W271 Group Lab 1

Ren Tu, Erin Werner & Michael Zeng

September 20th, 2020

Contents

Part 1 - Exploratory Data Analysis

(a) Data at a Glimpse

```
library(ggplot2)
library(dplyr)
library(car)
library(readr)
library(expss)
library(scales)
library(kableExtra)
library(gridExtra)
library(grid)
knitr::opts_chunk$set(tidy.opts=list(width.cutoff=60))
set.seed(2020)
```

```
challenger_orig <- read_csv(paste0(here::here(),"/labs/lab_1/challenger.csv"))
```

```
## Parsed with column specification:
## cols(
##   Flight = col_double(),
##   Temp = col_double(),
##   Pressure = col_double(),
##   O.ring = col_double(),
##   Number = col_double()
## )
```

```
head(challenger_orig)
```

```
## # A tibble: 6 x 5
##   Flight  Temp Pressure O.ring Number
##    <dbl> <dbl>    <dbl>  <dbl>  <dbl>
## 1      1    66       50      0      6
## 2      2    70       50      1      6
## 3      3    69       50      0      6
## 4      4    68       50      0      6
## 5      5    67       50      0      6
## 6      6    72       50      0      6
```

```
summary(challenger_orig)
```

```
##      Flight          Temp          Pressure         O.ring           Number
##  Min.   : 1.0   Min.   :53.00   Min.   : 50.0   Min.   :0.0000   Min.   :6
##  1st Qu.: 6.5   1st Qu.:67.00   1st Qu.: 75.0   1st Qu.:0.0000   1st Qu.:6
##  Median :12.0   Median :70.00   Median :200.0   Median :0.0000   Median :6
##  Mean   :12.0   Mean   :69.57   Mean   :152.2   Mean   :0.3913   Mean   :6
##  3rd Qu.:17.5   3rd Qu.:75.00   3rd Qu.:200.0   3rd Qu.:1.0000   3rd Qu.:6
##  Max.   :23.0   Max.   :81.00   Max.   :200.0   Max.   :2.0000   Max.   :6
```

```
paste("Sample Size: ", nrow(challenger_orig))
```

```
## [1] "Sample Size:  23"
```

From the printout of the Challenger data set, we see:

- It is a small data set with only 23 observations and 5 columns.
- Column `Flight` goes from 1 to 23 for the 23 rows, thus is an identification column that shouldn't be

used for modeling.

- Column `Temp` is a numeric column, which typically consists of values above 60. This will be one of the main features that we will use to build our model.
- Column `Pressure` is a numerical column yet it only takes on 3 values, thus we could also consider modeling with it being a categorical feature. Those three values are 50, 100, and 200 psi. This column will act as an additional feature that we will consider in our model.
- Column `O.ring` takes on 3 integer values, which indicate the number of O-rings under stress per test launch. Zero indicates a success with no O-ring failures. Values greater than zero indicate the number of total O-ring failures per launch. The two possible types of failure are erosion and blow-by. According to Dalal_etal_1989, this variable should be considered as the binomial outcome with n = 6. This will act as the label of the exercise.
- Column `Number` is a constant 6, which indicates the total number of O-ring observations per launch, thus shouldn't be included in the model.

(b) Data Visualization and Preliminary Analysis

We will now conduct and exploratory data analysis through graphs and tables to gain a more in depth understanding of the data that we are working with in order to build our model. We added additional columns for visualization purposes.

```
challenger_orig$success_asint <- ifelse(challenger_orig$O.ring > 0, 0, 1)
challenger_orig$success <- as.factor(challenger_orig$success_asint)
challenger_orig$fail <- as.integer(ifelse(challenger_orig$O.ring > 0, 1, 0))
challenger_orig$proportion <- challenger_orig$O.ring/challenger_orig$Number
challenger_orig$oring_cat <- as.factor(challenger_orig$O.ring)
challenger_orig$pressure_cat <- as.factor(challenger_orig$Pressure)
```

At this point, we know that we are modeling the number of O-rings under pressure given a numerical column `Temp` and a categorical column `Pressure`. We made following tables and visualizations to further understand the data.

(b.1) Univariate: Pressure

```
challenger_orig = apply_labels(challenger_orig,pressure_cat = "Pressure (psi)",
                      oring_cat = "Results",Temp = "Temperature (°F)")
temp <- as.data.frame(cro(challenger_orig$pressure_cat, challenger_orig$O.ring))
colnames(temp) <- c("", "Success", "1 O-Ring Failure", "2 O-Ring Failure")
rownames(temp) <- c("50 psi", "100 psi", "200 psi", "Total")
temp$Total = coalesce(temp$Success, 0) + coalesce(temp$`1 O-Ring Failure`, 0) +
             coalesce(temp$`2 O-Ring Failure`,0)
temp[2:5]
```

```
##         Success 1 O-Ring Failure 2 O-Ring Failure Total
## 50 psi        5                1               NA     6
## 100 psi       2               NA               NA     2
## 200 psi       9                4                2    15
## Total        16                5                2    23
```

This table shows us the number of total O-ring failures per launch at each pressure level. We can see that there are a majority of successful launches. An important observation is that failures occur a both low and high pressure levels, implying that this feature may not be as influential in our model.

(b.2) Univariate: Temperature

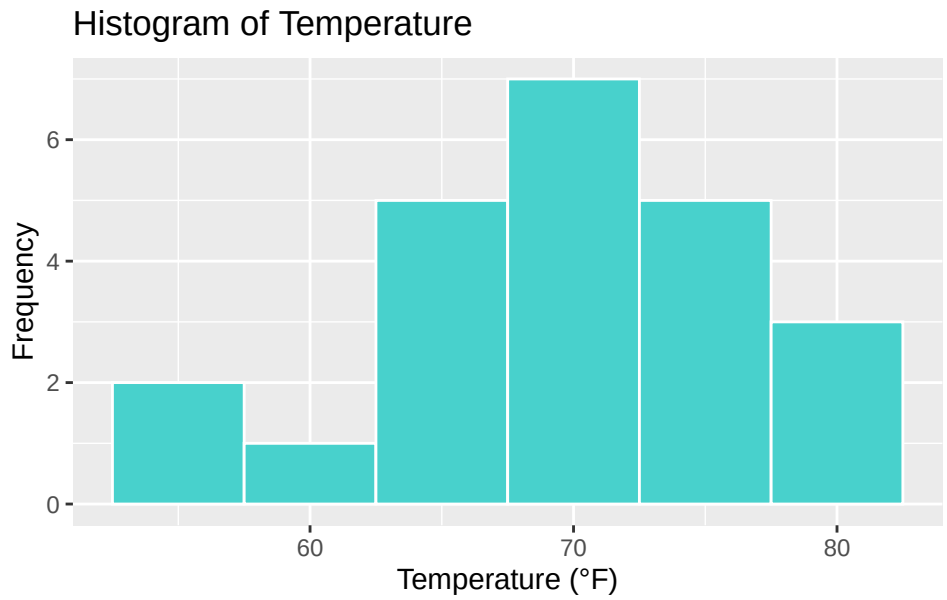
However, our primary focus will be the explanatory variable, `Temp`.

```
summary(challenger_orig$Temp)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   53.00   67.00   70.00   69.57   75.00   81.00
```

The temperatures in our data set range from 53°F to 81°F, with a mean of about 70°F. This means that most of the test launches occurred at reasonably warm temperatures compared to the 31°F temperature of the actual launch.

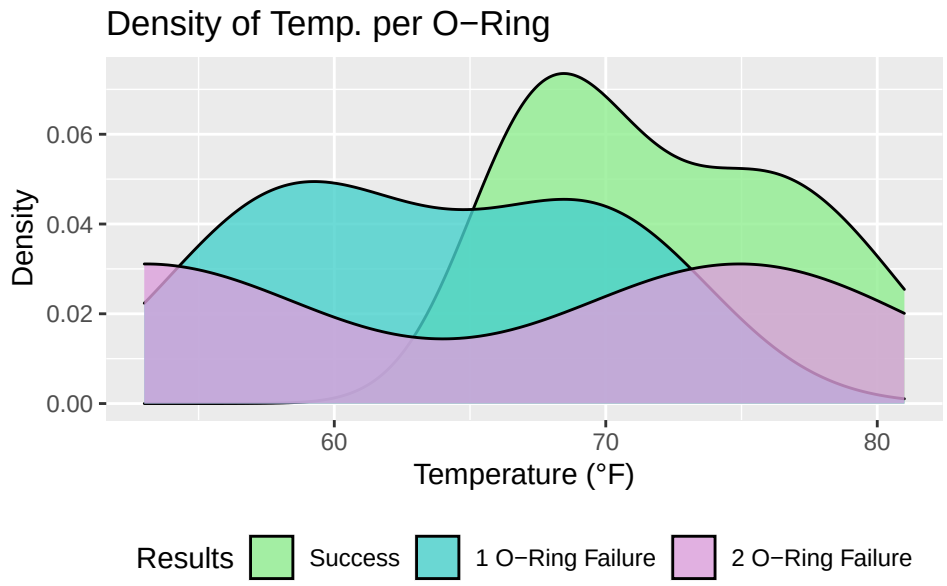
```
ggplot(challenger_orig, aes(Temp)) +
  geom_histogram(binwidth = 5, fill = "mediumturquoise",col="white",size = 0.5)+
  labs(title="Histogram of Temperature", x = "Temperature (°F)",y = "Frequency")
```



In the histogram above for our main explanatory variable `Temp`, there appears to be no major outliers or any fat tails. Overall, the distribution appears to be primarily uni-modal with a peak around the mean although there is a small peak at the lower end of the temperature distribution. We can see a mostly normal distribution, with slight negative skew towards higher temperature values. Yet, since we know that our sample size is only 23, which is less than 30, we can not rely on asymptotic assumptions of normality for our data. This will be important for our model assumptions moving forward.

Yet, this histogram only reveals the distribution of temperatures, without any indication of how successful the launch was at each given temperature. So, we can generate an additional plot to help provide this insight.

```
ggplot(challenger_orig, aes(Temp)) + theme(legend.position="bottom") +
  geom_density(aes(fill=factor(oring_cat)), alpha=0.8) +
  scale_fill_manual("Results", values=c("lightgreen","mediumturquoise","plum"),
                    labels=c("Success","1 O-Ring Failure","2 O-Ring Failure"))+
  labs(title="Density of Temp. per O-Ring", x="Temperature (°F)", y = "Density")
```

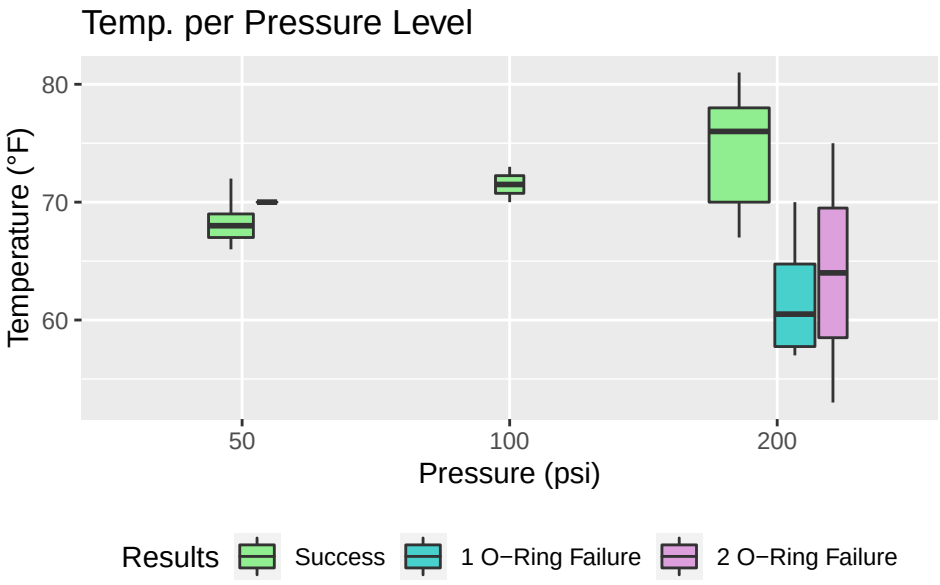


The density plot reveals a more detailed distribution of temperatures with regards to the total number of O-ring failures per launch. We can observe that most successful launches occur at higher temperatures, where as the failed O-ring instances have lower temperatures.

Now that we have examined both `Temp` and `Pressure` individually, we can examine both features together.

(b.3) Bivariate: Temperature + Pressure

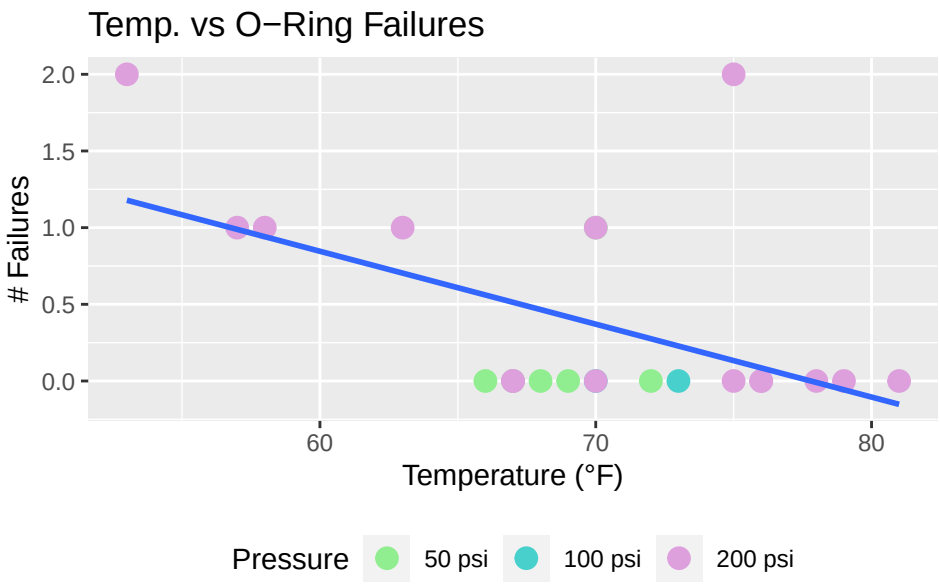
```
ggplot(challenger_orig,aes(pressure_cat,Temp))+theme(legend.position="bottom") +
  geom_boxplot(varwidth=T, aes(fill = factor(oring_cat))) +
  scale_fill_manual("Results",values=c("lightgreen","mediumturquoise","plum"),
                    labels=c("Success","1 O-Ring Failure","2 O-Ring Failure")) +
  labs(title="Temp. per Pressure Level",x="Pressure (psi)",y="Temperature (°F)")
```



From the box plot, we can see that most launches were performed with a pressure of 200 psi. It is also at this pressure that the majority of failed O-ring's occurred, most of which occurred at lower temperatures than that of the successes. This is a strong indication that temperature has a strong impact on the O-rings.

Another way to look at the relationship of these two features is to focus on the scatter plot.

```
ggplot(challenger_orig, aes(x=Temp, y=O.ring)) +
  geom_point(fill="gray", size = 3.5, aes(col=pressure_cat)) +
  geom_smooth(method="lm", se=F) + theme(legend.position="bottom") +
  scale_colour_manual("Pressure",values=c("lightgreen","mediumturquoise","plum"),
                      labels=c("50 psi", "100 psi", "200 psi")) +
  labs(title="Temp. vs O-Ring Failures",y="# Failures",x="Temperature (°F)")
```



Above, we notice the scatter plot of temperature vs. the number of failures as well as the predicted linear regression line, grouped by pressure level. There appears to be a negative relationship between the number of failures and temperature. This means that as temperature increases, O-ring failures are less likely to occur.

(c) Data Augmentation

Currently the label in the data-set is `O.ring`, which takes integer values 0, 1 and 2. Therefore, it is not suitable to directly fit into a logistic regression. One way to estimate the logistic regression is to use `O.ring/Number` as the label, which assumes each launch is independent and that the proportion of O-ring failures in each launch is an efficient estimator of failure rate. However, by doing this, we are still collapsing the unit of observation into a single launch, instead of single O-ring. It will yield sensible estimates, but due to the lower number of observations, the standard errors will be big and so will the confidence intervals.

So, to better estimate the logistic regression model, we had to transform the data such that each row represents an O-ring instead of an overall launch.

```
Temp <- c()
Pressure <- c()
O.ring <- c()

for(i in challenger_orig$Flight){
  t <- challenger_orig[which(challenger_orig$Flight == i),]
  if(t$O.ring == 0){O.ring <- c(O.ring, rep(0,6))}
  if(t$O.ring == 1){O.ring <- c(O.ring, 1, rep(0,5))}
  if(t$O.ring == 2){O.ring <- c(O.ring, 1, 1, rep(0,4))}
  Temp <- c(Temp, rep(t$Temp,6))
  Pressure <- c(Pressure, rep(t$Pressure,6))
}
```

```
challenger <- as.data.frame(cbind(Temp,Pressure,O.ring))
```

```
summary(challenger)
```

```
##       Temp          Pressure         O.ring
##  Min.   :53.00   Min.   : 50.0   Min.   :0.00000
##  1st Qu.:67.00   1st Qu.: 50.0   1st Qu.:0.00000
##  Median :70.00   Median :200.0   Median :0.00000
##  Mean   :69.57   Mean   :152.2   Mean   :0.06522
##  3rd Qu.:75.00   3rd Qu.:200.0   3rd Qu.:0.00000
##  Max.   :81.00   Max.   :200.0   Max.   :1.00000
```

```
paste("Sample Size: ", nrow(challenger))
```

```
## [1] "Sample Size:  138"
```

From the printout of the updated Challenger data set, we see:

- It is a larger data set with 138 observations and just 3 columns.
- Column `Temp` is still a numeric column, which is typically above 60.
- Column `Pressure` is also still a numerical column yet it only takes on 3 values, thus we could also consider modeling with it being a categorical feature.
- Column `O.ring` is now a binary column representing whether or not that individual O-ring was a failure or a success. Zero represents a success and one represents a failure. This data transformation ensures that the column is considered as a Bernoulli outcome and can serve as the label of the exercise.

Again, we can add additional columns for visualization purposes.

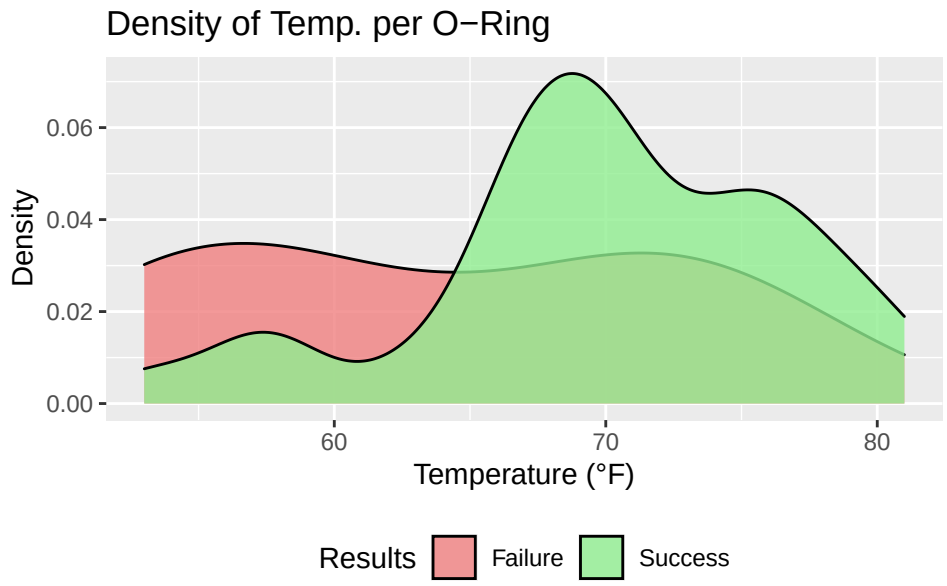
```
challenger$success_asint <- ifelse(challenger$O.ring > 0, 0, 1)
challenger$success <- as.factor(challenger$success_asint)
challenger$fail <- as.integer(ifelse(challenger$O.ring > 0, 1, 0))
challenger$pressure_cat <- as.factor(challenger$Pressure)
```

```
challenger = apply_labels(challenger,Temp = "Temperature (°F)")
temp <- as.data.frame(cro(challenger$pressure_cat, challenger$O.ring))
colnames(temp) <- c("", "Success", "Failure")
rownames(temp) <- c("50 psi", "100 psi", "200 psi", "Total")
temp$Total = coalesce(temp$Success, 0) + coalesce(temp$Failure, 0)
temp[2:4]
```

```
##         Success Failure Total
## 50 psi       35       1    36
## 100 psi      12      NA    12
## 200 psi      82       8    90
## Total       129       9   138
```

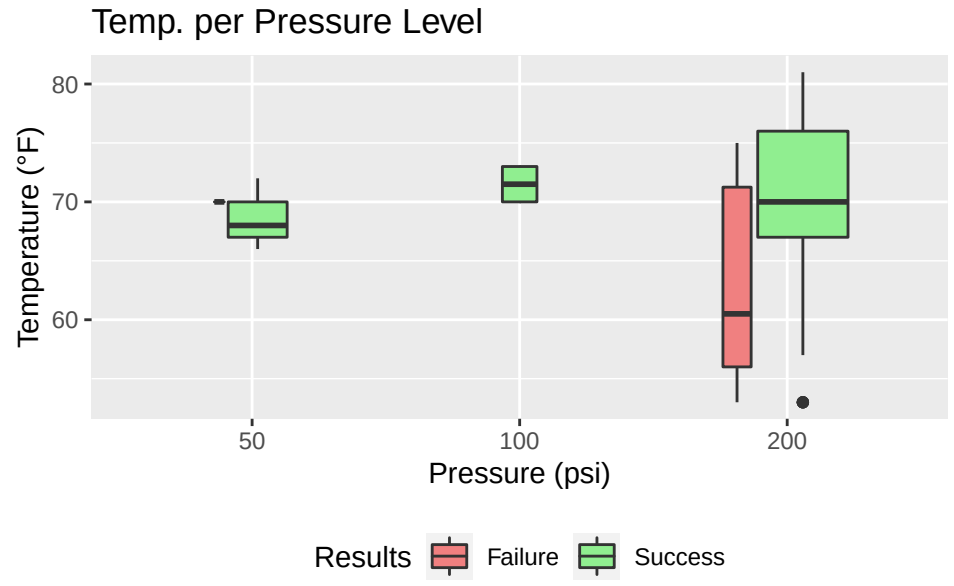
This table shows us the number of total successes and failures per O-ring at each pressure level.

```
ggplot(challenger, aes(Temp)) + theme(legend.position="bottom") +
  geom_density(aes(fill=factor(success)), alpha=0.8) +
  scale_fill_manual("Results", values = c("lightcoral", "lightgreen"),
                    labels=c("Failure", "Success")) +
  labs(title="Density of Temp. per O-Ring",x="Temperature (°F)",y="Density")
```



The density plot reveals the new distribution of temperatures with regards to the total number of O-ring failures and successes. We can observe that most successful O-rings occur at higher temperatures, where as the failed O-ring instances generally have lower temperatures. Yet, the distribution of successful O-rings is bi-modal, with a major peak around the mean temperature and a smaller peak at the lower end of the temperature distribution.

```
ggplot(challenger, aes(pressure_cat, Temp)) + theme(legend.position="bottom") +
    geom_boxplot(varwidth=T, aes(fill = factor(success))) +
    scale_fill_manual("Results", values = c("lightcoral", "lightgreen"),
                      labels=c("Failure", "Success")) +
    labs(title="Temp. per Pressure Level",x="Pressure (psi)",y="Temperature (°F)")
```



From the box plot, we can once again see that the majority of failed O-ring's occurred at lower temperatures than that of the successes. Successful O-ring's also have a greater density of points at higher temperatures. This strengthens the indication that temperature has an impact on the O-ring results.

Part 2

(a) Assumptions

The assumption that each O-ring is independent is essential for the author to structure the problem with logistic regression. Even though the logistic regression does not inherit every assumption in the linear regression setup, it does require each observation to be independent of the other. As a result, logistic regression assumes that each response has a binomial distribution, and independence of trials is required for the binomial. However, because three O-rings are on each rocket (6 total), there could be different dependencies in their success or failure, such as the method of installation or failure in one possibly leading to failure in another. Yet, we will assume independence in our model.

(b) Estimate Logistic Regression

To estimate the logistic regression model, we will use the augmented data set. We can the run the regression $logit(\pi) = \beta_0 + \beta_1 Temp + \beta_2 Pressure$ as follows, and recover the same set of coefficients as shown in the paper.

```
mod.fit.orig <- glm(formula = O.ring ~ Temp + Pressure,
                    family = binomial(link = logit),data = challenger)
```

```
summary(mod.fit.orig)
```

```
##
## Call:
## glm(formula = O.ring ~ Temp + Pressure, family = binomial(link = logit),
##     data = challenger)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -0.7940  -0.3670  -0.2500  -0.2162   2.8127
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  2.520195   3.486822   0.723   0.4698
## Temp        -0.098297   0.044890  -2.190   0.0285 *
## Pressure     0.008484   0.007677   1.105   0.2691
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 66.540  on 137  degrees of freedom
## Residual deviance: 58.856  on 135  degrees of freedom
## AIC: 64.856
##
## Number of Fisher Scoring iterations: 6
```

(c) LRT

Likelihood ratio tests (LRTs) can be performed on the model in two ways.

Firstly, we used the `Anova` function from `car` package, which performs a Type II Error test, which is the non-rejection of a false null hypothesis. Both the `Temp` row and the `Pressure` row are based on the model $logit(\pi) = \beta_0 + \beta_1 Temp + \beta_2 Pressure$, while the `Temp` row gives a test of $H_0 : \beta_1 = 0$ vs. $H_1 : \beta_1 \neq 0$ and the `Pressure` row gives a test of $H_0 : \beta_2 = 0$ vs. $H_1 : \beta_2 \neq 0$.

```
Anova(mod = mod.fit.orig, test = "LR")
```

```
## Analysis of Deviance Table (Type II tests)
##
## Response: O.ring
##          LR Chisq Df Pr(>Chisq)
## Temp       5.1838  1     0.0228 *
## Pressure   1.5407  1     0.2145
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Secondly, we use the `anova()` function from the `stat` package, which performs a Type I Error test, which is the incorrect rejection of a true null hypothesis. The `Temp` row gives a test of $H_0 : logit(\pi) = \beta_0$ vs. $H_a : logit(\pi) = \beta_0 + \beta_1 Temp$, while the `Pressure` row tests $H_0 : logit(\pi) = \beta_0 + \beta_1 Temp$ vs. $H_a : logit(\pi) = \beta_0 + \beta_1 Temp + \beta_2 Pressure$. We see that `Temp` is significant, while `Pressure` is not with `Temp` already considered in the model.

```
anova(object = mod.fit.orig, test = 'Chisq')
```

```
## Analysis of Deviance Table
##
## Model: binomial, link: logit
##
## Response: O.ring
##
## Terms added sequentially (first to last)
##
##
##          Df Deviance Resid. Df Resid. Dev Pr(>Chi)
## NULL                       137     66.540
## Temp      1   6.1440       136     60.396  0.01319 *
## Pressure  1   1.5407       135     58.856  0.21452
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Both tests point to similar results as `Temp` has significant p-values and `Pressure` does not. This indicates that `Temp` is important in explaining the failure of O-ring, while `Pressure` is not when `Temp` is already considered. Note that the `Pressure` row from both `Anova` and `anova` function yields the exact same information, because it is essentially the same test.

(d) Remove Pressure

`Pressure` does not seem to have much effect on the model. Because the p-value corresponding to `Pressure` is large, there is not sufficient evidence to indicate that the variable is important given that `Temp` is in the model. So, based on the LRT tests above, we agree with the author that `Pressure` could be removed from the model. A potential problem of removing such variable is that the in-sample fit of the model will be worse compared to when `Pressure` is included. Another potential problem is that the explanatory variable still may be important with respect to an interaction term or a transformation.

Part 3

(a) Estimate Model with `Temp` only

We will now run the simplified model $logit(\pi) = \beta_0 + \beta_1 Temp$, where π is the probability of an O-ring failure. This model also recovers the same set of coefficients as shown in the paper.

```
mod.fit.temp <- glm(formula = O.ring ~ Temp, family = binomial(link = logit),
                    data = challenger)
```

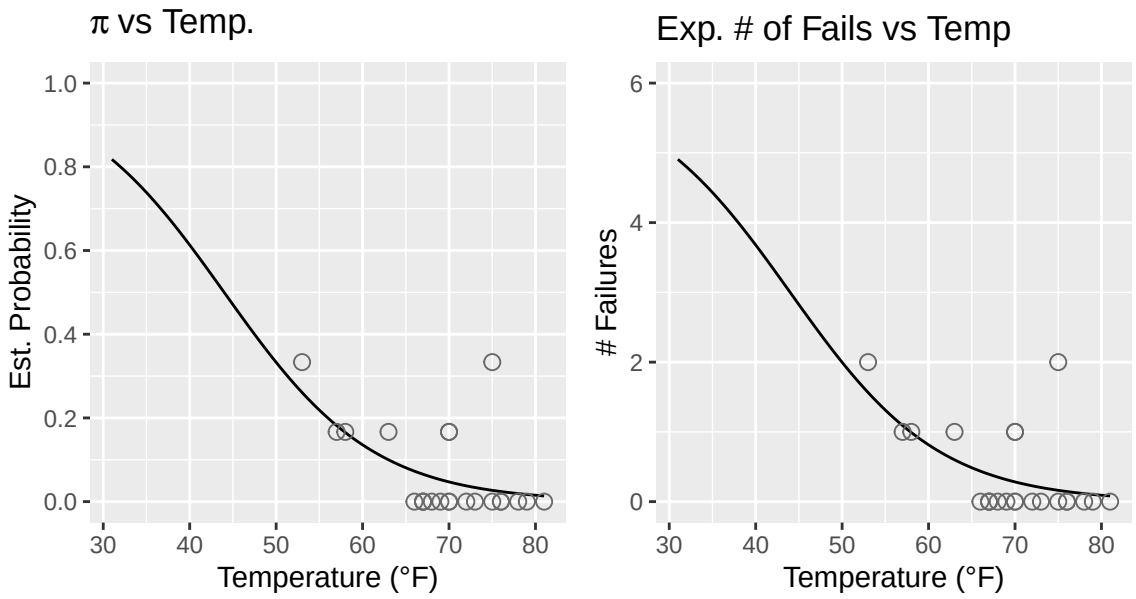
```
summary(mod.fit.temp)
```

```
##
## Call:
## glm(formula = O.ring ~ Temp, family = binomial(link = logit),
##     data = challenger)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -0.7774  -0.3677  -0.3106  -0.2209   2.6879
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  5.08498    3.05248   1.666   0.0957 .
## Temp        -0.11560    0.04702  -2.458   0.0140 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 66.540  on 137  degrees of freedom
## Residual deviance: 60.396  on 136  degrees of freedom
## AIC: 64.396
##
## Number of Fisher Scoring iterations: 6
```

(b) Plot Prediction vs Temp

Now, we can visualize the probability distribution for failure with respect to just temperature.

```
beta_1 = mod.fit.temp$coefficients[1]
beta_2 = mod.fit.temp$coefficients[2]
p1 <- ggplot(challenger_orig, aes(Temp, O.ring/6)) +
  stat_function(fun=function(x) exp(beta_1+beta_2*x)/(1+exp(beta_1+beta_2*x))) +
  geom_point(colour = "dimgrey", size = 2.5, shape = 1) + xlim(31,81) +
  scale_y_continuous(breaks=c(0,0.2, 0.4, 0.6, 0.8,1), limits = c(0,1)) +
  labs(title=expression(pi~"vs Temp."),y="Est. Probability",x="Temperature (°F)")
p2 <- ggplot(challenger_orig, aes(Temp, O.ring)) +
  stat_function(fun=function(x)6*exp(beta_1+beta_2*x)/(1+exp(beta_1+beta_2*x)))+
  geom_point(colour="dimgrey",size = 2.5,shape = 1) + xlim(31,81) + ylim(0,6) +
  labs(title="Exp. # of Fails vs Temp",y="# Failures", x="Temperature (°F)")
egg::ggarrange(p1, p2, nrow = 1)
```

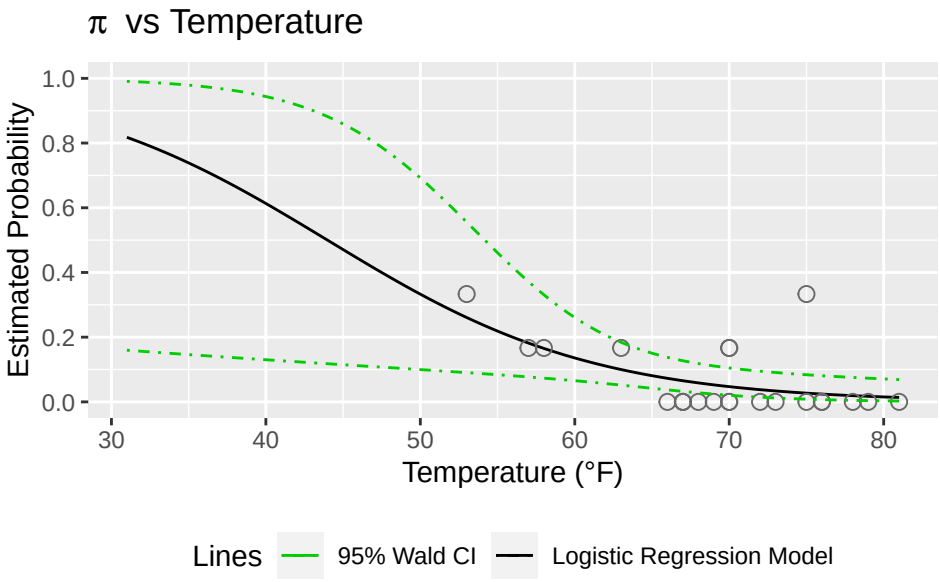


(c) Plot 95% Wald CI

Next, we can include the 95% Wald confidence interval bands for π on the probability distribution plot.

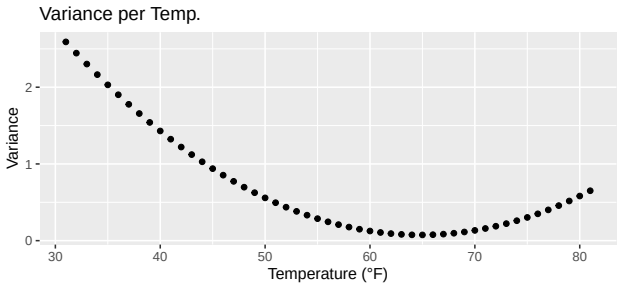
```
ci.pi <- function(newdata, mod.fit.obj, alpha){
  linear.pred<-predict(object=mod.fit.obj,newdata=newdata,type="link",se=TRUE)
  CI.lin.pred.lower <- linear.pred$fit - qnorm(p = 1-alpha/2)*linear.pred$se
  CI.lin.pred.upper <- linear.pred$fit + qnorm(p = 1-alpha/2)*linear.pred$se
  CI.pi.lower <- exp(CI.lin.pred.lower)/(1 + exp(CI.lin.pred.lower))
  CI.pi.upper <- exp(CI.lin.pred.upper)/(1 + exp(CI.lin.pred.upper))
  list(lower = CI.pi.lower, upper = CI.pi.upper)}
```

```
ggplot(challenger_orig, aes(Temp, O.ring/6)) + xlim(31,81) +
  stat_function(aes(colour = "Logistic Regression Model"), show.legend = TRUE,
    fun=function(x) exp(beta_1+beta_2*x)/(1+exp(beta_1+beta_2*x))) +
  geom_point(colour = "dimgrey", size = 2.5, shape = 1) +
  stat_function(aes(colour = "95% Wald CI"), show.legend = TRUE,lty = "dotdash",
    fun = function(x) ci.pi(newdata = data.frame(Temp = x),
                      mod.fit.obj = mod.fit.temp, alpha = 0.05)$lower) +
  stat_function(aes(colour = "95% Wald CI"), show.legend = FALSE,lty ="dotdash",
    fun = function(x) ci.pi(newdata = data.frame(Temp = x),
                      mod.fit.obj = mod.fit.temp, alpha = 0.05)$upper) +
  scale_y_continuous(breaks=c(0,0.2, 0.4, 0.6, 0.8,1), limits = c(0,1)) +
  theme(legend.position="bottom") +
  labs(title=expression(pi~" vs Temperature"),
       y="Estimated Probability", x="Temperature (°F)") +
  scale_colour_manual("Lines", values = c("green3", "black", "green"))
```



We can also consider the effect of variance in the model, between β_0 and β_1.

```
vartemp <- function(x) {0.0022*x*x - 2*0.1426*x + 9.3177}
dfvar<-data.frame(tempseq=seq(31,81,1),tempvar=sapply(seq(31,81,1), vartemp))
ggplot(dfvar, aes(x=tempseq, y=tempvar)) + geom_point() +
  labs(title="Variance per Temp.", x="Temperature (°F)", y="Variance")
grid.newpage()
grid.table(vcov(mod.fit.temp), theme = ttheme_minimal())
```



(a) Variance as function of Temperature

	(Intercept)	Temp
(Intercept)	9.31766397704148	−0.142565478730408
Temp	−0.142565478730408	0.00221124084616461

(b) Variance-Covariance Matrix

Figure 1: Variance Results

Overall, the 95% Wald confidence interval for $\beta_0+\beta_1 x$ is $\hat{\beta}_0+\hat{\beta}_1 x \pm Z_{1-\alpha/2}\sqrt{\hat{Var}(\hat{\beta}_0+\hat{\beta}_1 x)}$. And $\hat{Var}(\hat{\beta}_0+\hat{\beta}_1 x) = 0.0022x^2 - 2*0.1426x + 9.3177$, where x stands for temperature, is a decreasing function of x until x is around 65°F. Thus, the as temperature goes lower toward 30°F, the wider the confidence interval is. This is most likely because there are fewer observations, or none at all, for the lower temperatures. Therefore, there is more uncertainty about the estimates which is reflected by having wider intervals.

(d) 95% Wald CI at 31°F

When temperature is at 31°F, the estimated probability of an O-ring failure is $\pi = \frac{\exp(\hat{\beta}_0+\hat{\beta}_1 x)}{1+\exp(\hat{\beta}_0+\hat{\beta}_1 x)}$, with a corresponding confidence interval, which is calculated as follows:

```
newdata <- with(challenger, data.frame(Temp = c(31)))
print(paste0("Est. Probability: ",predict(mod.fit.temp,newdata,type="response")))
```

```
## [1] "Est. Probability: 0.817774406282599"
```

```
print(paste0("Lower CI: ",ci.pi(newdata,mod.fit.obj=mod.fit.temp,alpha=0.05)$lower))
```

```
## [1] "Lower CI: 0.159601356232889"
```

```
print(paste0("Upper CI: ",ci.pi(newdata,mod.fit.obj=mod.fit.temp,alpha=0.05)$upper))
```

```
## [1] "Upper CI: 0.990658280397187"
```

As a result, $\hat{\pi} = 0.8178$ and the 95% confidence interval is $0.1596 < \hat{\pi} < 0.9906$. This is because we do not observe responses below 53°F, so we need to assume that the same trend from 53°F to 81°F, as characterized by a logistic regression model, also occurs at this lower temperature.

The assumptions needed in order to apply the inference procedures are:

- normally distributed residuals
- N identical and independent trials with two possible outcomes for each trial
- the probability of success remains constant
- the number of successes is the variable of interest

Each of these assumptions have been satisfied for our model.

(e) Bootstrap CI

The bootstrap is done in the following steps:

1. Sample 23 temperature values out of the temperature data points with replacement
2. Predict 23 probabilities from the sampled temperature
3. Sample 23 of binomial outcomes with n = 6, if all of the 23 binomial outcomes are 0, go back to step 3
4. Expand the binomial outcomes as labels and repeat each sampled temperature 6 times as feature, then model it with logistic regression
5. Get the prediction from the newly fitted model as outcome for this step of the bootstrap
6. Repeat for a sufficient number of times (10K in this context) and get the 5th and 95th quantiles from the bootstrapped predictions

```
bootstrap = function(original_model=mod.fit.temp,n=23,t=10000,temp=31){
  pi_hat = predict(original_model,newdata=data.frame(Temp=temp),type='response',se=F)
  pi_bootstrap = rep(NA, t)
  for (ti in seq(t)){
    O.ring = 0
    while(max(O.ring) == 0){
      temp_sim = sample(challenger_orig$Temp, size = 23, replace = T)
      pi_hat_sim=predict(original_model,newdata=data.frame(Temp=temp_sim),
                         type='response',se=F)
      oring_sim = sapply(1:23,function(xi){rbinom(n =1,size=6,prob=pi_hat_sim[xi])})
      O.ring=unlist(sapply(oring_sim,function(xi){c(rep(1,xi),rep(0,6-xi))},simplify=F))
    }
    model_sim = glm(formula = O.ring ~ Temp, family = binomial(link = 'logit'),
                    data=data.frame(O.ring=O.ring,Temp=rep(temp_sim,each=6)))
    pi_bootstrap[ti] = predict(model_sim, newdata = data.frame(Temp = temp),
                               type = 'response', se = F)
  }
```

```
 return(data.frame(`lower`=quantile(pi_bootstrap,.05),`upper`=quantile(pi_bootstrap,.95)))
}
```

The 90% confidence interval when temperature is 31°F could be calculated as:

```
bs31 <- bootstrap(original_model = mod.fit.temp, n = 23, t = 10000, temp = 31)
print(bs31)
```

```
##        lower     upper
## 5% 0.1255383 0.9929607
```

The 90% confidence interval when temperature is 72°F could be calculated as:

```
bs72 <- bootstrap(original_model = mod.fit.temp, n = 23, t = 10000, temp = 72)
print(bs72)
```

```
##         lower      upper
## 5% 0.01084895 0.06975593
```

(f) Quadratic Term

In order to determine if a quadratic term is needed, we fit a model with a quadratic term, named `mod.fit.quad` and we used the `Anova` and `anova` function for the test. We also plotted the distributions to visually compare the results.

```
mod.fit.quad <- glm(formula = O.ring ~ Temp + I(Temp^2),
                    family = binomial(link = logit), data=challenger)
```

```
summary(mod.fit.quad)
```

```
##
## Call:
## glm(formula = O.ring ~ Temp + I(Temp^2), family = binomial(link = logit),
##     data = challenger)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -0.8832  -0.3279  -0.2901  -0.2530   2.6202
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) 22.126148  23.794552   0.930    0.352
## Temp        -0.650885   0.740761  -0.879    0.380
## I(Temp^2)    0.004141   0.005692   0.727    0.467
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 66.540  on 137  degrees of freedom
## Residual deviance: 59.902  on 135  degrees of freedom
## AIC: 65.902
##
## Number of Fisher Scoring iterations: 6
```

```
Anova(mod.fit.quad, test = "LR")
```

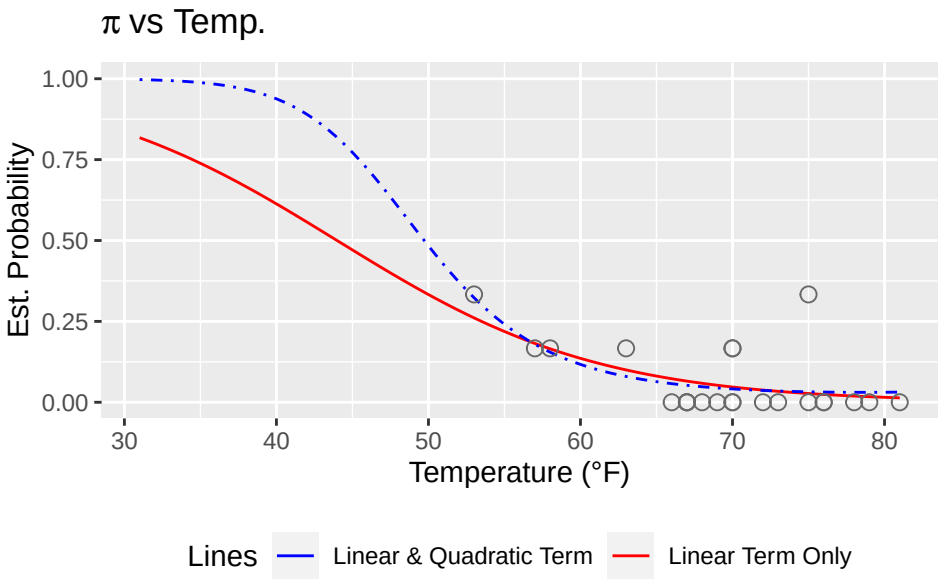
```
## Analysis of Deviance Table (Type II tests)
##
## Response: O.ring
```

```
##           LR Chisq Df Pr(>Chisq)
## Temp       0.71878  1     0.3965
## I(Temp^2)  0.49470  1     0.4818
```

```
anova(mod.fit.temp, mod.fit.quad, test = "Chisq")
```

```
## Analysis of Deviance Table
##
## Model 1: O.ring ~ Temp
## Model 2: O.ring ~ Temp + I(Temp^2)
##   Resid. Df Resid. Dev Df Deviance Pr(>Chi)
## 1       136     60.396
## 2       135     59.902  1   0.4947   0.4818
```

```
beta_q_1 = mod.fit.quad$coefficients[1]
beta_q_2 = mod.fit.quad$coefficients[2]
beta_q_3 = mod.fit.quad$coefficients[3]
ggplot(challenger_orig, aes(Temp, O.ring/6)) + xlim(31,81) + ylim(0,1) +
  stat_function(aes(colour = "Linear Term Only"), show.legend = TRUE,
                fun=function(x) exp(beta_1+beta_2*x)/(1+exp(beta_1+beta_2*x))) +
  stat_function(aes(colour = "Linear & Quadratic Term"),show.legend=FALSE,lty=4,
                fun=function(x) exp(beta_q_1+ beta_q_2*x+beta_q_3*x^2)/
                  (1+exp(beta_q_1+ beta_q_2*x+beta_q_3*x^2))) +
  geom_point(colour = "dimgrey", size = 2.5, shape = 1) +
  labs(title=expression(pi~"vs Temp."),y="Est. Probability",x="Temperature (°F)")+
  theme(legend.position="bottom") +
  scale_colour_manual("Lines", values = c("blue", "red"))
```



Using a LRT for the parameter of the quadratic term (given the linear term is in the model), we obtain -2log(Λ) = 0.4947 with a p-value of 0.4818. Because the p-value is large, there is not sufficient evidence of a quadratic relationship. Furthermore, the test statistic suggests that there is no significant evidence that the quadratic terms have an effect in explaining the O-ring failures. As a result, we do not recommend adding a quadratic term.

Part 4 - Discussion of Linear Model

With Temp as the only feature in the final model, the linear model will be:

```
mod.fit.lin <- lm(formula = O.ring ~ Temp, data = challenger)
```

```
summary(mod.fit.lin)
```

```
##
## Call:
## lm(formula = O.ring ~ Temp, data = challenger)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.19647 -0.08554 -0.06177 -0.01423  0.97784
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.616402   0.209118   2.948  0.00377 **
## Temp        -0.007923   0.002991  -2.649  0.00904 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2425 on 136 degrees of freedom
## Multiple R-squared:  0.04905,    Adjusted R-squared:  0.04206
## F-statistic: 7.016 on 1 and 136 DF,  p-value: 0.009036
```

The linear regression fitted a downward sloping line between O-ring failure and Temp. The intercept is 0.616402, meaning that at temperature of 0°F, the probability of an O-ring failure is 61.64%. And the slope of -0.007923 shows that with every 1 degree of temperature increase, the probability of failure goes down by .79%. Both coefficients are statistically significant at 0.01 level.

Furthermore, we can consider the practical significance of our linear regression results.

```
paste("Cohen's:",summary(mod.fit.lin)$r.square/(1-summary(mod.fit.lin)$r.square))
```

```
## [1] "Cohen's: 0.0515852277457017"
```

While the model does have statistically significant coefficients, it has rather small practical significance with a Cohen's effect size of 0.05.

To statistically test the assumption of normality, we can use the Shapiro-Wilk test. In this context, H_o is that the errors are normal.

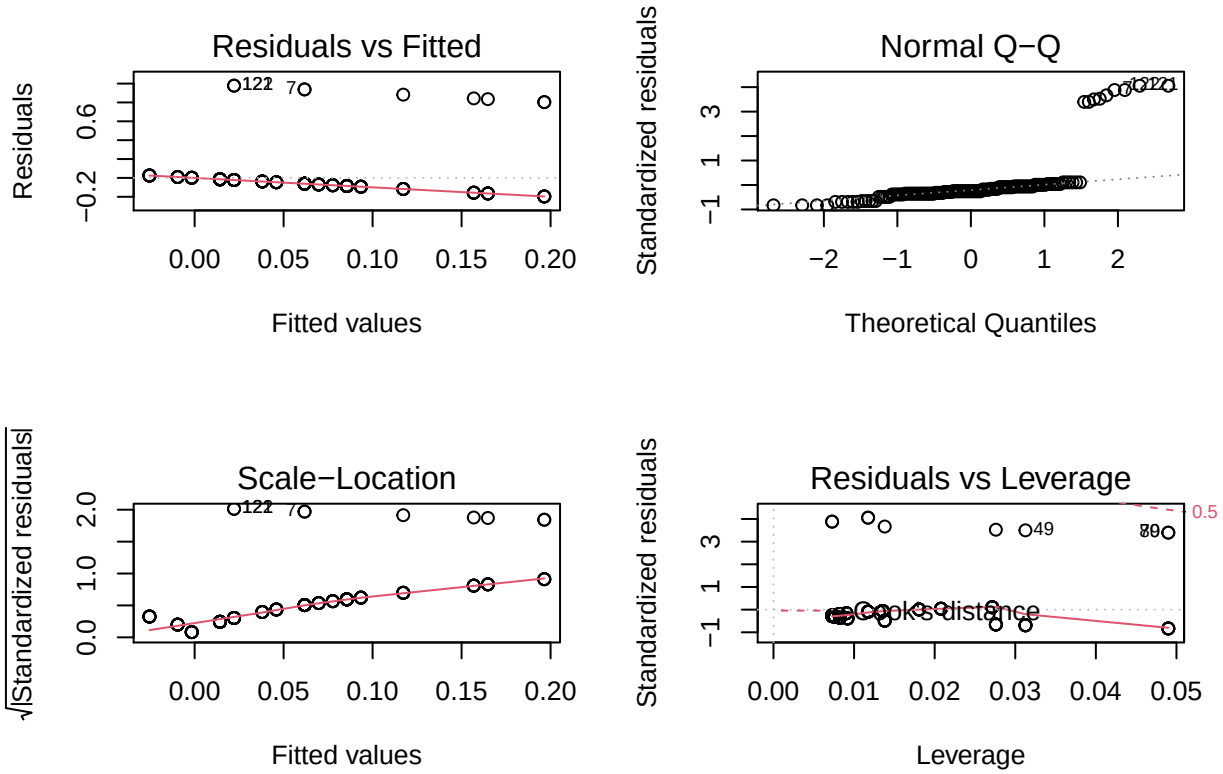
```
shapiro.test(mod.fit.lin$residuals)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  mod.fit.lin$residuals
## W = 0.46777, p-value < 2.2e-16
```

The p-value is less than alpha = 0.05 which means that the result is statistically significant and we can reject the null hypothesis that the errors are normal.

Now we will assess the six classical linear regression model assumptions:

```
par(mfrow = c(2, 2))
plot(mod.fit.lin)
```



1. Linear in Parameters: This assumption is met. The model is intended to express a linear relationship between `Temp` and `O.ring` and the error term is unconstrained.
2. Random Sampling: This assumption does apply. Our model assumes that the data set represents O-ring outcomes that are independent of each other and come from the same population of launches that are identically distributed.
3. No Perfect Collinearity: This assumption is satisfied. With only one independent variable, this model by definition cannot have multiple variables that are collinear with each other.
4. Zero Conditional Mean: This assumption is not met. In the residuals vs. fitted plot, we can see that the residuals are unevenly distributed around the zero line and the red mean curve is on a decreasing trend away from zero, suggesting a conditional mean that does not stay at zero.
5. Homoskedasticity: This assumption is not satisfied. The scale-location plot suggests that the residuals are on an increasing trend, which means that variance of the error term is not constant.
6. Normality of Error Term: This assumption fails. According to the normal Q-Q plot, many residuals deviate significantly from normality. The Shapiro-Wilk test has a similar conclusion with a highly significant p-value of 2.2e-16, thus we reject the null hypothesis of the normality of residuals.

Overall, given the diagnosis of the linear model, there are obvious violations of the necessary assumptions. Compared to the logistic model, it has its obvious short-comings. For example, temperatures larger than a certain number will provide a negative forecast on the probability of O-ring failures which is clearly non-sensible. However, it is not necessarily a completely useless model: a decision maker should know that both models point to high probability of failure at lower temperatures given the launch temperature of 31°F. However, one should also be aware that logistic regression is the more suitable model in this case where the independent variable is not bounded but the dependent variable is.

Part 5 - Summary

(a) Probability and Odds of Failure

According to our final model `mod.fit.temp` with only `Temp` as the feature in its linear format, per our calculation in Part 3 Question D, the predicted probability of an O-ring failure is $\hat{\pi} = 81.78\%$ with the 95% confidence interval of $[15.96\%, 99.06\%]$. Additionally, our bootstrapped 90% confidence interval from 10,000 draws in Part 3 Question D resulted in a confidence interval of $[12.55\%, 99.29\%]$.

Assuming that each O-ring is independent, the probability of a failed launch should be:

$$\begin{aligned} P(\text{failed launch}) &= P(\text{at least 1 of 6 O-rings fail}) \\ &= 1 - (1 - \hat{\pi})^6 \\ &= 1 - (1 - 0.8177744)^6 \\ &= 0.9999634 \end{aligned}$$

The corresponding 95% Wald confidence interval is $[64.77\%, 100\%]$, while the 90% bootstrapped confidence interval is $[55.29\%, 100\%]$.

The corresponding odds are summarized together in the following table:

```
pi_hat = 0.8177744
pi_hat_95l = 0.1596014
pi_hat_95u = 0.9906583
pi_hat_90l = bs31$lower
pi_hat_90u = bs31$upper

table = data.frame(' ' = c('Probability', 'Odds', 'Probability', 'Odds'),
                   ' ' = c(percent(pi_hat, accuracy = .0001),
                     number(pi_hat/(1-pi_hat), accuracy=0.0001),
                     percent(1-(1-pi_hat)^6, accuracy=.0001),
                     scientific((1 - (1 - pi_hat)^6)/(1 - pi_hat)^6,digits=4)),
                   'Lower' = c(percent(pi_hat_95l, accuracy = .0001),
                     number(pi_hat_95l/(1-pi_hat_95l), accuracy = 0.0001),
                     percent(1 - (1 - pi_hat_95l)^6, accuracy = .0001),
                     scientific((1-(1-pi_hat_95l)^6)/(1-pi_hat_95l)^6,digits=4)),
                   'Upper' = c(percent(pi_hat_95u, accuracy = .0001),
                     number(pi_hat_95u/(1-pi_hat_95u), accuracy = 0.0001),
                     percent(1 - (1 - pi_hat_90u)^6, accuracy = .0001),
                     scientific((1-(1-pi_hat_95u)^6)/(1-pi_hat_95u)^6,digits=4)),
                   'CI Lower' = c(percent(pi_hat_90l, accuracy = .0001),
                     number(pi_hat_90l/(1-pi_hat_90l), accuracy = 0.0001),
                     percent(1 - (1 - pi_hat_90l)^6, accuracy = .0001),
                     scientific((1-(1-pi_hat_90l)^6)/(1-pi_hat_90l)^6,digits=4)),
                   'Upper' = c(percent(pi_hat_90u, accuracy = .0001),
                     number(pi_hat_90u/(1-pi_hat_90u), accuracy = 0.0001),
                     percent(1 - (1 - pi_hat_90u)^6, accuracy = .0001),
                     scientific((1-(1-pi_hat_90u)^6)/(1-pi_hat_90u)^6,digits=4)))

table %>%
  kable(caption = "Summary Table", col.names = NULL, "latex")  %>%
  kable_styling(bootstrap_options = c("striped", "hover")) %>%
  pack_rows("O-ring Failure", 0,1) %>%
  pack_rows('Lauch Failure', 2, 3) %>%
  add_header_above(c(" ","Estimate","95 Percent Wald CI"=2,"90 Percent Bootstrap CI"=2))
```

Table 1: Summary Table

	Estimate	95 Percent Wald CI		90 Percent Bootstrap CI	
O-ring Failure					
Probability	81.7774%	15.9601%	99.0658%	12.5538%	99.2961%
Odds	4.4877	0.1899	106.0469	0.1436	141.0589
Lauch Failure					
Probability	99.9963%	64.7701%	100.0000%	55.2859%	100.0000%
Odds	2.731e+04	1.838e+00	1.505e+12	1.236e+00	8.219e+12

(b) Model Selection

We have experimented with multiple variations of feature combinations for our model:

- logistic model with `Temp` and `Pressure`
- logistic model with `Temp`
- logistic model with `Temp` and `Temp` squared
- linear model with `Temp`

The main take-aways are:

- `Temp` is a significant factor affecting the O-ring failures, which is very robust for the model format and the inclusion of additional features
- Neither `Pressure` nor `Temp` squared turned out to be significant under the logistic setting
- A linear model, given a reasonable range of temperature, is a sensible model. It is able to characterize the negative relationship between temperature and probability of O-ring failure, and produces a reasonable estimate at 31°F. However, it is a inferior option because our independent variables supposedly is unbounded, while the dependent variable is bounded. For example, under the linear setup, the model will yield an negative estimate at 100°F, or an unreasonably high estimate at 0°F.

(c) Decision Making

Our final model points to a high probability of failure at 31°F, yet there are a few more subtle points we could offer:

- The model assumed that each O-ring is independent in the way the data-set was augmented for the logistic regression. The assumption is challenged by the fact that all 6 O-rings are attached to the same rocket, possibly produced by the same manufacturer, and one failed O-ring might set off failures in the others during a launch. When the O-ring failures are correlated, the probability that at least one O-ring will fail can be higher than the previous calculation with assumed independence
- The point estimate is high, yet the CI is extremely wide. The Wald CI should be interpreted as a series of independent and repeated experiments, where intervals constructed like this will cover the true π 95% of the time. Yet, even without any model, one would know a trivial estimate of an interval 0% to 100% will surely cover the true π, which is only slightly wider than the Wald interval.
- With these limitations in mind, the decision maker will have to weigh the pros and cons of this model. We recognize the fact that it is extremely clear to us in retrospect what happened, and we certainly do not claim that we understand the cost and related down-sides of postponing a launch of this much importance. However, our research has found convincing evidence that it only takes one failed O-ring to result in a failed launch, which significantly reduces the CI for the probability of a failed launch. According to the table above, the 95% CI is from 65% to 100%. Unless there is significant cost associated with postponing, 31°F is not a fit temperature for a launch.