```
library(dplyr)
```

```
## Warning: As of rlang 0.4.0, dplyr must be at least version 0.8.0.
## x dplyr 0.7.8 is too old for rlang 0.4.3.
## i Please update dplyr to the latest version.
## i Updating packages on Windows requires precautions:
##   <https://github.com/jennybc/what-they-forgot/issues/62>
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(tidyr)
library(ggplot2)
library(ggpubr)
```

```
## Warning: package 'ggpubr' was built under R version 3.5.3
```

```
## Loading required package: magrittr
```

```
##
## Attaching package: 'magrittr'
```

```
## The following object is masked from 'package:tidyr':
##
##     extract
```

```
library(ggfortify)
```

```
## Warning: package 'ggfortify' was built under R version 3.5.3
```

```
library(car)
```

```
## Warning: package 'car' was built under R version 3.5.3
```

```
## Loading required package: carData
```

```
## Warning: package 'carData' was built under R version 3.5.3
```

```
##
## Attaching package: 'car'
```

```
## The following object is masked from 'package:dplyr':
##
##     recode
```

1. Code importing and checking the dataframe (no marks for this, you should be pros at this).

```
myDa <-  read.csv("/Users/kelli/Desktop/Biol343/Assignment_7/mytilus.csv", fileEncodi
ng="UTF-8-BOM")

summary(myDa)
```

```
##   lap94y_freq       miles_east
##  Min.   :0.1100   Min.   : 1.00
##  1st Qu.:0.1600   1st Qu.:18.50
##  Median :0.3150   Median :42.00
##  Mean   :0.3171   Mean   :37.32
##  3rd Qu.:0.4600   3rd Qu.:54.00
##  Max.   :0.5450   Max.   :67.00
```

2. Report the reference of a reliable source indicating how variables like allele frequencies should be transformed, and apply this transformation to your data (using dplyr).

Warton, David I., and Francis KC Hui. “The arcsine is asinine: the analysis of proportions in ecology.” Ecology 92.1 (2011): 3-10.

The above paper by Warton and Hui shows that when analyzing proportional data the logit transformation is the go to transformation over the traditionally popular arcsine transformation. They found coefficients of the logit transformation to be more readily interpretable, while the arcsine transformation lead to more problems in extrapolation beyond the fitted range.

```
myDat <- myDa %>%  mutate(logit_94y = logit(lap94y_freq))
```

```
## Warning: The `printer` argument is deprecated as of rlang 0.3.0.
## This warning is displayed once per session.
```

```
summary(myDat)
```

```
##   lap94y_freq       miles_east      logit_94y       
##  Min.   :0.1100   Min.   : 1.00   Min.   :-2.0907  
##  1st Qu.:0.1600   1st Qu.:18.50   1st Qu.:-1.6582  
##  Median :0.3150   Median :42.00   Median :-0.7768  
##  Mean   :0.3171   Mean   :37.32   Mean   :-0.8863  
##  3rd Qu.:0.4600   3rd Qu.:54.00   3rd Qu.:-0.1603  
##  Max.   :0.5450   Max.   :67.00   Max.   : 0.1805
```

3. Use lm() to fit linear, quadratic, cubic, quartic and null models to variation in transformed allele frequencies.

```
myDat <- myDat %>%  mutate(miles_east_2 = (miles_east-mean(miles_east))^2,
                           miles_east_3 = (miles_east-mean(miles_east))^3,
                           miles_east_4 = (miles_east-mean(miles_east))^4)



null <- lm(logit_94y ~ 1, data = myDat)
linear <- lm(logit_94y ~ miles_east, data = myDat)
quadratic <- lm(logit_94y ~ miles_east + miles_east_2, data = myDat)
cubic <- lm(logit_94y ~ miles_east + miles_east_2 + miles_east_3 , data = myDat)
quartic <- lm(logit_94y ~ miles_east + miles_east_2 + miles_east_3 + miles_east_4, da
ta = myDat)
```

4. Backwards selection to determine the MAM, plus interpretation of the results

```
anova(quartic, cubic)
```

```
## Analysis of Variance Table
##
## Model 1: logit_94y ~ miles_east + miles_east_2 + miles_east_3 + miles_east_4
## Model 2: logit_94y ~ miles_east + miles_east_2 + miles_east_3
##   Res.Df     RSS Df Sum of Sq      F Pr(>F)
## 1     12 0.88216
## 2     13 0.93049 -1 -0.048337 0.6575 0.4332
```

```
anova(cubic, quadratic)
```

```
## Analysis of Variance Table
##
## Model 1: logit_94y ~ miles_east + miles_east_2 + miles_east_3
## Model 2: logit_94y ~ miles_east + miles_east_2
##   Res.Df     RSS Df Sum of Sq      F   Pr(>F)
## 1     13 0.93049
## 2     14 1.58618 -1  -0.65569 9.1607 0.009727 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

No significant difference in modeling power between quartic/cubic (p = 0.433), significant difference between cubic and quadratic (p= 0.0097) cubic is MAM.

5. AICc-based model selection to determine the top model. Be careful because not all possible combinations of model terms are legitimate models on their own. You'll have to do some research to figure out how to use AICc to evaluate different polynomial regression models. Make sure you calculate and interpret the Akaike weights and evidence ratios for all the models in the appropriate model set.

The models ranked by AICc are as follows: 1) cubic (AICc = 14.3,evid ratio = 5.21, weight = 0.134) 2) quartic (AICc = 18.3, evid ratio = 39.22, weight = 0.018) 3) linear (AICc = 18.7, evid ratio = 46.1 , weight = 0.011) 4) quadratic (AICc = 19.3, evid ratio = 61.7 , weight = 0.003)

AICc based model selection determines cubic to be the best model 13.4% of the time, having an AICc value 4 points lower than the next best model (quartic). Linear is 3.22x more powerful than the next best model, and 154x more effective than the least powerful model (quartic).

```
library(MuMIn)
```

```
## Warning: package 'MuMIn' was built under R version 3.5.3
```

```
options(na.action = "na.fail")
(dd <-dredge(quartic, extra = "R^2", beta = "sd") )
```

```
## Fixed term is "(Intercept)"
```

```
## Global model call: lm(formula = logit_94y ~ miles_east + miles_east_2 + miles_east
_3 +
##     miles_east_4, data = myDat)
## ---
## Model selection table
##    (Int) mls_est mls_est_2 mls_est_3 mls_est_4     R^2 df  logLik AICc
## 6      0  1.5000             -0.6584           0.90960  4   0.163 11.0
## 8      0  1.4770   0.07553   -0.6010           0.91390  5   0.573 14.3
## 14     0  1.4710             -0.6009   0.05091 0.91110  5   0.301 14.9
## 10     0  1.0020                       0.21330 0.86260  4  -3.395 18.1
## 16     0  1.5620   0.28600   -0.7333  -0.25860 0.91830  6   1.026 18.3
## 2      0  0.9087                                0.82580  3  -5.412 18.7
## 4      0  0.9760   0.17850                      0.85320  4  -3.961 19.3
## 12     0  1.0060  -0.10100             0.31000 0.86380  5  -3.320 22.1
## 5      0                       0.6890           0.47470  3 -14.795 37.4
## 13     0                       0.8807   0.31330 0.53620  4 -13.737 38.8
## 7      0           0.18990     0.7812           0.50230  4 -14.336 40.0
## 15     0          -0.52390     0.9550   0.85050 0.56540  5 -13.183 41.8
## 1      0                                        0.00000  2 -20.267 45.4
## 9      0                               -0.22540 0.05079  3 -19.824 47.5
## 3      0          -0.18920                      0.03578  3 -19.957 47.8
## 11     0           0.18750            -0.40140 0.05498  4 -19.786 50.9
##    delta weight
## 6   0.00  0.697
## 8   3.30  0.134
## 14  3.85  0.102
## 10  7.12  0.020
## 16  7.34  0.018
## 2   7.66  0.015
## 4   8.25  0.011
## 12 11.09  0.003
## 5  26.43  0.000
## 13 27.80  0.000
## 7  29.00  0.000
## 15 30.81  0.000
## 1  34.38  0.000
## 9  36.49  0.000
## 3  36.75  0.000
## 11 39.90  0.000
## Models ranked by AICc(x)
```

```
(evid.ratio = max(dd$weight)/dd$weight)
```

```
##  model weights
##  [1]         1.000         5.210         6.840        35.084        39.247
##  [6]        46.100        61.783       255.619    547827.283   1087991.845
## [11]   1980416.413   4909198.721  29260297.105  83734074.464  95680427.990
## [16] 461100390.801
```

6. Did likelihood-ratio tests and AICc lead you to the same MAM?

Both tests identified the cubic model model (logit_94y ~ miles_east + miles_east_2 + miles_east_3) as the MAM.

7. Publication-quality graph + caption illustrating geographic variation in allele frequency with the various regression models superimposed. Use colours to highlight the models indicated as the MAM by the two approaches to model selection.

```
p1 = ggplot(myDat, aes(x = miles_east, y = logit_94y)) +
  geom_point(size = 5) +
  geom_smooth(method = lm, formula = y~poly(x,2), colour = "yellow", alpha = 0.2, se
= F) +
  geom_smooth(method = lm, formula = y~poly(x,3), colour = "green", alpha = 0.2, se =
F) +
  geom_smooth(method = lm, formula = y~poly(x,4), colour = "orange", alpha = 0.2, se
= F) +
  geom_smooth(method = lm, formula = y~x, alpha = 0.5,colour = "red", se = F) +
  theme_minimal() +
  labs(x = "Distance east of Southport CT (miles)", y = "Proportional frequency of th
e 94y allele (logit)")
p1
```

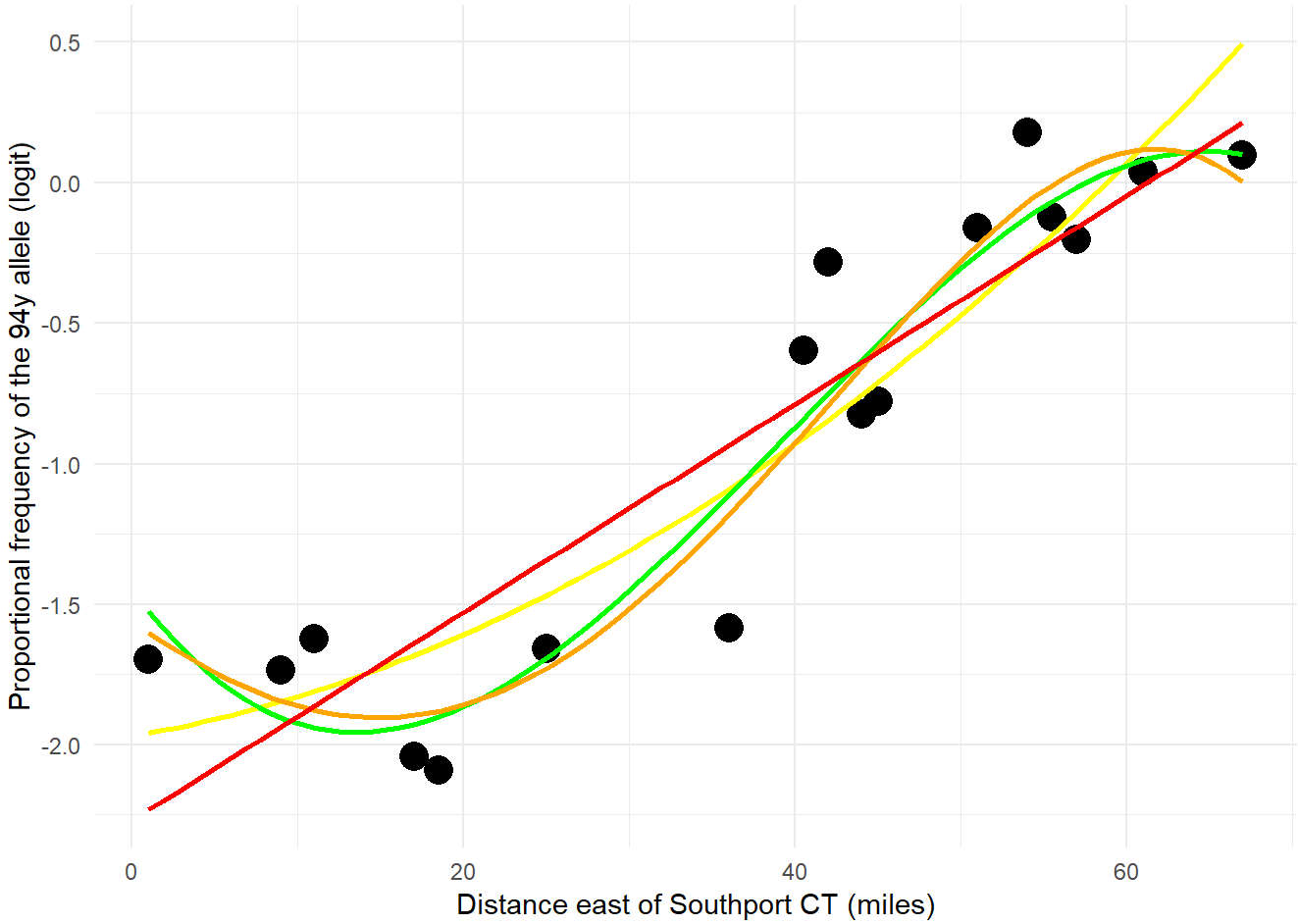


Figure 1. Proportional frequency of the 94y allele (logit transformed), by distance east of Southport CT. Plotted overtop of the data are four models: cubic (green) was found to be the MAM from both backwards selection

and likelyhood ratio tests, linear (yellow), quadratic (orange), and quartic models (red) are also plotted.

8. Comparison of how transformed vs. untransformed data meet model assumptions and how transformation influences statistical results. You'll want to see how transformation affected your choice of MAM, how it affected the analysis of residuals and whether it tamed particularly influential data points. In the end you'll have to state whether the data transformation is appropriate.

Both transformed and untransformed data resulted in the cubic linear model becoming the MAM. Transforming the data resulted in limited impact on the assumptions. Assumption 1 (normality) was a bit better in the transformed data, but assumption 2 actually appeared to be worse in the transformed data. The transformation appeared to make very little impact on the graphing of the linear model, and totally failed to reign in influential points, actually making them worse (see next question).

This transformation of the data did not accomplish all our objectives of reigning in influential data, and improving assumptions, therefore it was not appropriate.

```
library(ggpubr)

nullut <- lm(lap94y_freq ~ 1, data = myDat)
linearut <- lm(lap94y_freq ~ miles_east, data = myDat)
quadraticut <- lm(lap94y_freq ~ miles_east + miles_east_2, data = myDat)
cubicut <- lm(lap94y_freq ~ miles_east + miles_east_2 + miles_east_3 , data = myDat)
quarticut <- lm(lap94y_freq ~ miles_east + miles_east_2 + miles_east_3 + miles_east_
4, data = myDat)

#testing for MAM
anova(quarticut, cubicut)
```

```
## Analysis of Variance Table
##
## Model 1: lap94y_freq ~ miles_east + miles_east_2 + miles_east_3 + miles_east_4
## Model 2: lap94y_freq ~ miles_east + miles_east_2 + miles_east_3
##   Res.Df      RSS Df  Sum of Sq      F Pr(>F)
## 1     12 0.035081
## 2     13 0.039036 -1 -0.0039544 1.3527 0.2674
```

```
anova(cubicut, quadraticut)
```

```
## Analysis of Variance Table
##
## Model 1: lap94y_freq ~ miles_east + miles_east_2 + miles_east_3
## Model 2: lap94y_freq ~ miles_east + miles_east_2
##   Res.Df      RSS Df Sum of Sq      F  Pr(>F)
## 1     13 0.039036
## 2     14 0.055848 -1 -0.016812 5.5988 0.03418 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
anova(quadraticut, linearut)
```

```
## Analysis of Variance Table
##
## Model 1: lap94y_freq ~ miles_east + miles_east_2
## Model 2: lap94y_freq ~ miles_east
##   Res.Df      RSS Df Sum of Sq    F  Pr(>F)
## 1     14 0.055848
## 2     15 0.072562 -1 -0.016715 4.19 0.05991 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
anova(linearut, nullut)
```

```
## Analysis of Variance Table
##
## Model 1: lap94y_freq ~ miles_east
## Model 2: lap94y_freq ~ 1
##   Res.Df     RSS Df Sum of Sq      F    Pr(>F)
## 1     15 0.07256
## 2     16 0.41555 -1  -0.34299 70.903 4.557e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
(dd <-dredge(quarticut, extra = "R^2", beta = "sd") )
```

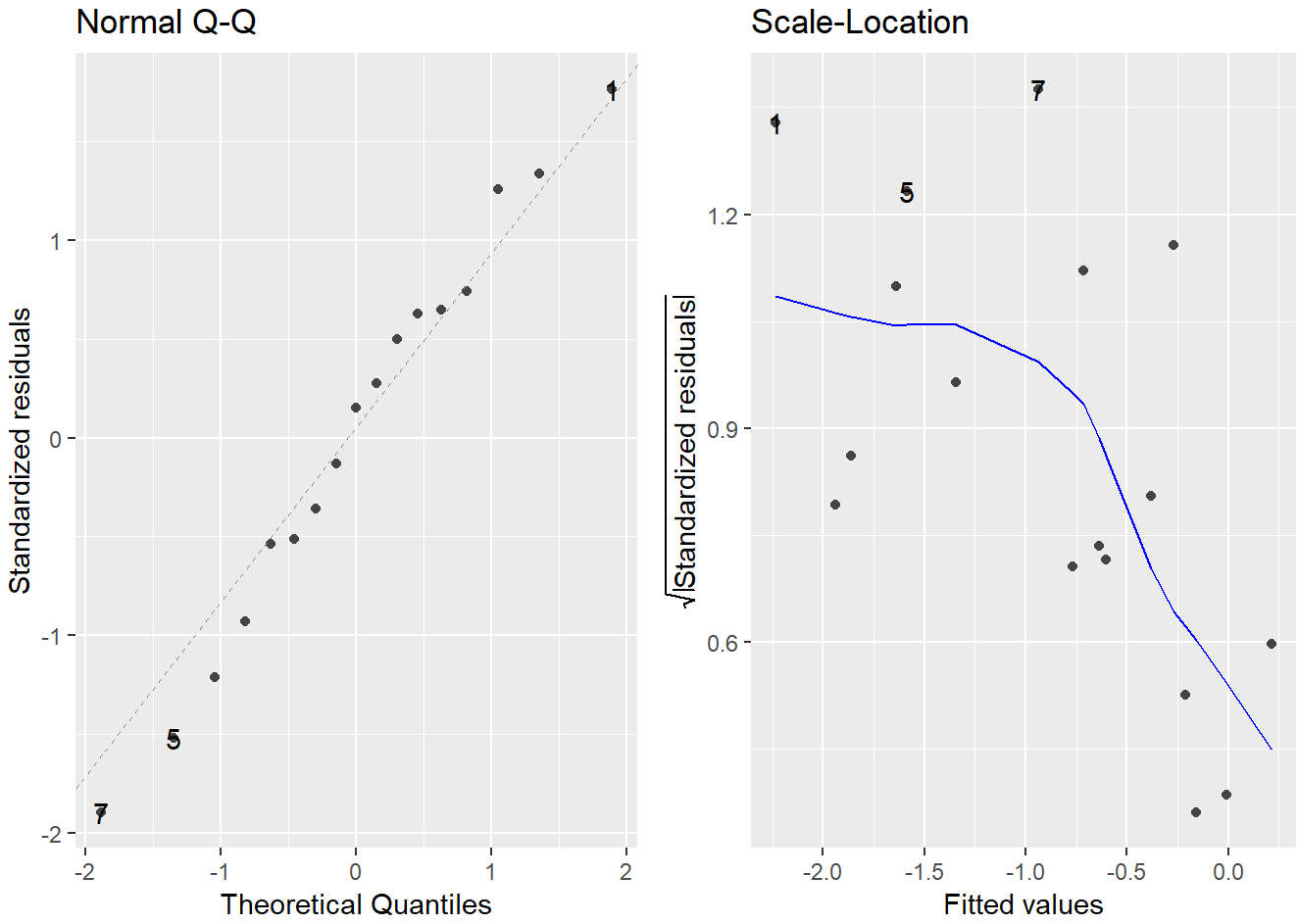
```
## Fixed term is "(Intercept)"
```

```
## Global model call: lm(formula = lap94y_freq ~ miles_east + miles_east_2 + miles_ea
st_3 +
##     miles_east_4, data = myDat)
## ---
## Model selection table
##    (Int) mls_est mls_est_2 mls_est_3 mls_est_4     R^2 df logLik  AICc
## 6      0  1.4400             -0.5914           0.89300  4 26.419 -41.5
## 8      0  1.3990   0.13250   -0.4906           0.90610  5 27.528 -39.6
## 14     0  1.3830             -0.4801    0.0984 0.89840  5 26.862 -38.3
## 10     0  1.0080                        0.2282 0.86750  4 24.603 -37.9
## 4      0  0.9901   0.21650                     0.86560  4 24.484 -37.6
## 2      0  0.9085                               0.82540  3 22.259 -36.7
## 16     0  1.5230   0.43940   -0.6836   -0.3771 0.91560  6 28.436 -36.5
## 12     0  1.0050   0.07862              0.1529 0.86820  5 24.650 -33.8
## 5      0                     0.7017           0.49240  3 13.189 -18.5
## 13     0                     0.9129    0.3452 0.56700  4 14.539 -17.7
## 7      0           0.24090   0.8186           0.53680  4 13.966 -16.6
## 15     0          -0.35030   0.9626    0.7043 0.58010  5 14.800 -14.1
## 1      0                                      0.00000  2  7.425 -10.0
## 9      0                               -0.2133 0.04548  3  7.820  -7.8
## 3      0          -0.15640                    0.02447  3  7.635  -7.4
## 11     0           0.36680             -0.5575 0.06151  4  7.964  -4.6
##    delta weight
## 6   0.00  0.480
## 8   1.90  0.185
## 14  3.24  0.095
## 10  3.63  0.078
## 4   3.87  0.069
## 2   4.83  0.043
## 16  5.03  0.039
## 12  7.66  0.010
## 5  22.97  0.000
## 13 23.76  0.000
## 7  24.91  0.000
## 15 27.36  0.000
## 1  31.51  0.000
## 9  33.71  0.000
## 3  34.08  0.000
## 11 36.91  0.000
## Models ranked by AICc(x)
```

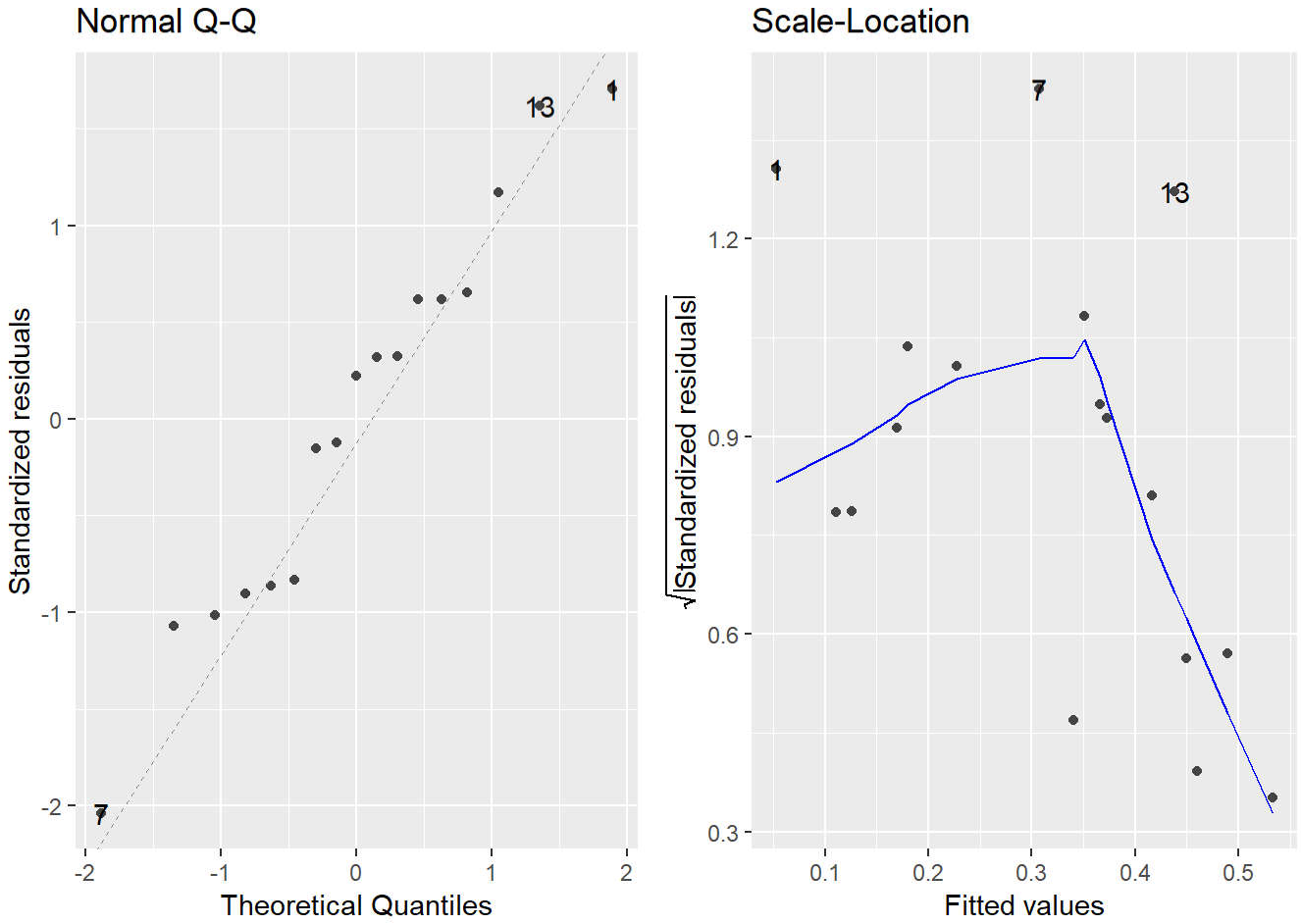
```
(evid.ratio = max(dd$weight)/dd$weight)
```

```
##  model weights
##  [1]         1.000         2.589         5.041         6.147         6.923
##  [6]        11.209        12.380        46.051     97391.777    144311.806
## [11]    255999.468    872942.855   6962986.817  20894285.817  25141996.918
## [16] 103447350.036
```

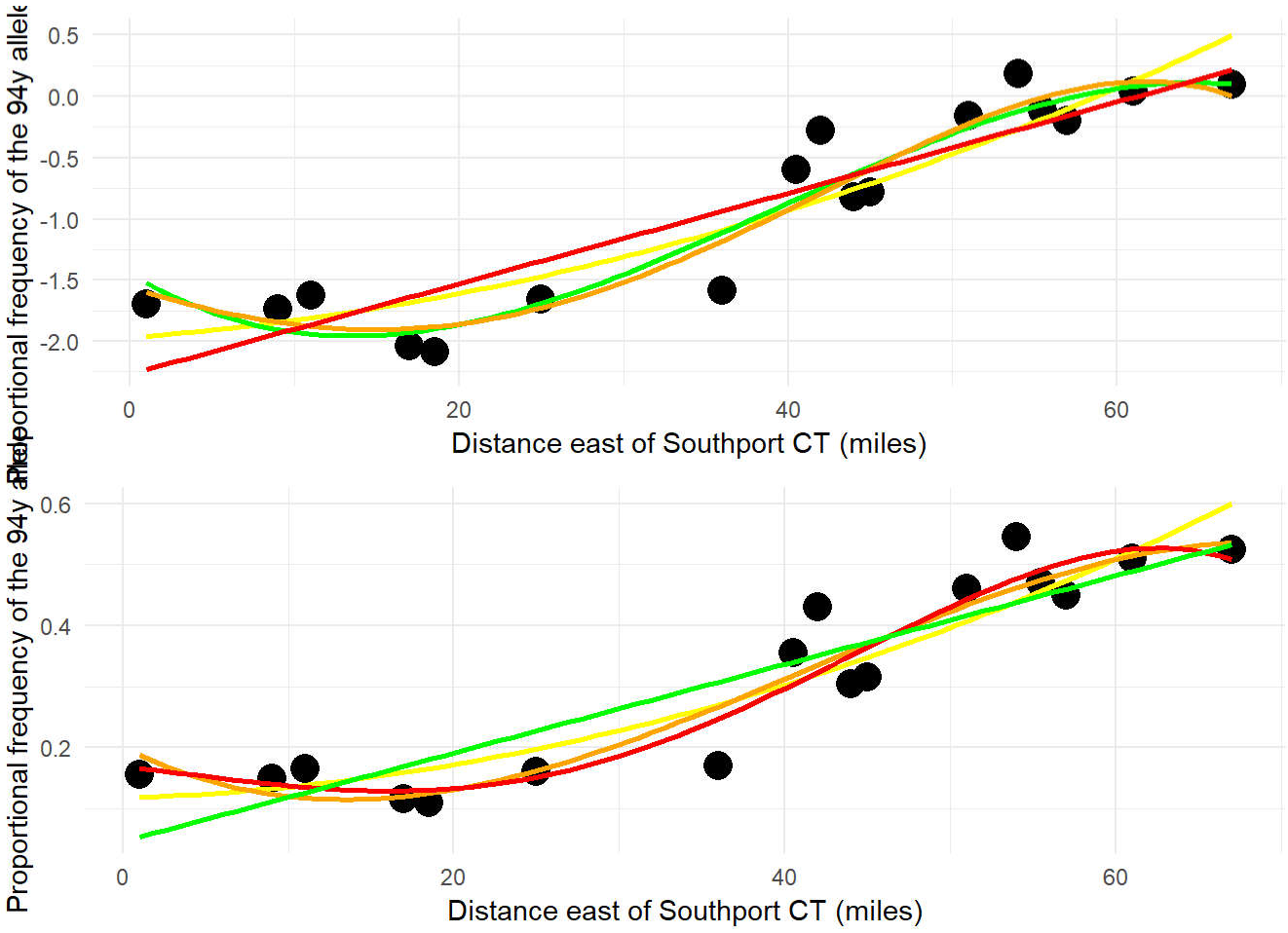
```
#testing assumptions for both glms
autoplot(linear, c(2,3))
```




```
autoplot(linearut, c(2,3))
```

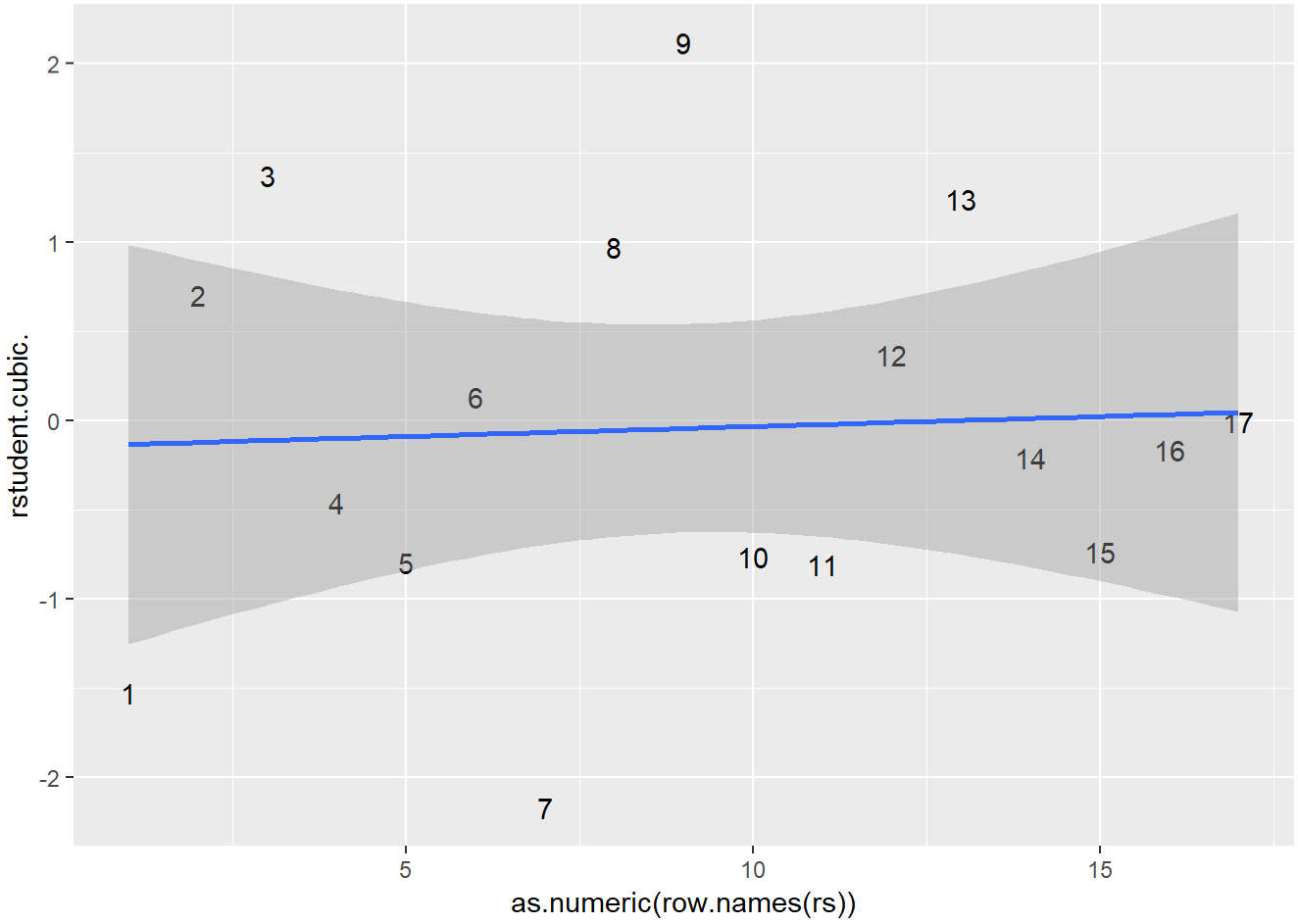



```
p2 <- ggplot(myDa, aes(x = miles_east, y = lap94y_freq)) +
  geom_point(size = 5) +
  geom_smooth(method = lm, formula = y~poly(x,2), colour = "yellow", alpha = 0.2, se
= F) +
  geom_smooth(method = lm, formula = y~poly(x,3), colour = "orange", alpha = 0.2, se
= F) +
  geom_smooth(method = lm, formula = y~poly(x,4), colour = "red", alpha = 0.2, se =
F) +
  geom_smooth(method = lm, formula = y~x, alpha = 0.5,colour = "green", se = F) +
  theme_minimal() +
  labs(x = "Distance east of Southport CT (miles)", y = "Proportional frequency of th
e 94y allele")
 ggarrange(p1,p2, ncol = 1, nrow = 2)
```



9. If you find an influential data point, then identify it and explore how its inclusion in the data affects the results and conclusions.

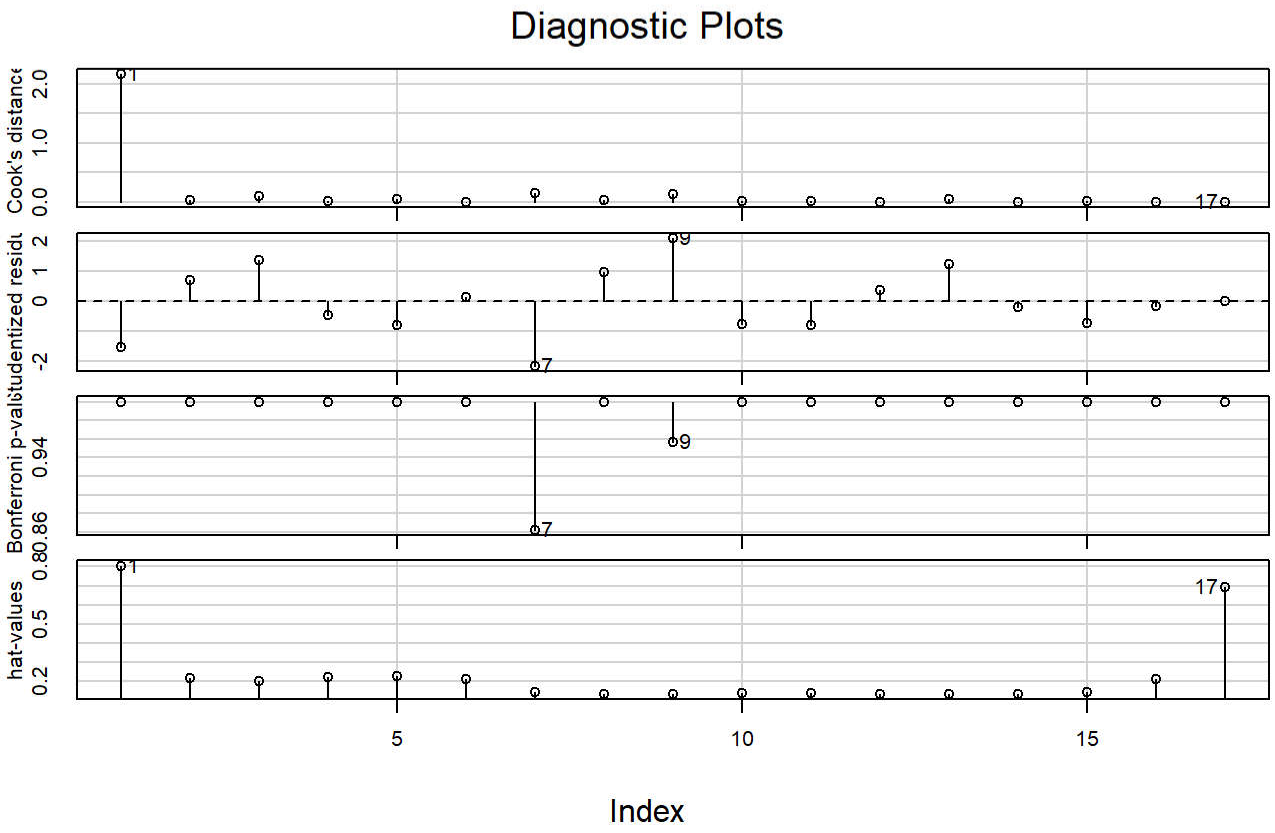
```
rs <- data.frame(rstudent(cubic))
ggplot(rs, aes(x= as.numeric(row.names(rs)), y = rstudent.cubic.)) +
  geom_text(aes(label=rownames(rs)))+
  geom_smooth(method = lm)
```



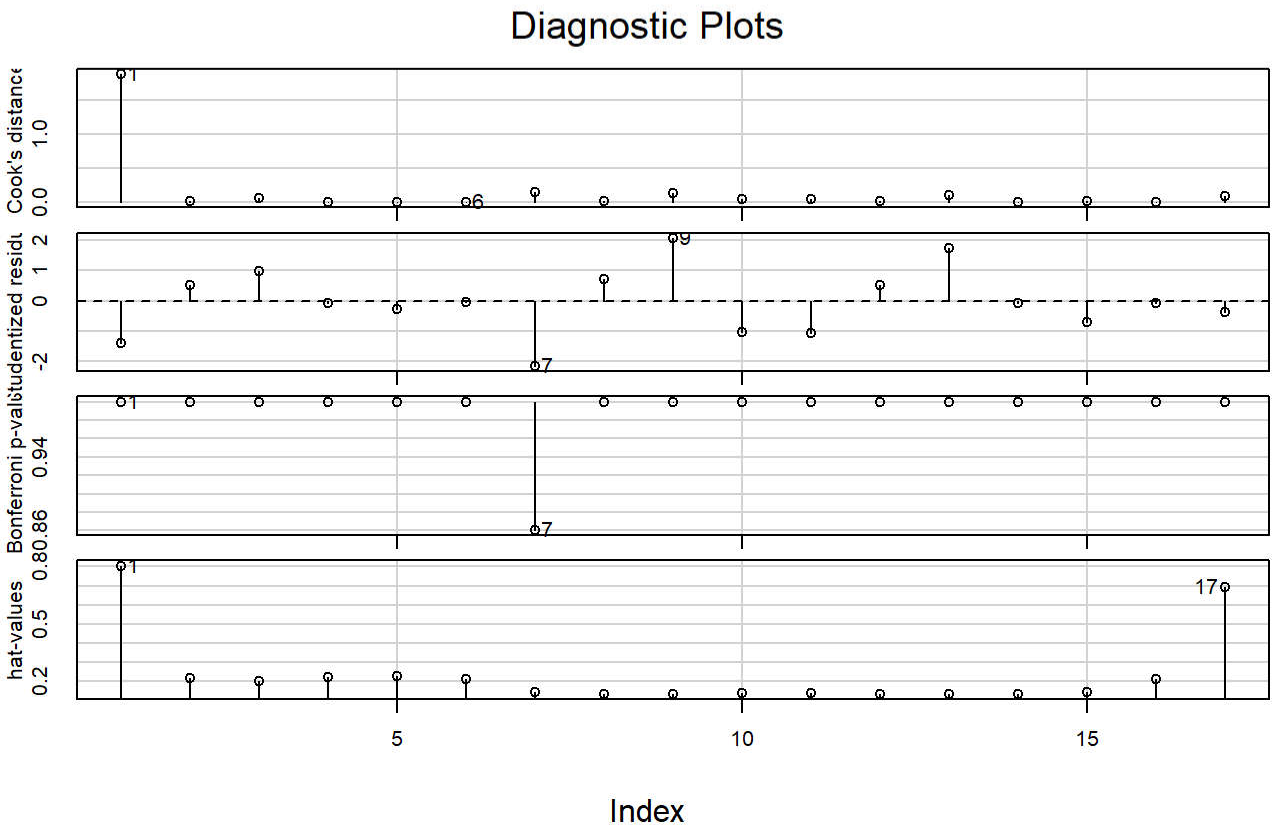

```
outlierTest(cubic)
```

```
## No Studentized residuals with Bonferroni p < 0.05
## Largest |rstudent|:
##    rstudent unadjusted p-value Bonferroni p
## 7 -2.171037           0.050701      0.86191
```

```
influenceIndexPlot(cubic)
```




```
influenceIndexPlot(cubicut)
```



The most influential data points in the set are points 1 and 7. Point one has a cooks distance of about 2 (significant influence on the linear model) and a hat value of about 0.8. This hat value suggests that the point is quite far from the predicted values and that the graph will shift significantly if data point 1 is dropped from the model. Meanwhile 7 is the closest to a significant p-value, having an uncorrected p-value of 0.05

10. Figuring out points 8 and 9 might require a bit of supplemental graphical analysis. If so, you don't have to provide formal captions for these graphs but the graphs do have to appear in your R Notebook.