W271 Lab1

Bas Hendri, Nishant Velagapudi & Dan Volk

January 21, 2018

Contents

Intro

The shuttle Challenger accident was a nationally noted disaster: the shuttle's distruction was linked to the failure of an "O-ring" to prevent fuel from leaking through a joint into a booster rocket. It was known prior to launch that cold temperature had a detrimental effect on O-ring function. Dalal et al. (1989) explored the 23 test launches of the space shuttle to find whether or not a statistical analysis of the relationship between temperature and O-ring failure would have informed staff that the Challenger launch needed to be delayed.

Here, we explore the test flight dataset and model the probability of O-ring failure on the basis of launch temperature and nozzle pressure. We evaluate the significance of our variables in our model before settling on a final specification, and include coefficient interpretations for our final model.

```r
library(car)
library(plyr)
library(dplyr)
library(ggplot2)
library(gridExtra)
```

```r
#change to whatever wd you need
#setwd("C:\\users\\nishray\\documents\\Berkeley_MIDS\\W271\\Lab1\\true\\")
#setwd("~/Berkeley_MIDS/W271/Lab1/true")
challenger <- read.csv("challenger.csv")
```

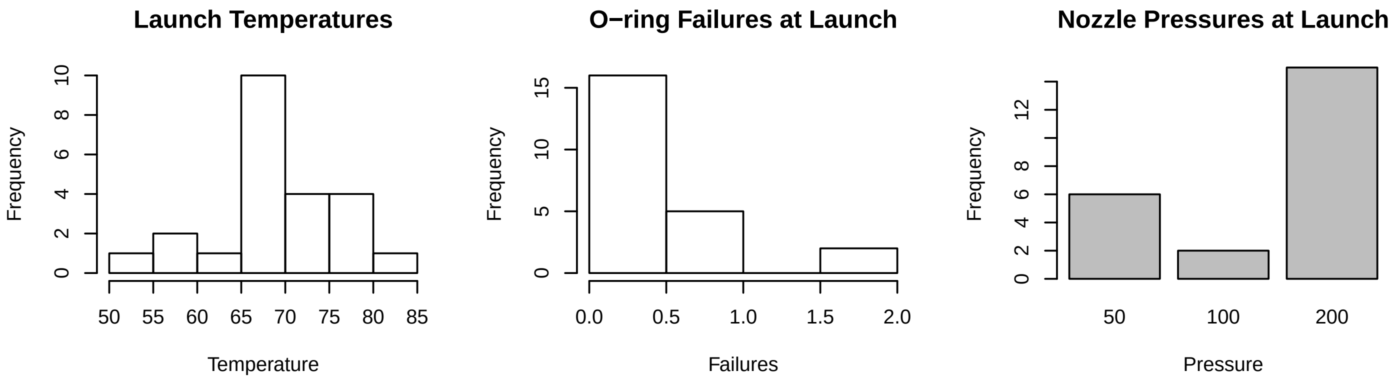
Exploratory Analysis

Univariate Analyses

First, we explore the features of our dataset. We have 5 variables in the set. The **`Number`** variable represents the number of O-rings in the launch: which is invariantly 6 for all observations. The **`Flight`** variable is an identity property. Since it could describe the order in which the pre-accident launches occurred, it represents potentially confounding variable (e.g. relationship between Flight and Failures is positive, relationship between Flight and Temperature is positive leads to bias).

`Pressure` is an integer variable, but only takes 3 values, that is `Pressure` $\in \{50, 100, 200\}$. As a result, we'll convert this variable to a factor when creating the requisite models. The **`O.ring`** variable represents the number of O-rings that fail in any given launch. In some flights, we have more than one failing O-ring.

```r
par(mfrow = c(1,3))
hist(challenger$Temp, main="Launch Temperatures", xlab="Temperature")
hist(challenger$O.ring, main="O-ring Failures at Launch", xlab="Failures")
plot(challenger$Pressure, main = "Nozzle Pressures at Launch", xlab="Pressure",ylab="Frequency")
```



Visualization of individual variables reveals a few interesting features. First, the coldest temperature in our test launch dataset is 53°F: the majority of test flights were performed between 70°F and 80°F. Proportionally, most flights had no O-ring failures (16), of the remaining 7 flights, 5 had one failure, while 2 flights had two failures apiece. Finally, the majority of test launches used a nozzle pressure of 200psi.

Bivariate Analyses

First, we look to see if there are any relationships between the flight number and our variables of interest (Pressure, temperature, and number of failures). This is interesting because it might reveal a temporal relationship. For example, if later flights are less likely to have an O-ring failure and also happened to see an increase in temperature (which is possible if the tests started in spring/winter time and continued until summer), then the reduced probability of O-ring failure could be wrongly attributed to temperature due to this confounding effect.

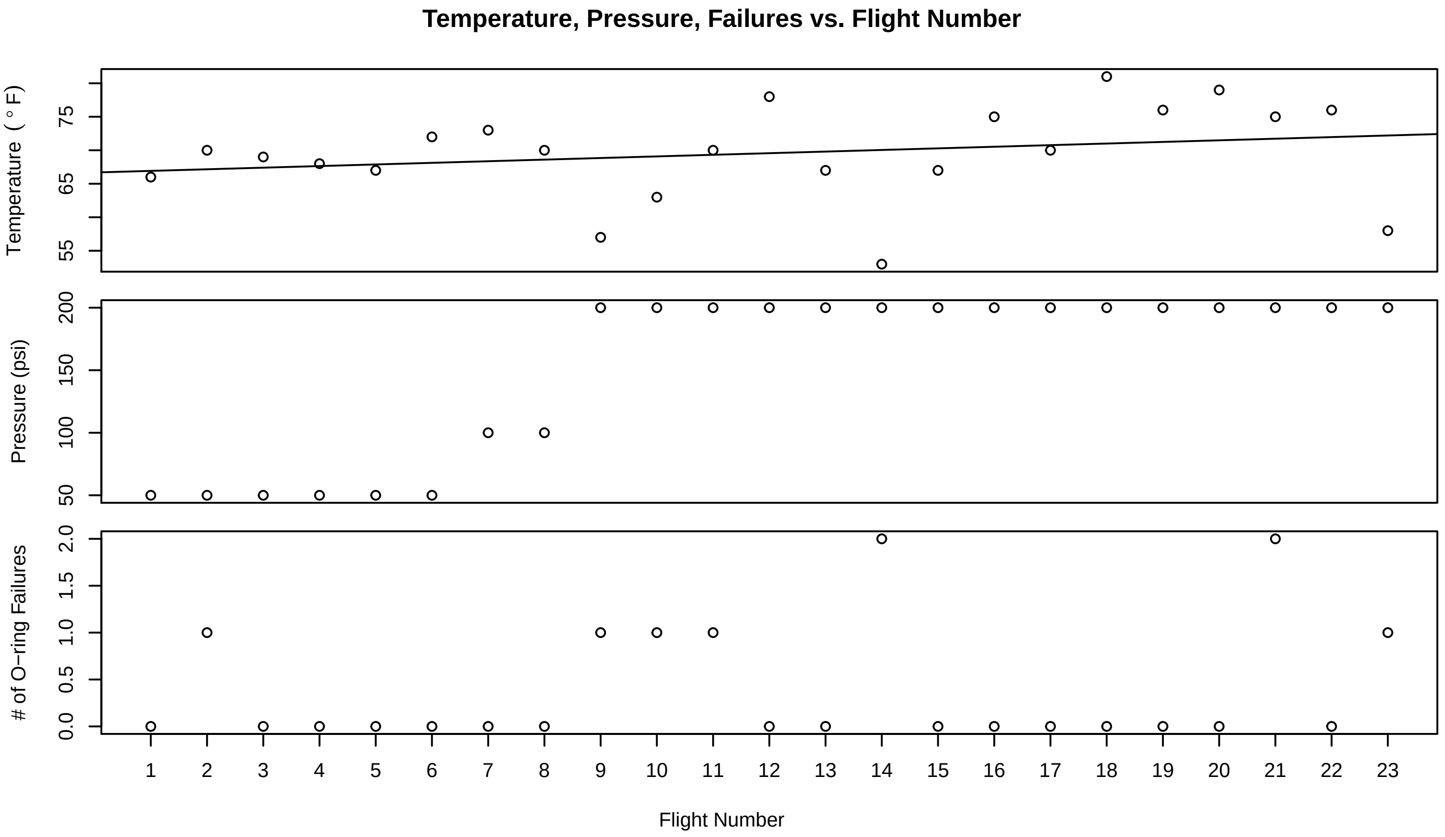
```r
# check for relationship between Flight and other explanatory/response variables (any temporal pattern)
par(mfrow=c(3,1), oma=c(5,0,2,0) + 0.1, mar=c(0,4,1,0) + 0.1)

plot(challenger$Flight, challenger$Temp, ylab=expression(Temperature*phantom(x)*(~degree~F)), xaxt="n")
abline(lm(Temp ~ Flight, data=challenger))

plot(challenger$Flight, as.numeric(levels(challenger$Pressure)[challenger$Pressure]), ylab="Pressure (psi)",
     xaxt="n")

plot(challenger$Flight, challenger$O.ring, xaxt="n", ylab="# of O-ring Failures")
axis(1, at=challenger$Flight)

title(main="Temperature, Pressure, Failures vs. Flight Number", xlab="Flight Number", outer=TRUE)
```

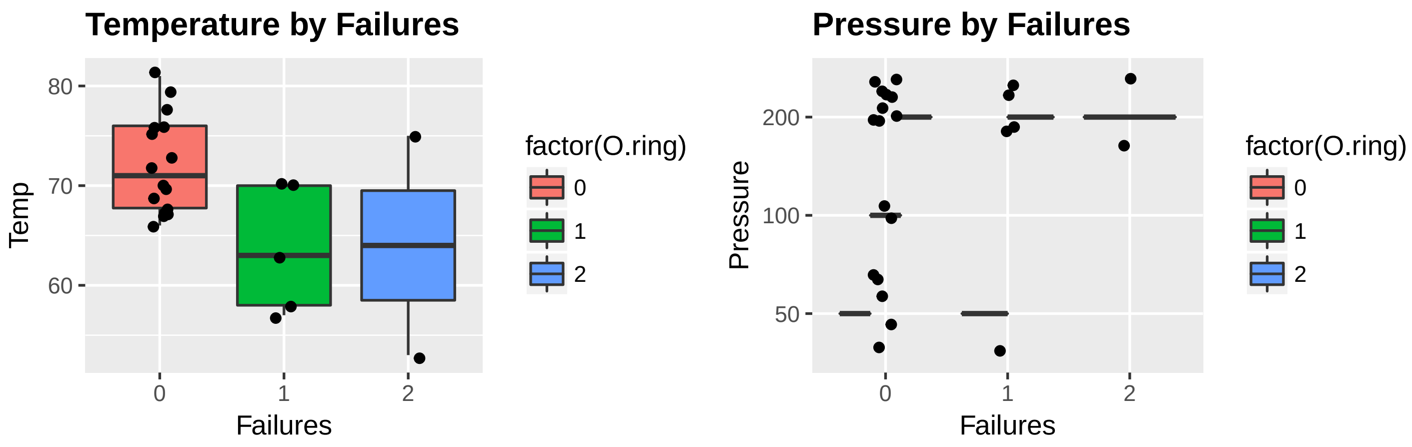


We can see that generally temperature is increasing with each subsequent flight. Additionally, pressure seems to have been incremented at various flight numbers (the first 6 flights are at a pressure of 50, the next two at 100, and the remainder at 200). Finally, there does not appear to be visual evidence of a trend between Flight number and O-ring failures.

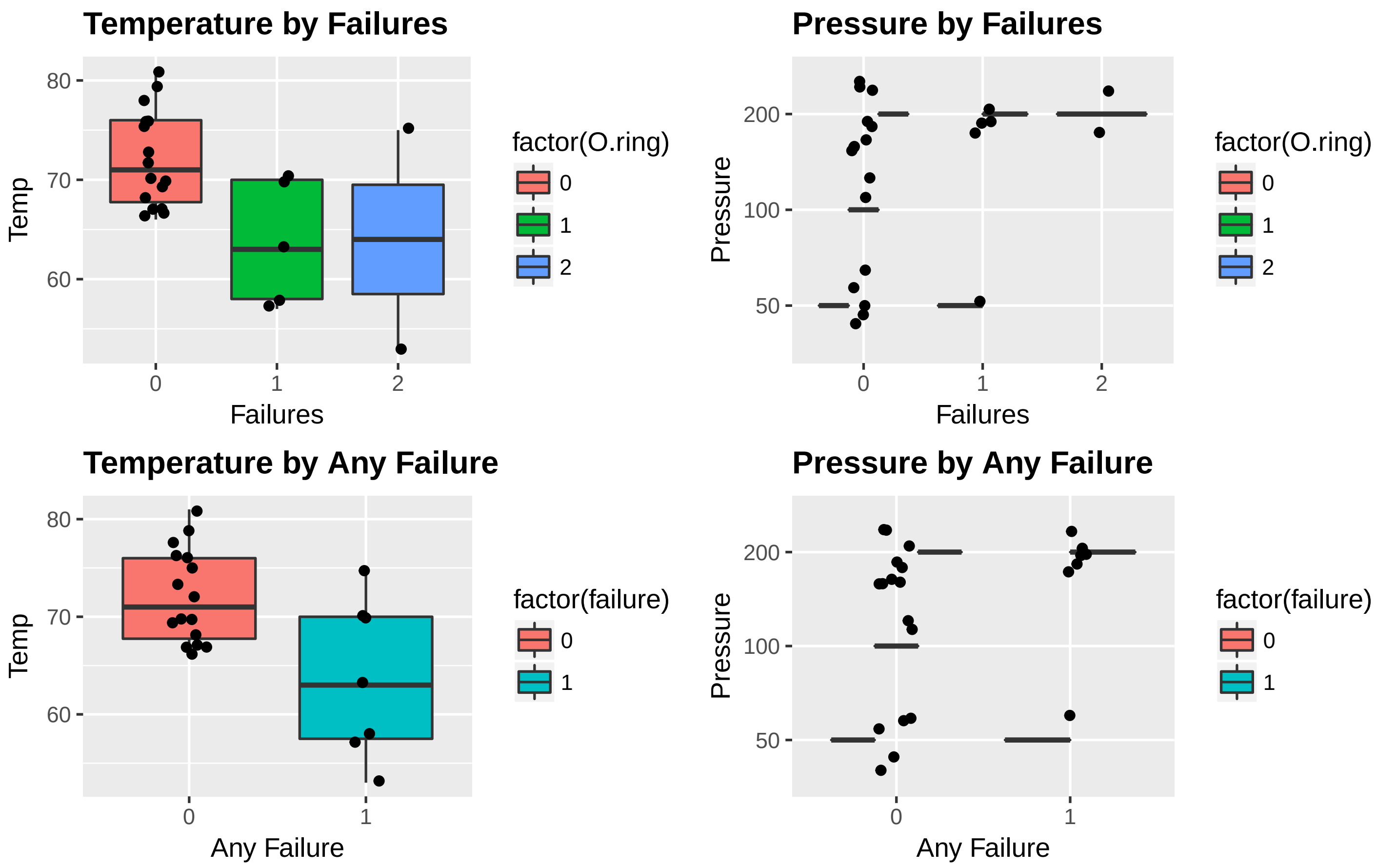
Given that we will look to predict probability of failure overall, we will create an indicator variable for overall failure (tracking if there are any failures in the test launch at all, referred to as "Any Failure"). We will include this variable in subsequent explorations. We now visualize and tabulate bivariate relationships of interest.

```
challenger$failure <- 1*(challenger$O.ring>0)

# Plot the O-Ring failure by temperature
plot1 <- ggplot(challenger, aes(factor(O.ring), Temp)) + geom_boxplot(aes(fill=factor(O.ring))) +
         geom_jitter(width=0.1) + xlab("Failures")     + ggtitle("Temperature by Failures")     +
         theme(plot.title=element_text(lineheight=1, face="bold"))
# Plot the O-Ring failure by Pressure
plot2 <- ggplot(challenger, aes(factor(O.ring), Pressure)) + geom_boxplot(aes(fill=factor(O.ring))) +
         geom_jitter(width=0.1) + xlab("Failures")         + ggtitle("Pressure by Failures")        +
         theme(plot.title=element_text(lineheight=1, face="bold"))
grid.arrange(plot1, plot2, ncol=2)
```



```
# Plot the O-Ring failure by temperature
plot3 <- ggplot(challenger, aes(factor(failure), Temp)) + geom_boxplot(aes(fill=factor(failure))) +
         geom_jitter(width=0.1) + xlab("Any Failure")   + ggtitle("Temperature by Any Failure")   +
         theme(plot.title=element_text(lineheight=1, face="bold"))
# Plot the O-Ring failure by Pressure
plot4 <- ggplot(challenger, aes(factor(failure), Pressure)) + geom_boxplot(aes(fill=factor(failure))) +
         geom_jitter(width=0.1) + xlab("Any Failure")       + ggtitle("Pressure by Any Failure")      +
         theme(plot.title=element_text(lineheight=1, face="bold"))
grid.arrange(plot1, plot2, plot3, plot4, ncol=2)
```



It immediately stands out that of the the six test launches during which an O-ring failure was observed, four were during the the four coldest launches (Temperature by Failures plot). Proportionally, the most failures also occur at a pressure of 200, but it is difficult to immediately discern whether that is an individual effect of nozzle pressure or due to an overlap between the coldest launches at nozzle pressures of 200.

We will use cross tabulations to create contingency tables for the purpose of summarizing our bivariate relationships of interest. We start with the relationship between the temperature and failure variables

```
# cross tabulations
# one failure at least overall versus temperature
xtabs(Temp ~ failure, aggregate(Temp ~ failure, challenger, mean))
```

```
## failure
##        0        1
## 72.12500 63.71429
```

```
# check significance of mean temperature difference
t.test(challenger[challenger$failure == "1", "Temp"], challenger[challenger$failure == "0", "Temp"])
```

```
##
##  Welch Two Sample t-test
##
## data:  challenger[challenger$failure == "1", "Temp"] and challenger[challenger$failure == "0", "Temp"]
## t = -2.5387, df = 7.9166, p-value = 0.03507
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -16.0644855  -0.7569431
## sample estimates:
## mean of x mean of y
##  63.71429  72.12500
```

```
# bucketing temperature
challenger['TempBucket'] <- transform(challenger, group=cut(as.numeric(sub('[%]', '', Temp)),
    breaks=c(31, 40, 50, 60, 70, 80, 90),
    labels=c('31-40', '41-50', '51-60', '61-70', '71-80', '81-90')))$group

# summary of temperature bucket versus overall failure
data.frame(ProportionOfFailures=table(challenger$TempBucket, challenger$failure)[,"1"] /
             rowSums(table(challenger$TempBucket, challenger$failure)),
           Flights=rowSums(table(challenger$TempBucket, challenger$failure)),
           FailureFlights=table(challenger$TempBucket, challenger$failure)[,"1"])
```

```
##       ProportionOfFailures Flights FailureFlights
## 31-40                  NaN       0              0
## 41-50                  NaN       0              0
## 51-60            1.0000000       3              3
## 61-70            0.2727273      11              3
## 71-80            0.1250000       8              1
## 81-90            0.0000000       1              0
```

We can see that the average temperature when an O-ring failure was observed was 63.7°F, while the average temperature when no O-ring failure was observed was 72.2°F. A T-test describes this difference as statistically signifance, with a p-value of .035. We also note again that we have no observations below 50°F.

Bucket temperatures were then bucketed into discrete categories. We visualize the distribution of test launches within each temperature bucket and summarize the number of failures per bucket. We see that all launches with a temperature between 51°F and 60°F involved at least one O-ring failure: the proportion of launches with an O-ring failure steadily increases as the temperature bucket decreases.

Finally, we examine the average temperature at each discrete nozzle pressure in case of O-ring failure or lack thereof:

```
#failure vs temperature and pressure
xtabs(Temp ~ failure + Pressure, aggregate(Temp ~ failure + Pressure, challenger, mean))
```

```
##        Pressure
## failure       50      100      200
##       0 68.40000 71.50000 74.33333
##       1 70.00000  0.00000 62.66667
```

```
xtabs(Number/6 ~ failure + Pressure ,data=challenger)
```

```
##        Pressure
## failure 50 100 200
##       0  5   2   9
##       1  1   0   6
```

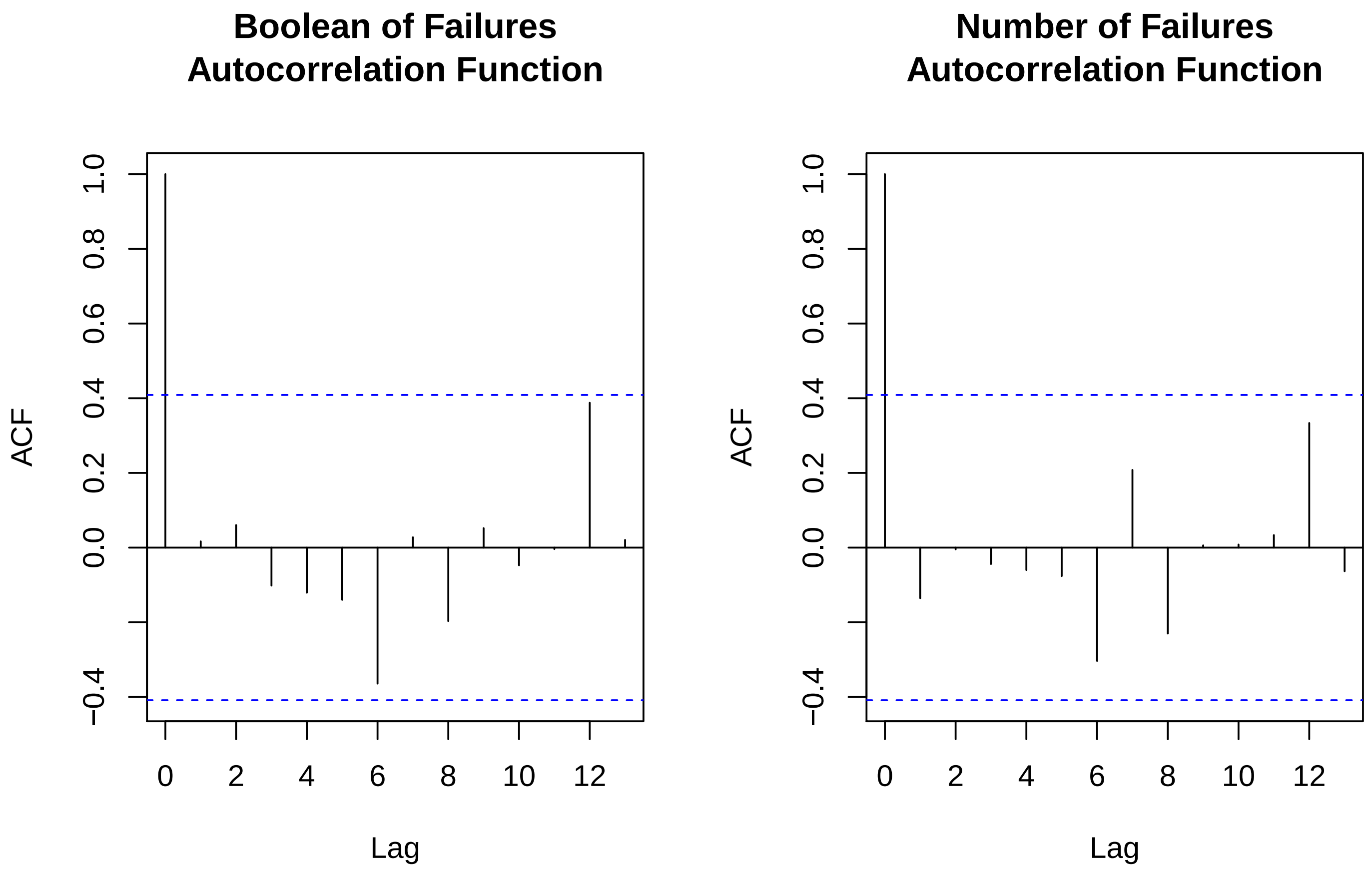
```
#overall rate of failures - 7 failure cases out of 23 total
#two cases where two rings failed
```

We observe that the temperature gap between launches with and without an O-ring failure appears to be smaller at a pressure of 50 than a pressure of 200. However, we lack sufficient sample size to place much confidence in this observation: we only have 6 observations at a pressure of 50, as compared to a sample of 15 observations at a pressure of 200.

Temporal Analysis

In this exploration, we'd like to verify that O-ring failures are random and invariant to the order of testing. We attempt to estimate the autocorrelation function of **failure** and **O.ring** and observe if any amount of lag introduces significant correlation. As we can see in the correlograms, None of the spikes breach 5% significance blue line (except for `lag=0` which is by design).

```
par(mfrow=c(1,2))
acf(challenger$failure, ylab="ACF", main=paste0("Boolean of Failures","\n","Autocorrelation Function"))
acf(challenger$O.ring, ylab="ACF", main=paste0("Number of Failures","\n","Autocorrelation Function"))
```



Literature Questions 4 & 5

Question 4: Logistic regression model of O-ring failures

4.a. Independence assumption

In order to use linear/logistic regression, we have to have a two-level response variable and we have to assume independence of each trial within the dataset. The former requires casting the number of O-ring failures into a boolean specifying whether or not there was a failure rather than number of failures.

The latter requires that each O-ring is IID with respect to likelihood of failure on any given trial. These two assumptions are fine in the initial model: but the authors convert from probability of overall failure p_{FF} back to the probability of a single ring failing p_F, defined as $p_F = 1 - (1 - p_{FF})^{\frac{1}{6}}$. This is potentially problematic: given that O-ring failures within a single flight may not actually be independent: the failure of one O-ring might compromise another. Further, we assume that there is no relationship between flights and that knowledge of a previous flight's O-ring failure was not used in any way in the subsequent test flights.

The authors argue through contrasting against estimated binary-logistic model that both binomial- and binary-logistic models would yield similar result in terms of predicting number of O-ring failures (Figure 4 in Dalal et al. (1989)).

4.b. Logistic regression model using explanatory variables in a linear form.

```
# Fit a model using the explanatory variables Temperature and Pressure
chal.glm1 <- glm(formula=failure ~ Temp + Pressure, family=binomial, data=challenger)
summary(chal.glm1)
```

```
##
## Call:
```

```
## glm(formula = failure ~ Temp + Pressure, family = binomial, data = challenger)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.2093  -0.6044  -0.4151   0.3635   2.0479
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)   13.5106     7.4288   1.819   0.0690 .
## Temp          -0.2211     0.1078  -2.050   0.0403 *
## Pressure100  -15.2969  2761.7586  -0.006   0.9956
## Pressure200    1.3774     1.3154   1.047   0.2950
## ---
## 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: 18.214  on 19  degrees of freedom
## AIC: 26.214
##
## Number of Fisher Scoring iterations: 16
```

4.c. Evaluating variable importance

According to the LRT below temperature is a significant variable while pressure is not. This is in line with the results from the paper as well as our EDA above.

```
Anova(chal.glm1)
```

```
## Analysis of Deviance Table (Type II tests)
##
## Response: failure
##          LR Chisq Df Pr(>Chisq)
## Temp       7.3827  1   0.006585 **
## Pressure   2.1008  2   0.349793
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

4.d. Removing `Pressure` variable

Based on LRTs performed with pressure as a integer and pressure as a factor, it seems that pressure is not used significantly in these models. These results concur with the author's findings: we can remove this explanatory variable from the regression without a significant impact on model fit.

We investigate if removal of Pressure leads to a complete separate situation. This does not appear to be the case (see figure "Temperature by Failures" above).

Question 5: Simplified logistic regression model of O-ring failures

Simplified model: $\text{logit}(\pi) = \beta_0 + \beta_1 \texttt{Temp}$, where π is the probability of an O-ring failure.

5.a. Model estimate

```
chal.glm2 <- glm(formula=failure ~ Temp, family=binomial(link="logit"), data=challenger)
summary(chal.glm2)
```

```
##
## Call:
## glm(formula = failure ~ Temp, family = binomial(link = "logit"),
##     data = challenger)
```

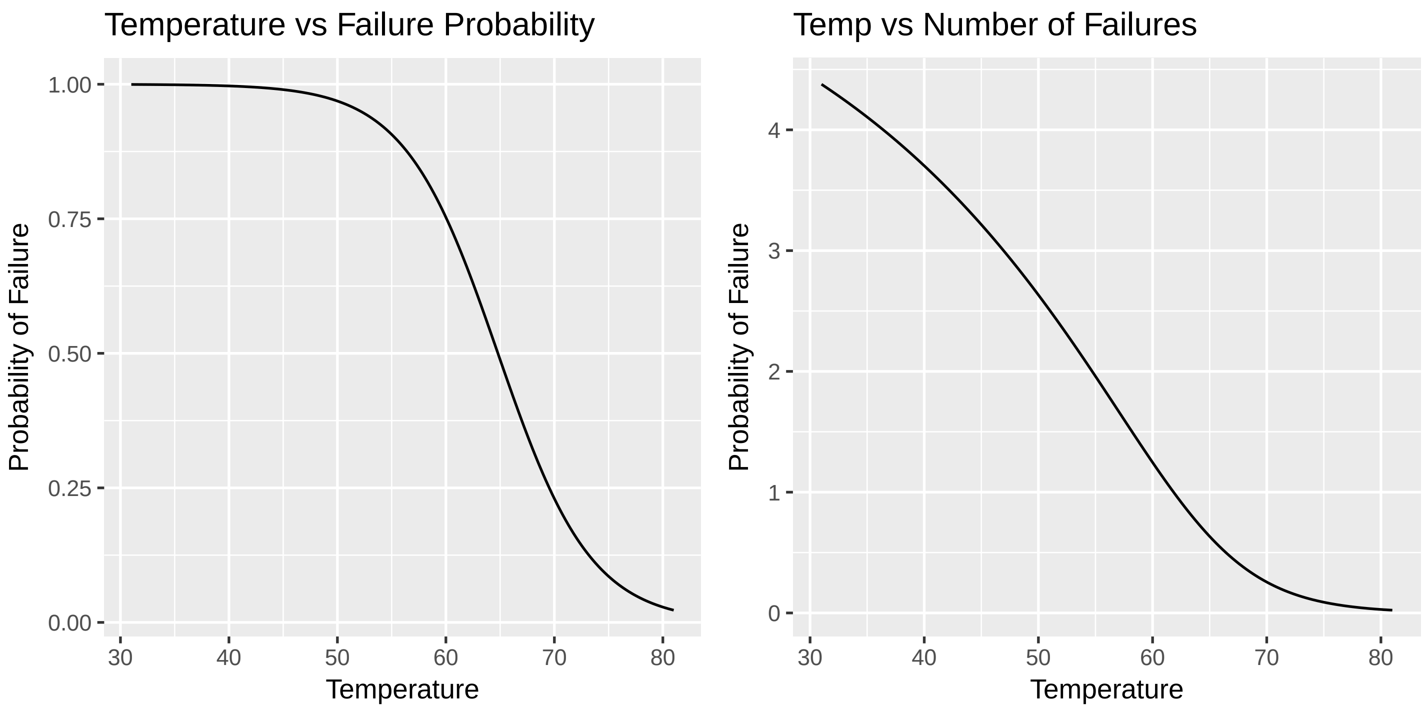
```
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.0611  -0.7613  -0.3783   0.4524   2.2175
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  15.0429     7.3786   2.039   0.0415 *
## Temp         -0.2322     0.1082  -2.145   0.0320 *
## ---
## 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: 20.315  on 21  degrees of freedom
## AIC: 24.315
##
## Number of Fisher Scoring iterations: 5
```

5.b. Model plots

```r
logit_transform <- function(toTransform){
  return(exp(toTransform)/(1+exp(toTransform)))
}

#Probability of Failure
predict_data <- data.frame(Temp=seq(from=31, to=81, by=0.1))
prediction_bool <- predict(chal.glm2, newdata=predict_data, type="link",se=TRUE)
plot5.b.1 <- ggplot() + geom_line(aes(x=predict_data$Temp, y=logit_transform(prediction_bool$fit))) +
             xlab("Temperature") + ylab("Probability of Failure") + ggtitle("Temperature vs Failure Probability")
#Number of Failures
#pff = 1 - (1-p_f)^6, from paper : p_f = 1 - (1-pff)^(1/6)
prediction_bool_OneFail <- (1 - (1 - logit_transform(prediction_bool$fit))^(1/6)) * 6
plot5.b.2 <- ggplot() + geom_line(aes(x=predict_data$Temp, y=prediction_bool_OneFail)) + xlab("Temperature") +
             ylab("Probability of Failure") + ggtitle("Temp vs Number of Failures")

grid.arrange(plot5.b.1, plot5.b.2, ncol=2)
```



5.c. Confidence interval

The equation for a Wald confidence interval is:

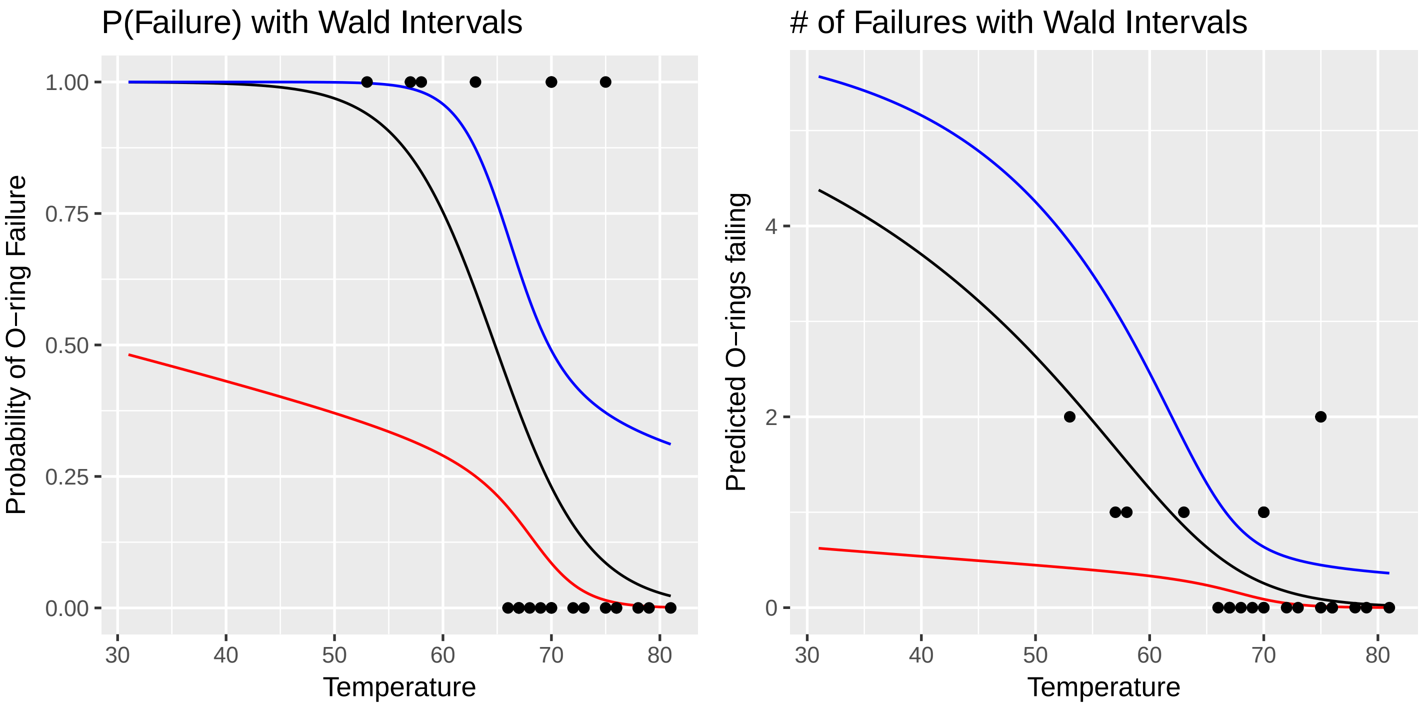
$$\hat{\pi} - Z_{1-\alpha/2}\sqrt{\frac{\hat{\pi}(1-\hat{\pi})}{n}} < \pi < \hat{\pi} + Z_{1-\alpha/2}\sqrt{\frac{\hat{\pi}(1-\hat{\pi})}{n}}$$

```
#Probability of Failure
LB <- as.numeric(prediction_bool$fit) + qnorm(p=.025) * prediction_bool$se
UB <- as.numeric(prediction_bool$fit) + qnorm(p=.975) * prediction_bool$se
bool_Wald <- data.frame(Temp=predict_data$Temp, prediction=logit_transform(prediction_bool$fit),
                        LB=logit_transform(LB), UB=logit_transform(UB))

plot1 <- ggplot() + geom_line(aes(x=bool_Wald$Temp,y=bool_Wald$prediction)) +
         geom_line(aes(x=bool_Wald$Temp,y=bool_Wald$LB),color="red") +
         geom_line(aes(x=bool_Wald$Temp,y=bool_Wald$UB),color="blue") +
         geom_point(aes(x=challenger$Temp,y=challenger$failure)) +
         xlab("Temperature") + ylab("Probability of O-ring Failure") + ggtitle("P(Failure) with Wald Intervals")
#Number of Failures

LB_OneFail <- (1 - (1 - logit_transform(LB))^(1/6)) * 6
UB_OneFail <- (1 - (1 - logit_transform(UB))^(1/6)) * 6
NumFail_DF <- data.frame(Temp=predict_data$Temp, prediction=prediction_bool_OneFail,
                           LB=LB_OneFail, UB=UB_OneFail)
plot2 <- ggplot() + geom_line(aes(x=NumFail_DF$Temp, y=NumFail_DF$prediction)) +
         geom_line(aes(x=NumFail_DF$Temp,y=NumFail_DF$LB), color="red") +
         geom_line(aes(x=NumFail_DF$Temp,y=NumFail_DF$UB), color="blue") +
         geom_point(aes(x=challenger$Temp,y=challenger$O.ring)) +
         xlab("Temperature") + ylab("Predicted O-rings failing") +
         ggtitle("# of Failures with Wald Intervals")

grid.arrange(plot1, plot2, ncol=2)
```



The bands are wider at a lower temperature because we have progressively less data at these points. However, the upper bound is constrained to 1 through the logit transformation: so initially the bands widen with increased temperature as the lower bound approaches zero and the upper bound remains fixed to one.

5.d. Model estimate for 1986 Challenger launch

```
lp.hat <- predict.glm(chal.glm2, newdata=data.frame(Temp=31),
                      type="link", se.fit=TRUE)
data.frame(ProbFailure=round(logit_transform(lp.hat$fit)[1],5),
           UpperBound=round(logit_transform(lp.hat$fit + 1.96 * lp.hat$se.fit), 5),
           LowerBound=round(logit_transform(lp.hat$fit - 1.96 * lp.hat$se.fit), 5))
```

```
##   ProbFailure UpperBound LowerBound
## 1     0.99961          1    0.48157
```

When fitting a logistic regression model we must satisfy the following four assumptions:

1) The dependent variable must by binary. (The conditional distribution of Y must follow a Bernoulli distribution)

```
table(challenger$failure)
```

```
##
##  0  1
## 16  7
```

The table above shows that this is the case with 16 flights without failures and only 7 flights with failures. We have converted from the native integer variable (which could take on three values: 0, 1, and 2) to an indicator for whether or not there were any O-ring failures in the flight.

2) Observations must be independent of one another. The error term must follow an independent and identically distributed random variable.

We can see below that there is no evidence of autocorrelation, however it does seem like we may not have normally distributed error terms. This may require a transformation to resolve.

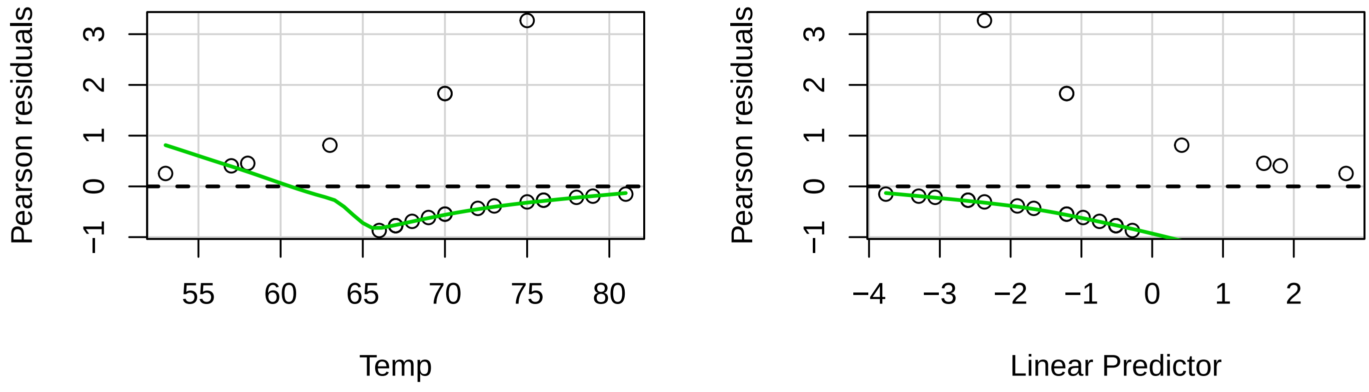
```
# Test for autocorrelation
durbinWatsonTest(chal.glm2)
```

```
##  lag Autocorrelation D-W Statistic p-value
##    1   -0.0008262438      1.927747   0.824
##  Alternative hypothesis: rho != 0
```

```
# Shapiro Wilkes
shapiro.test(chal.glm2$residuals)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  chal.glm2$residuals
## W = 0.62149, p-value = 1.567e-06
```

```
# Plot residuals
residualPlots(chal.glm2)
```



```
##      Test stat Pr(>|t|)
## Temp     0.926    0.336
```

3) No perfect collinearity

This is satisfied by the fact that we do not have multiple independent variables.

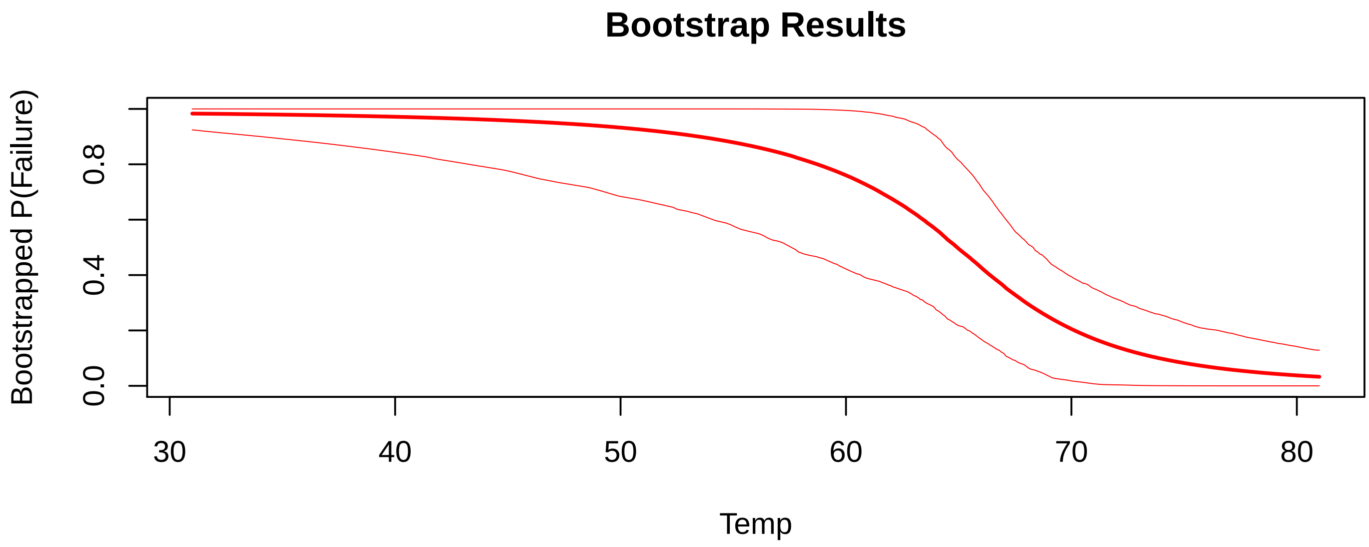
4) Linearity assumption. Independent variables must be linear with log odds ratio

The equation we have written can be specified as $log\ odds(failure) = \beta_0 + \beta_1 Temp + \epsilon$ and is by definition a linear relationship between the independent variables and the log odds of failure

5.e. Parametric bootstrap confidence interval

From secondary sources (NOAA: http://www.cpc.ncep.noaa.gov/products/precip/CWlink/pdf/southeast_temp.shtml) on the nearest weather station, Orlando, the temperature is distributed approximately skew normal. For simplicity, we will generate bootstrap replicate by estimating (non-skew) normal distribution over temperature instead. We will take the random temperature inputs and use our model to generate the fitted values for each temperature sample. We then use the `rbinom` function to simulate the number of failures experienced for each temperature sample given the fitted probability. We then transform these results back into a binary variable indicating presence of at least one O-ring failure. Using these simulated failures, we fit a new model which estimates the simulated output given the simulated input temperatures. The final is then use to get predicted values for all temperatures between 31 and 81 degrees. We use the 10th and 90th percentiles to define the upper and lower bounds of our parametric bootstrap.

```
temp.mean <- mean(challenger$Temp)
temp.sd <- sd(challenger$Temp)
replicate.size <- 23
bootstrap.estimate <- c()
bootstrap.n <- 1000
for (bootstrap.replicate in 1:bootstrap.n)
{
  replicate.temp <- rnorm(replicate.size, temp.mean, temp.sd)
  replicate.failure.p <- predict.glm(chal.glm2, newdata=data.frame(Temp=replicate.temp), type="response")
  replicate.failure <- sapply((1-(1-replicate.failure.p) ** (1/6)), function(p) rbinom(1,6,p) > 0)
  challenger.boot <- glm(formula=failure~Temp, data=data.frame(failure=replicate.failure, Temp=replicate.temp),
                    family='binomial')
  bootstrap.estimate <- c(bootstrap.estimate, predict.glm(challenger.boot, newdata=predict_data, type="response"))
}
bootstrap.estimate <- matrix(bootstrap.estimate,length(predict_data$Temp),bootstrap.n)
bootstrap.ci <- sapply(1:length(predict_data$Temp), function(t) quantile(bootstrap.estimate[t,],c(0.05,0.95)))
bootstrap.mean <- sapply(1:length(predict_data$Temp), function(t) mean(bootstrap.estimate[t,]))
plot(predict_data$Temp, bootstrap.mean, ylim=range(c(bootstrap.ci)),
     xlab="Temp", ylab="Bootstrapped P(Failure)", type='l', col='red', lwd=2, main='Bootstrap Results')
lines(predict_data$Temp, bootstrap.ci[1,], col='red', lwd=0.5)
lines(predict_data$Temp, bootstrap.ci[2,], col='red', lwd=0.5)
```



```
bootstrap.estimates <- matrix(NA, 3, 2, dimnames=list(c("90% CI lower bound","mean","90% CI upper bound"),
                                                      c("Temp=31","Temp=72")))
bootstrap.estimates[2,] <- bootstrap.mean[predict_data$Temp %in% c(31,72)]
bootstrap.estimates[c(1,3),] <- bootstrap.ci[,predict_data$Temp %in% c(31,72)]
bootstrap.estimates
```

```
##                       Temp=31    Temp=72
## 90% CI lower bound 0.9245076 0.00354936
## mean               0.9831552 0.14069923
## 90% CI upper bound 1.0000000 0.31351214
```

5.f. Quadratic effect of temperature

The Shapiro test returns as highly statistically significant (in the assumption testing section), suggested that error residuals are not normal. This suggests that variable transformations or a new model specification would be useful. However, after updating the model specification to have a quadratic term, we see that neither of the variable coefficients are statistically significant. While the model with a quadratic term has a better fit, the model without the quadratic term has a lower AIC score.

We plot our predicted probabilities from both models (with and without the temperature quadratic term). We see that the probabilities output by the model including the quadratic term are visually less accurate than those of the model without the additional term. The reduction in residual deviance seems to mostly come from bringing the predicted probability closer to 0.5, thus reducing the penalty of an incorrect guess in either direction.

If you exend the plot out further, you notice that the positive coeffiecient on the quadratic term actually leads to increasing probabilities of failure at temperatures above 80°. This does not agree with our understanding of the O-rings which fail to seal at lower temperatures and therefore the model is not a good representation of reality.

```
chal.glmQuad <- glm(formula=failure ~ Temp + I(Temp^2), data=challenger)

anova(chal.glm2, chal.glmQuad, test="LRT")
```

```
## Analysis of Deviance Table
##
## Model 1: failure ~ Temp
## Model 2: failure ~ Temp + I(Temp^2)
##   Resid. Df Resid. Dev Df Deviance  Pr(>Chi)
## 1        21    20.3152
## 2        20     3.1279  1   17.187 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

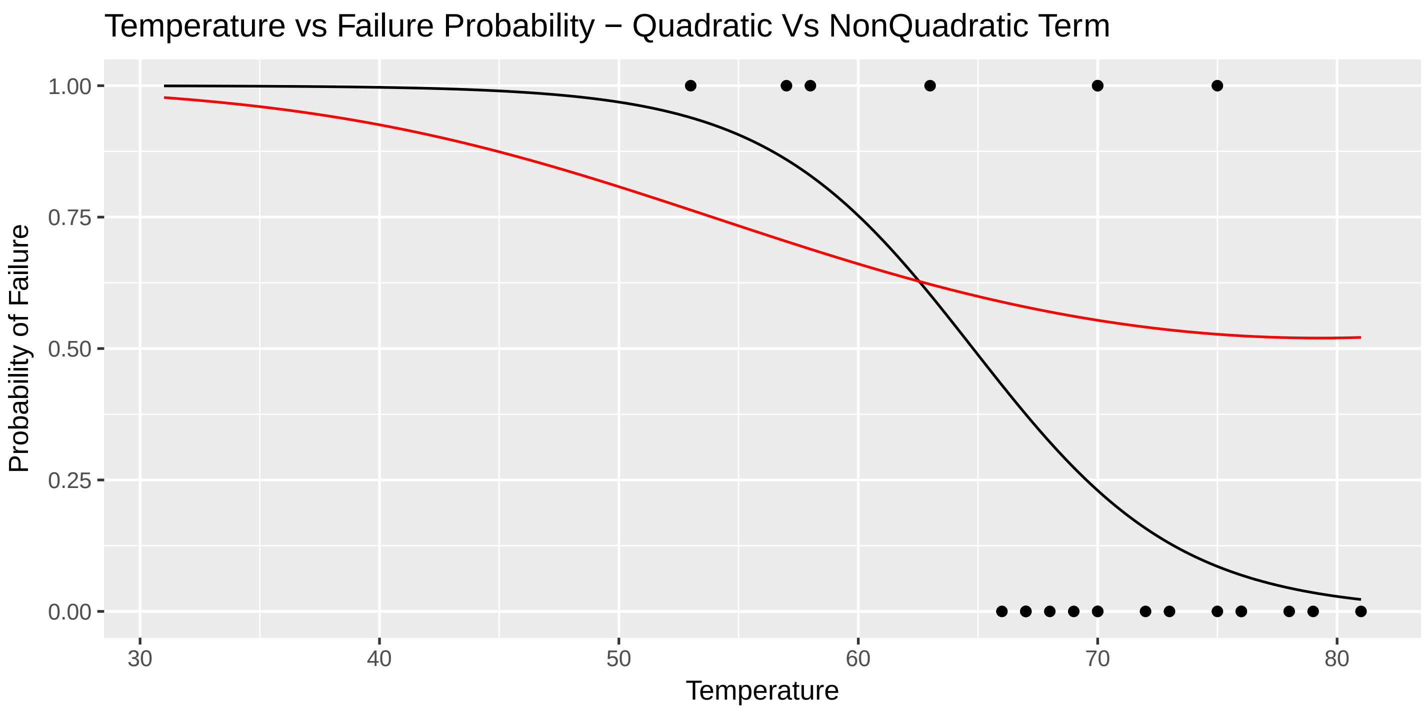
```
Anova(chal.glmQuad)
```

```
## Analysis of Deviance Table (Type II tests)
##
## Response: failure
##           LR Chisq Df Pr(>Chisq)
## Temp        1.8522  1     0.1735
## I(Temp^2)   1.3472  1     0.2458
```

```
data.frame(TemperatureOnly_AIC=AIC(chal.glm2), QuadraticTerm_AIC=AIC(chal.glmQuad))
```

```
##   TemperatureOnly_AIC QuadraticTerm_AIC
## 1            24.31519          27.38297
```

```
ggplot() + geom_line(aes(x=predict_data$Temp, y=logit_transform(prediction_bool$fit))) +
           xlab("Temperature") + ylab("Probability of Failure") +
           ggtitle("Temperature vs Failure Probability - Quadratic Vs NonQuadratic Term") +
           geom_line(aes(
             x=predict_data$Temp,
             y=logit_transform(predict(chal.glmQuad, newdata=predict_data, type="link"))), color="red") +
           geom_point(aes(x=challenger$Temp,y=challenger$failure))
```



Summary

Odds and probability of failure

```r
chal.glm2$coefficients
```

```
## (Intercept)        Temp 
##  15.0429016  -0.2321627
```

According to the model estimate, the odds of failure changes by 0.7928 for every 1°F increase in temperature, and it halves for every 2.9856°F increase in temperature.

```r
temp.OR <- exp(chal.glm2$coefficients['Temp']);      names(temp.OR) <- "Temp OR"
temp.HL <- log(.5) / chal.glm2$coefficients['Temp']; names(temp.HL) <- "Temp HL"
c(temp.OR, temp.HL)
```

```
##   Temp OR   Temp HL 
## 0.7928171 2.9856090
```

In terms of probability,

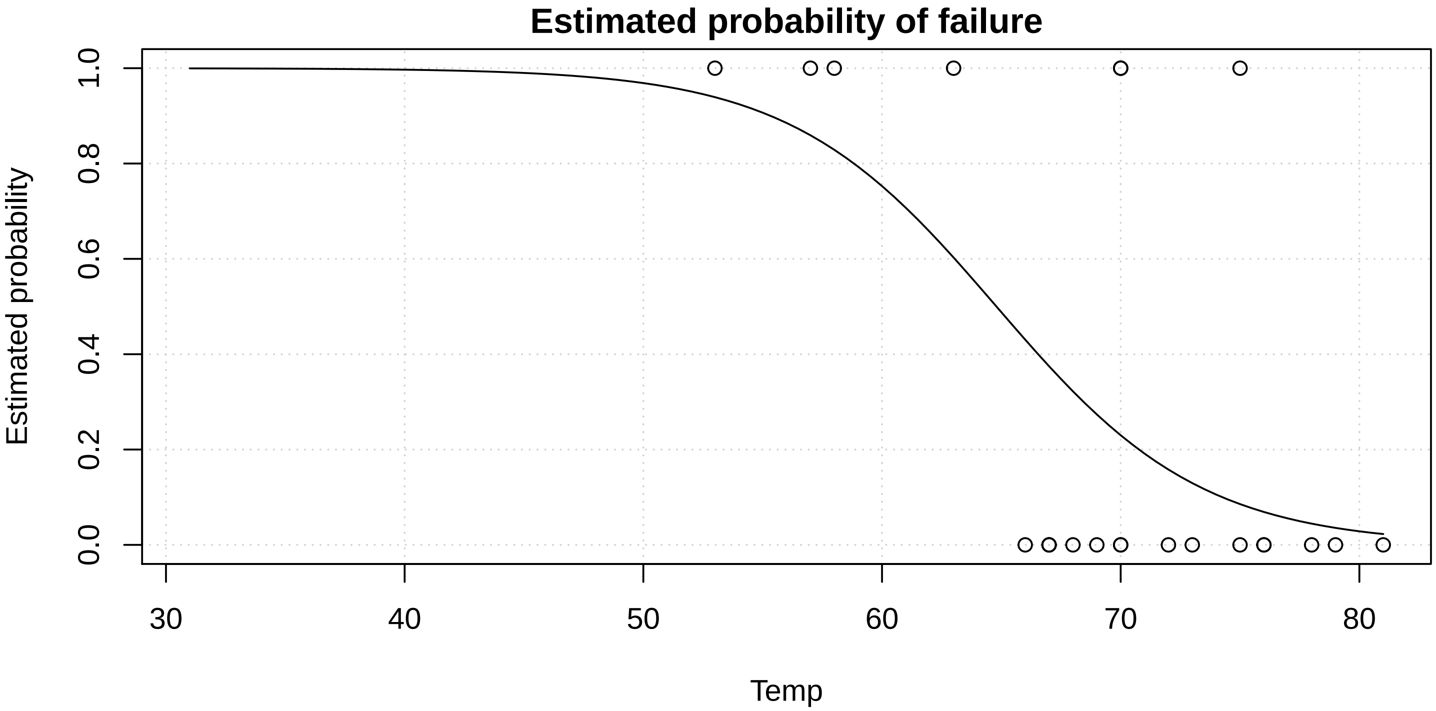
$$\hat{\pi} = \frac{\exp\left(15.0429016 - 0.2321627\text{Temp}\right)}{1 + \exp\left(15.0429016 - 0.2321627\text{Temp}\right)}$$

```r
temp.probs <- t(sapply(c(t(aggregate(Temp ~ failure, data=challenger,
                                     FUN=function(x) c(U=max(x), L=min(x)))$Temp), 31),
       function(x) c(x, exp(chal.glm2$coefficients[1] + chal.glm2$coefficients[2] * x) /
                       (1 + exp(chal.glm2$coefficients[1] + chal.glm2$coefficients[2] * x)))))
colnames(temp.probs) <- c("Temp", "Pred. prob. of failure")
rownames(temp.probs) <- c("Max. dataset temperature (failure=0)", "Min. dataset temperature (failure=0)",
                          "Max. dataset temperature (failure=1)", "Min. dataset temperature (failure=1)",
                          "Challenger 1986 launch (failure=1)"); temp.probs
```

```
##                                      Temp Pred. prob. of failure
## Max. dataset temperature (failure=0)   81             0.02270329
## Min. dataset temperature (failure=0)   66             0.43049313
## Max. dataset temperature (failure=1)   75             0.08554356
## Min. dataset temperature (failure=1)   53             0.93924781
## Challenger 1986 launch (failure=1)     31             0.99960878
```

Main effect of final model

```
par(mar=c(4,4,1.5,0) + 0.1)
curve(expr=exp(chal.glm2$coefficients[1] + chal.glm2$coefficients[2]*x) /
           (1 + exp(chal.glm2$coefficients[1] + chal.glm2$coefficients[2]*x)),
      main="Estimated probability of failure", ylim=c(0,1), xlim=range(c(31,challenger$Temp)),
      ylab="Estimated probability", xlab="Temp", panel.first=grid())
points(x=challenger$Temp, y=challenger$failure)
```


Estimated probability of failure
Estimated probability
Temp