Assignment: Explore regression functions for delta.temp and Year polynomials up to full 2nd order for Wildfire.Count, Drought.Count, All.Disasters.Count, and Severe.Storm.Count of the NOAAGISSWD dataset. Use Logistic regression or linear regression as appropriate using lmboot or logitboot, and identify plausible models which with 95% confidence the coefficients are not 0. Use AIC or PRESS to decide between models.

Choosing Regression Models

For this assignment, I first analyzed the four different variables of the NOAAGISSWD dataset (Wildfire.Count, Drought.Count, All.Disasters.Count, Severe.Storm.Count) and then applied the correct regression models based on my findings.

```
── Variable type: numeric ──────────────────────────────────────────────────────────────────────────────────────────────
   skim_variable          n_missing complete_rate     mean     sd      p0     p25      p50    p75    p100 hist
 1 Year                           0             1 2002.    12.8   1980    1991.   2002.    2012.  2023  ▇▇▇▇▇
 2 Drought.Count                  0             1    0.705  0.462    0       0       1        1      1   ▇▁▁▁▇
 3 Flooding.Count                 0             1    1      1.10     0       0       1        2      4   ▇▃▂▁▁
 4 Freeze.Count                   0             1    0.205  0.462    0       0       0        0      2   ▇▁▂▁▁
 5 Severe.Storm.Count             0             1    4.23   4.36     0       1       2        7     19   ▇▃▂▁▁
 6 Tropical.Cyclone.Count         0             1    1.41   1.56     0       0       1        2      7   ▇▂▁▁▁
 7 Wildfire.Count                 0             1    0.5    0.506    0       0       0.5      1      1   ▇▁▁▁▇
 8 Winter.Storm.Count             0             1    0.5    0.665    0       0       0        1      2   ▇▁▆▁▁
 9 All.Disasters.Count            0             1    8.55   6.22     0       4.75    6.5     11     28   ▇▆▃▂▁
10 delta.temp                     0             1    0.556  0.268    0.12    0.33    0.545    0.7    1.17 ▇▇▆▃▂
```

```
wildfire_model <- glm(Wildfire.Count ~ delta.temp + Year + delta.temp:Year + I(delta.temp^2) + I(Year^2),
                      data = weather_data, family = binomial(link = "logit"))

drought_model <- glm(Drought.Count ~ delta.temp + Year + delta.temp:Year + I(delta.temp^2) + I(Year^2),
                     data = weather_data, family = binomial(link = "logit"))

all_disasters_model <- lm(All.Disasters.Count ~ delta.temp + Year + delta.temp:Year + I(delta.temp^2) + I(Year^2),
                          data = weather_data)

severe_storm_model <- lm(Severe.Storm.Count ~ delta.temp + Year + delta.temp:Year + I(delta.temp^2) + I(Year^2),
                         data = weather_data)
```

The Wildfire and Drought variables used the logistic regression model. This is because these variables only contain integer values between 0 and 1, as seen above, and so they can be classified as binary data. By looking at the summarized histogram of each of the two variables, it can be seen that there are spikes at 0 and 1 (the very beginning and end of the graph). This means that that data does not follow a normal distribution, and thus linear regression can not be applied as there is no variance. Thus, the glm model with the binomial family and the logit link is suitable here for showing the logistic regression.

The All Disasters and Severe Storm variables, on the other hand, used the linear regression model. This is because these variables contain a range of values (0 to 28 for All Disasters and 0 to 19 for Severe Storm). The summarized histograms show that these two variables follow some sort of skewed distribution, but it is clear that it is not a bimodal distribution and so the data is not binary. Also, the variables are continuous, and so linear regression can be used. Thus, the lm function is suitable here for showing the linear regression.

Both of the regression functions included delta.temp and Year, as well as their interaction term (delta.temp:Year) and polynomial terms up to the full 2nd order as required by the assignment (I(delta.temp^2) and I(Year^2)). This allows for exploring the relationship between the response variables and the predictors, including non-linear interactions.

```
> summary(wildfire_model)

Call:
glm(formula = Wildfire.Count ~ delta.temp + Year + delta.temp:Year +
    I(delta.temp^2) + I(Year^2), family = binomial(link = "logit"),
    data = weather_data)

Coefficients:
                  Estimate Std. Error z value Pr(>|z|)
(Intercept)      7.634e+04  8.117e+04   0.941    0.347
delta.temp       4.724e+03  4.300e+03   1.098    0.272
Year            -7.769e+01  8.227e+01  -0.944    0.345
I(delta.temp^2)  6.864e+01  5.750e+01   1.194    0.233
I(Year^2)        1.976e-02  2.084e-02   0.948    0.343
delta.temp:Year -2.397e+00  2.179e+00  -1.100    0.271

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 60.997  on 43  degrees of freedom
Residual deviance: 41.831  on 38  degrees of freedom
AIC: 53.831

Number of Fisher Scoring iterations: 6
```

```
> summary(drought_model)

Call:
glm(formula = Drought.Count ~ delta.temp + Year + delta.temp:Year +
    I(delta.temp^2) + I(Year^2), family = binomial(link = "logit"),
    data = weather_data)

Coefficients:
                  Estimate Std. Error z value Pr(>|z|)
(Intercept)      8.932e+04  7.468e+04   1.196    0.232
delta.temp       4.114e+03  3.760e+03   1.094    0.274
Year            -9.049e+01  7.567e+01  -1.196    0.232
I(delta.temp^2)  4.313e+01  4.699e+01   0.918    0.359
I(Year^2)        2.292e-02  1.917e-02   1.196    0.232
delta.temp:Year -2.080e+00  1.905e+00  -1.092    0.275

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 53.413  on 43  degrees of freedom
Residual deviance: 44.580  on 38  degrees of freedom
AIC: 56.58

Number of Fisher Scoring iterations: 5
```

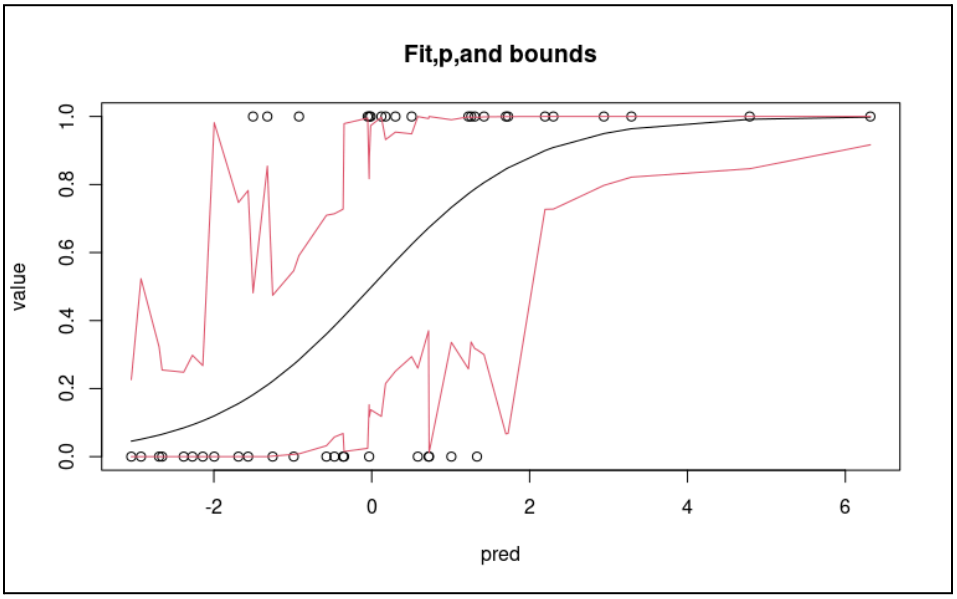
```
summary(all_disasters_model)

all:
lm(formula = All.Disasters.Count ~ delta.temp + Year + delta.temp:Year +
   I(delta.temp^2) + I(Year^2), data = weather_data)

esiduals:
   Min     1Q  Median     3Q    Max
3.7011 -1.9905 -0.5961  1.2394  5.8747

oefficients:
                  Estimate Std. Error t value Pr(>|t|)
Intercept)      2.188e+05  6.920e+04   3.162  0.00307 **
elta.temp       9.667e+03  3.503e+03   2.760  0.00885 **
ear            -2.218e+02  7.011e+01  -3.163  0.00306 **
(delta.temp^2)  1.386e+02  4.397e+01   3.151  0.00317 **
(Year^2)        5.620e-02  1.776e-02   3.164  0.00306 **
elta.temp:Year -4.908e+00  1.774e+00  -2.766  0.00871 **
--
ignif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

esidual standard error: 2.479 on 38 degrees of freedom
ultiple R-squared:  0.8597,    Adjusted R-squared:  0.8413
-statistic: 46.58 on 5 and 38 DF,  p-value: 3.433e-15
```

```
> summary(severe_storm_model)

Call:
lm(formula = Severe.Storm.Count ~ delta.temp + Year + delta.temp:Year +
    I(delta.temp^2) + I(Year^2), data = weather_data)

Residuals:
    Min      1Q  Median      3Q     Max
-2.8189 -1.0013  0.0217  0.9033  3.4247

Coefficients:
                  Estimate Std. Error t value Pr(>|t|)
(Intercept)      1.302e+05  4.353e+04   2.991  0.00486 **
delta.temp       5.147e+03  2.203e+03   2.336  0.02488 *
Year            -1.319e+02  4.411e+01  -2.989  0.00488 **
I(delta.temp^2)  7.484e+01  2.766e+01   2.706  0.01015 *
I(Year^2)        3.338e-02  1.117e-02   2.988  0.00490 **
delta.temp:Year -2.614e+00  1.116e+00  -2.342  0.02453 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.56 on 38 degrees of freedom
Multiple R-squared:  0.8867,    Adjusted R-squared:  0.8718
F-statistic: 59.48 on 5 and 38 DF,  p-value: < 2.2e-16
```

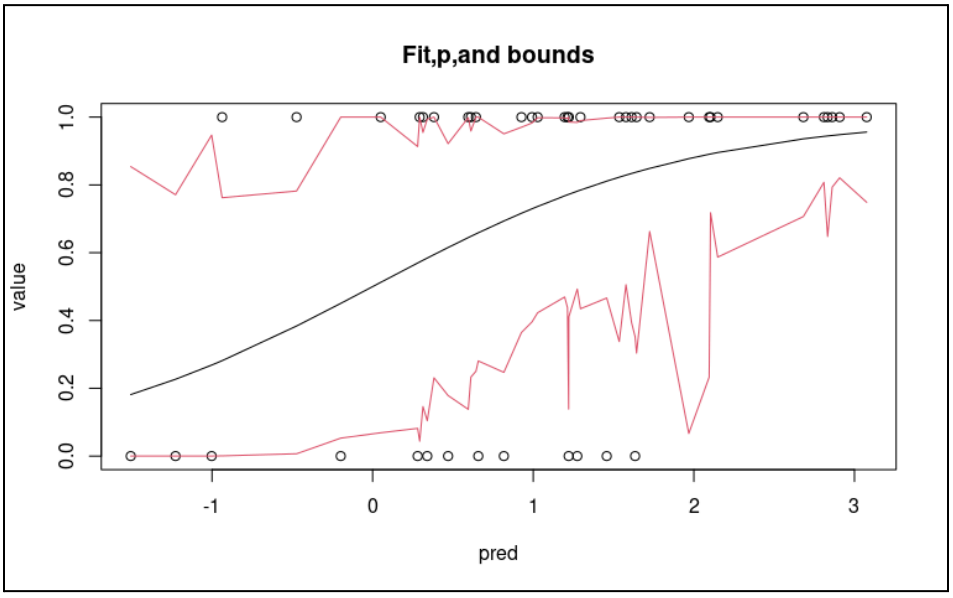
Using lmboot and logitboot Functions

I used logistic regression (lmboot) or linear regression (logitboot) as appropriate to identify models which with 95% confidence the coefficients are not 0.

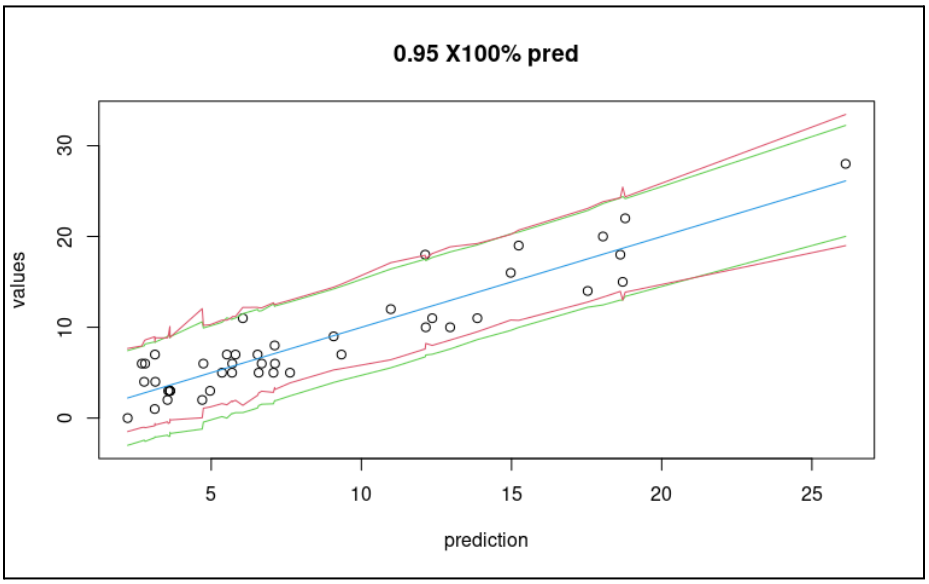
```
wildfire_boot <- logitboot(wildfire_model, DF = weather_data, nboot =
10000, alpha = 0.05)
```



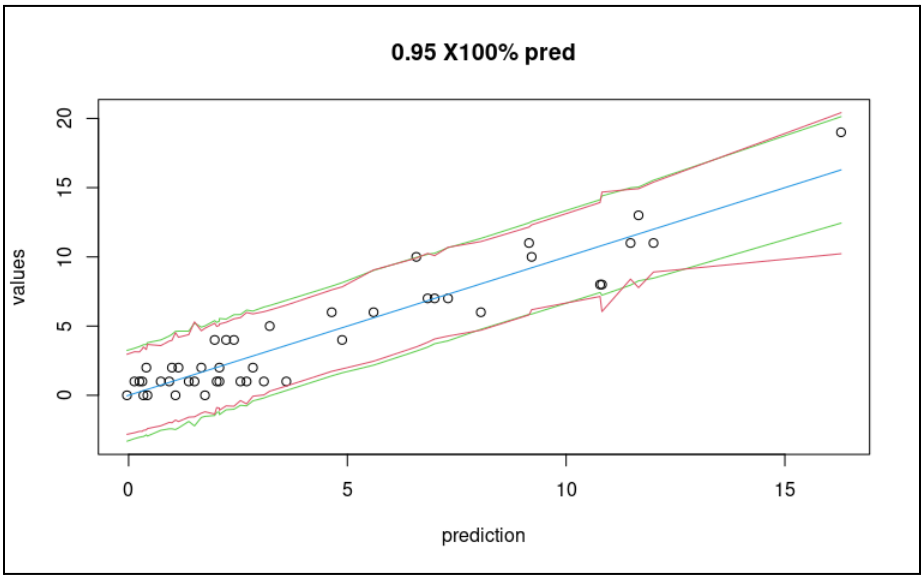

```
drought_boot <- logitboot(drought_model, DF = weather_data, nboot =
10000, alpha = 0.05)
```



```
all_disasters_boot <- lmboot(all_disasters_model, DF = weather_data,
nboot = 10000, alpha = 0.05)
```




```
severe_storm_boot <- lmboot(severe_storm_model, DF = weather_data,
nboot = 10000, alpha = 0.05)
```



The code and resulting output plots above show how I was able to apply the lmboot and logitboot functions to the dataset variables. I continued to use the four model variables I created in the beginning as these are the variables that hold the correct models applied on each of the dataset's variables. The plots display green lines which represent the 0.95 prediction interval bounds, and red lines which represent the fitted lines. This allows for a visual analysis of the models' predictive performance.

The first and second plots, which are wildfire_boot and drought_boot, are diagnostic plots for the linear regression models fit on each of the variables. The plots display the fitted values, residuals, and prediction bounds. These plots help analyze the variance of residuals and the presence of outliers.

The third and fourth plots, which are the all_disasters_boot and severe_storm_boot plots, show scatter plots that compare the predicted values against the actual values for each of their respective models.

For all of the models, the coefficients and confidence intervals are provided for the intercept, delta.temp, Year, interaction term, and the quadratic terms, as seen below.

```
> print(wildfire_boot$coef)
    (Intercept)      delta.temp            Year I(delta.temp^2)       I(Year^2) delta.temp:Year
   7.634483e+04    4.723529e+03   -7.768639e+01    6.863539e+01    1.976133e-02   -2.396904e+00
> print(drought_boot$coef)
    (Intercept)      delta.temp            Year I(delta.temp^2)       I(Year^2) delta.temp:Year
   8.931845e+04    4.113925e+03   -9.048894e+01    4.312665e+01    2.291802e-02   -2.079660e+00
> print(all_disasters_boot$coef)
    (Intercept)      delta.temp            Year I(delta.temp^2)       I(Year^2) delta.temp:Year
   2.188306e+05    9.667064e+03   -2.217899e+02    1.385640e+02    5.619854e-02   -4.908488e+00
> print(severe_storm_boot$coef)
    (Intercept)      delta.temp            Year I(delta.temp^2)       I(Year^2) delta.temp:Year
   1.302037e+05    5.146517e+03   -1.318537e+02    7.484359e+01    3.338186e-02   -2.614145e+00
```

Here are all of the coefficients from fitting each of the regression models. The bootstrap coefficients and corresponding 95% confidence intervals for each predictor variable in the models are seen.

The way that the bootstrap method works is it resamples the data thousands of times and refits the model on each resampled dataset. This estimates the uncertainty in the coefficients, and is used to identify models where the coefficients are statistically significant at the 95% confidence level. If the 95% confidence interval for a particular coefficient does not contain zero, then the corresponding predictor variable can be considered a possible contributor to the model. To calculate the confidence intervals for each of the variables, I wrote the following code. I was able to view the structure of each of the model variables and extract the confidence intervals for each of them.

```
wildfire_boot_ci <- wildfire_boot$pointwiseCI
drought_boot_ci <- drought_boot$pointwiseCI
all_disasters_boot_ci <- all_disasters_boot$coef.point
severe_storm_boot_ci <- severe_storm_boot$coef.point
```

```
> print(wildfire_boot_ci)
      (Intercept) delta.temp      Year I(delta.temp^2)   I(Year^2) delta.temp:Year
2.5%    -191872.9  -5575.233 -664.5454       -50.83208 -0.04793164      -21.543113
97.5%    653156.1  42473.923  192.3701       628.48329  0.16844999        2.801913
> print(drought_boot_ci)
      (Intercept) delta.temp        Year I(delta.temp^2)   I(Year^2) delta.temp:Year
2.5%    -62468.71   -4370.63 -1847.23236        -66.6652 -0.01612785      -39.618388
97.5%  1827603.36   78356.07    63.49266        844.3201  0.46671496        2.222229
> print(all_disasters_boot_ci)
      (Intercept) delta.temp       Year I(delta.temp^2)  I(Year^2) delta.temp:Year
2.5%     63432.39   1551.622 -422.22064        38.34409 0.01628902      -9.7970253
97.5%   417026.78  19322.948  -64.28183       257.57509 0.10685292      -0.7977287
> print(severe_storm_boot_ci)
      (Intercept) delta.temp       Year I(delta.temp^2)   I(Year^2) delta.temp:Year
2.5%     33884.72  -129.8058 -239.99517        3.383615 0.008559076     -5.29539926
97.5%   236857.89 10432.6155  -34.07573      141.985336 0.060798677      0.06179907
```

Wildfire Model:

- Intercept
  - CI from -191872 to 653156
  - High uncertainty
- delta.temp
  - CI from -5575 to 42473
  - Suggests significant positive effect
- Year
  - CI from -664 to 192.3701
  - Crosses zero, potentially insignificant
- I(delta.temp^2)
  - CI from -50 to 628
  - Uncertain
- I(Year^2)
  - CI from -0.04793164 to 0.16844999
  - Narrow, closer to zero
- delta.temp:Year
  - CI from -21.5431 to 2.8019
  - Crosses zero, potentially insignificant

Drought Model:

- Intercept
  - CI from -62468 to 1827603
  - High uncertainty
- delta.temp
  - CI from -4370 to 78356
  - Possible positive effect
- Year
  - CI from -1847 to 63
  - Crosses zero, potentially insignificant.
- I(delta.temp^2) and I(Year^2)

    - Wide ranges, crosses zero, potentially insignificant.
- delta.temp:Year
    - CI from -39.6183 to 2.22229
    - Crosses zero

<u>All Disasters Model:</u>

- Intercept
    - CI from 63432 to 417026
    - More certain
- delta.temp
    - CI from 1551 to 19322
    - Significant positive effect
- Year
    - CI from -422 to -64
    - Crosses zero, potentially insignificant
- I(delta.temp^2) and I(Year^2)
    - Does not cross zero, narrower intervals, potentially significant
- delta.temp:Year
    - CI from -9.7970 to -0.7977
    - Suggests significance

<u>Severe Storm Model:</u>

- Intercept
    - CI from 33884 to 236857
    - More certain
- delta.temp
    - CI from -129 to 10432
    - Crosses zero, potentially insignificant
- Year
    - CI from -239 to -34
    - Negative effect
- I(delta.temp^2) and I(Year^2)
    - Does not cross zero, narrower ranges, potentially significant
- delta.temp:Year
    - CI from -5.2953 to 0.0617
    - Crosses zero, potentially insignificant

<u>Model Coefficients and AIC</u>

Next, I checked the model coefficients and calculated the Akaike information criterion (AIC) for each of the variables. Based on the summaries, the following findings are made:

- For the Wildfire and Drought variables that used logistic regression, none of the coefficients are statistically significant at the 5% level.

- For the All Disasters variable that used linear regression, all coefficients are statistically significant at the 1% level.

- Lastly, for the Severe Storm variable that also used linear regression, all coefficients are statistically significant, with some at the 1% level and some at the 5% level.

```
> AIC(wildfire_model)
[1] 53.83097
> AIC(drought_model)
[1] 56.58047
> AIC(all_disasters_model)
[1] 212.3122
> AIC(severe_storm_model)
[1] 171.5219
```

To decide between models, I decided to use AIC. AIC is a measure of the quality of statistical models for a dataset; lower AIC indicates a better model. Based on the AIC values calculated for these variables, the Wildfire variable has the best fit, followed by the Drought model.

To double-check that using glm for the Wildfire and Drought variables, and using lm for the All Disasters and Severe Storm variables is the most optimal, I also calculated the AIC for each variable using the opposite regression model.

```
wildfire_model_v2 <- lm(Wildfire.Count ~ delta.temp + Year + delta.temp:Year + I(delta.temp^2) + I(Year^2),
                        data = weather_data)

drought_model_v2 <- lm(Drought.Count ~ delta.temp + Year + delta.temp:Year + I(delta.temp^2) + I(Year^2),
                       data = weather_data)

all_disasters_model_v2 <- glm(All.Disasters.Count ~ delta.temp + Year + delta.temp:Year + I(delta.temp^2) + I(Year^2),
                              data = weather_data, family = binomial(link = "logit"))

severe_storm_model_v2 <- glm(Severe.Storm.Count ~ delta.temp + Year + delta.temp:Year + I(delta.temp^2) + I(Year^2),
                             data = weather_data, family = binomial(link = "logit"))
```

```
> AIC(wildfire_model_v2)
[1] 58.20765
> AIC(drought_model_v2)
[1] 60.37368
```

Instantly, I can see that the AIC is higher for using the lm model for both the Wildfire and Drought variables.

Additionally, I am not able to apply the glm model to the All Disasters and Severe Storm variables as these are not binary variables; the values do not fall between 0 and 1 inclusive, as seen below:

```
> all_disasters_model_v2 <- glm(All.Disasters.Count ~ delta.temp + Year + delta.temp:Year + I(delta.temp^2) + I(Year^2),
+                               data = weather_data, family = binomial(link = "logit"))
Error in eval(family$initialize) : y values must be 0 <= y <= 1
> severe_storm_model_v2 <- glm(Severe.Storm.Count ~ delta.temp + Year + delta.temp:Year + I(delta.temp^2) + I(Year^2),
+                              data = weather_data, family = binomial(link = "logit"))
Error in eval(family$initialize) : y values must be 0 <= y <= 1
```

Just to be sure, I decided to convert the values to a binary outcome and apply the model anyways, and check the AIC as well.

```
All_Disasters_Binary <- ifelse(weather_data$All.Disasters.Count > 0, 1, 0)
all_disasters_model_v2 <- glm(All_Disasters_Binary ~ delta.temp + Year + delta.temp:Year + I(delta.temp^2) + I(Year^2),
                              data = weather_data, family = binomial(link = "logit"))

Severe_Storm_Binary <- ifelse(weather_data$Severe.Storm.Count > 0, 1, 0)
severe_storm_model_v2 <- glm(Severe_Storm_Binary ~ delta.temp + Year + delta.temp:Year + I(delta.temp^2) + I(Year^2),
                             data = weather_data, family = binomial(link = "logit"))

AIC(all_disasters_model_v2)
AIC(severe_storm_model_v2)
```

However, when attempting to do this, I get the following warning:

```
Warning messages:
1: glm.fit: algorithm did not converge
2: glm.fit: fitted probabilities numerically 0 or 1 occurred
```

The first warning suggests that the model is overfitting, and there is possible data separation within the variable's values. The second warning suggests that some of the data points are being converted to exactly 0 or 1, which is unusual for a logistic regression model, as values should be between these two extremes rather than just at the extremes. I considered using regularization to help the overfitting issue, but overall, it would make more sense to just apply the lm model over glm; the data for these variables themselves are not suitable to be adapted to a binary format.

Nested Model Comparisons

One more thing I did was a nested model comparison. These comparisons are used to test whether adding or removing variables within the same models incites change. As seen in class, they are tested by using chi-square difference tests, and comparing changes in the model's fit.

To accomplish this, I created a reduced model fit for each variable. These fits only used delta.temp and Year as the predictors. The pchisq() function that finds the chi-square difference takes in the difference in deviances between the reduced model and the full model, so I computed that as well. Since I want to find the p-values, using lower.tail = FALSE tells the function to calculate the upper-tail probability (which is the p-value itself). Ultimately, the output of the pchisq() function is the p-value associated with the difference in deviances. I set the

degrees of freedom to 3 since this is the difference in the number of parameters between the two models.

```
wildfire_model_reduced <- glm(Wildfire.Count ~ delta.temp + Year, data = weather_data, family = binomial(link = "logit"))
dev_diff_wildfire <- deviance(wildfire_model_reduced) - deviance(wildfire_model)
p_value_wildfire <- pchisq(dev_diff_wildfire, df = 3, lower.tail = FALSE)

drought_model_reduced <- glm(Drought.Count ~ delta.temp + Year, data = weather_data, family = binomial(link = "logit"))
dev_diff_drought <- deviance(drought_model_reduced) - deviance(drought_model)
p_value_drought <- pchisq(dev_diff_drought, df = 3, lower.tail = FALSE)

all_disasters_model_reduced <- lm(All.Disasters.Count ~ delta.temp + Year, data = weather_data)
dev_diff_all_disasters <- deviance(all_disasters_model_reduced) - deviance(all_disasters_model)
p_value_all_disasters <- pchisq(dev_diff_all_disasters, df = 3, lower.tail = FALSE)

severe_storm_model_reduced <- lm(Severe.Storm.Count ~ delta.temp + Year, data = weather_data)
dev_diff_severe_storm <- deviance(severe_storm_model_reduced) - deviance(severe_storm_model)
p_value_severe_storm <- pchisq(dev_diff_severe_storm, df = 3, lower.tail = FALSE)
```

I then checked whether the p-value for each variable exceeded a significance level of 0.01 or not. If the p-value < 0.01, that means adding the terms improvies the normal model fit (compared to the reduced model fit). If the p-value > 0.01, that would mean that it did not improve the model fit. My findings were as follows:

- Adding the higher-order terms does not significantly improve the model fit for the Wildfire and Drought variables.
- Adding the higher-order terms does significantly improve the model fits for the All Disasters and Severe Storm variables.

These results suggest that for the Wildfire and Drought variables, the simpler model with just delta.temp and Year as predictors is enough, and adding the higher-order terms does not significantly improve the model. For the All Disasters and Severe Storm variables, including the higher-order terms (the polynomial terms seen in the original model I used) significantly improves the model fit, meaning that these extra terms help show important relationships in the data not fully seen by the simpler model.