w271 Lab 1: Investigation of the 1989 Space Shuttle Challenger Accident

Christina Min, Shishir Agarwal , Edward Tong

31/01/2021

Background

An article "Risk Analysis of the Space Shuttle: Pre-Challenger Prediction of Failure by Dalal et al (1989)" was created to investigate the accident of the space shuttle Challenger explosion on January 28, 1986 right after the launch. The authors used 23 preaccidents launch data with conditions of each flight to perform statistical analysis and evidence in order to find out the possible root cause of failure so that such catastrophic accident can be avoided in the future launch.

Since the article already covered the detail thorough statistical analysis with findings and conclusion, our team's work is based on the article information with the pre-accidents data set to perform other statistical analysis with obtained results to address each part (total 5 Parts) of the questions in this report.

Part 1

We have performed a thorough EDA on the data set to have an overview so that we can examine any anomalies, missing values as well as univariate, bivariate and multivariate analysis before performing the modeling in subsequent sections.

```
library(knitr)
opts_chunk$set(tidy.opts=list(width.cutoff=60),tidy=TRUE)
setwd("~/MIDS/Courses/w271/w271_lab1") #christina
#  setwd("/home/jovyan/r_bridge/student_work/shagarwa/Lab#1") #shishir
# setwd("../w271_lab1/") # edward
```

Load the required libraries and the dataset for further exploration and analysis.

```
# Check and install missing packages
list.of.packages <- c("car", "dplyr", "Hmisc", "skimr", "ggplot2", "stargazer",
                      "mcprofile", "gridExtra", "binom","grid")
new.packages <- list.of.packages[!(list.of.packages %in%
                                     installed.packages()[,"Package"])]
if(length(new.packages)) install.packages(new.packages)

# Start with a clean R environment
rm(list = ls())

# Load Libraries
```

```
lapply(c("car", "dplyr", "Hmisc", "skimr", "ggplot2", "stargazer",
         "mcprofile","gridExtra", "binom", "grid"),
       require, character.only = TRUE)
```

We then run the basic EDA to find 23 observations and 5 features in the dataset. Given the context from the above paper, we observed the response variable of o.ring takes values of 0,1,2 failures from 6 trials. We also notice explanatory variable temperature data ranges from 53 to 81 and is evenly distributed since mean and median are roughly the same. We also notice the explanatory variable pressure takes values between 50, 100, 200. Lastly, we notice Flight is not providing any additional information and will not use it for our analysis.

```
# load data as "df"

df <- read.table(file = "challenger.csv", header = TRUE, sep = ",")
# df <- read.table(file = "./Lab#1/challenger.csv", header = TRUE, sep = ",")

# basic information and statistics of variables
str(df)
```

```
## 'data.frame':    23 obs. of  5 variables:
##  $ Flight  : int  1 2 3 4 5 6 7 8 9 10 ...
##  $ Temp    : int  66 70 69 68 67 72 73 70 57 63 ...
##  $ Pressure: int  50 50 50 50 50 50 100 100 200 200 ...
##  $ O.ring  : int  0 1 0 0 0 0 0 0 1 1 ...
##  $ Number  : int  6 6 6 6 6 6 6 6 6 6 ...
```

```
# check for basic distribution
summary(df)
```

```
##      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
```

```
# check for missing values
paste("there are", sum(is.na(df)), "missing values.")
```

```
## [1] "there are 0 missing values."
```

To aid further analysis, we created a new binary response variable, "Stress" by using O.ring for binary modeling and also transformed the Pressure from numerical to factor variable.

```
df$Pressure.factor <- df$Pressure
df$Pressure.factor <- as.factor(df$Pressure.factor)
# Stress as response variable for binay modeling
df <- mutate(df,
       Stress = ifelse(O.ring > 0, 1, 0),
      )
```

```r
# flight # is treated as a string (no rank)
df$Flight <- as.character(df$Flight)
# check the data again
skim(df)
```

Table 1: Data summary

Name	df
Number of rows	23
Number of columns	7
Column type frequency:	
character	1
factor	1
numeric	5
Group variables	None

Variable type: character

skim_variable	n_missing	complete_rate	min	max	empty	n_unique	whitespace
Flight	0	1	1	2	0	23	0

Variable type: factor

skim_variable	n_missing	complete_rate	ordered	n_unique	top_counts
Pressure.factor	0	1	FALSE	3	200: 15, 50: 6, 100: 2

Variable type: numeric

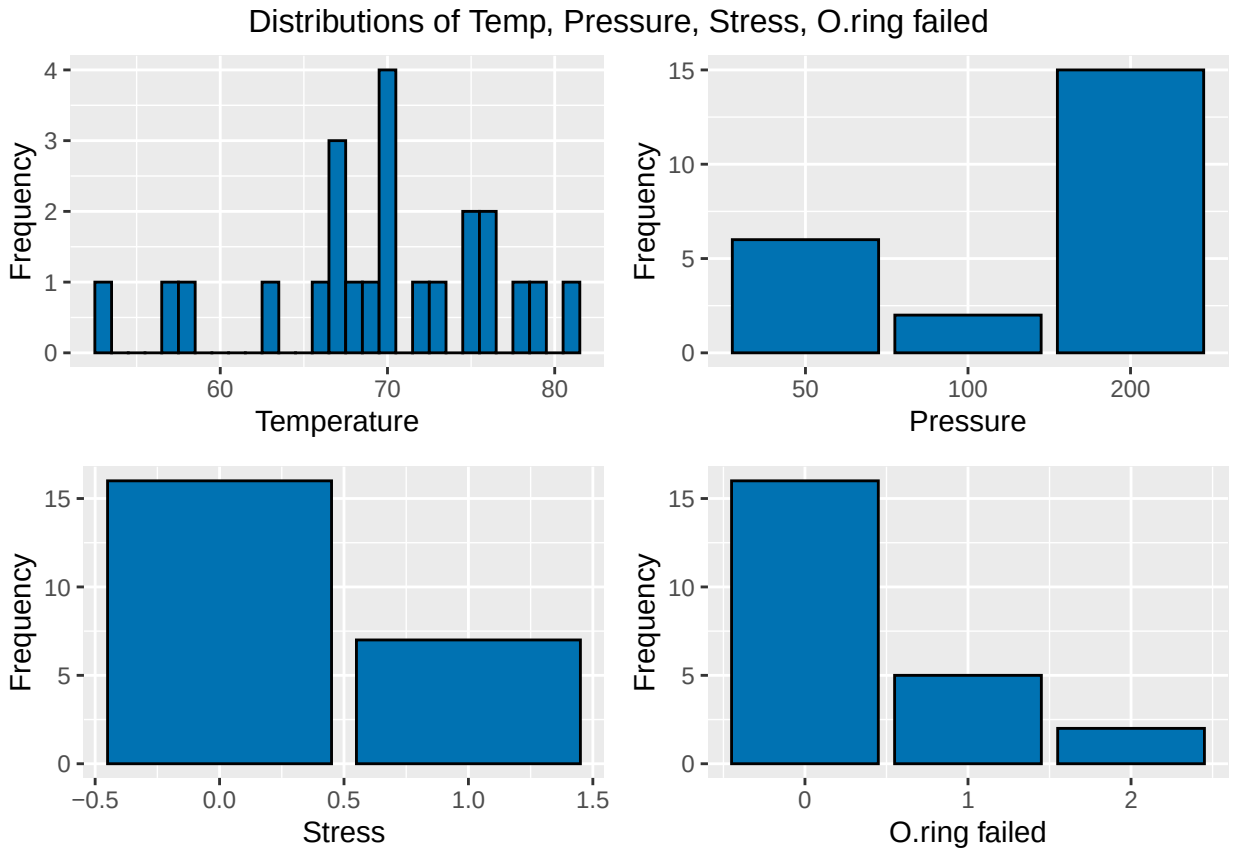
skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100	hist
Temp	0	1	69.57	7.06	53	67	70	75	81	
Pressure	0	1	152.17	68.22	50	75	200	200	200	
O.ring	0	1	0.39	0.66	0	0	0	1	2	
Number	0	1	6.00	0.00	6	6	6	6	6	
Stress	0	1	0.30	0.47	0	0	0	1	1	

Next we examined the distributions of the univariate variables of O.ring/Stress, Temperature and Pressure. We find Temperature has no skew and most data is associated with Pressure equal to 200 psi. In addition , we also observed the frequency of failure to success is 7/23 with one outliner case at Temp 72 with two rings failure.

```r
# Univariate
# Proportional Table for Response Variable Stress
prop.table(table(df$Stress))
```

```
##
##         0         1
## 0.6956522 0.3043478
```

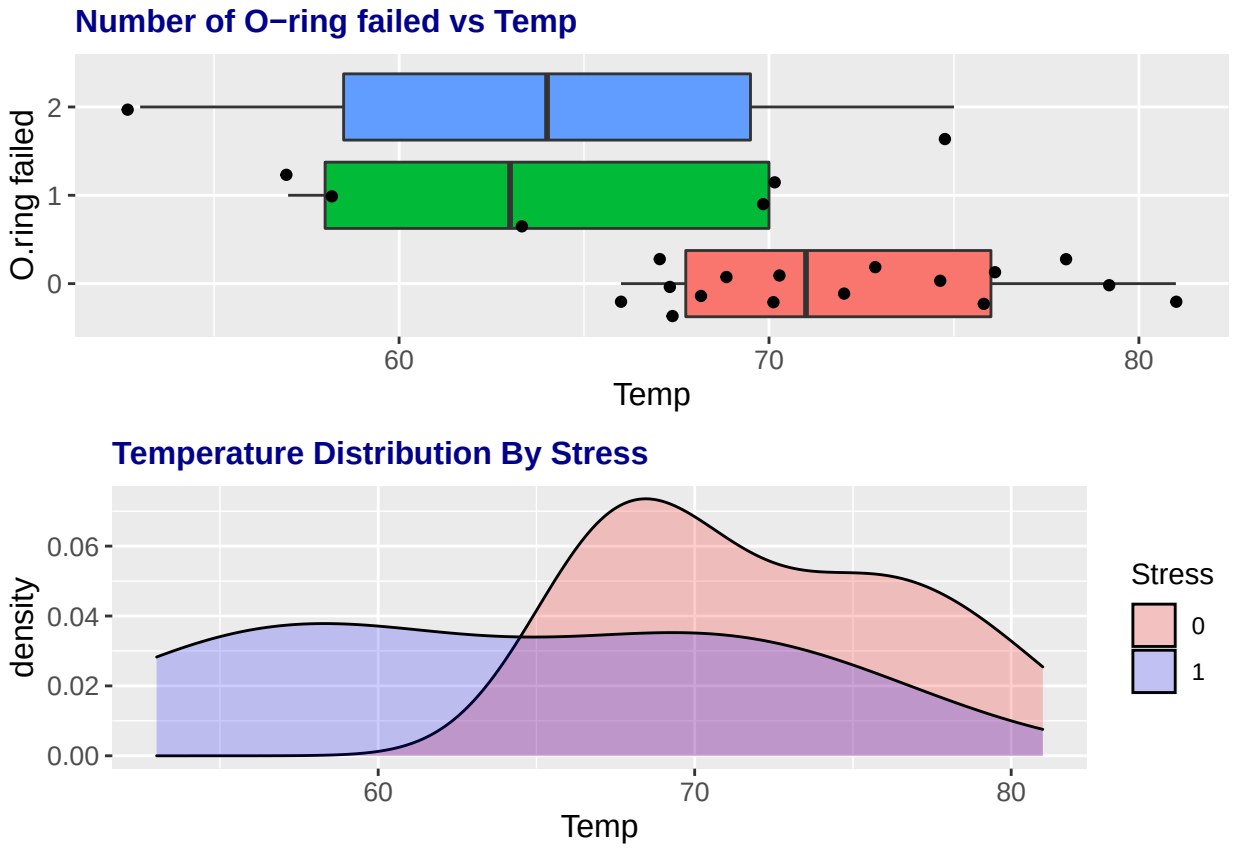
```r
# Temp plot
temp.plot <- ggplot(df, aes(x = Temp)) +
  geom_histogram(aes(x = Temp), binwidth = 1, fill="#0072B2", colour="black") +
  xlab("Temperature") +
  ylab("Frequency") + theme(plot.title = element_text(lineheight=1, face="bold",
                                                      color ="dark blue"))

# Pressure Plot
pres.plot <- ggplot(df, aes(x = Pressure.factor)) +
  geom_bar(aes(x = Pressure.factor), fill="#0072B2", colour="black") +
  xlab("Pressure") +
  ylab("Frequency") +
  theme(plot.title = element_text(lineheight=1, face="bold", color ="dark blue"))

# Stress Plot
stress.plot <- ggplot(df, aes(x = Stress)) +
  geom_bar(aes(x = Stress), fill="#0072B2", colour="black") +
  xlab("Stress") +
  ylab("Frequency") + theme(plot.title = element_text(lineheight=1, face="bold",
                                                      color ="dark blue"))

# O.ring failed Plot
oring.plot <- ggplot(df, aes(x = O.ring)) +
  geom_bar(aes(x = O.ring), fill="#0072B2", colour="black") +
  xlab("O.ring failed") +
  ylab("Frequency") + theme(plot.title = element_text(lineheight=1, face="bold",
                                                      color ="dark blue"))

grid.arrange(temp.plot, pres.plot, stress.plot, oring.plot,
             top="Distributions of Temp, Pressure, Stress, O.ring failed" ,
            ncol=2)
```



With bivariate distribution charts on O.ring/Stress against Temperature and Pressure; it shows at lower temperature, we have very few data points compared to at higher temperatures; the number of O.ring failed are inversely proportional to the temperature; the proportion of failure to success is higher at temperature below 60. Lastly, we also see outliner at 75 Temperature where we get 2 O.ring failure.

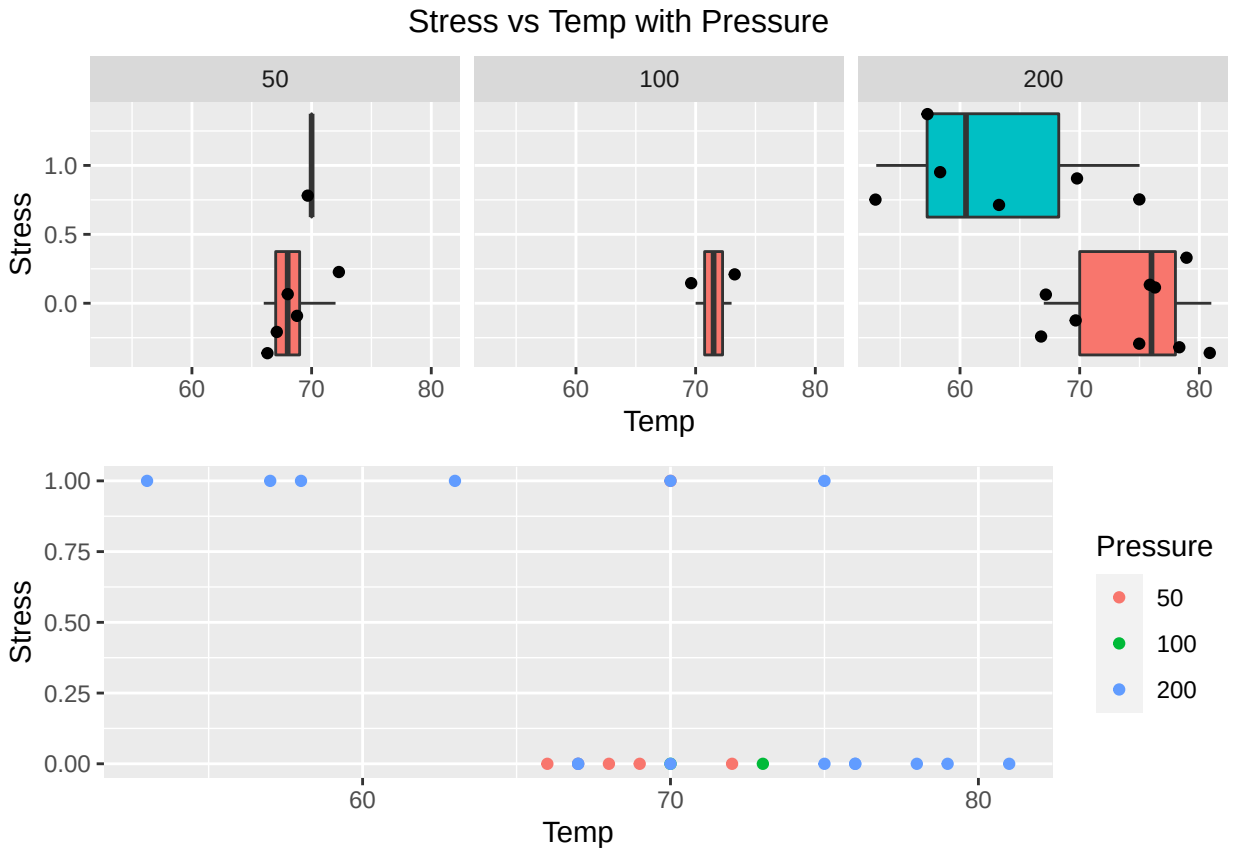
```
# temperature and O.ring Stress incidents
ring_temp.box <- ggplot(df, aes(Temp, factor(O.ring))) +
  geom_boxplot(aes(fill = factor(O.ring))) + geom_jitter() +
  guides(fill=FALSE) + ggtitle("Number of O-ring failed vs Temp") +
  ylab("O.ring failed") +
  theme(axis.text = element_text(size=10),
        axis.title = element_text(size=12),
        plot.title = element_text(lineheight=1, size =12, face="bold", color ="dark blue"))

# temperature distribution by Stress
temp_stress.dist <- ggplot(df, aes(Temp, fill = factor(Stress)))+
  scale_fill_manual(name = "Stress",values = c('red','blue'))+
  geom_density(alpha=0.2) +
  ggtitle("Temperature Distribution By Stress") +
  theme(legend.position = 'right', axis.text = element_text(size=10),
        axis.title = element_text(size=12),
        plot.title = element_text(lineheight=1, size =12, face="bold",
                                  color ="dark blue")) +
```

```
  scale_color_manual(name = "Stress", values = c('red','blue'))

grid.arrange(ring_temp.box, temp_stress.dist, nrow=2)
```



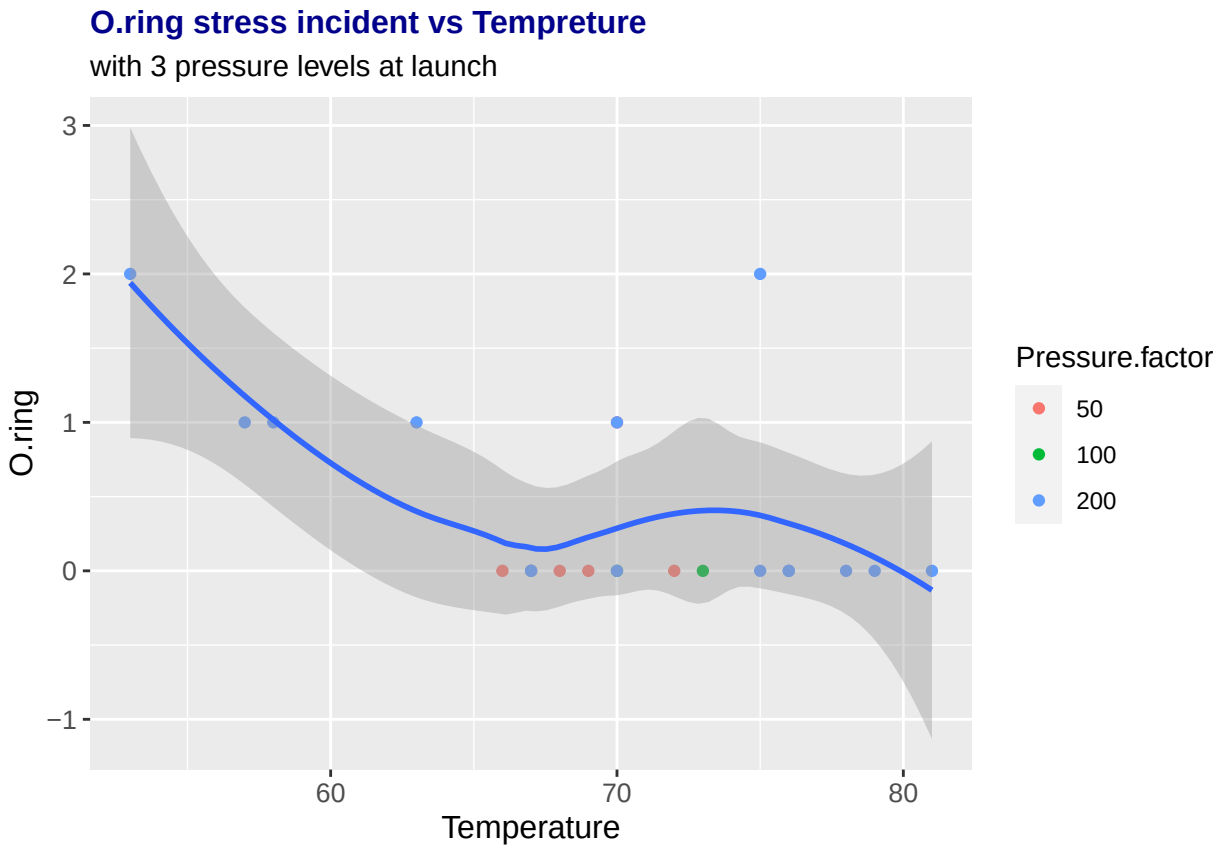
Following shows the multivariate distributions. We notice most of the data points are with 200 psi pressure and within the 200 psi we notice as many data points with failure as with success. So we do not think pressure is relevant here but we will check it later with regression analysis

```
ringstree_temp_pres.box <- ggplot(df, aes(Temp, Stress)) +
  geom_boxplot(aes(fill = factor(Stress))) +
  geom_jitter() + guides(fill=FALSE) +
  facet_wrap(~ Pressure, nrow = 1) +
  theme(plot.title = element_text(lineheight=1, face="bold", color ="dark blue"))

dotplot <- ggplot(data = df) +
  geom_point(mapping = aes(x = Temp, y = Stress, color = factor(Pressure))) +
  labs(color ="Pressure") +
 theme(plot.title = element_text(lineheight=1, face="bold", color ="dark blue"))
grid.arrange(ringstree_temp_pres.box, dotplot, top = "Stress vs Temp with Pressure",
             nrow=2)
```




```
# explore more interaction between O.ring stress, Temperature and Pressure levels
ringstress.temp.press.all <- ggplot(data = df, mapping = aes(x = Temp, y = O.ring))+
geom_point(mapping = aes(color = Pressure.factor)) + geom_smooth() +
  labs( x = 'Temperature', y = 'O.ring',
        title ="O.ring stress incident vs Tempreture",
        subtitle = "with 3 pressure levels at launch") +
  theme(axis.text = element_text(size=10),
         axis.title = element_text(size=12),
         plot.title = element_text(lineheight=1, size =12, face="bold",
                                   color = "dark blue"))
ringstress.temp.press.all
```

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```



In summary,the dataset has no missing values but with one outlier. We alerady observed that Temperature is negatively correlated with O.ring failures in our EDA and will further validate our assumption, i.e. low temperature is the culprit of this launch disaster in our statistical analysis in later part of our report.

Part 2 Answer the following from Question 4 of Bilder and Loughin Section 2.4 Exercises (page 129):

(a) The authors use logistic regression to estimate the probability an O-ring will fail. In order to use this model, the authors needed to assume that each O-ring is independent for each launch. Discuss why this assumption is necessary and the potential problems with it. Note that a subsequent analysis helped to alleviate the authors' concerns about independence.

For both binary and binomial analysis, we have to assume each of the launch is independent and identically distributed. Moreover, for binomial analysis, we have to assume each of the six O.rings is independent to one another and with the same probability of failure. Based on their binary and binomial analysis, there is no material difference between them which alleviate their concern on independence on each o.ring. Thus, we can relax this assumption when using either of two models.

(b) Estimate the logistic regression model using the explanatory variables in a linear form.

We have performed both binomial linear and binary logistic model(refer to Appendix for binary model) with Temp and Pressure explanatory variables as follows:

```
glm.mod.binom1 <- glm(O.ring/Number ~ Temp + Pressure, data = df,
                      family = binomial(link="logit"),weights=Number)
summary(glm.mod.binom1)
```

```
##
## Call:
## glm(formula = O.ring/Number ~ Temp + Pressure, family = binomial(link = "logit"),
##     data = df, weights = Number)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.0361  -0.6434  -0.5308  -0.1625   2.3418
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  2.520195   3.486784   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: 24.230  on 22  degrees of freedom
## Residual deviance: 16.546  on 20  degrees of freedom
## AIC: 36.106
##
## Number of Fisher Scoring iterations: 5
```

(c) Perform LRTs to judge the importance of the explanatory variables in the model.

LRT for Binomial Model

```
Anova(glm.mod.binom1, test = "LR")
```

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

Based on LRT, Temperature is a significant factor for predicting o.ring failures with p-value of 0.005 with Pressure in the model; while pressure is not significant with high p-value of 0.21 with temperature in the model.

(d) The authors chose to remove Pressure from the model based on the LRTs. Based on your results, discuss why you think this was done. Are there any potential problems with removing this variable?

Follow up on our result in part c on LRT with Chisq test, with 95% confidence, Pressure has insignificant impact on predicting O.rings failure holding temperature constant.

By removing the Pressure variable, the potential problem could affect the model Goodness of Fits, i.e. decrease the R^2 with one explanatory variable (Temp) instead of two (Temp + Pressure).

Binomial Model is:

$$logit(Proportion) = \beta_0 + \beta_1 Temp$$

Binary Model is:

$$logit(Stress) = \beta_0 + \beta_1 Temp$$

Part 3

Answer the following from Question 5 of Bilder and Loughin Section 2.4 Exercises (page 129-130):

Continuing Exercise 4, consider the simplified model $logit(\pi) = \beta_0 + \beta_1 Temp$, where π is the probability of an O-ring failure. Complete the following:

(a) Estimate the model. We analyzed both the binary and binomial model but due to page limit will focus our efforts on the binomial model and for binary model can refer to Appendix

```
glm.mod.binom2 <- glm(O.ring/Number ~ Temp, data = df,
                      family = binomial(link = "logit"), weights = Number)
summary(glm.mod.binom2)
```

```
##
## Call:
## glm(formula = O.ring/Number ~ Temp, family = binomial(link = "logit"),
##     data = df, weights = Number)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -0.95227  -0.78299  -0.54117  -0.04379   2.65152
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  5.08498    3.05247   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: 24.230  on 22  degrees of freedom
## Residual deviance: 18.086  on 21  degrees of freedom
## AIC: 35.647
##
## Number of Fisher Scoring iterations: 5
```

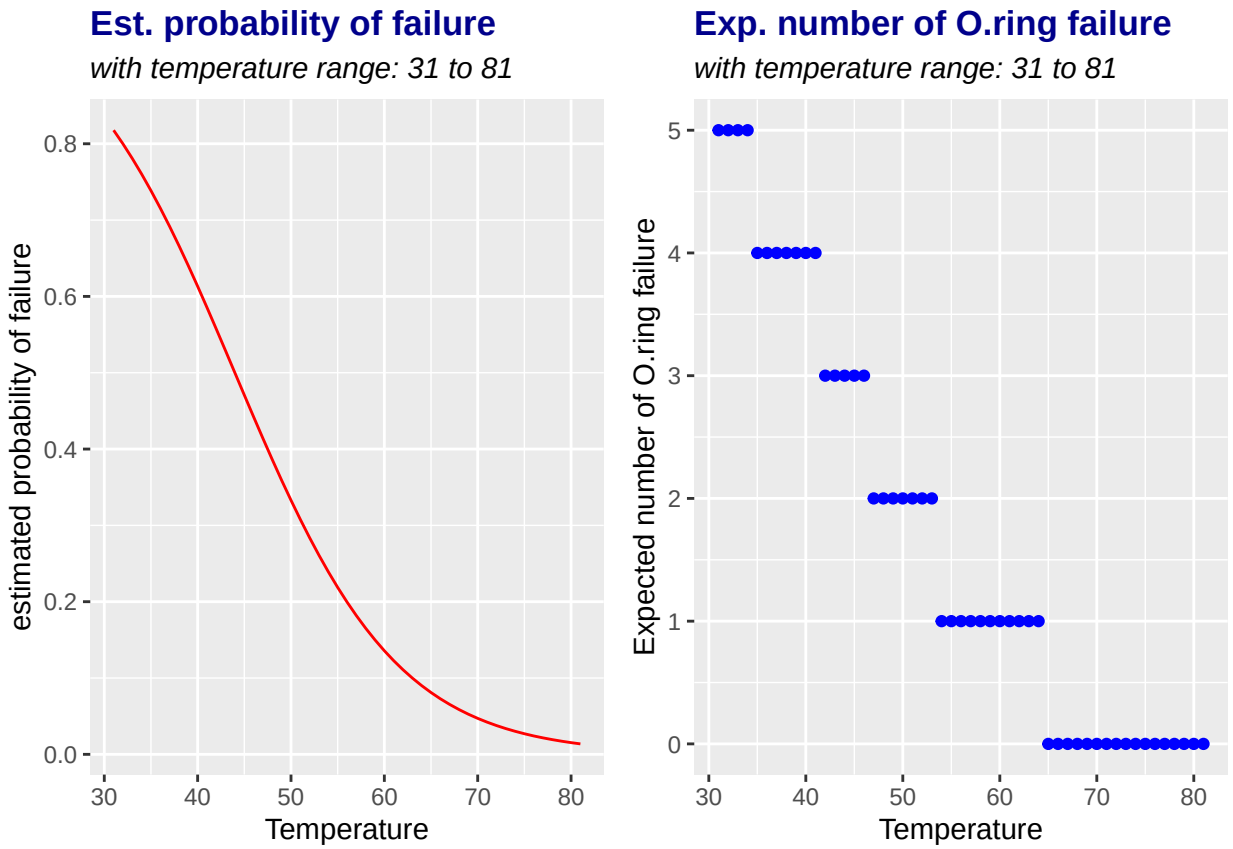
Thus our binomial model is:

$$logit\left(\pi_i\right) = 5.08498 - 0.11560 Temp$$

(b) Construct two plots: (1) π vs. Temp and (2) Expected number of failures vs. Temp. Use a temperature range of 31° to 81° on the x-axis even though the minimum temperature in the data set was 53°.

```
alpha = 0.05
# set up new data for Temperature
predict.data <- data.frame(Temp = seq(from = 31, to = 81, by = 1))
# predict linear predictor (log of odds ratio) at each Temperature point
linear.pred <- predict(object = glm.mod.binom2, newdata = predict.data,
                       type = "link", se.fit = TRUE)
# estimated probability of failure
pi.hat <- exp(linear.pred$fit) / (1 + exp(linear.pred$fit))

# compute confidence interval
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))

# dataframe for all
pi.df <- data.frame(pi.hat, lower = CI.pi.lower, upper = CI.pi.upper)
```

```
# pi vs temperature
prob.plot <- ggplot() +
  geom_line(mapping = aes(x = predict.data$Temp, y = pi.hat), color = 'red') +
  labs( x = 'Temperature', y = 'estimated probability of failure',
        title ="Est. probability of failure",
        subtitle = "with temperature range: 31 to 81") +
  theme(plot.title = element_text(lineheight=1, face="bold", color = "dark blue"))+
  theme(plot.subtitle=element_text(face="italic", color="black"))

# expected number of failures
n = df$Number[1]
O.ring.hat <- round(n * pi.hat)
# %>% round(.,digits = 0)
est.num.plot <- ggplot() +
  geom_point(mapping = aes(x = predict.data$Temp, y = O.ring.hat), color = 'blue') +
  labs(x = 'Temperature', y = 'Expected number of O.ring failure',
        title = "Exp. number of O.ring failure",
        subtitle = "with temperature range: 31 to 81") +
  theme(plot.title = element_text(lineheight=1, face="bold", color = "dark blue"))+
  theme(plot.subtitle=element_text(face="italic", color="black"))

grid.arrange(prob.plot, est.num.plot, ncol=2)
```

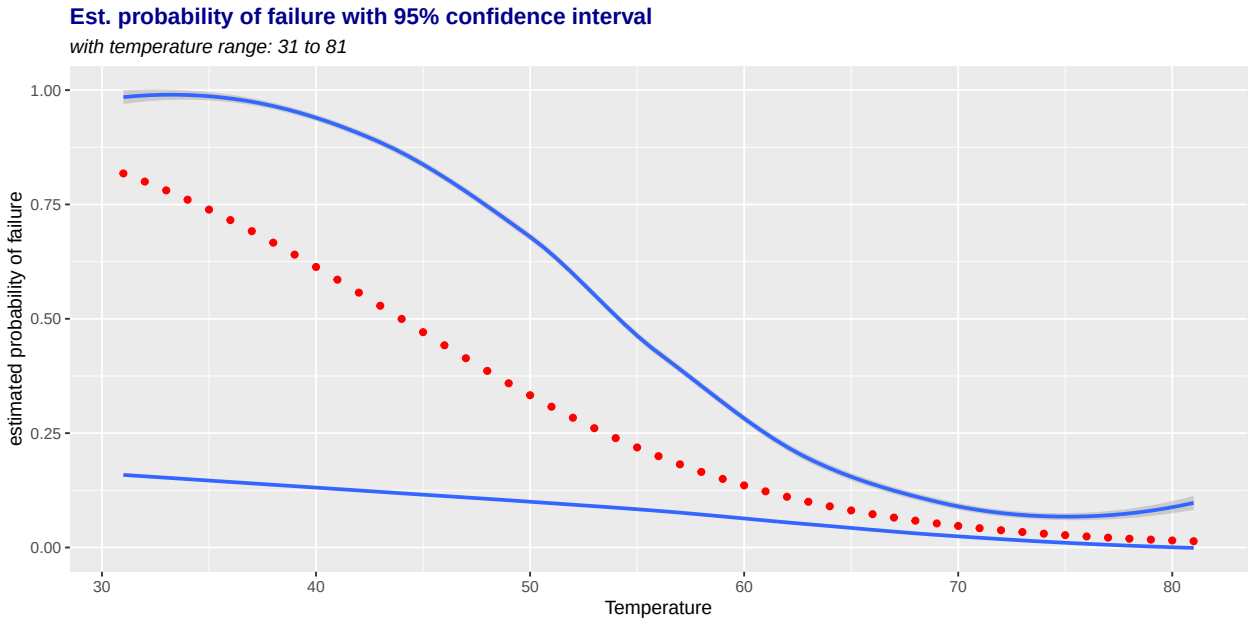


(c) Include the 95% Wald confidence interval bands for π on the plot. Why are the bands much wider for lower temperatures than for higher temperatures?

As the CI is used to measure the certainty of model prediction, there are fewer data points at lower temperature than at higher temperature. This resulted the wider band of CI due to uncertainty and variability at lower temperature for the same 95% CI in comparison to higher temperature with much narrow CI band.

```
alpha = 0.05

ggplot() +
  geom_point(mapping = aes(x = predict.data$Temp, y = pi.hat), color = 'red') +
  geom_smooth(mapping = aes(x = predict.data$Temp, y = CI.pi.lower)) +
  geom_smooth(mapping = aes(x = predict.data$Temp, y = CI.pi.upper)) +
  labs( x = 'Temperature', y = 'estimated probability of failure',
       title ="Est. probability of failure with 95% confidence interval",
       subtitle = "with temperature range: 31 to 81") +
  theme(plot.title = element_text(lineheight=1, face="bold", color = "dark blue")) +
  theme(plot.subtitle=element_text(face="italic", color="black"))
```

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```



(d) The temperature was 31° at launch for the Challenger in 1986. Estimate the probability of an O-ring failure using this temperature, and compute a corresponding confidence interval. Discuss what assumptions need to be made in order to apply the inference procedures.

We find at 31F the estimated probability of O.ring failure is 81.8% with 95% confidence interval of 16% to 99%.

The assumptions that need to be made in order to apply the inference procedures are: a. Each launch has to be independent; b. distribution has to be the same normal distribution; c. the model coefficient is normally distributed.

```
pi.hat.T31 <- pi.hat[1]
CI.pi.lower.T31 <- CI.pi.lower[1]
CI.pi.upper.T31 <-  CI.pi.upper[1]
data.frame(pi.hat.T31, lower = CI.pi.lower.T31, upper = CI.pi.upper.T31)
```

```
##   pi.hat.T31     lower     upper
## 1  0.8177744 0.1596025 0.9906582
```

(e) Rather than using Wald or profile LR intervals for the probability of failure, Dalal et al. (1989) use a parametric bootstrap to compute intervals. Their process was to (1) simulate a large number of data sets (n = 23 for each) from the estimated model of Temp; (2) estimate new models for each data set, say and (3) compute at a specific temperature of interest. The authors used the 0.05 and 0.95 observed quantiles from the simulated distribution as their 90% confidence interval limits. Using the parametric bootstrap, compute 90% confidence intervals separately at temperatures of 31° and 72°.27

```
# beta from glm.mod.binom2
beta.hat <- glm.mod.binom2$coefficients
# set.seed(8558)
Temp.sim <- sample(x = df$Temp, size = 23, replace = TRUE)
# plug in the 2nd model - estimated pi
pi.sim<- exp(beta.hat[1] + beta.hat[2]*Temp.sim) / (1 + exp(beta.hat[1] +
```

```
                                                    beta.hat[2]*Temp.sim))
# sample from pi with binomial distribution
# set.seed(8118)
y <- rbinom(23, size = 6, prob = pi.sim)

df1 <- data.frame(y,Temp.sim, pi.sim)
head(df1,5)
```

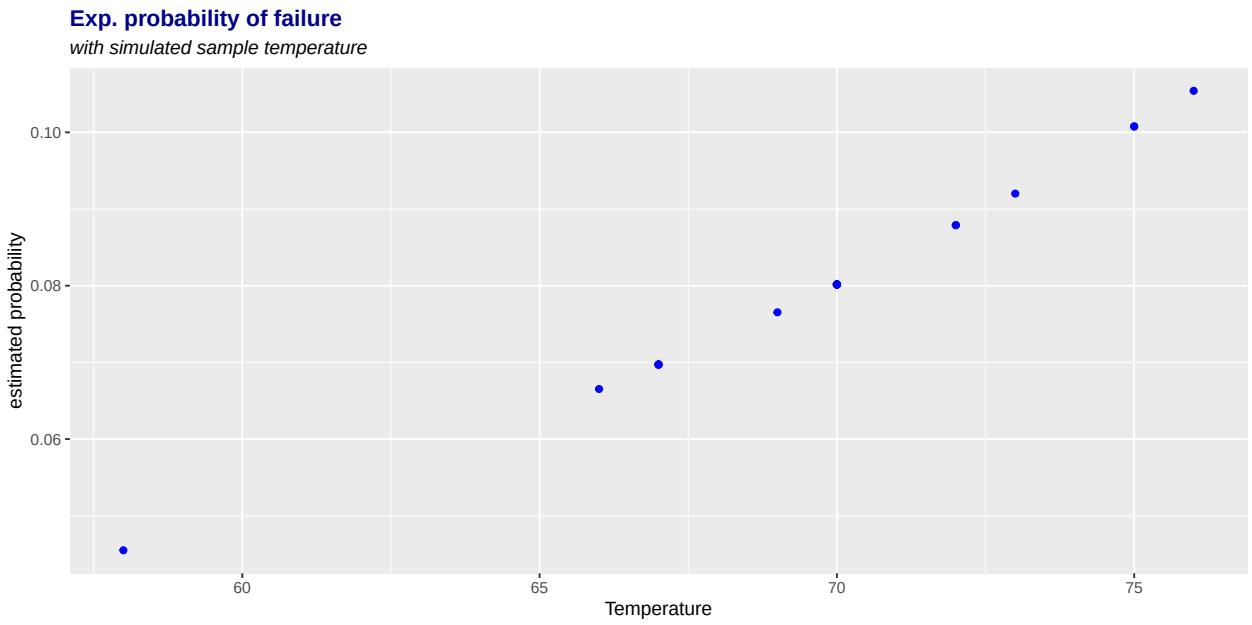
```
##   y Temp.sim     pi.sim
## 1 2       67 0.06535695
## 2 0       70 0.04710595
## 3 0       67 0.06535695
## 4 0       66 0.07278337
## 5 1       67 0.06535695
```

```
# fit it to the new model
mod.fit.sim <- glm(formula = y/6 ~ Temp.sim, family = binomial(link = "logit"),
                   weights = rep(6, 23))
summary(mod.fit.sim)
```

```
##
## Call:
## glm(formula = y/6 ~ Temp.sim, family = binomial(link = "logit"),
##     weights = rep(6, 23))
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.1289  -1.0013  -0.9313   0.6136   1.8956
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -5.95984    6.48786  -0.919    0.358
## Temp.sim     0.05028    0.09239   0.544    0.586
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 23.621  on 22  degrees of freedom
## Residual deviance: 23.312  on 21  degrees of freedom
## AIC: 44.518
##
## Number of Fisher Scoring iterations: 5
```

```
# extract model coefficients
beta.hat.sim <- mod.fit.sim$coefficients
# estimate the model
y.hat.sim <- predict(mod.fit.sim, newdata = data.frame(Temp=Temp.sim), type =
                       "response", se=TRUE)
O.ring.hat <- y.hat.sim$fit*6
```

```
sim.plot <- ggplot() +
  geom_point(mapping = aes(x = Temp.sim, y = y.hat.sim$fit), color = 'blue') +
  labs(x = 'Temperature', y = 'estimated probability',
       title = "Exp. probability of failure",
       subtitle = "with simulated sample temperature") +
  theme(plot.title = element_text(lineheight=1, face="bold", color = "dark blue")) +
  theme(plot.subtitle=element_text(face="italic", color="black"))
sim.plot
```




```
set.seed(8558)
save.pi_31 <- rep(NA, 1000)
save.pi_72 <- rep(NA, 1000)
for (i in 1:1000) {
  beta.hat <- glm.mod.binom2$coefficients
  Temp.sim <- sample(x = df$Temp, size = 23, replace = TRUE)
  pi<-exp(beta.hat[1] + beta.hat[2]*Temp.sim) / (1 + exp(beta.hat[1] + beta.hat[2]*Temp.sim))
  y <- rbinom(23, size = 6, prob = pi)
  mod.fit.sim <- glm(formula = y/6 ~ Temp.sim, family = binomial(link = "logit"),
                     weights = rep(6, 23))
  beta.hat.sim <- mod.fit.sim$coefficients
  y.hat.sim <- predict(mod.fit.sim, newdata = data.frame(Temp=Temp.sim), type =
                         "response", se=TRUE)
  (pi_31 = as.numeric(exp(beta.hat.sim[1]+beta.hat.sim[2]*31) /
                        (1+exp(beta.hat.sim[1]+beta.hat.sim[2]*31))))
  (pi_72 = as.numeric(exp(beta.hat.sim[1]+beta.hat.sim[2]*72.27) /
                        (1+exp(beta.hat.sim[1]+beta.hat.sim[2]*72.27))))
  (save.pi_31[i] <- pi_31)
  (save.pi_72[i] <- pi_72)
  CI.df <- data.frame(pi.31 = save.pi_31, pi.72 = save.pi_72)
}
```

```r
# print mean and CI for T31 and T72
CI.T31 <- quantile(save.pi_31, c(0.05, 0.95))
CI.T72 <- quantile(save.pi_72, c(0.05, 0.95))
paste("mean of Temp 31 is:",mean(save.pi_31), "mean of Temp 72 is:",mean(save.pi_72))
```

```
## [1] "mean of Temp 31 is: 0.719327898963717 mean of Temp 72 is: 0.0364817562496689"
```

```r
paste("CI of Temp 31:", "Low" , CI.T31[1], "High:",CI.T31[2])
```

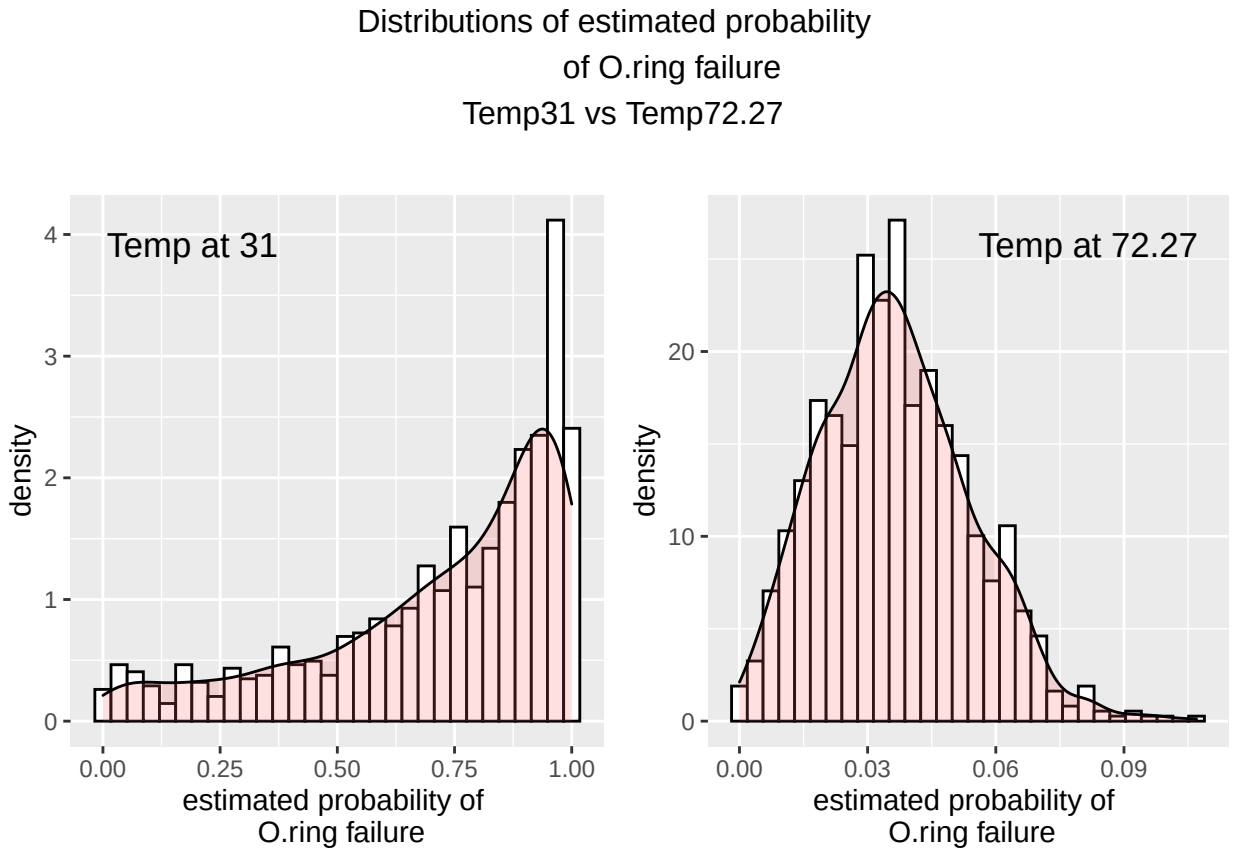
```
## [1] "CI of Temp 31: Low 0.125582095953187 High: 0.991530896235629"
```

```r
paste("CI of Temp 72:", "Low" , CI.T72[1], "High:",CI.T72[2])
```

```
## [1] "CI of Temp 72: Low 0.010124187441672 High: 0.0658125339678034"
```

```r
#plot the distribution of two CIs
plot.T31 <- ggplot(CI.df, aes(x=save.pi_31)) +
    geom_histogram(aes(y=..density..), colour="black", fill="white") +
    geom_density(alpha=0.2, fill="#FF6666") +
    labs(x = 'estimated probability of \n O.ring failure', y = 'density') +
  ggtitle("Temp at 31") +
  theme(plot.title = element_text(vjust= -10,hjust= 0.1))

plot.T72 <- ggplot(CI.df, aes(x=save.pi_72)) +
    geom_histogram(aes(y=..density..), colour="black", fill="white") +
    geom_density(alpha=0.2, fill="#FF6666") +
    labs(x = 'estimated probability of \n O.ring failure', y = 'density') +
  ggtitle("Temp at 72.27") +
  theme(plot.title = element_text(vjust= -10,hjust= 0.9))

grid.arrange(plot.T31, plot.T72, top = "Distributions of estimated probability
             of O.ring failure \n Temp31 vs Temp72.27", ncol=2)
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



Based on our 1000 iteration of bootstrap simulation, with 90% confidence, the estimated probability of O.ring failures is between 12.6% and 99.2% at Temperature 31 with mean at 72%. It resembles the similar pattern of wide range we observed in part d, indicating uncertainty of this estimation; however, the left skewed distribution of probabilities demonstrated higher probability of failure towards 100%. On the contrary, for temperature at 72.27, the probability distribution is more like a normal distribution with mean centered at 3.6% and CI is between 1% and 6.6% for estimated probability of failure and we are more certain that O.ring is not likely to fail at launch with temperature at 72.27.

(f) Determine if a quadratic term is needed in the model for the temperature.

We included quadratic term, Temp^2 in both the binomial model for testing.

```
glm.mod.binom3 <- glm(O.ring/Number ~Temp + I(Temp^2), data = df,
                      family = binomial(link = 'logit'), weights = Number)
summary(glm.mod.binom3)
```

```
##
## Call:
## glm(formula = O.ring/Number ~ Temp + I(Temp^2), family = binomial(link = "logit"),
##     data = df, weights = Number)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -0.84320  -0.72385  -0.61980  -0.01335   2.52101
##
```

```
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) 22.126148  23.794426   0.930    0.352
## Temp        -0.650885   0.740756  -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: 24.230  on 22  degrees of freedom
## Residual deviance: 17.592  on 20  degrees of freedom
## AIC: 37.152
##
## Number of Fisher Scoring iterations: 5
```

```
Anova(glm.mod.binom3, test = "LR")
```

```
## Analysis of Deviance Table (Type II tests)
##
## Response: O.ring/Number
##           LR Chisq Df Pr(>Chisq)
## Temp       0.71878  1     0.3965
## I(Temp^2)  0.49470  1     0.4818
```

Quadratic term of temperature is included in the binomial model for evaluation. We found this added term with high p-value is not only statistically insignificant,but also drive Temp term to insignificant, hence, we decided not to include this term in the final model.

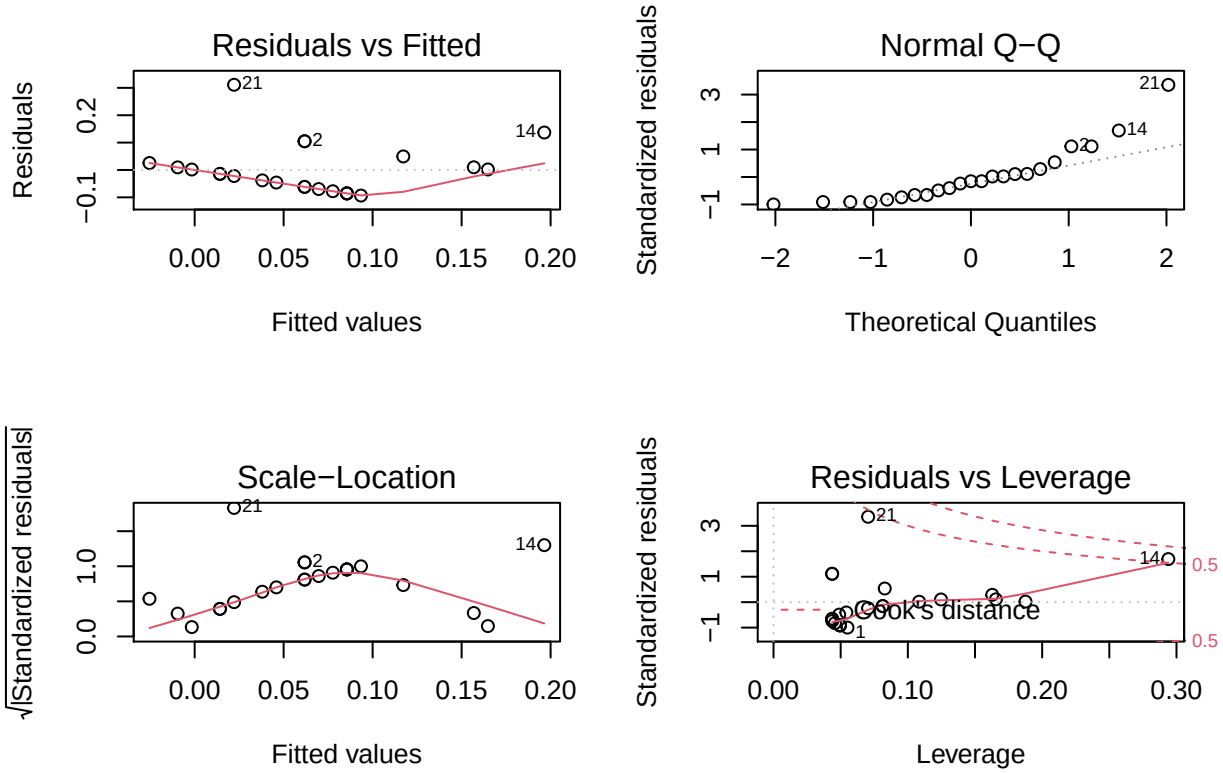
Part 4 With the same set of explanatory variables in your final model, estimate a linear regression model. Explain the model results; conduct model diagnostic; and assess the validity of the model assumptions. Would you use the linear regression model or binary logistic regression in this case? Explain why.

```
# estimate a linear regression model.
mod.fit.lr <- lm(formula = O.ring/Number ~ Temp, data = df, weights = Number)
summary(mod.fit.lr)
```

```
##
## Call:
## lm(formula = O.ring/Number ~ Temp, data = df, weights = Number)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.22894 -0.16102 -0.03486  0.04311  0.76223
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.616402   0.203252   3.033  0.00633 **
## Temp        -0.007923   0.002907  -2.725  0.01268 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## Residual standard error: 0.2357 on 21 degrees of freedom
## Multiple R-squared:  0.2613, Adjusted R-squared:  0.2261
## F-statistic: 7.426 on 1 and 21 DF,  p-value: 0.01268
```

```
# Linear model diagnosis with residual , QQ plots
par(mfrow=c(2,2))
plot(mod.fit.lr)
```



From the plots above, they show the relationship between the explanatory variable, Temp and output variable violates the assumption of homogeneity of variance and linear relationship as find from the scale location plot and residuals vs fitted plot respectively, moreover the normal QQ plot also shows the residuals are not normal distribution, linear model cannot be used for the prediction in this case. We also observed from the plots which identified the influential observation as #2, 14 and 21, however, our analysis would include these observations given the size of the data set is small and extreme conditions is recommended to be included for prediction in order to manage the risk and final decision (Go/No Go) on the shuttle launch in the future.

Based on the linear diagnosis plots analysis, it violates the linear model assumptions, moreover, the relationship between o.ring failure and temp is not linear, thus, binary logistic regression model is recommended for this case.

Part 5

Interpret the main result of your final model in terms of both odds and probability of failure. Summarize the final result with respect to the question(s) being asked and key takeaways from the analysis.

We evaluated three models (binomial, binary, linear) with the pre-accident launch data. We dis-

counted the linear regression model because some of the key assumptions for linear regression model were violated. We do not present the binary model results because they are similar to binomial.

We got following results for our binomial model:

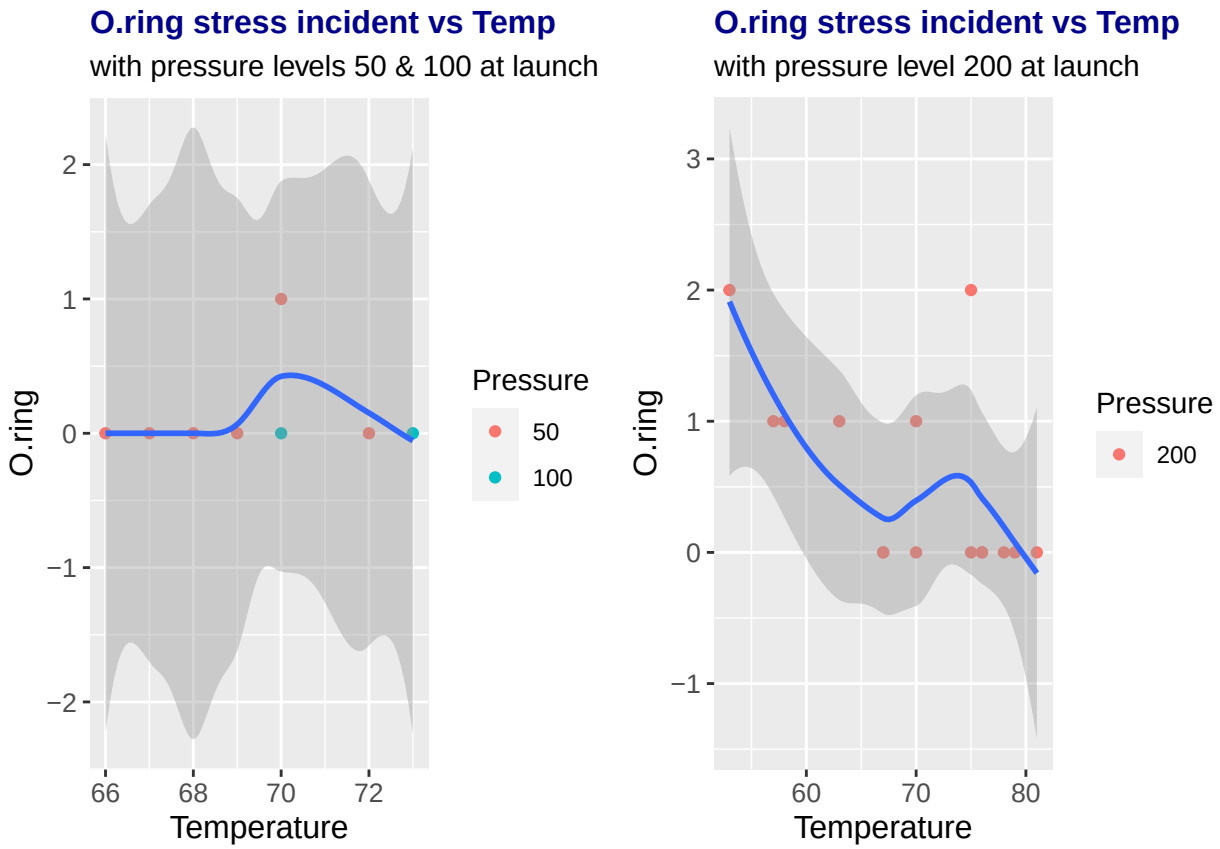
- The odds of failure decrease by 0.315 with every 10 degree increase in temperature.
- At 31F we estimate mean probability of failure of O.ring to be 81.2% with 16% to 99% as the 95% Confidence Interval.
- At 72.27F we estimate mean probability of failure of O.ring to be 3.66% with 1.3% to 93.7% as the 95% Confidence Interval.

Therefore, based on the statistical analysis and this report, we believe there is strong statistical evidence of temperature's effect on O.ring failure. At 31F, the temperature at which Challenger was launched, there is at least 16% chance of O.ring failure. Postponement to 72.27F would have reduced the chance of O.ring failure to at the most 9.37%. This postponement would have decreased the probability of failure by minimum 42% calculated by (0.1596025-0.09370094)/0.1596025. Thus, we agree with the author that statistical science plays an important role in space-shuttle risk management process and could have been used to predict failure prior to the launch of Challenger.

Apendix More EDA

```
ringstress.temp.press.50.100 <- ggplot(data = filter(df, Pressure.factor != '200'),
                                       mapping = aes(x = Temp, y = O.ring)) +
  geom_point(mapping = aes(color =  Pressure.factor)) +
  labs(color = "Pressure") +
  geom_smooth(data = filter(df, Pressure.factor != '200')) +
  labs( x = 'Temperature', y = 'O.ring',
       title ="O.ring stress incident vs Temp",
       subtitle = "with pressure levels 50 & 100 at launch") +
  theme(axis.text = element_text(size=10),
        axis.title = element_text(size=12),
        plot.title = element_text(lineheight=1, size = 12, face="bold",
                                  color = "dark blue"))

# examine the pressure at 200 level
ringstress.temp.press.200 <- ggplot(data = filter(df, Pressure.factor == '200'),
  mapping = aes(x = Temp, y = O.ring)) +
  geom_point(mapping = aes(color = Pressure.factor)) +
  labs(color = "Pressure") +
  geom_smooth(
  data = filter(df, Pressure.factor == '200')
  ) +
  labs( x = 'Temperature', y = 'O.ring',
       title ="O.ring stress incident vs Temp",
       subtitle = "with pressure level 200 at launch") +
  theme(axis.text = element_text(size=10),
        axis.title = element_text(size=12),
        plot.title = element_text(lineheight=1, size =12, face="bold",
                                  color ="dark blue"))

grid.arrange(ringstress.temp.press.50.100, ringstress.temp.press.200,
             ncol=2)
```

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```



(Binary Model)

```
glm.mod1 <- glm(Stress ~ Temp + Pressure,
                family = binomial(link = logit),
                data = df)
summary(glm.mod1)
```

```
##
## Call:
## glm(formula = Stress ~ Temp + Pressure, family = binomial(link = logit),
##     data = df)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.1993  -0.5778  -0.4247   0.3523   2.1449
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) 13.292360   7.663968   1.734   0.0828 .
## Temp        -0.228671   0.109988  -2.079   0.0376 *
## Pressure     0.010400   0.008979   1.158   0.2468
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
```

```
##
##     Null deviance: 28.267  on 22  degrees of freedom
## Residual deviance: 18.782  on 20  degrees of freedom
## AIC: 24.782
##
## Number of Fisher Scoring iterations: 5
```

LRT for Binary Model

```
Anova(glm.mod1, test = 'LR')
```

```
## Analysis of Deviance Table (Type II tests)
##
## Response: Stress
##          LR Chisq Df Pr(>Chisq)
## Temp       7.7542  1   0.005359 **
## Pressure   1.5331  1   0.215648
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Comparing Binary Model with and without Pressure

```
glm.mod2 <- glm(Stress ~ Temp, family = binomial, data = df)
anova(glm.mod2, glm.mod1)
```

```
## Analysis of Deviance Table
##
## Model 1: Stress ~ Temp
## Model 2: Stress ~ Temp + Pressure
##   Resid. Df Resid. Dev Df Deviance
## 1        21     20.315
## 2        20     18.782  1   1.5331
```