Carson Slater
STA 5381 Homework #5: Due 12pm on Friday, May 3.

1. **Bootstrapping Time-Series:** Bootstrapping can be adapted to time series regression, but, as in the case of fixed-X resampling, the procedure makes strong use of the model fit to the data—in particular, the manner in which the serial dependency in the data is modeled. Suppose that the errors in the linear model

$$Y_t = \beta_0 + \beta_1 x_{1t} + \ldots + \beta_p x_{pt} + e_t$$

follow a first order autoregressive process, $e_t = \phi e_{t-1} + \nu_t$, where $\nu_t \sim$ iid $N(0, \sigma_\nu^2)$, and maximum likelihood (ML) estimates of ϕ and $\boldsymbol{\beta}$ are denoted $\hat{\phi}$ and $\hat{\boldsymbol{\beta}}$. From the residuals, $\hat{e}_t = Y_t - \left(\hat{\beta}_0 + \hat{\beta}_1 x_{1t} + \cdots + \hat{\beta}_p x_{pt}\right)$, we can estimate ν_t as $\hat{\nu}_t = \hat{e}_t - \hat{\phi} e_{t-1}$ for $t = 2, 3, \ldots, n$, taking $\hat{\nu}_1 = \hat{e}_1$. To obtain a bootstrapped sample,

   - Sample n values with replacement from the $\hat{\nu}_t$, call them $(\hat{\nu}^*_{b1}, \hat{\nu}^*_{b2}, \ldots, \hat{\nu}^*_{bn})$.
   - Using the $\hat{\nu}^*_{bt}$, construct residuals as $\hat{e}^*_{b1} = \hat{\nu}^*_{b1}$, and $\hat{e}^*_{bt} = \hat{\phi}\hat{e}^*_{b,t-1} + \hat{v}^*_{bt}$ for $t = 2, 3, \ldots, n$.
   - From these residuals and the original fitted values, $\hat{Y}_t = \hat{\beta}_0 + \hat{\beta}_1 x_{1t} + \cdots + \hat{\beta}_p x_{pt}$, construct bootstrapped Y_t values, $Y^*_{bt} = \hat{Y}_t + \hat{e}^*_{bt}$.

   The Y^*_{bt} are used along with the original $\mathbf{x}'_t$ to obtain bootstrap replicates of $\hat{\beta}^*_b$ of the ML coefficient estimates.

   The data you will use to answer the following questions is in the car library and is called Hartnagel. It contains data on Canadian crime-rates for each year from 1931 to 1968. The following variables in this dataset will be needed:

   fconvict: Female indictable-offense conviction rate per 100,000

   tfr: Total fertility rate per 1000 women

   partic: Women's labor-force participation rate per 1000

   The goal is to model the female indictable-offense conviction rate per 100,000 as a function of tfr and partic.

   (a) Report estimates of ϕ and $\boldsymbol{\beta}$ for the data.

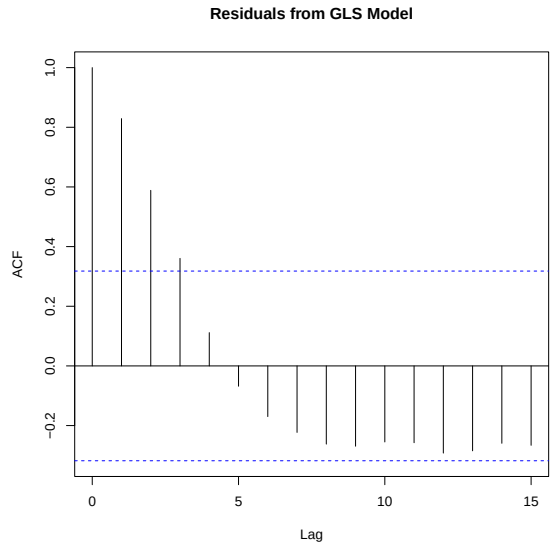
```
ts_data <- Hartnagel

convict_mod <- gls(fconvict ~ tfr + partic, correlation =
   corAR1(form = ~year), data = ts_data, method = "ML")

# (a)

summary(convict_mod)$coefficients

##  (Intercept)          tfr       partic
## 179.28150414  -0.03240886   0.04241892

nlme::intervals(convict_mod)
```

```
## Approximate 95% confidence intervals
##
##  Coefficients:
##                   lower         est.       upper
## (Intercept) 45.87641850 179.28150414 312.6865898
## tfr         -0.06114292  -0.03240886  -0.0036748
## partic      -0.20513290   0.04241892   0.2899708
##
##  Correlation structure:
##        lower      est.    upper
## Phi 0.486054 0.8155952 0.942072
##
##  Residual standard error:
##    lower     est.    upper
## 12.32070 21.05685 35.98750

# Extracting the correlation structure
convict_mod$modelStruct$corStruct

## Correlation structure of class corAR1 representing
##       Phi
## 0.8155952
```

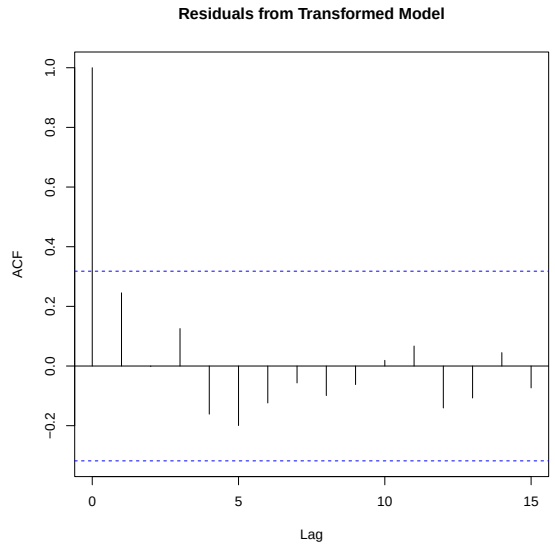
Here we see from the generalized least squares model that $\boldsymbol{\beta} = (179.2815, -0.0324, 0.04242)$, and $\phi = 0.8156$.

(b) Is the AR(1) correlation structure sufficient to remove autocorrelation from the residuals? Justify your answer.

```
phi <- 0.8155952

# ACF Plot for Residuals of Model
acf(convict_mod$residuals, main = "Residuals from GLS Model")
```



```r
# ACF For Transformed Model
convict_lm <- lm(fconvict ~ tfr + partic, data = ts_data)
convict_X <- model.matrix(convict_lm)
convict_Sigma <- matrix(0, nrow = 38, ncol = 38)
convict_Sigma <- phi^abs(row(convict_Sigma) - col(convict_Sigma))
convict_sm <- chol(convict_Sigma)
convict_sm_inv <- solve(t(convict_sm))
convict_xstar <- convict_sm_inv %*% convict_X
convict_ystar <- convict_sm_inv %*% ts_data$fconvict
convict_mod_trans <- lm(convict_ystar ~ convict_xstar-1)
rstandard(convict_mod_trans) |> acf(main = "Residuals from Transformed Model")
```



After transforming the GLS model by right-multiplying by the inverse of the Cholesky decomposition, the model seems to contain no significant serial correlation in the residuals. Hence, we conclude that the the AR(1) correlation structure sufficient to remove autocorrelation from the residuals.

(c) Employ the bootstrapping procedure outlined above with 2,000 bootstrap replicates to

```r
set.seed(612)
n <- nrow(ts_data) # number of bootstraps samples in each replication
b <- 2000 # number of replications

# function
get_nuhat <- function(mod, phi) {
 nu_hat <- numeric(length(mod$residuals))
 nu_hat[1] <- convict_mod$residuals[1]
 for (i in seq_along(mod$residuals)[-1]) {
   nu_hat[i] <- convict_mod$residuals[i] - phi*nu_hat[(i-1)]
 }
 nu_hat
}
```

```
nu_hat <- get_nuhat(convict_mod, phi = 0.8155952)

tfr_coef <- numeric(b)
partic_coef <- numeric(b)

# pretty sure this works
for(i in 1:b) {
 nu <- sample(nu_hat, n, TRUE)

 nu_star <- numeric(length(nu))
 nu_star[1] <- nu[1]
 for (j in seq_along(nu)[-1]) {
   nu_star[j] <- phi*nu_star[(j-1)] + nu[j]
 }

 data <- ts_data |> dplyr::mutate(fconvict = fconvict + nu_star)
 m <- gls(fconvict ~ tfr + partic, correlation = corAR1(form = ~year), data = da

 tfr_coef[i] <- unname(coef(m))[2]
 partic_coef[i] <- unname(coef(m))[3]
}
```

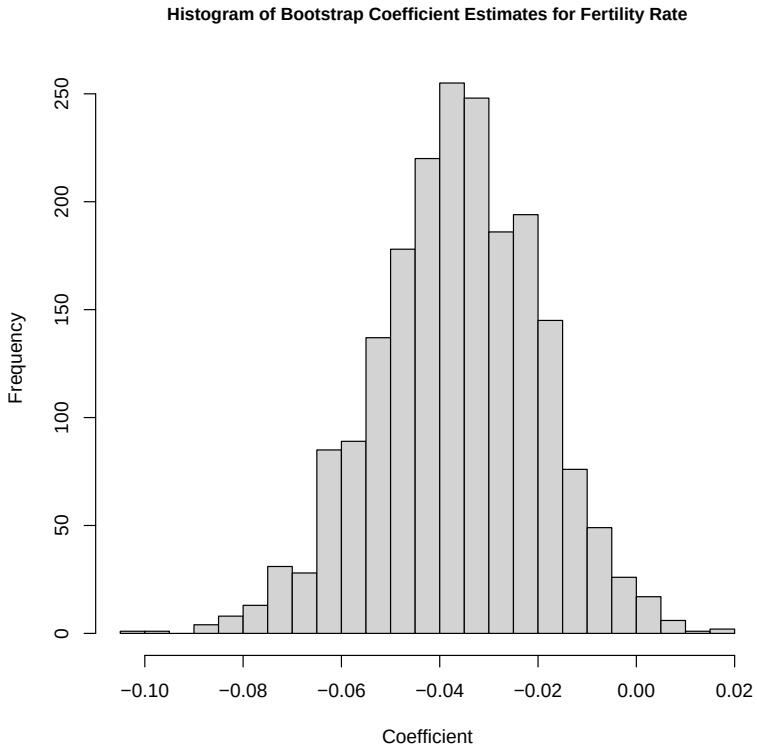
i. Find standard errors of the coefficient estimates.

```
(tfr_se <- sd(tfr_coef) |> round(3))
## [1] 0.017
(partic_se <- sd(partic_coef) |> round(3))
## [1] 0.139
```

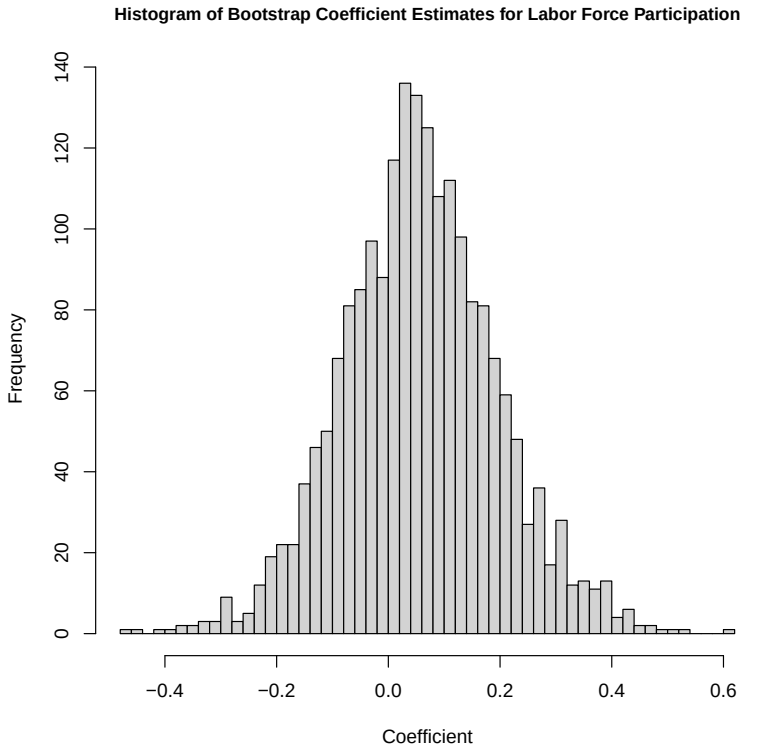
The standard error of the bootstrap coefficient estimates for the tfr variable is 0.017, and for the partic variable is 0.139.

ii. Generate a plot of the distribution of the bootstrap replicates of the tfr and partic coefficients.

```
hist(tfr_coef, breaks = "FD",
     main = "Histogram of Bootstrap Coefficient Estimates for Fertility Rate"
     xlab = "Coefficient",
     cex.main = 0.9)
```




```
hist(partic_coef, breaks = "FD",
    main = "Histogram of Bootstrap Coefficient Estimates for Labor Force Part
    xlab = "Coefficient",
    cex.main = 0.9)
```



iii. Report the 95% percentile bootstrap intervals for β_{tf} and β_{partic}.

```
(tfr_q <- quantile(tfr_coef, c(0.025, 0.975)))
##          2.5%        97.5%
## -0.070952684 -0.004634822
(partic_q <- quantile(partic_coef, c(0.025, 0.975)))
##       2.5%      97.5%
## -0.2092794  0.3529638
```

The bootstrap 95% quantile interval for tfr is $(-0.0709, -0.0046)$, whereas the bootstrap 95% interval quantile for partic is $(-0.2093, 0.3529)$.

iv. Which percentile interval would change the most (the one for β_{tf} or β_{partic}) if the adjusted percentile intervals were computed? Justify your answer.

```
#Adjusted Percentile Interval `tfr`
z <- qnorm((length(which(tfr_coef < convict_mod$coef[2])))/b)

tfr_coef_bar <- mean(tfr_coef)

A <- sum((tfr_coef - tfr_coef_bar)^3)/(6*(sum((tfr_coef - tfr_coef_bar)^2))^1

z_alpha_2 <- qnorm(.975)
A1 <- pnorm(z + (z - z_alpha_2) / (1 - A*(z - z_alpha_2)))
A2 <- pnorm(z + (z + z_alpha_2) / (1 - A*(z + z_alpha_2)))

tfr_lower <- round(b*A1)
tfr_upper <- round(b*A2)

round(sort(tfr_coef)[tfr_lower], 4)
## [1] -0.0622
round(sort(tfr_coef)[tfr_upper], 4)
## [1] 0.0019
#Adjusted Percentile Interval `partic`
z <- qnorm((length(which(partic_coef < convict_mod$coef[3])))/b)

partic_coef_bar <- mean(partic_coef)

A <- sum((partic_coef - partic_coef_bar)^3)/(6*(sum((partic_coef - partic_coe

z_alpha_2 <- qnorm(.975)
A1 <- pnorm(z + (z - z_alpha_2) / (1 - A*(z - z_alpha_2)))
A2 <- pnorm(z + (z + z_alpha_2) / (1 - A*(z + z_alpha_2)))

partic_lower <- round(b*A1)
partic_upper <- round(b*A2)

round(sort(partic_coef)[partic_lower], 4)
```

```
## [1] -0.2317
round(sort(partic_coef)[partic_upper], 4)
## [1] 0.3096
```

The bootstrap 95% quantile interval for tfr is $(-0.0709, -0.0046)$, whereas the adjusted bootstrap 95% interval quantile for tfr is $(-0.0622, 0.0019)$, which is shorter and less conservative. The bootstrap 95% quantile interval for partic is $(-0.2093,\ 0.3529)$, whereas the adjusted bootstrap 95% interval quantile for partic is $(-0.2317, 0.3096)$, which is shorter and less conservative.

The 95% quantile interval for the coefficient for partic changed the most in length, probably because the boostrap sample for partic is less symmetric in distribution.

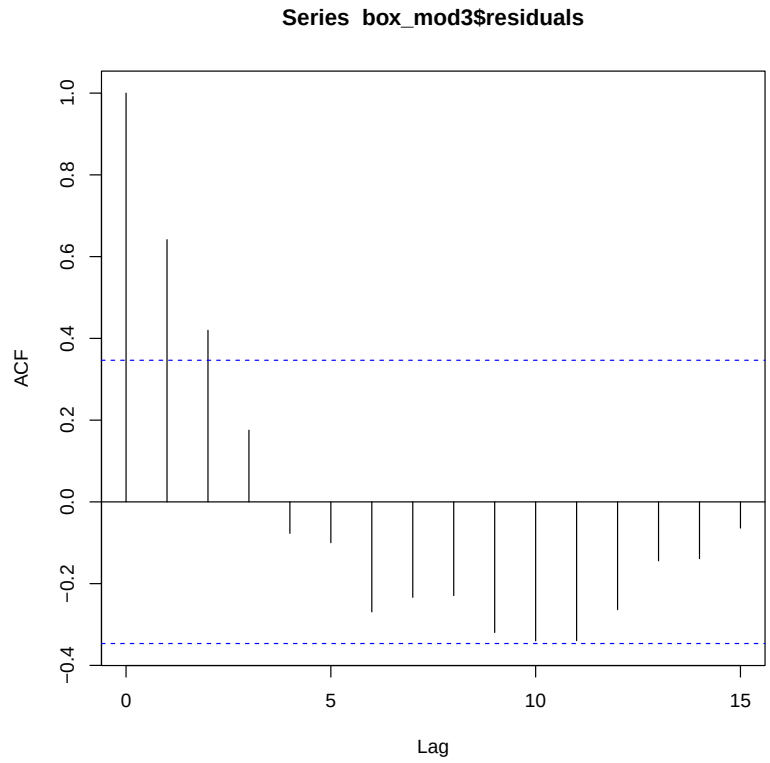
2. Question 1 from Chapter 9 of the textbook.

   (a) This would appear to be modeled well by an AR(1) process. The staffer seems to have solid intuition to make the transformation using the Cholesky decomposition of the $\mathbf{\Sigma}$ matrix to take away serial correlation. Here the residual standard errors seems to have a nonlinear relationship which indicates there is something that is not being accounted for in the model.

   (b) After trying a few different models, the cubic model seems to adequately fit the data. The diagnostic plots for this particular model look good too (see page 11).

```
boxoffice <- read.delim("boxoffice.txt", header = TRUE) |>
  mutate(YearsS1975 = year - 1975)

box_mod3 <- gls(GrossBoxOffice ~ YearsS1975 + I(YearsS1975^2) + I(YearsS1975^3),
            correlation = corAR1(form = ~year), data = boxoffice, method = "ML")
summary(box_mod3)

## Generalized least squares fit by maximum likelihood
##   Model: GrossBoxOffice ~ YearsS1975 + I(YearsS1975^2) + I(YearsS1975^3)
##   Data: boxoffice
##        AIC      BIC   logLik
##   325.9921 334.7865 -156.996
##
## Correlation Structure: AR(1)
##  Formula: ~year
##  Parameter estimate(s):
##       Phi
## 0.6401496
##
## Coefficients:
##                     Value Std.Error   t-value p-value
## (Intercept)     141.33353  52.40362  2.697019  0.0117
## YearsS1975      -20.51780  13.73789 -1.493520  0.1465
## I(YearsS1975^2)   2.87300   0.97456  2.947996  0.0064
## I(YearsS1975^3)  -0.04606   0.01938 -2.377184  0.0245
##
##  Correlation:
##                 (Intr) YS1975 I(YS1975^2
## YearsS1975      -0.817
## I(YearsS1975^2)  0.674 -0.962
## I(YearsS1975^3) -0.583  0.903 -0.984
##
## Standardized residuals:
##        Min         Q1        Med         Q3        Max
## -1.4742360 -0.7002665 -0.2800461  0.9519966  2.0686930
##
## Residual standard error: 42.20712
## Degrees of freedom: 32 total; 28 residual
```

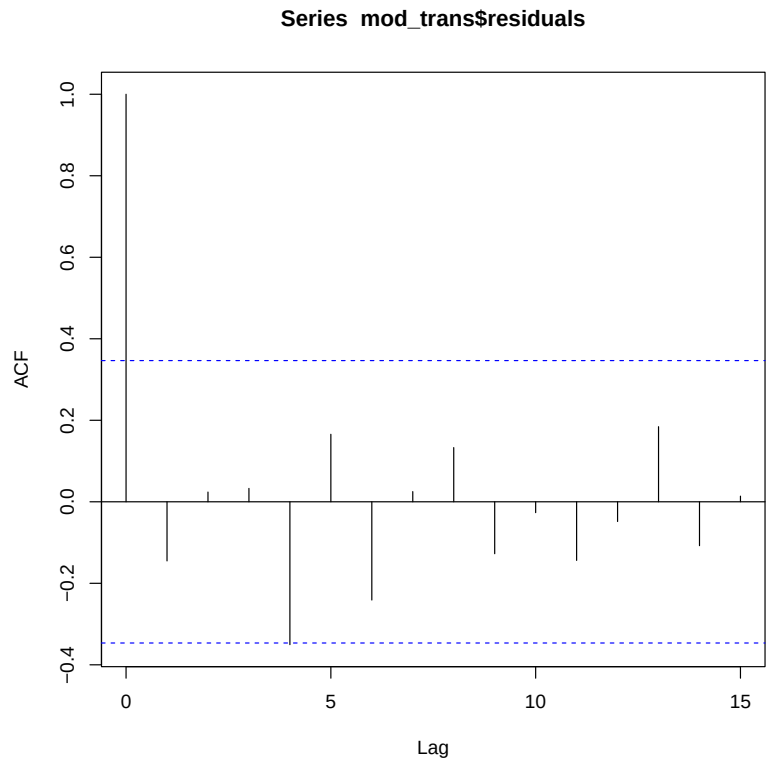
```
acf(box_mod3$residuals)
```



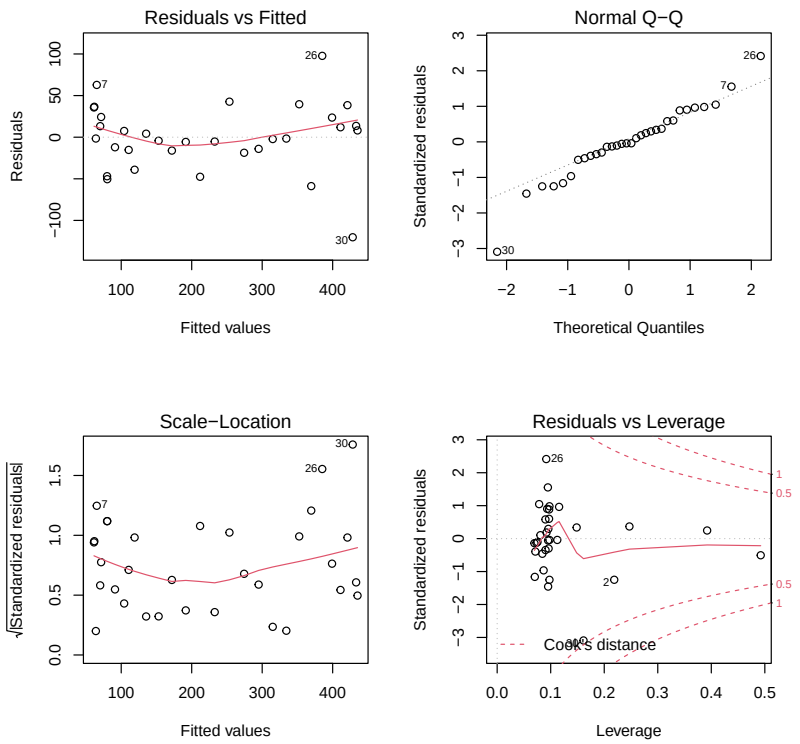

```
box_mod3$modelStruct$corStruct

## Correlation structure of class corAR1 representing
##       Phi
## 0.6401496

mod3 <- lm(GrossBoxOffice ~ YearsS1975 + I(YearsS1975^2) + I(YearsS1975^3),
           data = boxoffice)
phi <- 0.6401496  #Estimate of phi comes from the summary of initial model
X <- model.matrix(mod3)

# Cholesky Transformation
Sigma <- matrix(0, nrow = 32, ncol = 32)
Sigma <- phi^abs(row(Sigma) - col(Sigma))
sm <- chol(Sigma)
sm.inv <- solve(t(sm))
xstar <- sm.inv %*% X
ystar <- sm.inv %*% boxoffice$GrossBoxOffice
new_data <- data.frame("ystar"    = ystar,
                       "xstar"    = xstar[,2],
                       "xstar_sq" = xstar[,3],
                       "xstar_cb" = xstar[,4])
mod_trans <- lm(ystar ~ xstar + xstar_sq + xstar_cb, data = new_data)
summary(mod_trans)
```

```
##
## Call:
## lm(formula = ystar ~ xstar + xstar_sq + xstar_cb, data = new_data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -120.25  -15.49   -1.63   23.75   97.73
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 166.77102   48.49925   3.439 0.001849 **
## xstar       -61.19296   21.93878  -2.789 0.009395 **
## xstar_sq      5.09800    1.34525   3.790 0.000736 ***
## xstar_cb     -0.08279    0.02452  -3.376 0.002172 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 42.47 on 28 degrees of freedom
## Multiple R-squared:  0.9214,Adjusted R-squared:  0.913
## F-statistic: 109.4 on 3 and 28 DF,  p-value: 1.433e-15
```

```
acf(mod_trans$residuals)
```




```
par(mfrow = c(2, 2))
plot(mod_trans)
```



In our analysis, we found that the third-order polynomial fit of YearsS1975 captures the nonlinear relationship in the data. The ACF plot of the standardized residuals of our transformed third-order model. In the tails, there are violations of normality in the diagnostic QQ-plots, but those are present in all reasonable model fits. We found that our residuals vs. fitted plot shows a nonlinear relationship, but it was not taken out by adding fourth and higher order terms - this was also exacerbated in the first and second degree models.

(c)
```
predict(mod_trans, newdata = data.frame("xstar"    = 33,
                                        "xstar_sq" = 33^2,
                                        "xstar_cb" = 33^3
                                        ))

##        1
## 723.7348
```

The model predicts that in 2008, GrossBoxOffice will be 723.73.

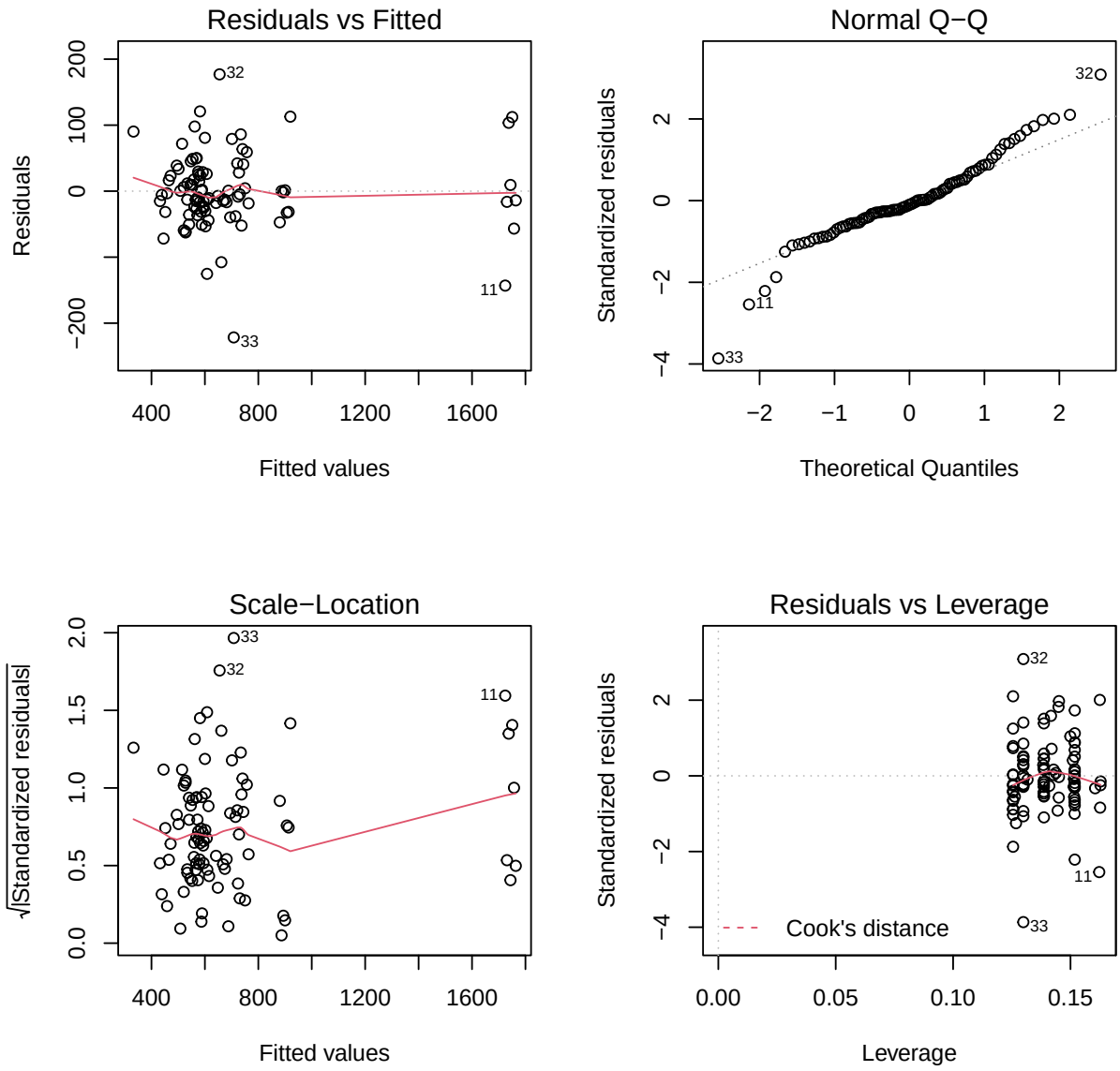
(d) Looking at the model diagnostic plots in part (b), observation 26 is noted as an observation that possesses a high residual value from the model; however, it does not have a high leverage. In a technical sense, it is not an outlier. Also it does not have the highest residual value. In short, in a coloquial sense it could be consider 'out of the ordinary,' but it is not an outlier in a technical sense.

3. Question 2 from Chapter 9 of the textbook. Add part (c): Instead of including month as a dummy variable, would harmonics fit the monthly trend in the data? Use two pairs of harmonics to fit that trend in the sales, and plot the fitted trend on top of the sales by month. Try three pairs of harmonics as well, and comment on the difference in fit between it and the two pair model.

```
books <- read.delim("bookstore.txt", header = TRUE)

books_lm <- lm(Sales ~ Time + Month_2 + Month_3 + Month_4 + Month_5 + Month_6 +
           data = books)
lmtest::dwtest(books_lm, alternative = "two.sided")

##
##  Durbin-Watson test
##
## data:  books_lm
## DW = 2.4091, p-value = 0.06354
## alternative hypothesis: true autocorrelation is not 0
```

The Durbin-Watson test for serial correlation is not significant at the $\alpha = 0.05$ level, but it was close. We will use generalized least squares to model this, to see if the confidence interval for the ϕ value contains zero.

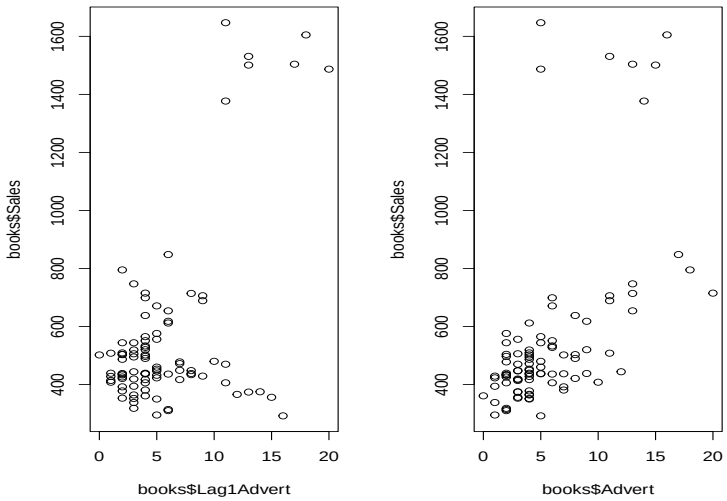
```
books_mod <- gls(Sales ~ Time + Month_2 + Month_3 + Month_4 + Month_5 + Month_6
                 correlation = corAR1(form = ~Time), data = books, method = "ML"

books_mod$modelStruct$corStruct

## Correlation structure of class corAR1 representing
##        Phi
## -0.2330868

phi <- -0.2330868
X <- model.matrix(books_lm)

# Cholesky Transformation
Sigma <- matrix(0, nrow = 93, ncol = 93)
Sigma <- phi^abs(row(Sigma) - col(Sigma))
sm <- chol(Sigma)
sm.inv <- solve(t(sm))
xstar <- sm.inv %*% X
ystar <- sm.inv %*% books$Sales
mod_trans <- lm(ystar ~ xstar-1)
par(mfrow = c(2,2))
plot(mod_trans)
```



(a) From this initial model, we observe deviations from normality with right skewness in the QQ-Plot of the residuals, we also observe what appears to be heteroscedastic variance in the plot of the square root of the standardized residuals vs. the fitted values (bottom right).

(b) We will fit a second model considering the amount of advertising the bookstore has done in that respective month and the prior month.

```
par(mfrow = c(1,2))
plot(books$Lag1Advert, books$Sales)
plot(books$Advert, books$Sales)
```




```
par(mfrow = c(1,1))

books_mod2 <- gls(Sales ~ Time + Advert + Lag1Advert + Month_2 + Month_3 + Month
              correlation = corAR1(form = ~ Time), data = books, method = "ML")

books_lm2 <- lm(Sales ~ Time + Advert + Lag1Advert + Month_2 + Month_3 + Month_4
                data = books) # this has gotta be my final model.
summary(books_lm2)
```

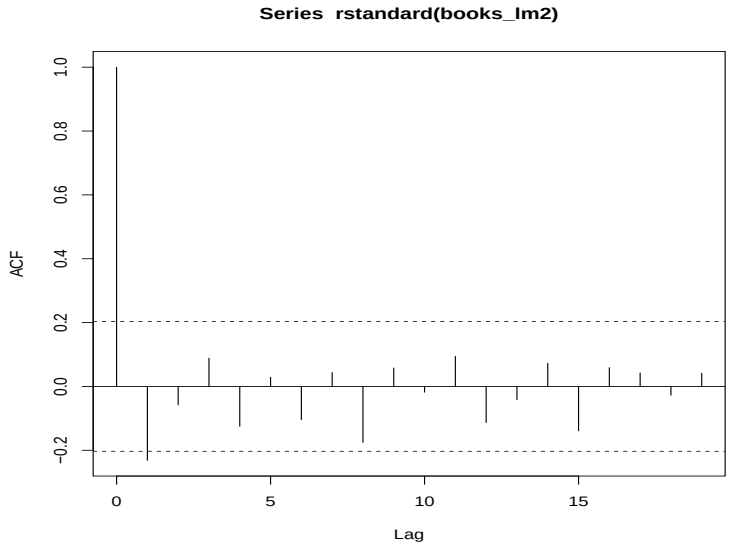
```
##
## Call:
## lm(formula = Sales ~ Time + Advert + Lag1Advert + Month_2 + Month_3 +
##     Month_4 + Month_5 + Month_6 + Month_7 + Month_8 + Month_9 +
##     Month_10 + Month_10 + Month_11 + Month_12, data = books)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -240.283  -28.203   -6.308   28.352  167.624
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  392.3619    40.8122   9.614 6.99e-15 ***
## Time           0.4322     0.2377   1.818 0.072853 .
## Advert         2.2999     2.5743   0.893 0.374385
## Lag1Advert    -4.5174     2.6625  -1.697 0.093744 .
## Month_2      -45.8840    37.8276  -1.213 0.228799
## Month_3       26.6907    40.5869   0.658 0.512720
## Month_4       81.0497    33.5470   2.416 0.018030 *
## Month_5       59.3321    36.7394   1.615 0.110361
## Month_6       59.3564    36.4203   1.630 0.107184
## Month_7      -12.4035    37.4233  -0.331 0.741203
## Month_8       60.0235    35.9118   1.671 0.098648 .
```

```
## Month_9      129.0218    37.2445   3.464 0.000867 ***
## Month_10     165.7306    35.4326   4.677 1.20e-05 ***
## Month_11     324.7993    43.5922   7.451 1.08e-10 ***
## Month_12    1149.1212    39.4806  29.106  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 60.55 on 78 degrees of freedom
## Multiple R-squared:  0.9653,Adjusted R-squared:  0.9591
## F-statistic:   155 on 14 and 78 DF,  p-value: < 2.2e-16
```

```
lmtest::dwtest(books_lm2, alternative = "two.sided")
```

```
##
##  Durbin-Watson test
##
## data:  books_lm2
## DW = 2.4223, p-value = 0.05437
## alternative hypothesis: true autocorrelation is not 0
```

```
acf(rstandard(books_lm2))
```

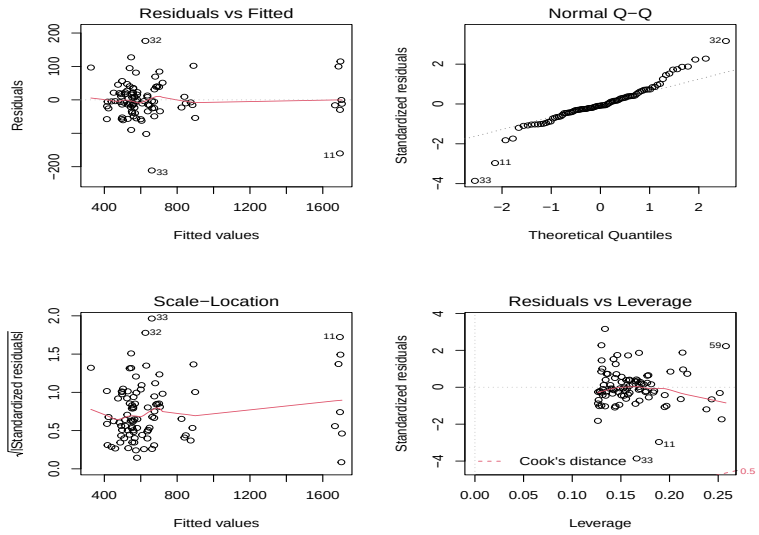



```
phi <- -0.1879788  #Estimate of phi comes from the summary of initial model
X <- model.matrix(books_lm2)

# Cholesky Transformation
Sigma <- matrix(0, nrow = 93, ncol = 93)
Sigma <- phi^abs(row(Sigma) - col(Sigma))
sm <- chol(Sigma)
sm.inv <- solve(t(sm))

xstar <- sm.inv %*% X
```

```r
ystar <- sm.inv %*% books$Sales
mod_trans <- lm(ystar ~ xstar-1)
summary(mod_trans)
```

```
##
## Call:
## lm(formula = ystar ~ xstar - 1)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -211.333  -24.438   -4.582   22.491  176.406
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## xstar(Intercept)  393.5653    38.4834  10.227 4.63e-16 ***
## xstarTime           0.4382     0.1959   2.237 0.028122 *
## xstarAdvert         2.4619     2.5039   0.983 0.328541
## xstarLag1Advert    -4.9399     2.5957  -1.903 0.060713 .
## xstarMonth_2      -45.8964    38.9412  -1.179 0.242137
## xstarMonth_3       24.7561    40.2397   0.615 0.540202
## xstarMonth_4       81.7743    33.2084   2.462 0.016004 *
## xstarMonth_5       58.8279    36.2561   1.623 0.108717
## xstarMonth_6       59.1464    35.7768   1.653 0.102308
## xstarMonth_7      -13.1917    37.0138  -0.356 0.722504
## xstarMonth_8       59.9402    35.3096   1.698 0.093577 .
## xstarMonth_9      128.1907    36.9314   3.471 0.000848 ***
## xstarMonth_10     165.3548    35.5134   4.656 1.30e-05 ***
## xstarMonth_11     321.9293    43.5977   7.384 1.45e-10 ***
## xstarMonth_12    1152.7657    40.6301  28.372  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 59.96 on 78 degrees of freedom
## Multiple R-squared:  0.9945,Adjusted R-squared:  0.9935
## F-statistic: 944.6 on 15 and 78 DF,  p-value: < 2.2e-16
```

```r
par(mfrow = c(2,2))
plot(mod_trans)
```

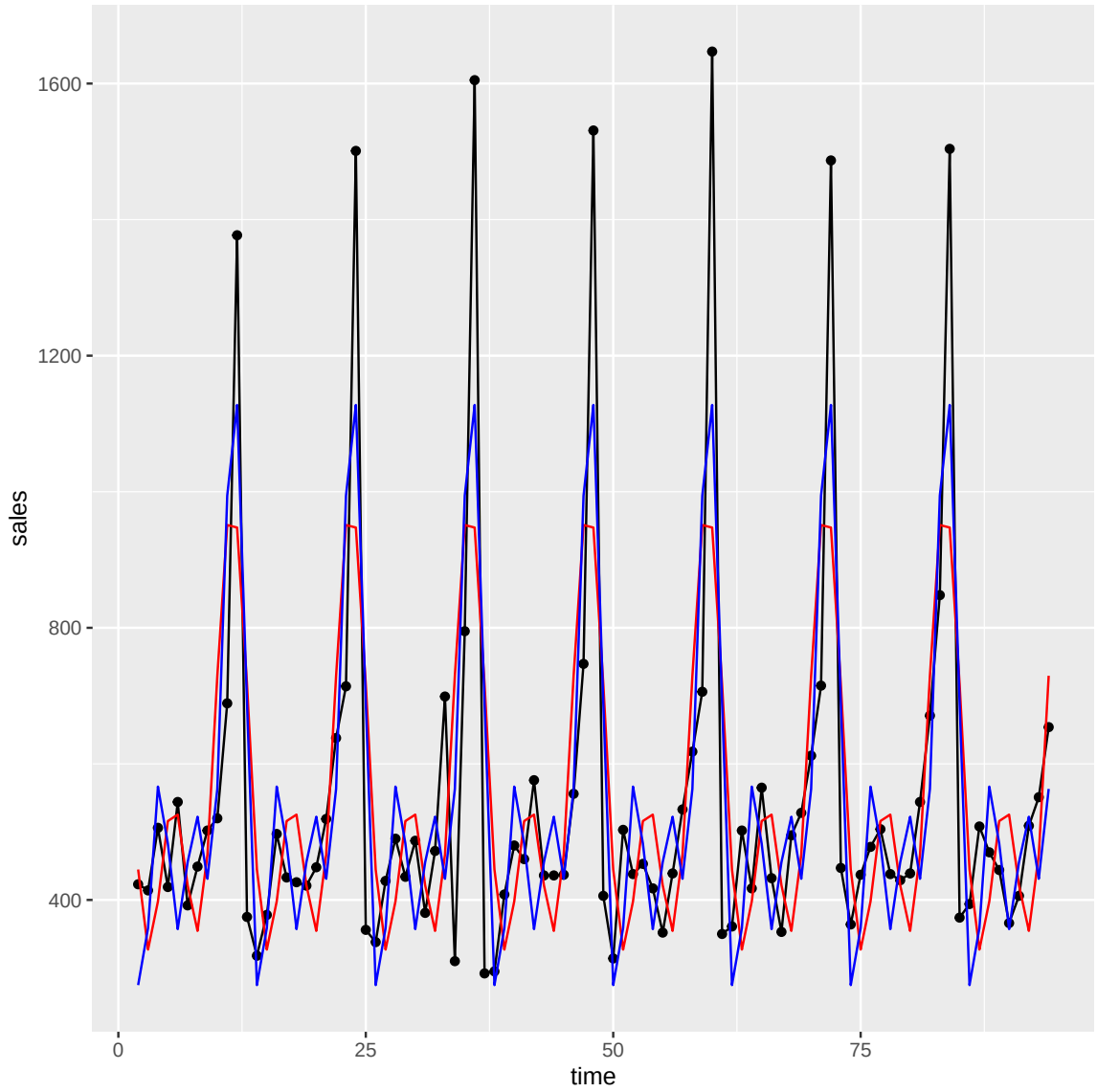



```
par(mfrow = c(1,1))
```

We fit a model including the advertising terms, but there exists some of the same problems in the prior model. In looking at the relationship between advertising terms and sales, the only significant relationship, as seen in the visualization and the transformed generalized least squares model, is the relationship between Lag1Advert and Sales. It is only marginally significant thought. Further in our analysis, we discovered that if we remove the Advert term, the Lag1Advert loses its significance, hence we elect to keep this model as our final model. We also discovered that higher order quadratics do not improve model fit, or post-hoc diagnostics.

(c) The harmonic fits are pictured below.

```
books <- books |> mutate(rads = (Time %% 12)* 30*pi/180)

mod_harm2 <- lm(Sales ~ cos(rads) + sin(rads) + cos(2*rads) + sin(2*rads), data
mod_harm3 <- lm(Sales ~ cos(rads) + sin(rads) + cos(2*rads) + sin(2*rads) + cos(

tibble(
  "time"  = books$Time,
  "fit2"  = predict(mod_harm2, newdata = books),
  "fit3"  = predict(mod_harm3, newdata = books),
  "sales" = books$Sales
) |> ggplot(aes(x = time, y = sales)) +
  geom_point() +
  geom_line() +
  geom_line(aes(time, fit2), color = "red") +
  geom_line(aes(time, fit3), color = "blue")
```

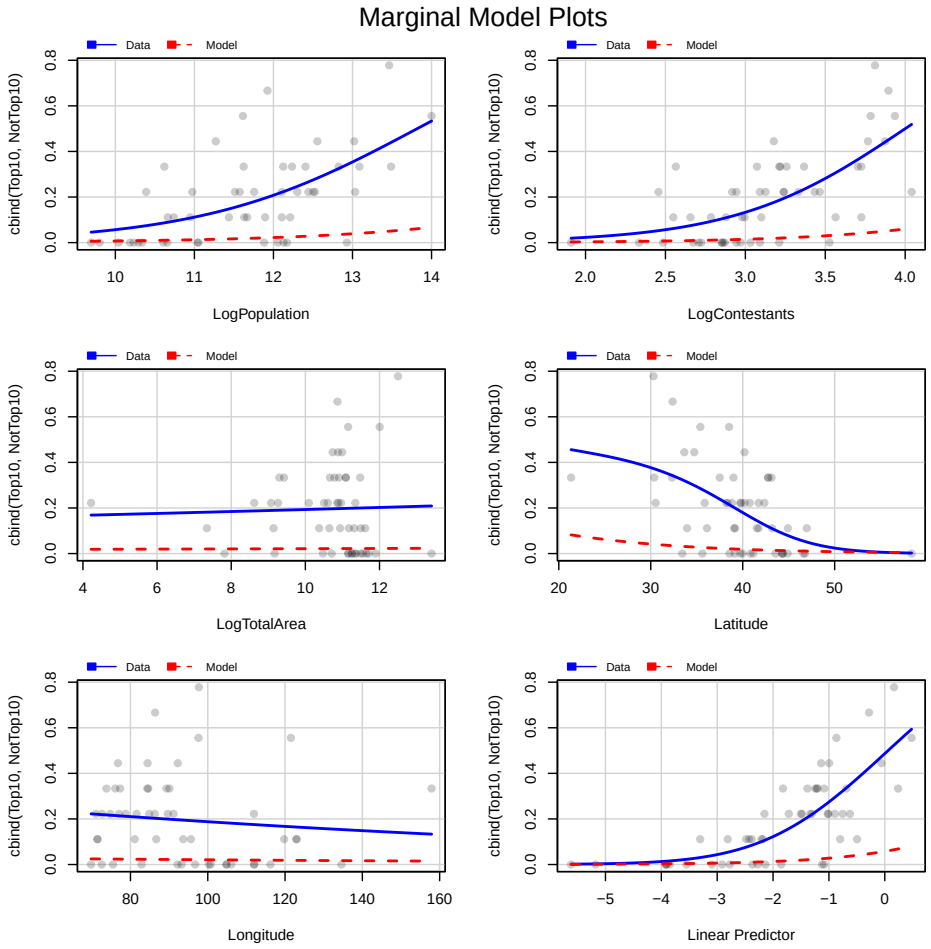


It appears the three pairs of harmonics fits better than the two pair harmonics, as it capture the peaks much more definitively.

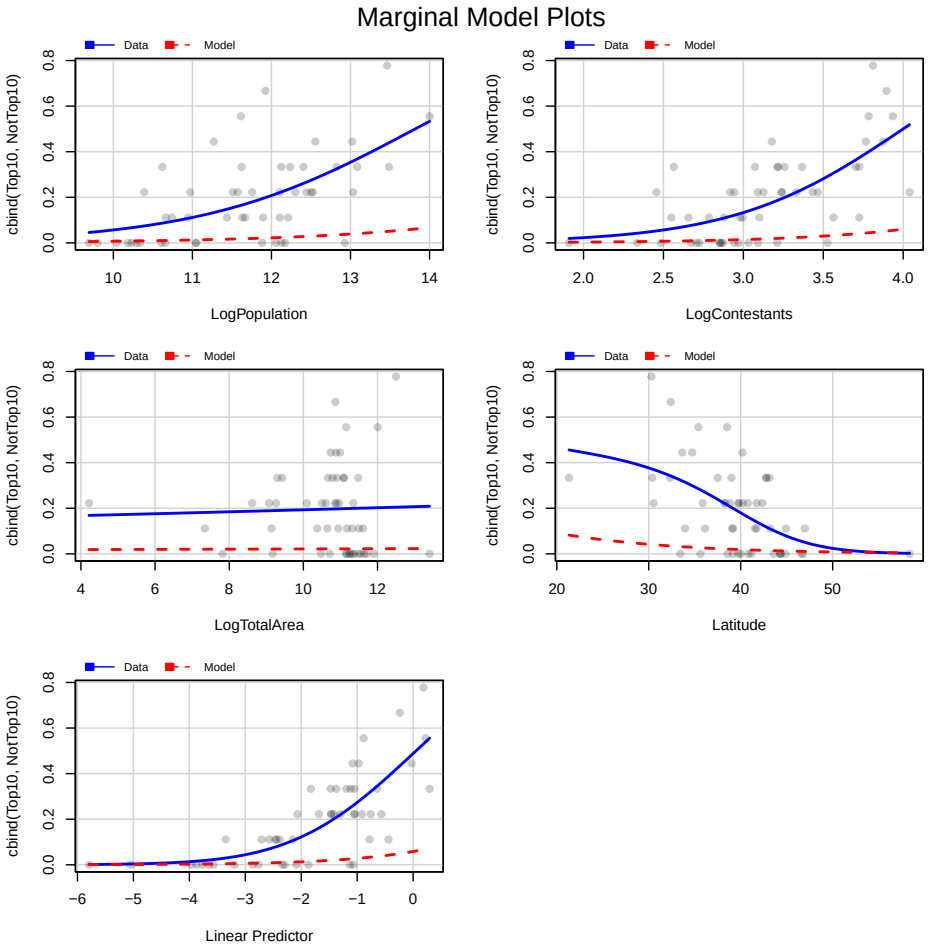
4. Problem 2 from Chapter 8 of the textbook.

   (a) We will begin with the full model and check if there are significant coefficients.

```
# (a)
ms_america <- read.delim("MissAmericato2008.txt", header = TRUE)

# transform response to be a proportion
ms_america <- ms_america |> mutate(NotTop10 = 9 - Top10,
                                   Top10Ever = ifelse(Top10 > 0, 1, 0),
                                   propTop10 = Top10/9,
                                   total = 9)

# kichen sink
log_reg_full <- glm(cbind(Top10, NotTop10) ~ LogPopulation + LogContestants + Lo

# got rid of longitude - looks good on paper
log_reg1 <- glm(cbind(Top10, NotTop10) ~ LogPopulation + LogContestants + LogTot
# Marginal Model Plots - pretty bad
mmps(log_reg_full, pch = 19, col = alpha(1, 0.2)) # full model
```



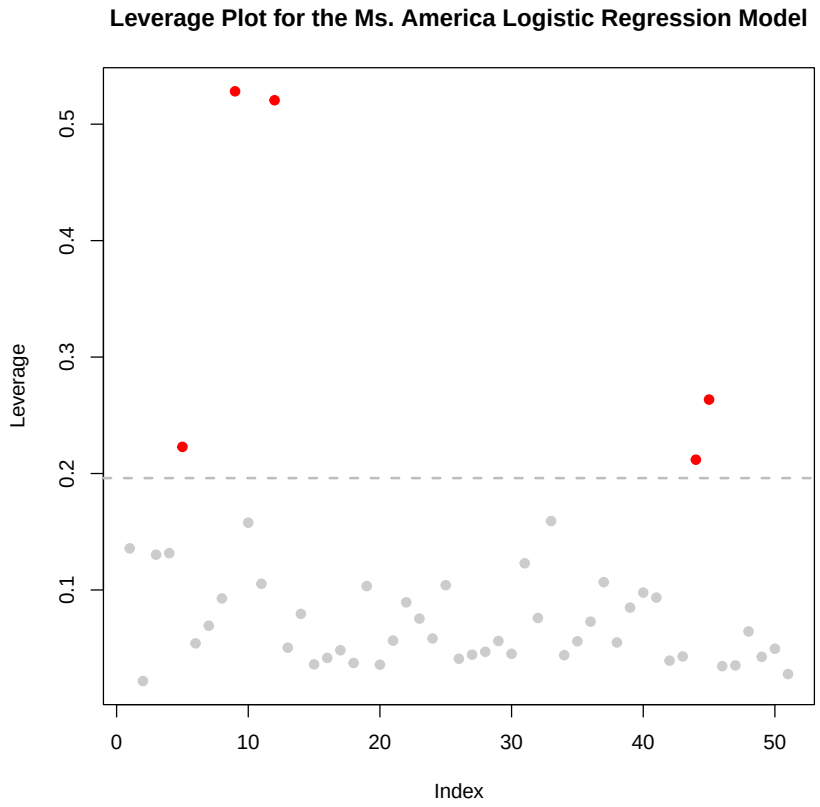
```
mmps(log_reg1, pch = 19, col = alpha(1, 0.2)) # reduced model
```



After fitting the full model, we discovered that the coefficient on Latitude was insignificant, so we fit a reduced model without it that yielded all significant coefficients. The marginal model plots for the full and reduced models look similar - they are both terrible, which indicates this model is not a good model.

(b) To find if there any high leverage observations, we can compute the leverage of each observation.

```
hii <- hatvalues(log_reg1)
cut.hii <- 2*5/nrow(ms_america)

plot(hii, pch = 19,
     col = ifelse(hii %in% hii[which(hii > cut.hii)],"red",alpha(1, 0.2)),
     main = "Leverage Plot for the Ms. America Logistic Regression Model",
     xlab = "Index",
     ylab = "Leverage")
abline(h = cut.hii, lty = 2, col = "gray", lwd = 2)
```




```
ms_america$abbreviation[which(hii > cut.hii)]

## [1] "CA" "DC" "HI" "TX" "UT"

rstandard(log_reg1)[which(hii > cut.hii)]

##           5           9          12          44          45
## -0.00103651  0.28854570  1.51378418  1.62963035 -0.28993791

# these are the states with high leverage observations
```

The states/territories with high leverage observations are California, Washington DC, Hawaii, Texas, and Utah. In our analysis, we conclude that none of the standardized residuals for those high leverage points are high (greater than two), so none of them are 'bad' leverage points, so to speak.

(c)

- The interpretation for the coefficient for x_1 is that for every 1 unit increase in the log of the population each state/territory, the odds of having a Ms. America pageant contestant from that respective state in the top 10 from 2000-2008 increases by a factor of $e^{0.5888} = 1.802$.
- The interpretation for the coefficient for x_3 is that for every 1 unit increase in the log of the geographic area of each state/territory, the odds of having a Ms. America pageant contestant from that respective state in the top 10 from 2000-2008 increases by a factor of $e^{1.3369} = 3.807$.
- The interpretation for the coefficient for x_3 is that for every 1 unit increase in the log of the geographic area of each state/territory, the odds of having a Ms.

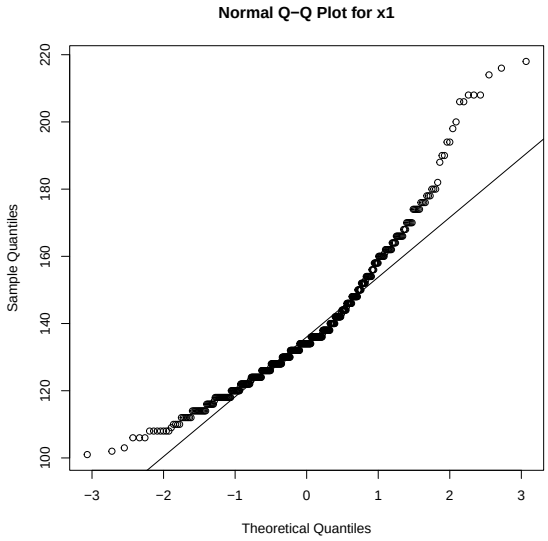
America pageant contestant from that respective state in the top 10 from 2000-2008 decreases by a factor of $e^{-0.3198} = 0.726$.

- The interpretation for the coefficient for x_4 is that for every 1 unit increase in the latitude of the geographic area of each state/territory, the odds of having a Ms. America pageant contestant from that respective state in the top 10 from 2000-2008 decreases by a factor of $e^{-0.0733} = 0.9293$.

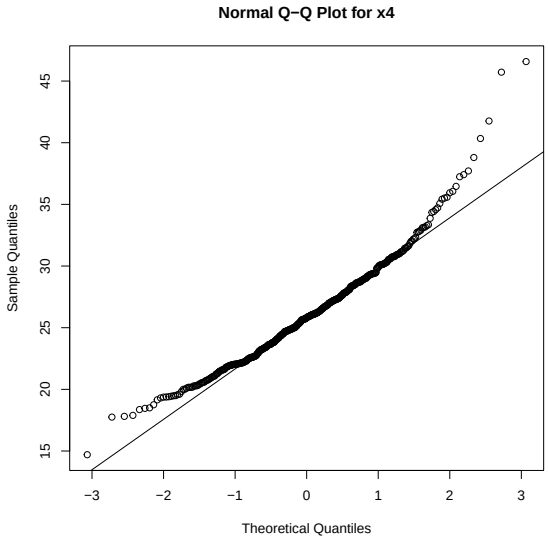
5. Problem 4 from Chapter 8 of the textbook.

   (a) This is a viable model; however, we recognize that the KDE's of x1 and x4 are right skewed, which violates normality assumptions. Perhaps a variance-stabilizing transformation for those two covariates could make a more robust model.

```
heart <- read.csv("HeartDisease.csv", header = TRUE)

heart_mod <- glm(formula = HeartDisease ~ x1 + x2 + x3 + x4 + x5, family = binom
qqnorm(heart$x1, main = "Normal Q-Q Plot for x1")
qqline(heart$x1)
```




```
qqnorm(heart$x4, main = "Normal Q-Q Plot for x4")
qqline(heart$x4)
```



   (b) Even the QQ plots for x1 and x4 demonstrate that they violate normality assumptions to fit a model. To violate the model assumptions, define f1x1 $= \log(x_1)$ and

f2x4 $= \log(x_4)$. Both of these transformations would condense the ranges of x_1 and x_4, shrinking and stabilizing the variance, as the KDE's of both x1 and x4 were found to be non-normal in part (a).

(c) Based on strictly the marginal model plots (which is what the question asks), this model is a valid model for the data. All of the marginal model plots indicate that the fit between the nonparametric curve and the model are very close to one another.

(d) The coefficient for x_3 in the second model is $\hat{\beta}_{x_3} = 0.904$. This means that if a person has a family history of heart disease, the odds of that patient having heart disease also increases by a factor of $e^{0.9411} = 2.5626$.

6. A traffic engineering study was conducted to evaluate the effects of three traffic signal types on traffic delay at intersections during both rush and non-rush hours. The three signal types were pre-timed, semi-actuated, and fully actuated signals. A sample of 4 intersections were selected to test the signal types, and estimated stop time per vehicle at the intersection was recorded along with the intersection, traffic signal type, and whether it was rush hour or not. Identify the response and the factors in this experiment. Also, identify whether each factor is fixed or random.

   (a) The response is the stop time of a car at an intersection. The factors are the rush hour vs. not rush hour (*fixed*), the three signal types (*fixed*), and the four different intersections (*random*).

7. An experiment was conducted to test the effects of nitrogen fertilizer on lettuce production. Five rates of ammonium nitrate were applied to four replicate plots in a completely randomized design. The data are the number of heads of lettuce harvested from each plot and are given in "lettuce.txt."

   (a) Write the linear model for this study, and explain the model components.

   The linear model is the treatment means model,

   $$Y_{ij} = \mu + \tau_i + \varepsilon_{ij}.$$

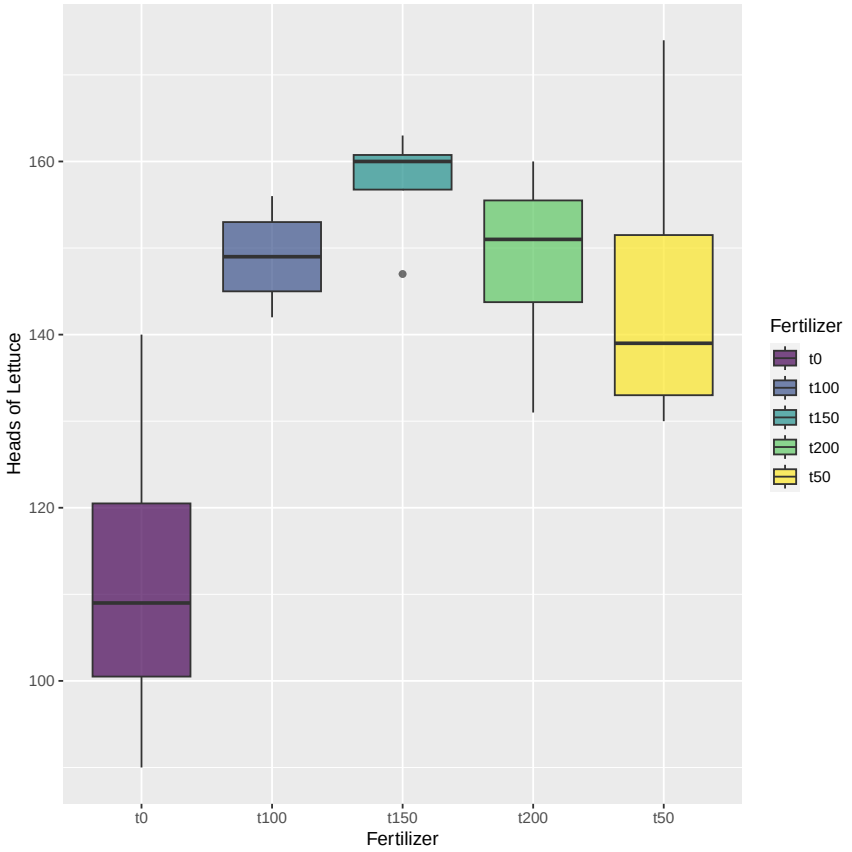
   where:

   $$\begin{aligned} Y_{ij} &: \text{Observed value in group } i \text{ and replicate } j \\ \mu &: \text{ Grand mean} \\ \tau_i &: \text{ Effect of treatment (fertilizer) } i \\ \varepsilon_{ij} &: \text{ Random error term for group } i \text{ and replicate } j \end{aligned}$$

   We also assume that $\varepsilon_{ij} \overset{iid}{\sim} \mathcal{N}(0, \sigma^2)$.

   (b) Display the side-by-side boxplot, and describe what you see.

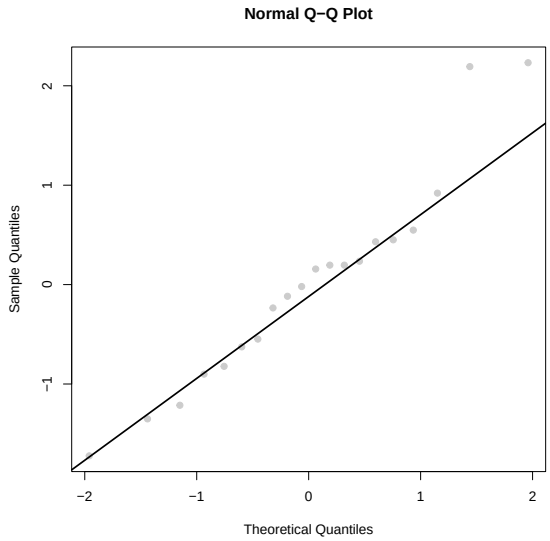
```
lettuce <- read.delim("lettuce.txt", header = TRUE, sep = '') |>
  mutate(fert = as.factor(fert))
lettuce_mod <- lm(heads ~ ., data = lettuce)

lettuce |> ggplot(aes(x = fert, y = heads, fill = fert)) +
  geom_boxplot(alpha = 0.7) +
  scale_fill_viridis_d() +
  labs(fill = "Fertilizer",
       x = "Fertilizer",
       y = "Heads of Lettuce")
```



In the boxplot, it is obvious that the yield of lettuce heads for `t0` is far lower than the other fertilizer types.

(c) Check the assumptions for the model.
For the model, we have to check if the standard deviations across predictors is the same, as well as if the residuals are normally distributed.

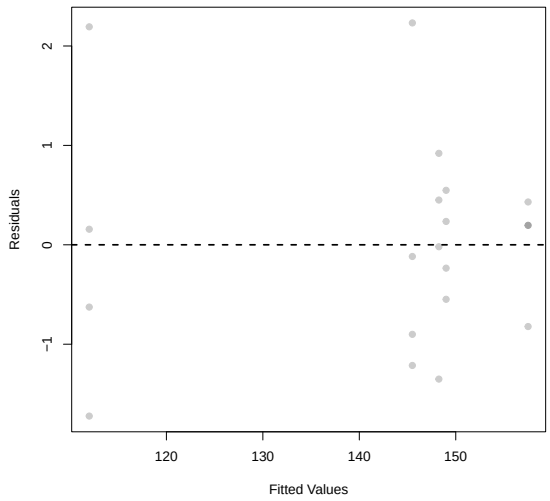
```
# checking for normal residuals
res <- rstandard(lettuce_mod)
qqnorm(res, pch = 19, col = alpha(1, 0.2))
qqline(res, lwd = 2)
```




```
shapiro.test(res)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  res
## W = 0.94246, p-value = 0.2668
```

```
# checking for constant variance
plot(fitted(lettuce_mod), res, pch = 19, xlab = "Fitted Values", ylab = "Residua
abline(h = 0, lwd = 2, lty = 2)
```




```
leveneTest(lettuce$heads, lettuce$fert)
```

```
## Levene's Test for Homogeneity of Variance (center = median)
##       Df F value Pr(>F)
## group  4  1.0234 0.4269
```

```
##       15
```

For the Shapiro-Wilk test for normality, we fail to reject the null hypothesis for all reasonable α values. The QQ-plot also appears normal, with a little right tail skewness. Additionally, we fail to reject Levene's Test for homogeneity of variance. The Plot of standardized residuals vs. fitted values indicates that there is constant variance, though this plot is not very helpful due to the large gap in the domain, as well as the small number of observations. These diagnostics indicate we can proceed with ANOVA with a clear conscience, since none of the required assumptions are clearly violated.

(d) If the assumptions are satisfied, compute the analysis of variance for this data.

```
# (d)
lettuce_anova <- aov(lettuce_mod)
# (at least) one of these is not like the others
summary(lettuce_anova)
```

```
##             Df Sum Sq Mean Sq F value  Pr(>F)
## fert         4   4958  1239.5   5.702 0.00538 **
## Residuals   15   3261   217.4
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Computing the ANOVA for this data yields a significant test result, which implies one of the treatments yields significantly different results than the others.

(e) Given the assumptions are satisfied and the effect of varying the amount of ammonium nitrate is significant, find which ammonium nitrate rate yields the highest number of lettuce heads using the appropriate multiple comparison technique.

```
TukeyHSD(lettuce_anova)
```

```
##   Tukey multiple comparisons of means
##     95% family-wise confidence level
##
## Fit: aov(formula = lettuce_mod)
##
## $fert
##              diff        lwr      upr     p adj
## t100-t0    37.00   4.806752 69.19325 0.0208421
## t150-t0    45.50  13.306752 77.69325 0.0042956
## t200-t0    36.25   4.056752 68.44325 0.0239316
## t50-t0     33.50   1.306752 65.69325 0.0395331
## t150-t100   8.50 -23.693248 40.69325 0.9218376
## t200-t100  -0.75 -32.943248 31.44325 0.9999931
## t50-t100   -3.50 -35.693248 28.69325 0.9969468
## t200-t150  -9.25 -41.443248 22.94325 0.8972296
## t50-t150  -12.00 -44.193248 20.19325 0.7776955
## t50-t200   -2.75 -34.943248 29.44325 0.9988068
```

Based on the Tukey multiple comparison of means, the treatment that appears to yield the most lettuce heads on average is the fertilizer t150, as it has a higher mean than every other. One caveat is that the only significant difference was the t0 comparison of means, and it was significantly different from every other treatment. Otherwise, none of the other treatment means were significantly (p-value ≤ 0.05) from each other. Using the information from this sample though we would conclude that the t150 fertilizer yields the most lettuce heads on average.

This comparison on its own does not account for the variability of the data from sampling. We will conduct contrasts using Hsu's method.

```
# hard coded Hsu's
diffs <- c(112.00 - 157.50,
           145.50 - 157.50,
           148.25 - 157.50,
           149.00 - 157.50,
           157.50 - 149.00)
names(diffs) <- unique(lettuce$fert)

# multivariate t quantile for alpha
# S estimated from MSE in ANOVA
# n is the number of obs in each factor

d <- 2.356*sqrt(217.4*2/4)


ints <- cbind(tapply(lettuce$heads, lettuce$fert, mean) |> sort() |> names(),
      "lower"   = ifelse(diffs - d > 0, 0, diffs - d),
      'central' = diffs,
      "upper"   = ifelse(diffs + d < 0, 0, diffs + d))
```

fertilizer	lower	central	upper
t0	-70.06349	-45.50	0.00000
t50	-36.56349	-12.00	12.56349
t100	-33.81349	-9.25	15.31349
t150	-16.06349	8.50	33.06349
t200	-33.06349	-8.50	16.06349

From Hsu's method we conclude that there is no true unique fertilizer that yields a maximum value of lettuce heads. We do conclude that the t150 fertilizer has the highest upper bound for its Hsu interval. This is neither a significant nor definitive statement though, and no sure conclusions should be drawn from this.

8. A company tested two chemistry methods for the determination of serum glucose. Three pools of serum were used for the experiment. Each pool contained different levels of glucose through the addition of glucose to the base level of an existing serum pool. Three samples of serum from each pool were prepared independently for each level of glucose with each of the two chemistry methods. The concentration of glucose (mg/dl) for all samples was measured on one run of a spectrophotometer, and the data is given in the file "serum.txt" on the website.

```
# some preliminary checking of assumptions and EDA

serum <- read.delim("serum.txt", header = TRUE, sep = '')

tapply(serum$serum, list(serum$method, serum$glucose), mean)

##          l1       l2       l3
## m1 42.90000 141.8333 181.4667
## m2 40.43333 131.7000 175.1000

tapply(serum$serum, list(serum$method, serum$glucose), sd)

##           l1       l2       l3
## m1 0.4000000 3.092464 1.342882
## m2 0.7094599 1.212436 1.609348

tapply(serum$serum, list(serum$method, serum$glucose), length)

##    l1 l2 l3
## m1  3  3  3
## m2  3  3  3
```

(a) Write a linear model for this experiment, and explain the terms.

The two-way ANOVA model can be represented as:

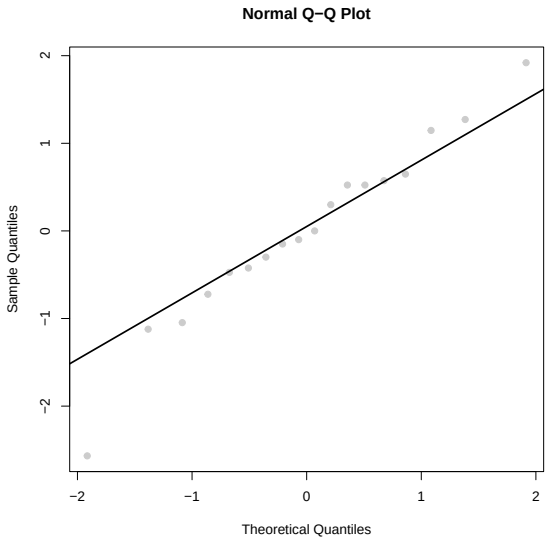
$$Y_{ijk} = \mu + \tau_i + \beta_j + (\tau\beta)_{ij} + \varepsilon_{ijk}$$

where:

Y_{ijk} : Observed value in group i, level j, and replicate k
μ : Overall mean
τ_i : Effect of drawing from pool i $(i = 1, 2)$
β_j : Effect of chemistry method j $(j = 1, 2, 3)$
$(\tau\beta)_{ij}$: Interaction effect between drawing from pool i and chemistry method j
ε_{ijk} : Random error term for drawing from pool i, chemistry method j, and replicate k

(b) Compute the residuals, and use them to check the assumptions. Does a transformation appear to be needed? If so, what transformation would be appropriate?

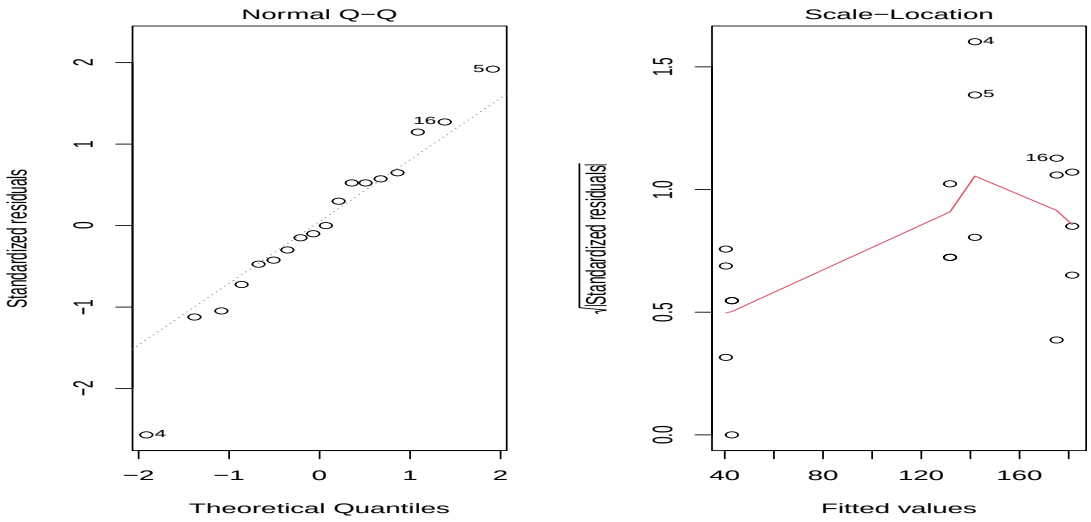
```
# checking normality of residuals
serum_mod <- lm(serum ~ method*glucose, data = serum)
summary(serum_mod)

##
## Call:
## lm(formula = serum ~ method * glucose, data = serum)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.4333 -0.6167 -0.0667  0.7500  2.5667
##
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)          42.9000     0.9455  45.375 8.56e-15 ***
## methodm2             -2.4667     1.3371  -1.845   0.0899 .
## glucosel2            98.9333     1.3371  73.992  < 2e-16 ***
## glucosel3           138.5667     1.3371 103.634  < 2e-16 ***
## methodm2:glucosel2   -7.6667     1.8909  -4.054   0.0016 **
## methodm2:glucosel3   -3.9000     1.8909  -2.062   0.0615 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.638 on 12 degrees of freedom
## Multiple R-squared:  0.9995,Adjusted R-squared:  0.9992
## F-statistic:  4407 on 5 and 12 DF,  p-value: < 2.2e-16

res <- rstandard(serum_mod)
qqnorm(res, pch = 19, col = alpha(1, 0.2))
qqline(res, lwd = 2)
```




```
shapiro.test(res)

##
##  Shapiro-Wilk normality test
##
## data:  res
## W = 0.9711, p-value = 0.8181

shapiro.test(rstandard(serum_mod)) # FTR normality null hypothesis

##
##  Shapiro-Wilk normality test
##
## data:  rstandard(serum_mod)
## W = 0.9711, p-value = 0.8181

# checking homogenous variance
# FTR
leveneTest(serum$serum, interaction(serum$method, serum$glucose))

## Levene's Test for Homogeneity of Variance (center = median)
##       Df F value Pr(>F)
## group  5  0.7826 0.5812
##       12
```

Based on the results of both the QQ-plots and the Shapiro-Wilk test for normality, we fail to reject the hypothesis that the residuals are normally distributed. Likewise, based on the Levene's test for homogenous variance, fail to reject the null that there is homogenous variance between treatments. We lastly create diagnostic plots for the model.

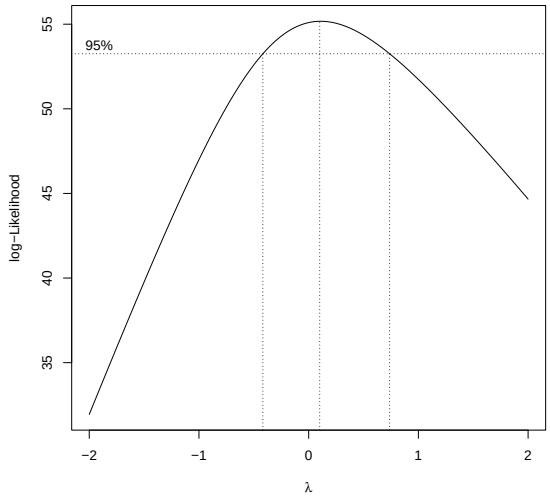
```
par(mfrow = c(1,2))
plot(serum_mod, which = c(2, 3))
```




```
par(mfrow = c(1,1))
```

Although Levene's test failed to reject at all reasonable α values, we notice in our model that there is most definitely non-constant variance (see right plot with the squre root of the absolute value of the standardized residuals). We suspect a log transformation of the outcome is needed.

```
MASS::boxcox(serum_mod)
```



The Box-Cox transform indicates a log-transform might be useful for stabilizing the variance.

```
serum_mod <- lm(log(serum) ~ method*glucose, data = serum)
summary(serum_mod)

##
## Call:
```

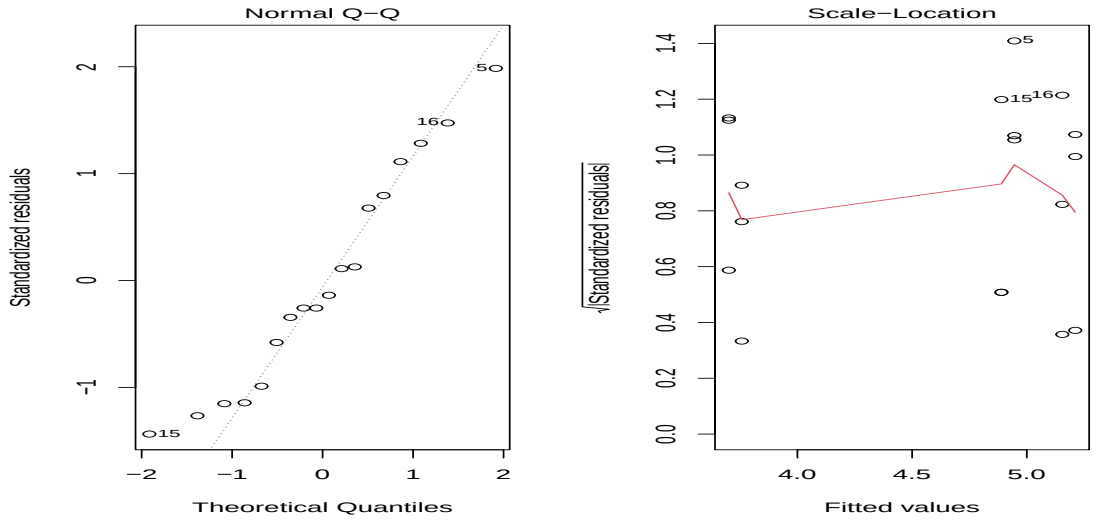
```
## lm(formula = log(serum) ~ method * glucose, data = serum)
##
## Residuals:
##        Min         1Q     Median         3Q        Max
## -0.0243452 -0.0077622 -0.0005429  0.0078871  0.0188860
##
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)         3.758843   0.007807 481.468  < 2e-16 ***
## methodm2           -0.059291   0.011041  -5.370 0.000168 ***
## glucosel2           1.195650   0.011041 108.294  < 2e-16 ***
## glucosel3           1.442211   0.011041 130.625  < 2e-16 ***
## methodm2:glucosel2 -0.014704   0.015614  -0.942 0.364900
## methodm2:glucosel3  0.023566   0.015614   1.509 0.157100
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.01352 on 12 degrees of freedom
## Multiple R-squared:  0.9997,Adjusted R-squared:  0.9996
## F-statistic:  7885 on 5 and 12 DF,  p-value: < 2.2e-16
```

None of the interactions in the model were significant - so we will remove them from the model.

```
serum_mod <- lm(log(serum) ~ method + glucose, data = serum)
summary(serum_mod)
```

```
##
## Call:
## lm(formula = log(serum) ~ method + glucose, data = serum)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.019488 -0.012027 -0.002687  0.010388  0.026923
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  3.757366   0.007251  518.19  < 2e-16 ***
## methodm2    -0.056337   0.007251   -7.77 1.92e-06 ***
## glucosel2    1.188298   0.008881  133.81  < 2e-16 ***
## glucosel3    1.453994   0.008881  163.73  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.01538 on 14 degrees of freedom
## Multiple R-squared:  0.9995,Adjusted R-squared:  0.9994
## F-statistic: 1.015e+04 on 3 and 14 DF,  p-value: < 2.2e-16
```

```
par(mfrow = c(1,2))
plot(serum_mod, which = c(2,3))
```




```
par(mfrow = c(1,1))

shapiro.test(rstandard(serum_mod))

##
##  Shapiro-Wilk normality test
##
## data:  rstandard(serum_mod)
## W = 0.94975, p-value = 0.4211
```

Per the diagnostic plots, it appears that the transformation enables us to proceed forward without any ANOVA assumptions being violated. The variace does not seem to vary as much across the fitted values and the residuals still seem normal.

(c) Compute the analysis of variance, and test the hypothesis of no method/glucose interaction. What do you conclude?

```
serum_aov <- aov(log(serum) ~ method*glucose, data = serum)
serum_aov |> summary()

##                Df Sum Sq Mean Sq   F value   Pr(>F)
## method          1  0.014   0.014    78.109 1.34e-06 ***
## glucose         2  7.193   3.597 19670.484  < 2e-16 ***
## method:glucose  2  0.001   0.001     3.057   0.0845 .
## Residuals      12  0.002   0.000
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Based on the ANOVA, we do not believe that there exists a significant interaction at the $\alpha = 0.05$ level.

(d) Should you test for main effects? If so, test them, and describe their effect on glucose concentration in the serum..

```
serum_aov <- aov(serum_mod)
serum_aov |> summary()

##             Df Sum Sq Mean Sq  F value   Pr(>F)
## method       1  0.014   0.014    60.37 1.92e-06 ***
## glucose      2  7.193   3.597 15202.28  < 2e-16 ***
## Residuals   14  0.003   0.000
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Because the interaction is insignificant but the main effects are in the ANOVA from (c), we ought to and do test for main effects. The ANOVA does indicate they have a significant impact on the mean glucose concentration in the serum.

(e) Test the appropriate differences in pairwise means, and describe which are significantly different.

```
TukeyHSD(aov(serum ~ method + glucose, data = serum))

##   Tukey multiple comparisons of means
##     95% family-wise confidence level
##
## Fit: aov(formula = serum ~ method + glucose, data = serum)
##
## $method
##            diff       lwr       upr    p adj
## m2-m1 -6.322222 -8.682072 -3.962372 5.06e-05
##
## $glucose
##            diff       lwr       upr p adj
## l2-l1  95.10000  91.57308  98.62692     0
## l3-l1 136.61667 133.08974 140.14359     0
## l3-l2  41.51667  37.98974  45.04359     0
```

Based on the Tukey multiple comparisons, all of the pairwise comparisons are significantly different from one another.

9. A study of the effect of temperature on percent shrinkage in dyeing fabrics was made on two replications for each of four fabrics in a completely randomized design. The data are the percent shrinkage of two replicate fabric pieces dried at each of four temperatures and is given on the class website in the "fabric.txt" file.

   (a) Write a linear model for the experiment, explain the terms, and compute the analysis of variance for the data.

   The two-way ANOVA model can be represented as:

   $$Y_{ijk} = \mu + \tau_i + \beta_j + (\tau\beta)_{ij} + \varepsilon_{ijk}$$

   where:

   Y_{ijk} : Observed value in group i, level j, and replicate k
   μ : Grand mean
   τ_i : Effect of using fabric i $(i = 1, 2, 3, 4)$
   β_j : Effect of drying at temperature j $(j = 1, 2, 3, 4)$
   $(\tau\beta)_{ij}$: Interaction effect between drying fabric i at temperature j
   ε_{ijk} : Random error term for drying fabric i, temperature j, and replicate k

   (b) Test the null hypothesis of no fabric/temperature interaction.

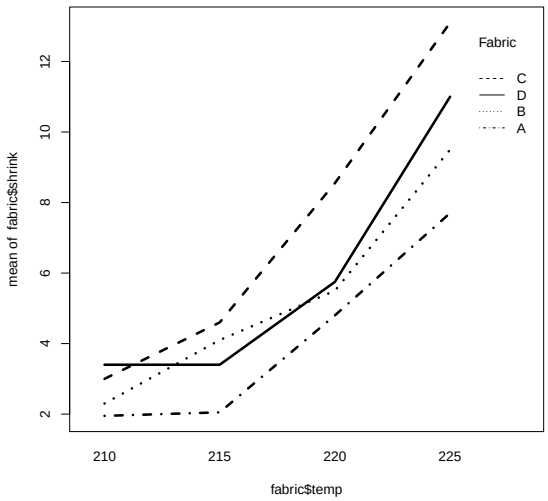
```
fabric <- read.delim("fabric.txt", header = TRUE, sep = '') |>
  mutate(temp = as.factor(temp),
         fabric = as.factor(fabric))

fabric_mod1 <- lm(shrink ~ temp*fabric, data = fabric)
anova_fabric_mod1 <- aov(fabric_mod1)
summary(anova_fabric_mod1)
```

```
##             Df Sum Sq Mean Sq F value   Pr(>F)
## temp         3 283.94   94.65 1892.91  < 2e-16 ***
## fabric       3  41.88   13.96  279.18 5.05e-14 ***
## temp:fabric  9  15.86    1.76   35.24 7.09e-09 ***
## Residuals   16   0.80    0.05
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

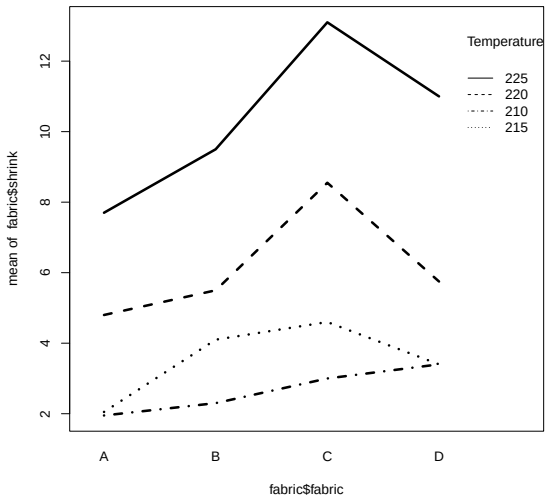
   Per the ANOVA summary, we see that the interaction between temp and fabric is in fact significant for all reasonable levels of α (p-value = 7.09e-9).

   (c) Plot the interaction plots of the cell means versus temperature for each fabric type. Interpret the plot.

```
interaction.plot(fabric$temp, fabric$fabric, fabric$shrink, lwd = 3, trace.label
```




```
interaction.plot(fabric$fabric, fabric$temp, fabric$shrink, lwd = 3, trace.label
```



According to the interaction plots, the lower one indicates that for higher temperatures (temp = 220, 225), the mean shrinkage does not appear to change across fabrics A, B, C, and D. The interactions for fabric and temp appears to be more pronounced at lower temperatures.

These plots also generally indicate that higher drying temperatures are indicative of greater shrinkage (top plot).

Also, fabric D appears to have similar shrinkage at lower temperatures, but has higher rates of increase of shrinkage at higher values (top plot).

(d) Which cell means are not significantly different from each other?

```
TukeyHSD(anova_fabric_mod1)$`temp:fabric` |>
  data.frame() |>
```

```
  dplyr::filter(`p.adj` >= 0.05)
```

Below is a table with all of the pairwise differences that were not significantly different from each other. Each Pair is written with the format temp:fabric - temp:fabric. Out of 120 pairwise comparisons of cell means, 15 were not significantly different from each other.

Pair	Difference	Lower	Upper	p-value
215:A-210:A	0.10	-0.795	0.995	1.000
210:B-210:A	0.35	-0.545	1.245	0.954
210:B-215:A	0.25	-0.645	1.145	0.998
215:B-220:A	-0.70	-1.595	0.195	0.219
220:B-220:A	0.70	-0.195	1.595	0.219
215:C-220:A	-0.20	-1.095	0.695	1.000
220:C-225:A	0.85	-0.045	1.745	0.072
210:C-210:B	0.70	-0.195	1.595	0.219
215:C-215:B	0.50	-0.395	1.395	0.666
210:D-215:B	-0.70	-1.595	0.195	0.219
215:D-215:B	-0.70	-1.595	0.195	0.219
220:D-220:B	0.25	-0.645	1.145	0.998
210:D-210:C	0.40	-0.495	1.295	0.889
215:D-210:C	0.40	-0.495	1.295	0.889
215:D-210:D	0.00	-0.895	0.895	1.000