Untitled

2022-12-03

0. Instructions for installing tinytex for PDF rendering: **https://yihui.org/tinytex/** (https://yihui.org/tinytex/)

```
install.packages('tinytex')
tinytex::install_tinytex()
```

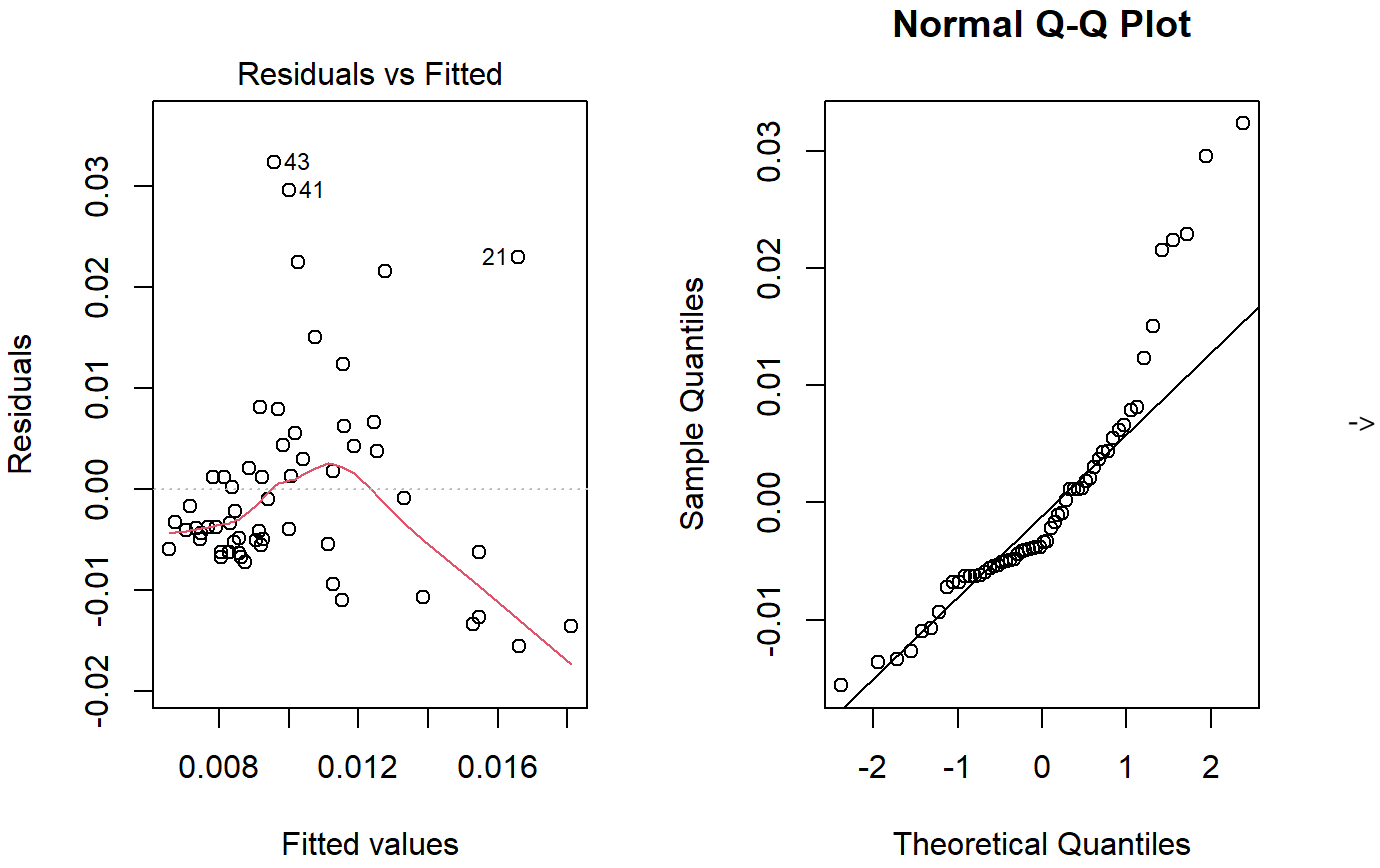
I.First section

We will begin by analyzing the data using simple linear regression. (1) PEC that fits a linear regression model for the proportion of vehicles that are electric as function of the proportion of people who work from home. Create a plot to assess the assumption of normally-distributed residuals. What do you notice?

```
car = read.csv("C:/Rlab6/electric_WFH.csv")


car_lm = lm(prop_electric ~ prop_WFH, data = car )

layout(matrix(2:1, ncol = 2))
qqnorm(resid(car_lm))
qqline(resid(car_lm))
plot(car_lm, which = 1)
```

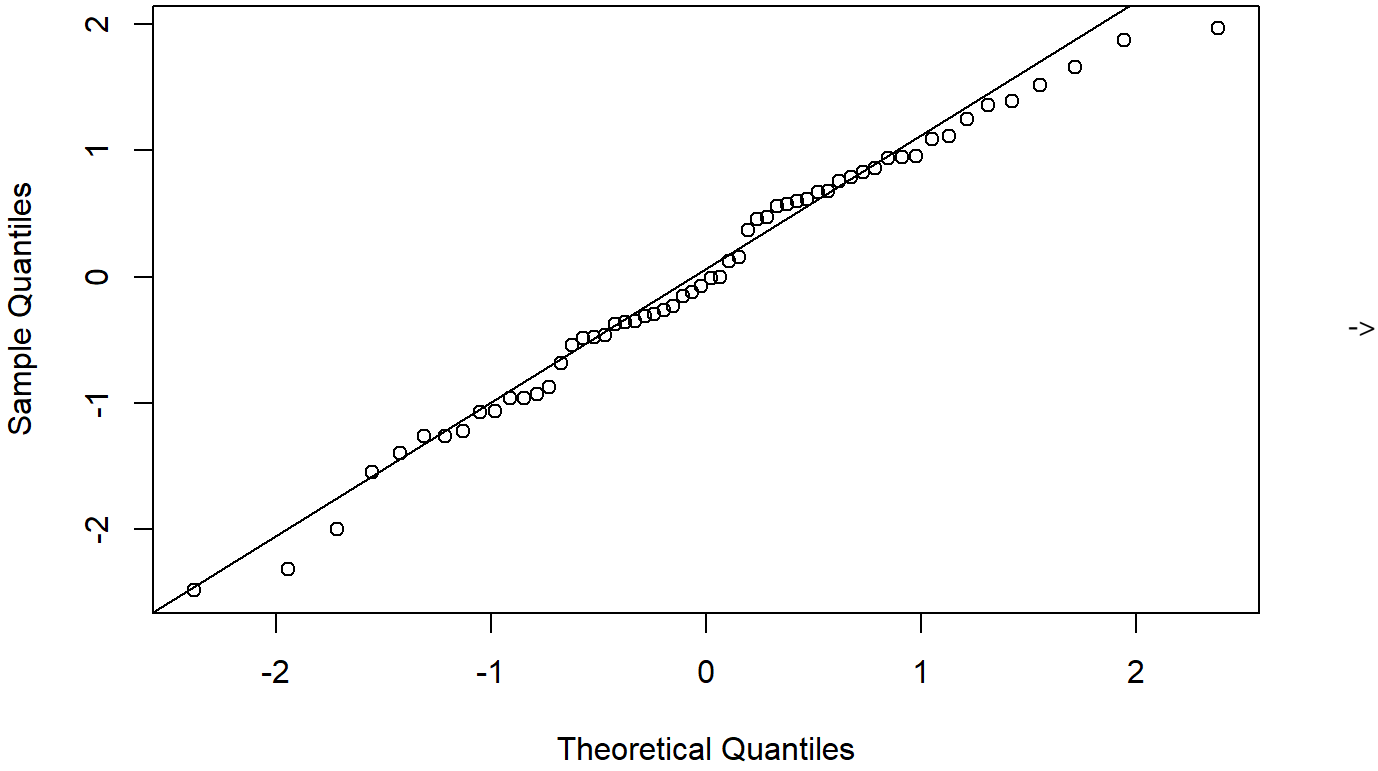


There are some outliers, so it is hard to say normally distributed completely.

2. [4 pts] PEC that fits a different linear model, this time for the log of the proportion of vehicles that are electric as a function of the proportion of people who work from home. Again check the assumption of normally-distributed residuals. What do you notice now?

```
log_proportion = log(prop_electric) ~ prop_WFH
fit_log = lm(log_proportion, data = car)

qqnorm(resid(fit_log))
qqline(resid(fit_log))
```

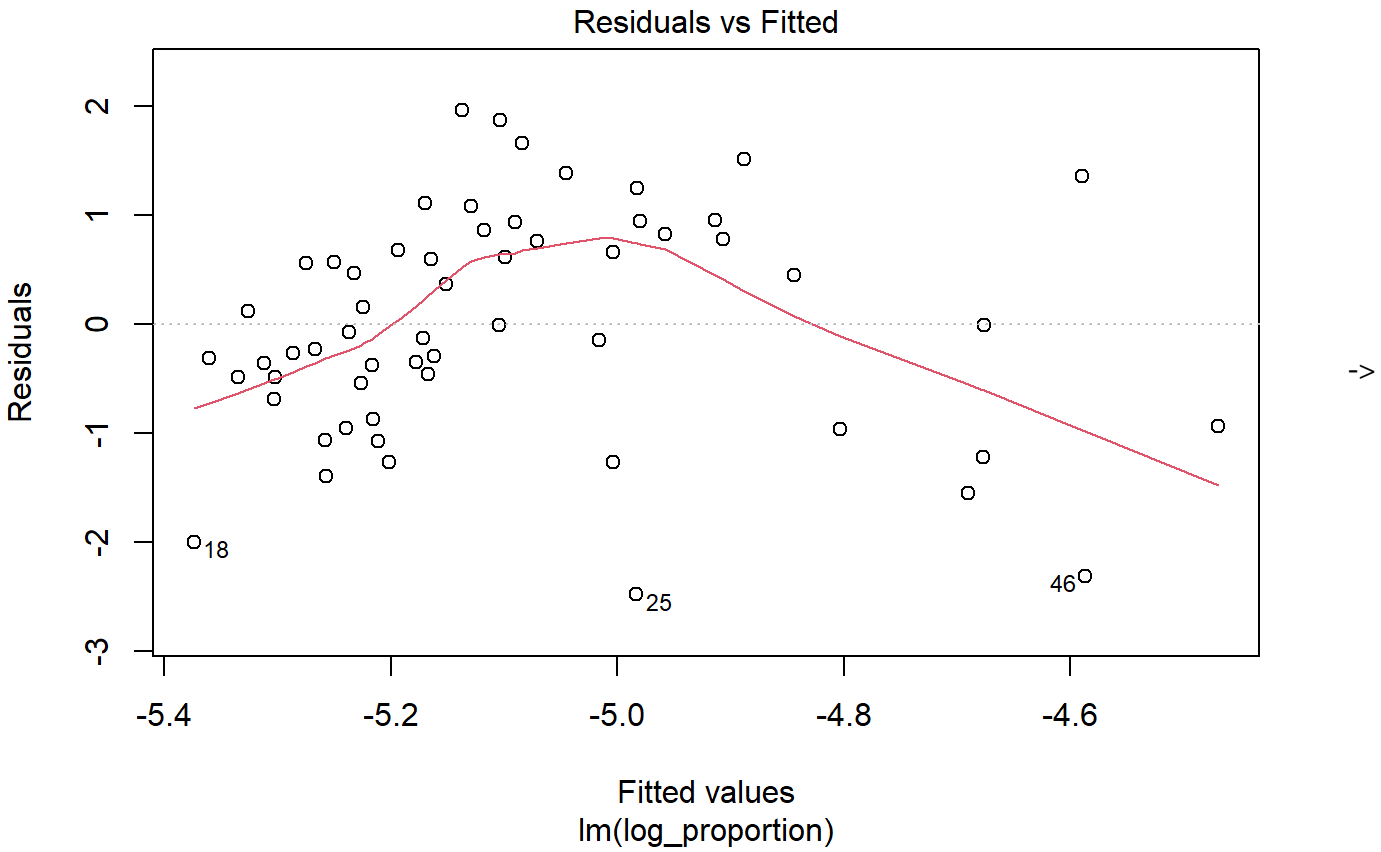


->

Yes ,can confirm normally-distributed residuals.

3. [4 pts] Make a plot of the residuals from your model fit in (2) as a function of the fitted values (hint: plot(…, which = 1)). What does your plot suggest about the validity of the assumptions of independent residuals and/or a true linear relationship between the response and predictor?

```
plot(fit_log, which = 1)
```



The residuals are mostly negative when the fitted value is small, positive when the fitted value is in the middle and negative when the fitted value is large. That is, the spread is approximately constant, but the conditional mean is not.

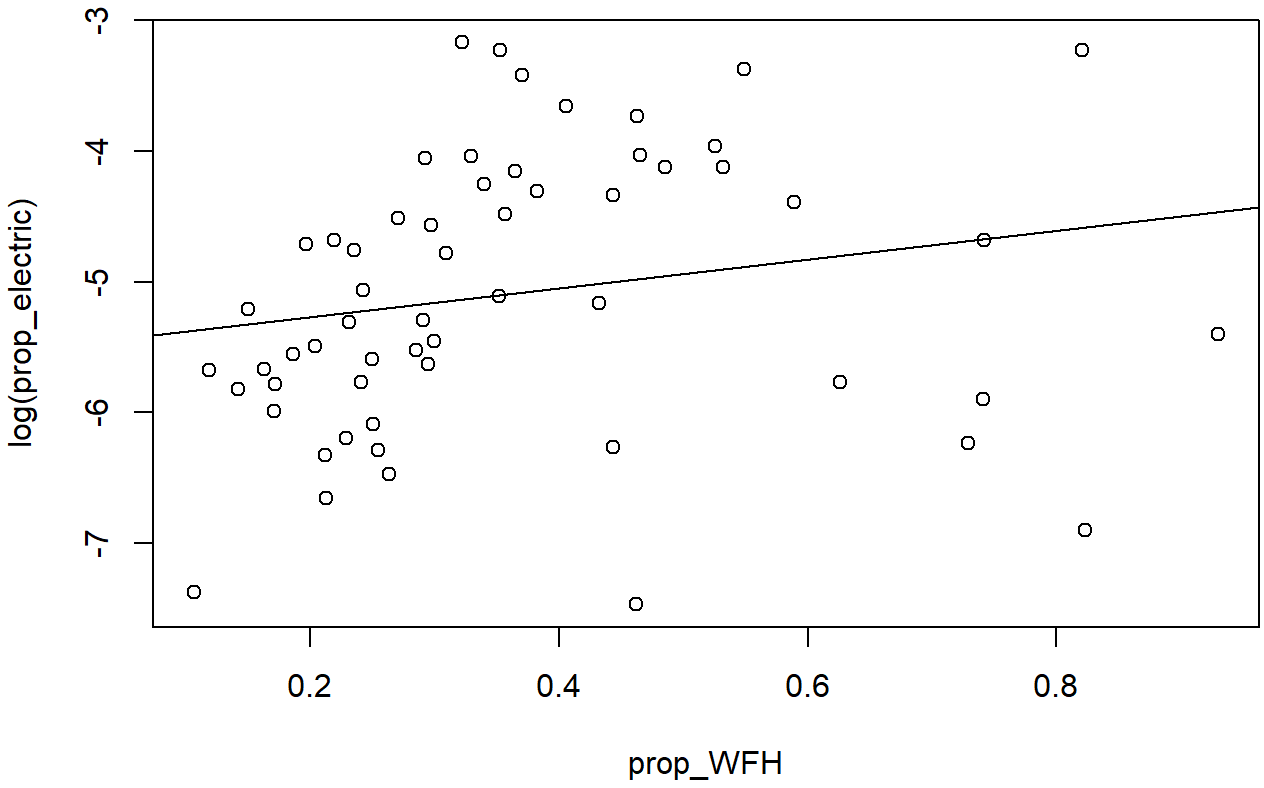
4. [4 pts] Make a plot of the log of the proportion of vehicles that are electric as a function of the proportion of people who work from home. Add a line that represents the fit from your model in (2) (hint: you might find the function abline() useful). Make a guess about why your plot in (3) looks the way it does.

```
summary(fit_log)
```

```
##
## Call:
## lm(formula = log_proportion, data = car)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.4831 -0.6482 -0.0397  0.7795  1.9651
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -5.4910     0.2965 -18.519   <2e-16 ***
## prop_WFH      1.0984     0.7205   1.524    0.133
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.043 on 56 degrees of freedom
## Multiple R-squared:  0.03985,    Adjusted R-squared:  0.0227
## F-statistic: 2.324 on 1 and 56 DF,  p-value: 0.133
```

```
plot(log(prop_electric)~ prop_WFH, data = car)

abline(a = -5.4910, b = 1.0984)
```



(5) [4 pts] What is the proportion of people who work from home in San Diego County? Use your model from

(2) to make a 95% confidence interval for the expected proportion of vehicles for counties with this proportion of people working from home. Make sure your prediction is for a proportion, not a log-proportion. How does your interval compare to the observed proportion of vehicles that are electric in San Diego County?

```
pred = predict(fit_log, newdata = data.frame(prop_WFH = 0.465493213), interval = "con
fidence")
exp(pred)
```

```
##           fit         lwr         upr
## 1 0.006876032 0.005041948 0.009377293
```

###II.Generalized Additive Model

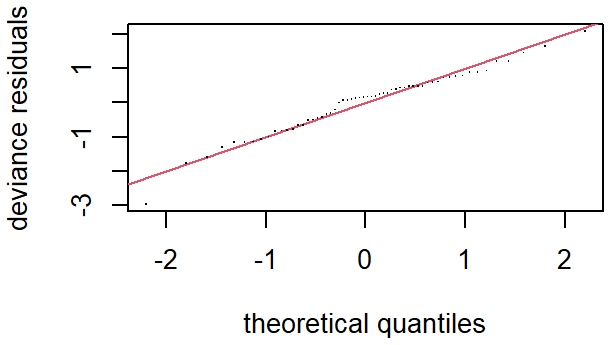
6. Use the mgcv package to fit a Generalized Additive Model (GAM) for the log of the proportion of vehicles that are electric as a flexible function of the proportion of people who work from home. Use the default smoothing basis of thin plate regression splines (i.e., bs = “tp”) and a basis dimension of 10. PEC. Make sure your coefficients match mine.

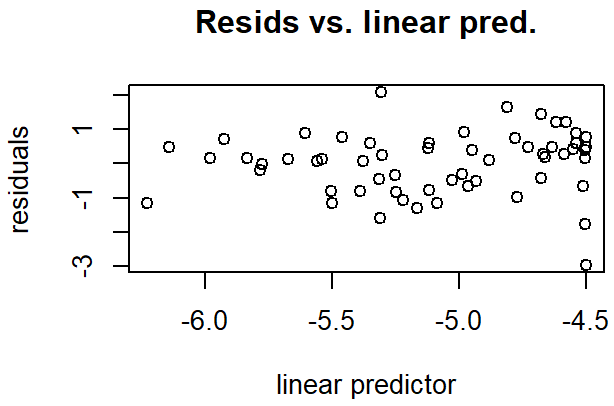
```
library(mgcv)
```

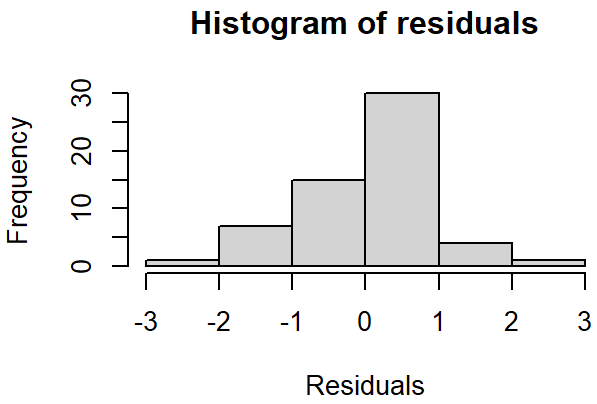
```
## 필요한 패키지를 로딩중입니다: nlme
```

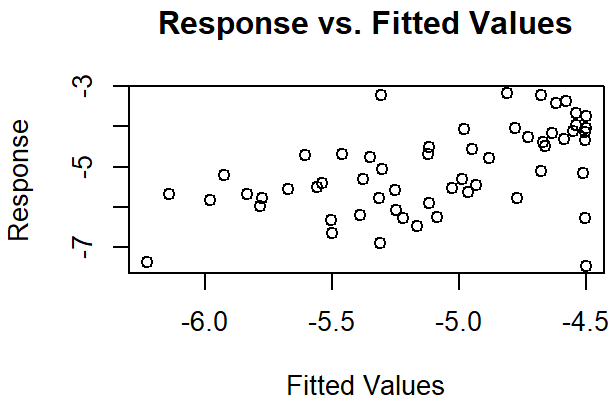
```
## This is mgcv 1.8-40. For overview type 'help("mgcv-package")'.
```

```
fit_gam = gam(log(prop_electric) ~ s(prop_WFH, k =10, bs = "tp"), data = car)
gam.check(fit_gam)
```







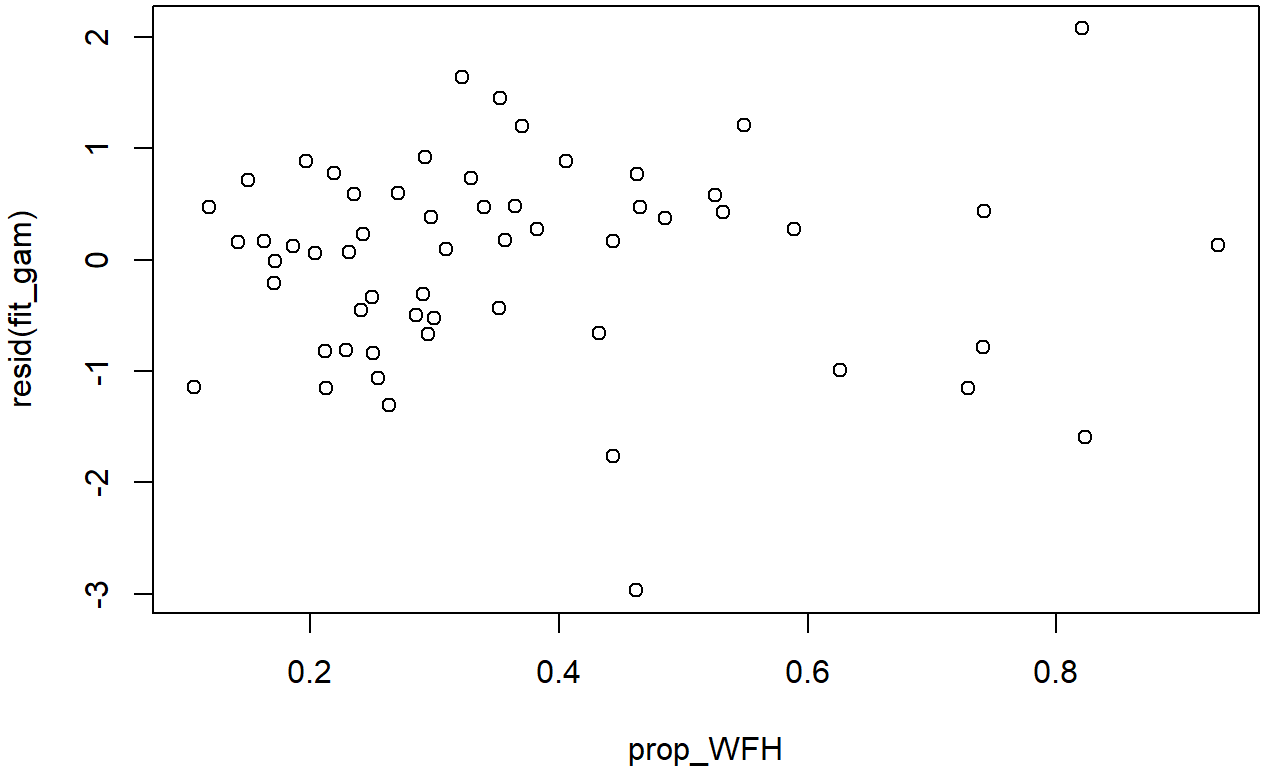



```
##
## Method: GCV   Optimizer: magic
## Smoothing parameter selection converged after 4 iterations.
## The RMS GCV score gradient at convergence was 3.68067e-05 .
## The Hessian was positive definite.
## Model rank =  10 / 10
##
## Basis dimension (k) checking results. Low p-value (k-index<1) may
## indicate that k is too low, especially if edf is close to k'.
##
##                k'  edf k-index p-value
## s(prop_WFH) 9.00 2.78    1.01    0.47
```

```
#fit$coefficients
```

7. Make a plot of residuals as a function of fitted values for your GAM in (6). How does it compare to the plot for the linear model in (3)? What does your new plot suggest about the validity of model assumptions?

```
plot(resid(fit_gam) ~ prop_WFH, data=car )
```



(8) [4 pts] Make a new 95% CI for the expected proportion of vehicles that are electric in counties with the same proportion of people working from home as in San Diego County using your GAM from (6). How does your interval compare to the one you made in (5)? To the observed proportion of vehicles that are electric in San Diego County?

```
pred = predict(fit_gam, newdata = data.frame(prop_WFH = 0.465493213), se.fit = TRUE)
CI = exp(pred$fit+ qnorm(c(0.025, 0.975))* pred$se.fit )
```

```
## Warning in qnorm(c(0.025, 0.975)) * pred$se.fit: Recycling array of length 1 in ve
ctor-array arithmetic is deprecated.
##   Use c() or as.vector() instead.
```

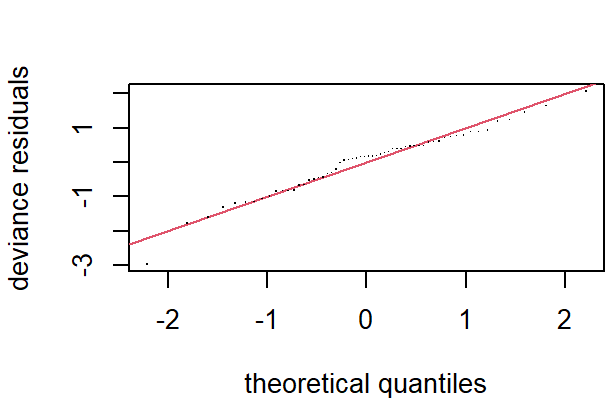
```
## Warning in pred$fit + qnorm(c(0.025, 0.975)) * pred$se.fit: Recycling array of len
gth 1 in array-vector arithmetic is deprecated.
##   Use c() or as.vector() instead.
```

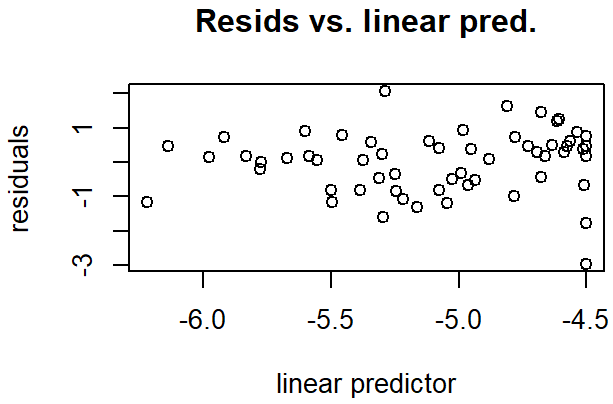
```
CI
```

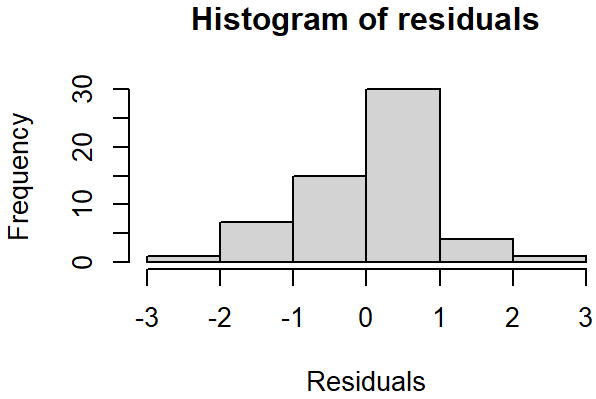
```
## [1] 0.007245468 0.017053922
```

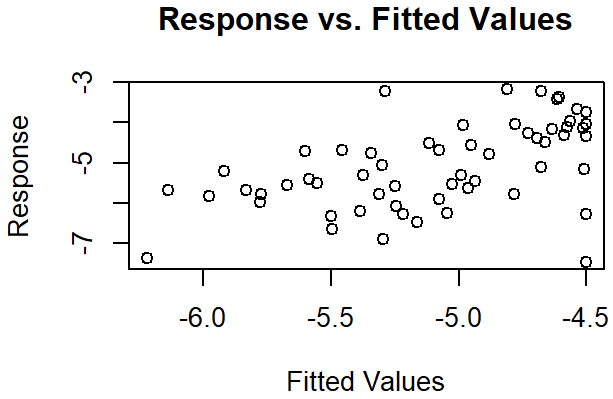
9. [4 pts] Fit a new GAM for the log proportion of vehicles that are electric using cubic regression splines instead of thin plate splines and still uses a basis dimension of 10. PEC.

```
library(mgcv)
fitte = gam(log(prop_electric) ~ s(prop_WFH, k =10, bs = "cr"), data = car)
gam.check(fitte)
```










```
##
## Method: GCV   Optimizer: magic
## Smoothing parameter selection converged after 10 iterations.
## The RMS GCV score gradient at convergence was 1.808044e-05 .
## The Hessian was positive definite.
## Model rank =  10 / 10
##
## Basis dimension (k) checking results. Low p-value (k-index<1) may
## indicate that k is too low, especially if edf is close to k'.
##
##                k'  edf k-index p-value
## s(prop_WFH) 9.00 2.65    1.01    0.47
```

10. [4 pts] Make a plot of proportion of vehicles that are electric as a function of the proportion of people who work from home. Add three curves representing the predicted proportion of vehicles that are electric according to the (1) linear, (2) GAM with thin plate regression splines and (3) GAM with cubic regression splines.

```
sequence = seq(from = min(car$prop_WFH), to = max(car$prop_WFH), length.out = 1000)
pred_1 =predict(fit_log, newdata = data.frame(prop_WFH = sequence), se.fit = T)
pred_2 =predict(fit_gam, newdata = data.frame(prop_WFH = sequence), se.fit = T)
pred_3 =predict(fitte, newdata = data.frame(prop_WFH = sequence), se.fit = T)
plot(prop_electric ~ prop_WFH, data = car)

lines(sequence, exp(pred_1$fit), lwd = 2, col = 1)
lines(sequence, exp(pred_2$fit), lwd = 2, col = 2)
lines(sequence, exp(pred_3$fit), lwd = 2, col = 3)
```