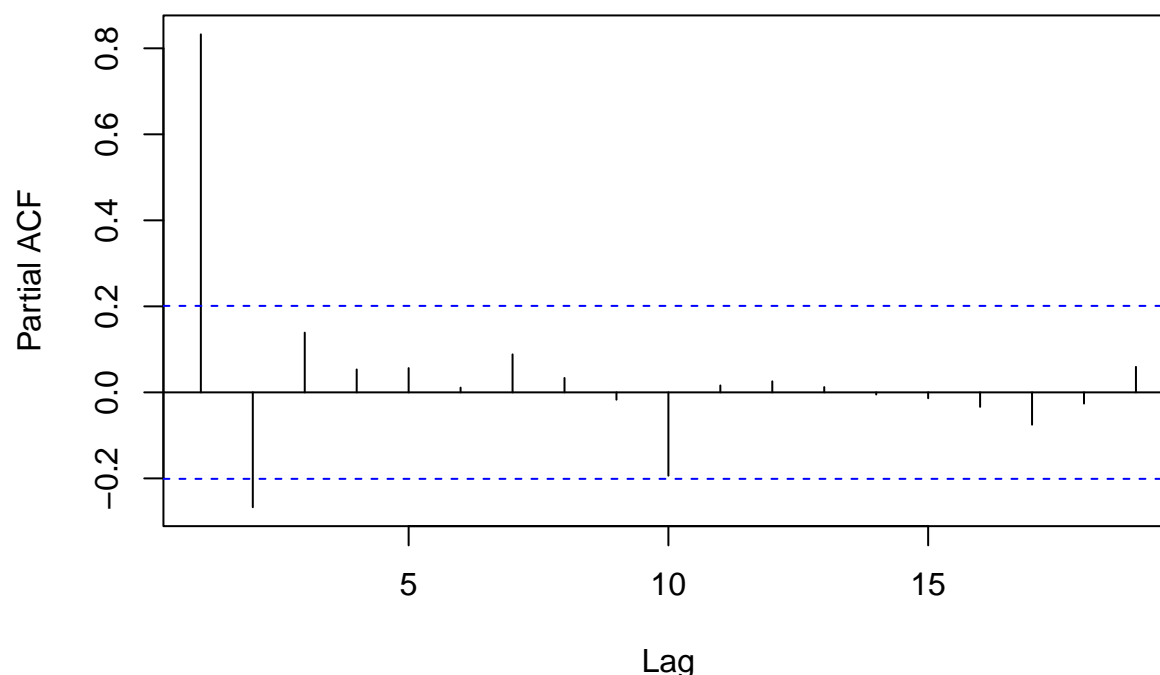
ACF of Annual Water Level of Lake Huron in Feet, 1875–1969



PACF:

```
pacf(train, main = "ACF of Annual Water Level of Lake Huron in Feet, 1875-1969")
```

ACF of Annual Water Level of Lake Huron in Feet, 1875–1969



Determine an appropriate ARMA model and explain your choice.

Firstly, as we are told to ignore the trend, there is no trend component m_t . Further, from inspection, there doesn't seem to be a seasonal component s_t . Hence, it seems logical to assume stationarity for this series.

Then, under stationarity assumptions, we examine the ACF. The ACF seems to be dominated by a slow-decaying trend (not, notably, a dampened sine curve) indicative of a potential AR component. Further, there is a “hint” of a dampened sine component to the positive ACF values. This could suggest a secondary α_2 which is negative. From this ACF, we don't really see a sharp tailing off of ACF values, so I would question the presence of a nonzero MA component in the ARMA process.

So, under the assumptions that an AR composition is sound, we inspect the PACF. As per our analysis of the ACF, there is a fairly large positive observed α_{11} as well as a secondary negative α_{22} . There also seems to be a decently large Partial ACF around lag 10; however, it is tough to visually determine if it is significant. Keeping a parsimonious model in mind, it seems most sensible to choose 2 as the p value such that $\forall k > p, |\hat{\alpha}_{kk}| < 2/\sqrt{n}$ which seems sensible given our assumptions and the observed PACF.

So, with all this in mind, we decide on an $\text{ARMA}(p = 2, q = 0)$ model.

Question 2

Fit the model you chose above using the function `arima()`. Write down the fitted model.

Prior to fitting, we have a theoretical model of the following form. Note that we assume nonzero μ .

$$\text{ARMA}_{p,q}(2,0) : X_t = \mu + Z_t + \alpha_1(X_{t-1} - \mu) + \alpha_2(X_{t-2} - \mu), \text{ where } Z_t \stackrel{\text{iid}}{\sim} \text{WN}(0, \sigma^2)$$

With this in mind, we fit the model (not assuming a zero mean.)

```
p = 2; d = 0; q = 0;
fit = arima(train, order = c(p, d, q), include.mean = TRUE)
(fit)
```

```
##
## Call:
## arima(x = train, order = c(p, d, q), include.mean = TRUE)
##
## Coefficients:
##          ar1          ar2  intercept
##          1.0617   -0.2707   579.0319
## s.e.    0.1006    0.1030    0.3339
##
## sigma^2 estimated as 0.484:  log likelihood = -101.01,  aic = 210.01
```

So, now we can write our fitted ARMA model as follows:

$$\widehat{\text{ARMA}}_{p,q}(2,0) : X_t = \hat{\mu} + Z_t + \hat{\alpha}_1(X_{t-1} - \hat{\mu}) + \hat{\alpha}_2(X_{t-2} - \hat{\mu}), \text{ where } Z_t \stackrel{\text{iid}}{\sim} \text{WN}(0, \hat{\sigma}^2)$$

From our model fitted, we have $\hat{\mu} \approx 579.03$, $\hat{\alpha}_1 \approx 1.06$, $\hat{\alpha}_2 \approx -0.27$ and $\hat{\sigma}^2 \approx 0.48$, hence we can write:

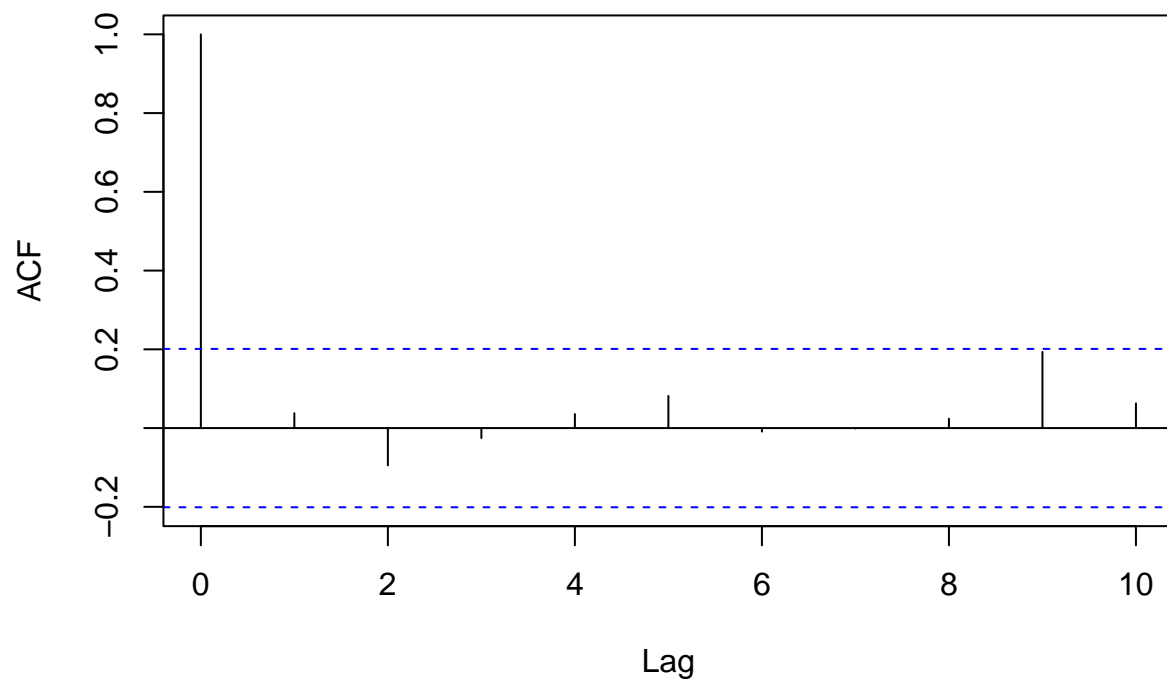
$$X_t = 579.03 + Z_t + 1.06(X_{t-1} - 579.03) - 0.27(X_{t-2} - 579.03), \text{ where } Z_t \stackrel{\text{iid}}{\sim} \text{WN}(0, 0.48)$$

Question 3

Examine appropriate diagnostics for your fitted model.

```
acf(fit$residuals, main = "ACF of Residuals for ARMA fit of Lake Huron Water Levels (in ft)",
    lag.max = 10)
```

ACF of Residuals for ARMA fit of Lake Huron Water Levels (in ft)

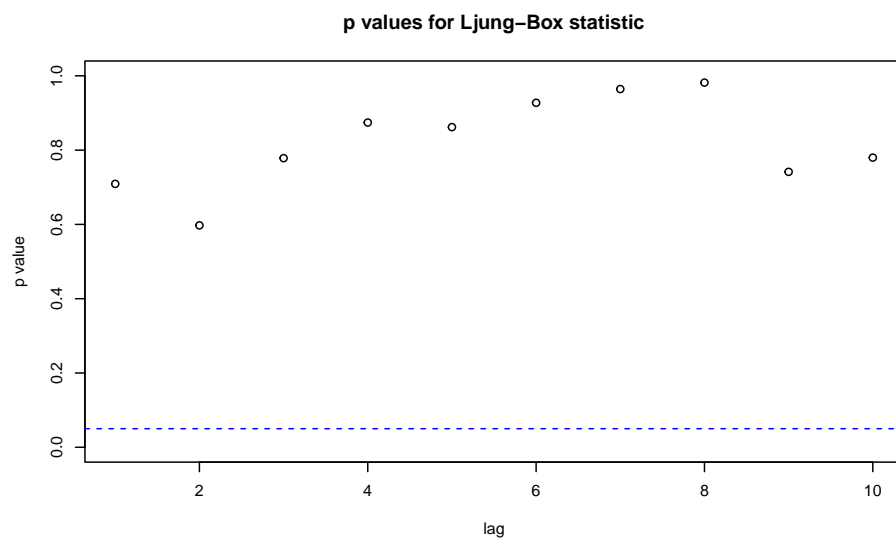
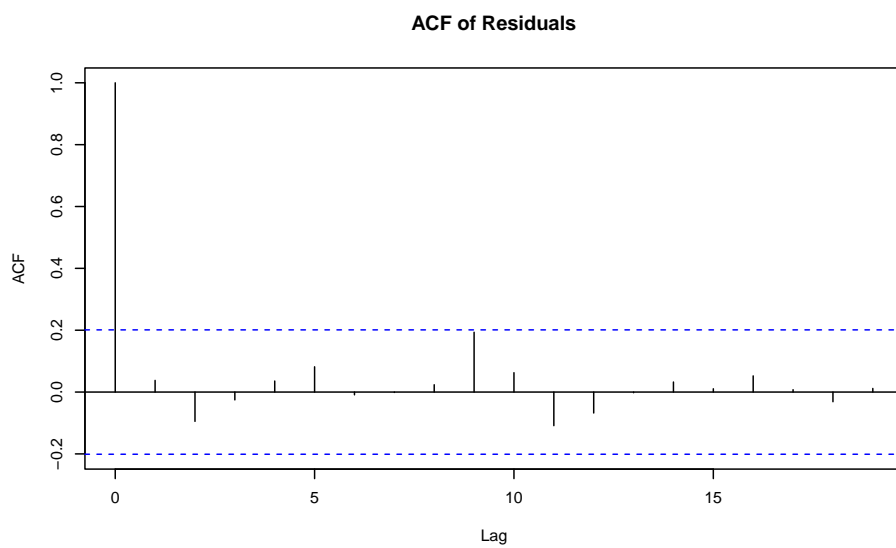
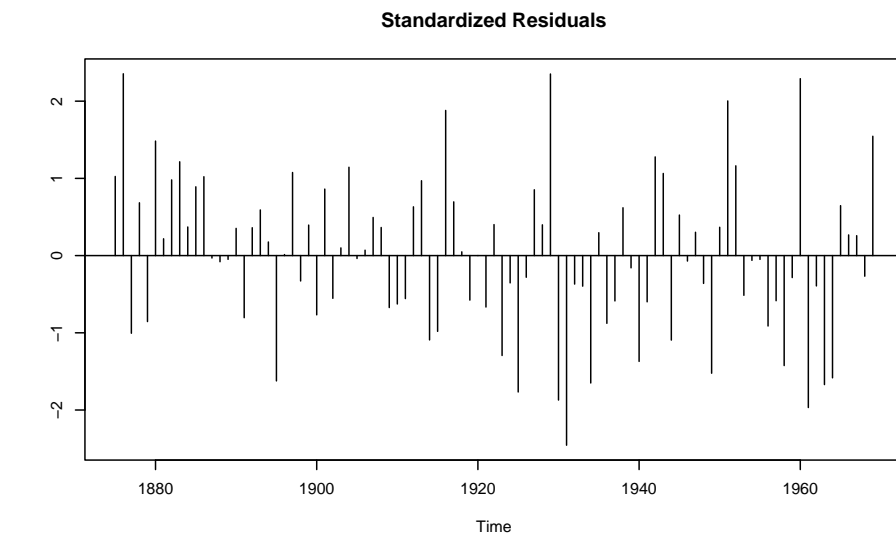


Specifically, first plot the acf of the residuals to see if there are significant autocorrelations after lag 0.

The above plot of the ACF of the residuals has a marginally significant value at around lag 9 (i.e. very nearly above the threshold), which we commented on earlier. While it is tough to determine above the white noise threshold, this value may indicate that there is some other component we are not capturing in our model, which may in fact be significant. It's important to note that the white noise threshold is just a “rule of thumb” and isn't foolproof.

*Then, use `tsdiag()` to observe the standardized residuals and the *p*-values of the Ljung-Box version for the portmanteau test. Report what you observe and then comment on the fit.*

```
tsdiag(fit)
```



From the diagnostic plots above, it seems like there is a lot of noise in the residuals, with lots of standardized values around ± 2 , which is not as low as we'd like to see. This could be due in part to the fact that there is that high acf value around lag 9. However, despite these discrepancies from a “perfect” model, the p -values of the Ljung-Box Portmanteau test do not fall below the α for any common testing levels (i.e. $\alpha \in \{0.01, 0.05, 0.10\}$) which indicates that there is not significant evidence to suggest a lack of fit in the model. However, it should be noted the p -values are lower than we've seen in previous labs with well-fitting models (but this isn't itself evidence for a lack-of-fit!)

Question 4

Use the `predict` command in R to forecast the Lake Huron level for the next three years, i.e., 1970, 1971, and 1972. Provide 95% prediction intervals for each forecast.

Rather than using the reported standard errors, we use the `forecast` package for additional precision

```
library(forecast)

results = data.frame(forecast(fit))

kable(results[1:3, c(1, 4, 5)])
```

	Point.Forecast	Lo.95	Hi.95
1970	579.9223	578.5587	581.2859
1971	579.7856	577.7968	581.7744
1972	579.5911	577.2847	581.8976

Question 5

Compare the forecast with the true values (in test dataset). Comment on what you find.

The values from the test data set are provided below:

```
as.numeric(test)
```

```
## [1] 579.31 579.89 579.96
```

So, the forecasted values are quite close to the true values. Indeed, each of them lie within their respective 95% prediction intervals, as we would hope for a well-fitting model.