Work completed by Joel Vinas

AI usage:

Gemini Search was used to produce R code which would:

1. *Avoid an axis alignment error*
2. *Produce a Title on plot function*
3. *Produce a Title on a Panel of Residuals*
4. *Properly import carData to use vif() function*

Links:

- Google CoLab:
  - https://colab.research.google.com/drive/1Iwn4sM5gFzTbp8JqQCry6g_UxzRq0PuI?usp=sharing
- GitHub:
  - https://github.com/joelvinas/COMP-SCI_5565/blob/main/Assignment%202/Output/Assignment_2_Linear_Regression.ipynb
- GitHub (Raw):
  - https://raw.githubusercontent.com/joelvinas/COMP-SCI_5565/refs/heads/main/Assignment%202/Output/Assignment_2_Linear_Regression.ipynb

This document has been modified from the source to improve readability in the PDF format.

(1) Select a dataset to implement your own version of the "Linear Regression" exercise above. Include your scripts, the results, and 2 relevant plots:

- Regression
- 4 Panel of residuals

```
#Data Package = MASS: https://cran.r-project.org/web/packages/MASS/MASS.pdf
#Cabbages: Page 23
# Data from a cabbage field trial
# The cabbages data set has 60 observations and 4 variables
#Format
#  This data frame contains the following columns:
#  Cult    Factor giving the cultivar of the cabbage, two levels: c39 and c52.
#  Date    Factor specifying one of three planting dates: d16, d20 or d21.
#  HeadWt  Weight of the cabbage head, presumably in kg.
#  VitC    Ascorbic acid content, in undefined units.
#Source
#  Rawlings, J. O. (1988) Applied Regression Analysis: A Research Tool. Wadsworth and Brooks/Cole. Example 8.4, page 219.
#  (Rawlings cites the original source as the files of the late Dr Gertrude M Cox.)
library(MASS)
head(cabbages)
attach(cabbages)
lm.fit <- lm(HeadWt ~ VitC, data = cabbages)
#lm.fit <- lm(VitC ~ HeadWt, data = cabbages)
lm.fit
summary(lm.fit)
```

Call:
lm(formula = HeadWt ~ VitC, data = cabbages)

Coefficients:

(Intercept)	VitC
5.92806	-0.05754

Call:
lm(formula = HeadWt ~ VitC, data = cabbages)

Residuals:

Min	1Q	Median	3Q	Max
-1.0150	-0.5117	-0.1575	0.4244	1.6095

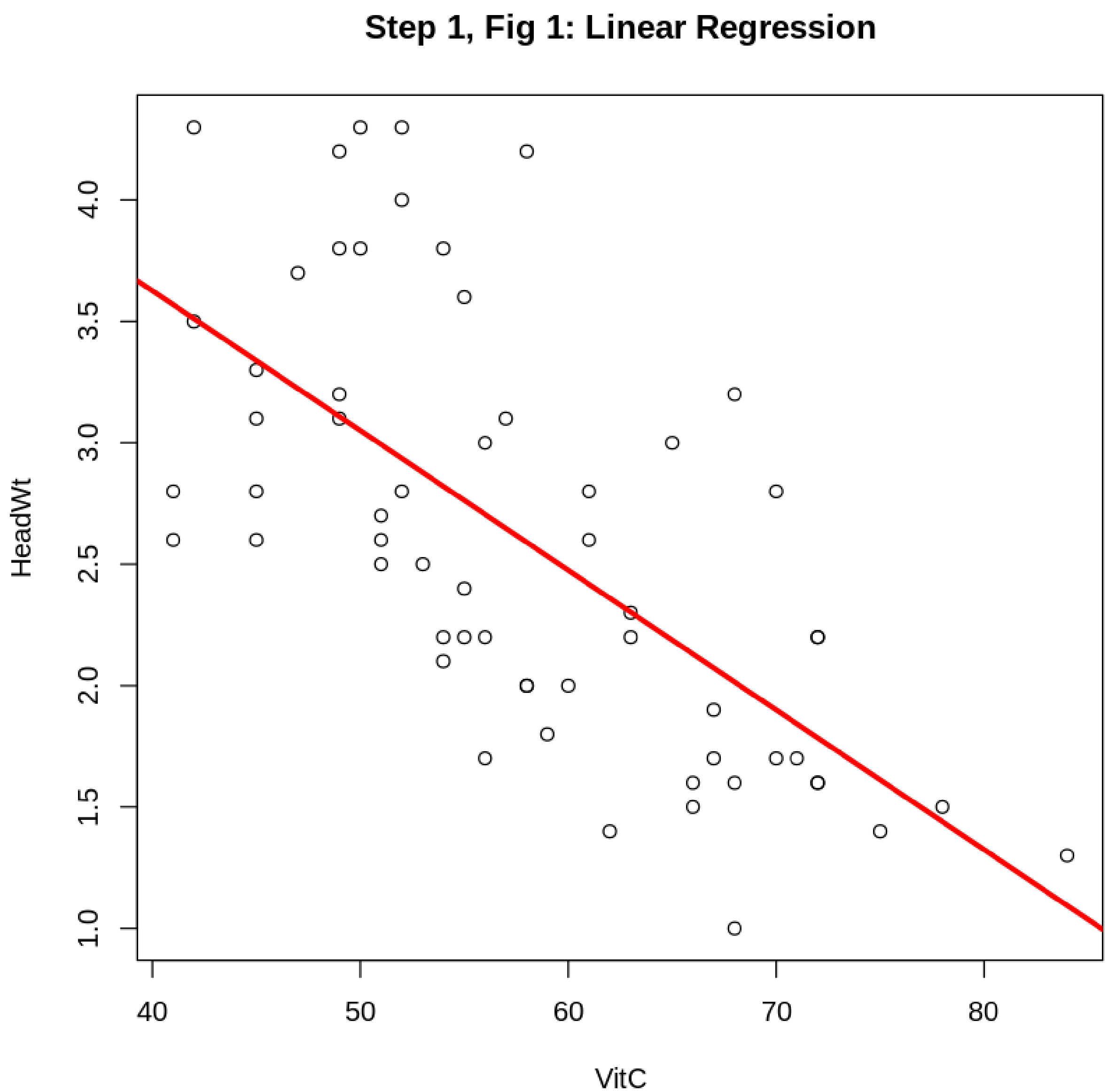
Coefficients:

	Estimate	Std. Error	t value	Pr(>|t|)	

(Intercept)	5.928059	0.505983	11.716	< 2e-16	***
VitC	-0.057545	0.008603	-6.689	9.75e-09	***
Signif. codes: *0 ‘***’* *0.001 ‘**’* *0.01 ‘*’* *0.05 ‘.’* *0.1 ‘ ’* *1*					

Residual standard error: 0.6687 on 58 degrees of freedom
Multiple R-squared: 0.4355
Adjusted R-squared: 0.4257
F-statistic: 44.74 on 1 and 58 DF
p-value: 9.753e-09

```
#Step 1.1: Linear Regression
plot(VitC, HeadWt)
abline(lm.fit, lwd = 3, col = "red")
title("Step 1, Fig 1: Linear Regression")
```



Notes on Panels of Residuals:

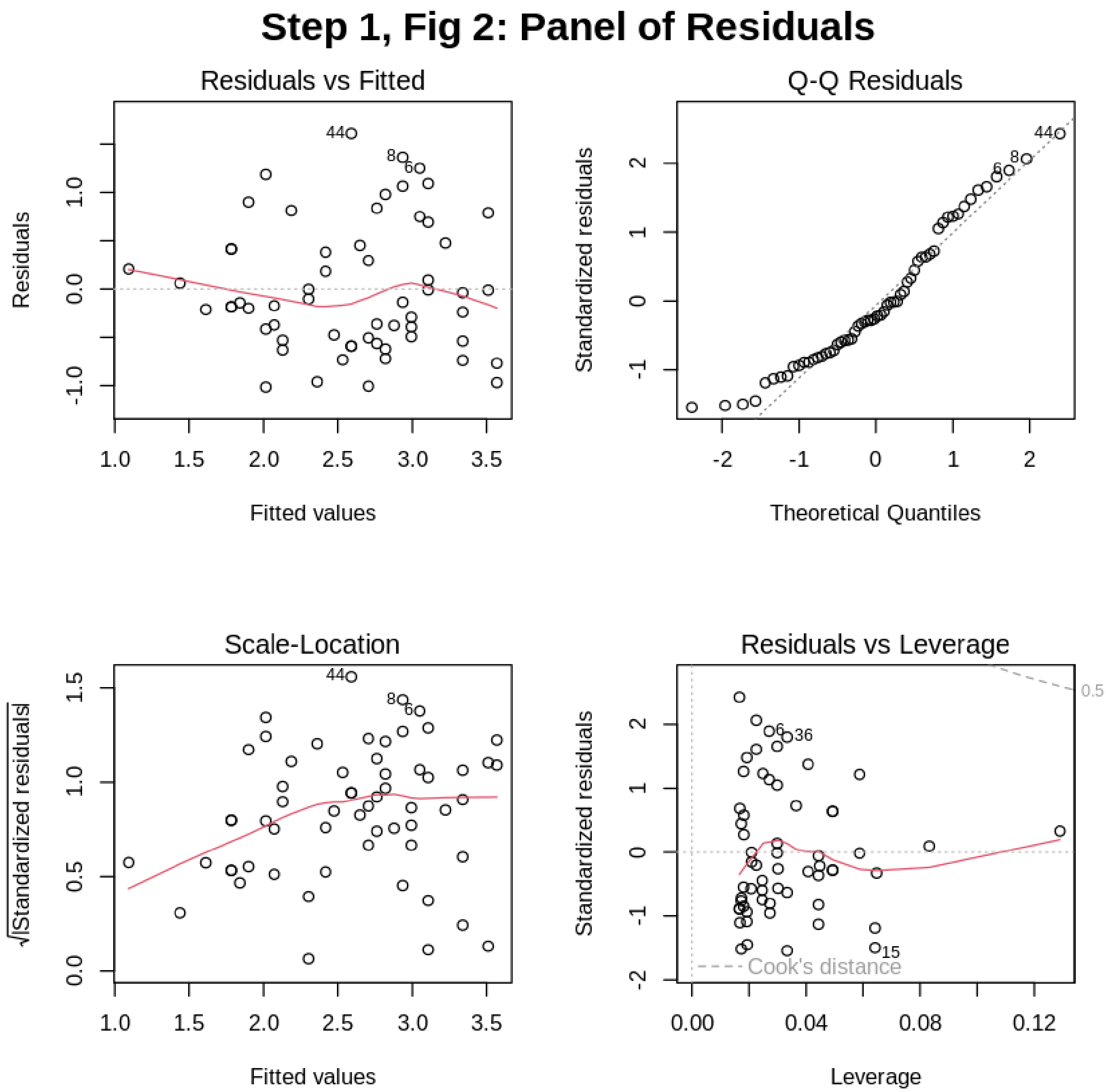
Four diagnostic plots are automatically produced by applying the plot() function directly to the output from lm(). In general, this command will produce one plot at a time, and hitting Enter will generate the next plot.

However, it is often convenient to view all four plots together. We can achieve this by using the par() and mfrow() functions, which tell R to split the display screen into separate panels so that multiple plots can be viewed simultaneously. For example, par(mfrow = c(2, 2)) divides the plotting region into a 2×2 grid of panels.

Before creating the plots, use par(oma) to set aside space in the outer margins for the main title. The third element of oma controls the top outer margin.

After all plots in the panel are created, use mtext() with outer = TRUE to place the title in the outer margin.

```
#Step 1.2: Panel of Residuals
par(mfrow = c(2, 2))
plot(lm.fit)
par(mfrow = c(2, 2), oma = c(0, 0, 3, 0), mar = c(4, 4, 2, 2) + 0.1)
# mar adjusts inner margins
mtext("Step 1, Fig 2: Panel of Residuals", side = 3, line = 1, outer
= TRUE, cex = 1.5, font = 2)
```



(2) Apply the methods of the "Multiple Linear" regression.

- Provide a 4 plot of the residuals, including the leverage.
- Provide the scripts and results.

#In order to fit a multiple linear regression model using least squares, we again use the lm() function.
#The syntax lm(y ~ x1 + x2 + x3) is used to fit a model with three predictors, x1, x2, and x3.
#The summary() function now outputs the regression coefficients for all the predictors.

#lm.fit <- lm(HeadWt ~ VitC + Date, data = cabbages)
#summary(lm.fit)
#The Cabbages data set contains 4 variables. Although we could type in these values, there is a better method using the following short-hand:

```
lm.fit <- lm(HeadWt ~ ., data = cabbages)
summary(lm.fit)
```

Call: lm(formula = HeadWt ~ ., data = cabbages)
Residuals: Min 1Q Median 3Q Max -1.03111 -0.48389 -0.09277 0.30036 1.41756

Coefficients:

	Estimate	Std. Error	t value	Pr(>|t|)	
(Intercept)	5.73790	0.67344	8.520	1.25e-11	***
Cultc52	0.07181	0.24103	0.298	0.766876	
Dated20	0.11317	0.21357	0.530	0.598309	
Dated21	-0.24140	0.22986	-1.050	0.298234	
VitC	-0.05415	0.01303	-4.155	0.000114	***
*Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1*					

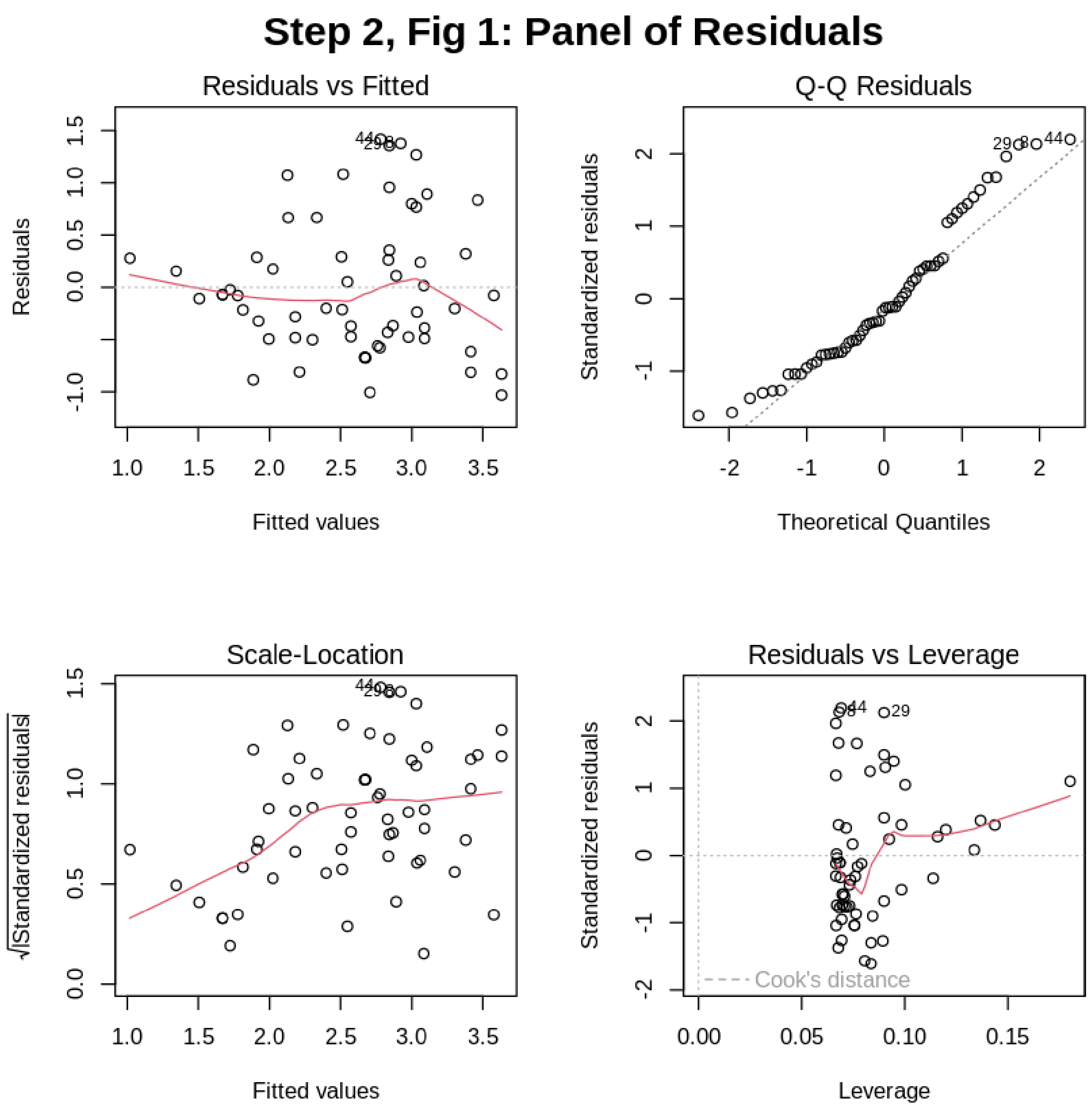
Residual standard error: 0.669 on 55 degrees of freedom
Multiple R-squared: 0.4642, Adjusted R-squared: 0.4252
F-statistic: 11.91 on 4 and 55 DF, p-value: 4.861e-07

```
#We can access the individual components of a summary object by name (type ?summary.lm to see what is available).
#Hence summary(lm.fit)$r.sq gives us the R2, and summary(lm.fit)$sigma gives us the RSE.
#The vif() function, part of the car package, can be used to compute variance inflation factors.
# (As the VIF values are below 5, most VIF's are low to moderate for this data)
install.packages("car") #The car package is not part of the base R installation so it must be downloaded the first time you use it via the install.packages() function in R.
library(car) ## Loading required package: carData
vif(lm.fit)
```

A matrix: 3 × 3 of type dbl

	GVIF	Df	GVIF^(1/(2*Df))
Cult	1.947162	1	1.395407
Date	1.345033	2	1.076919
VitC	2.292195	1	1.514000

```
#Step 2: Fig 1: Panel of Residuals with leverage
par(mfrow = c(2, 2))
plot(lm.fit)
par(mfrow = c(2, 2), oma = c(0, 0, 3, 0), mar = c(4, 4, 2, 2) + 0.1)
# mar adjusts inner margins
mtext("Step 2, Fig 1: Panel of Residuals", side = 3, line = 1, outer
= TRUE, cex = 1.5, font = 2)
```



#What would the Panel of Residuals look like without VitC?
#To run a regression excluding this predictor, use the following syntax to run a regression using all predictors except VitC.

```
lm.fit2 <- lm(HeadWt ~ . - VitC, data = cabbages)
summary(lm.fit2)

par(mfrow = c(2, 2))
plot(lm.fit2)
par(mfrow = c(2, 2), oma = c(0, 0, 3, 0), mar = c(4, 4, 2, 2) + 0.1)
# mar adjusts inner margins
mtext("Step 2, Fig 2: Panel of Residuals (w/o VitC)", side = 3, line
= 1, outer = TRUE, cex = 1.5, font = 2)
```

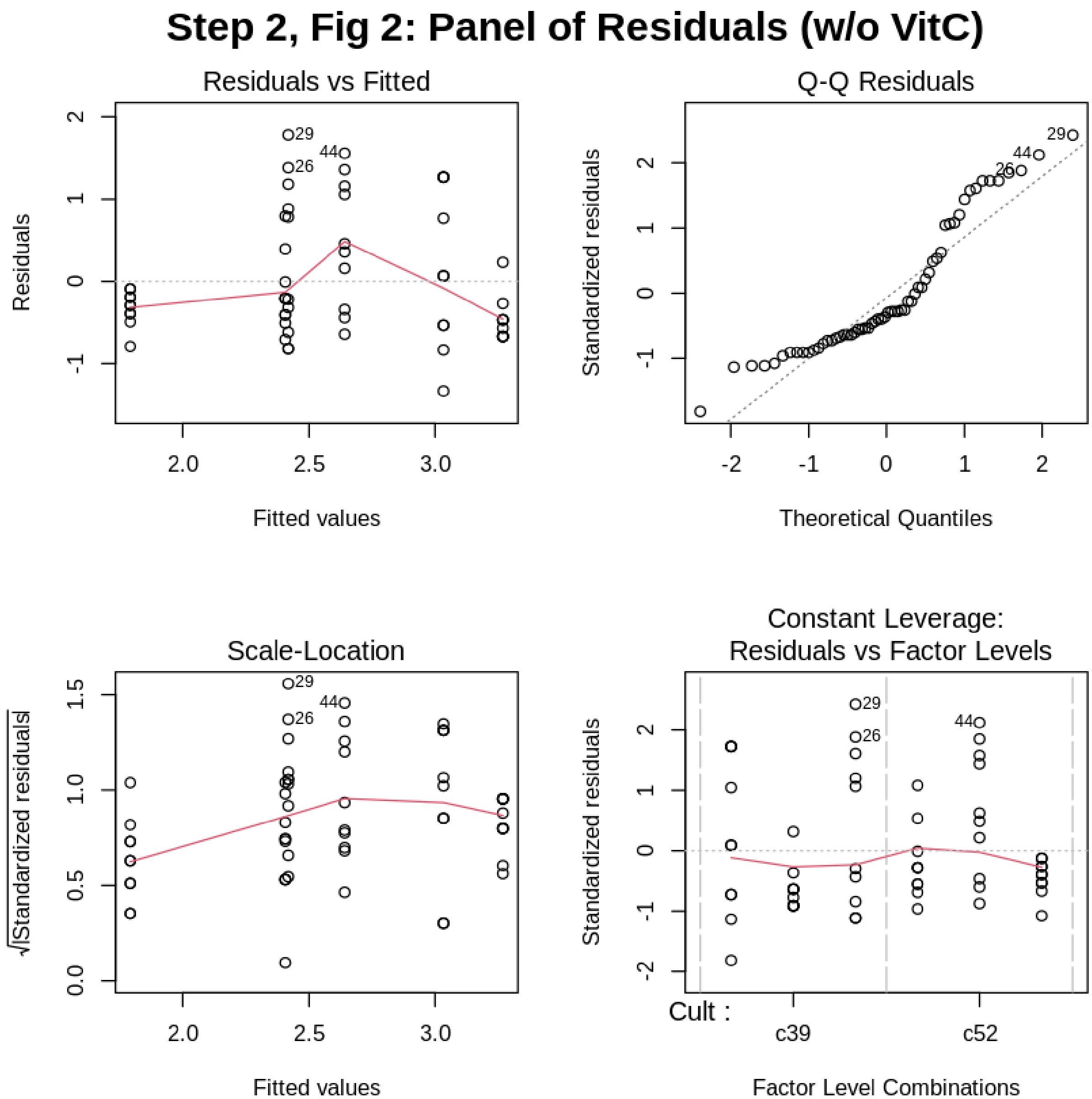
Call: lm(formula = HeadWt ~ . - VitC, data = cabbages)
Residuals:

Min	1Q	Median	3Q	Max
-1.3333	-0.5133	-0.2433	0.4096	1.7817

Coefficients:

	Estimate	Std. Error	t value	Pr(>|t|)	
(Intercept)	3.0333	0.1962	15.459	< 2e-16	***
Cultc52	-0.6267	0.1962	-3.194	0.00231	**
Dated20	0.2350	0.2403	0.978	0.33233	
Dated21	-0.6150	0.2403	-2.559	0.01322	*
Signif. codes: *0 '***'* *0.001 '**'* *0.01 '*'* *0.05 '.'* *0.1 ' '* *1*					

Residual standard error: 0.7599 on 56 degrees of freedom
Multiple R-squared: 0.296
Adjusted R-squared: 0.2583
F-statistic: 7.848 on 3 and 56 DF
p-value: 0.0001847


Step 2, Fig 2: Panel of Residuals (w/o VitC)
Residuals vs Fitted
Residuals
Fitted values
Q-Q Residuals
Standardized residuals
Theoretical Quantiles
Scale-Location
√|Standardized residuals|
Fitted values
Constant Leverage:
Residuals vs Factor Levels
Standardized residuals
Cult :
c39
c52
Factor Level Combinations

(3) Generate a paragraph describing the most significant finding from your personal experience with the exercise what do you think was most interesting? Did you discover, see in practice, or better understand any concept related to our class discussions?

For this data set, it is clear that there is a strong inverse relationship between the Ascorbic acid and the weight of the cabbage head. That is, as the amount of acid increases, the cabbage yields become smaller by weight. With a p-value of 9.753e-09, this relationship is clearly significant.

I was surprised to find that the differences between the Linear Regression and Multiple Linear Regression were hardly noticable. This is likely due to the impact that the Ascorbic Acid content had on the weight of the cabbage head. I attempted to remove the VitC to see if a second strong relationship could be found, but since the other factors were categorical rather than numeric, the data did not show a distinct trend.

I found the application of the Linear Regression model against this data set to be the most striking. The visualization clarified how a relationship can be defined between two factors. Observing the weakness of the relationships between weight and the cultivar (c39 or c52) or date of the planting (d16, d20 or d21) makes me curious to find other methods to extract meaning from these variables - or determine if no strong relationship exists.