By: Anthony Miriani and Wenyi Wang Date: 12/9/2015

**The Purpose:**

The background, the analysis and the prediction comes from a famous American television show named Game of Thrones. The purpose of our project is to predict if a specific character, Jon Snow, will die or not in the next season. We will exclude Jon Snow’s data in the dataset we use. Within a single episode there is one person at least that will be killed. The person who is killed each show seems to be completely random, however, so we will try to predict whether Jon Snow will be killed using predictive modeling.

**Our Dataset:**

We collected most of our data from the website “A Wiki of Ice and Fire” while a small amount of the data was collected from other websites. We also manually calculated several columns so that we could perform logistic regressions more accurately. Our dataset includes 19 variables, 17 of which we feel may be useful for predicting who will be killed. As for the remaining two columns, one is the name of all characters and the other one is who has already been killed versus who is still alive (for training purposes). The Name column could be considered the unique identifier for each row as there are no duplicates in this column. There are 904 rows of data in our dataset. We will build two models, one for the people who have been killed already and one for the people who are still alive.

1 Name: Factor w/ 904 levels – Name of the Character in Game Of Thrones

2 Lines: int – The number of times a character speaks in the show. One line represents when one character speaks until another character speaks or a new scene starts.

3 Allegiances: Factor w/ 12 levels – This is a categorical variable that represents which allegiance, or Kingdom, a character is a part of

4 Death.Year: int – If available, this variable shows when a dead character was killed.

5 Book.of.Death: int – If available, this variable shows in which book the character was killed.

6 Death.Chapter: int – If available, this variable shows in which chapter the character was killed.

7 Book.Intro.Chapter: int – This represents the chapter in the series in which the character is first introduced.

8 Book.Death.Percentage: num - This is a field that was found online to show a different model’s forecasted death %. There is very little data in this field.

9 gender: int – This is a categorical variable to show the gender in which a 1 represents a male while a 0 represents a female.

10 nobility: int – This is a categorical variable to show who is noble versus not. A 1 represents a noble character while a 0 represents a slave.

11 got: int – This is a binomial variable to show whether or not a character is in the book a Game of Thrones. A 1 shows a character being in that novel while a 0 shows that a character was not in it.

12 cok: int - This is a binomial variable to show whether or not a character is in the book a Clash of Kings. A 1 shows a character being in that novel while a 0 shows that a character was not in it.

13 sos: int - This is a binomial variable to show whether or not a character is in the book a Storm of Swords. A 1 shows a character being in that novel while a 0 shows that a character was not in it.

14 ffc: int - This is a binomial variable to show whether or not a character is in the book a Feast of Crows. A 1 shows a character being in that novel while a 0 shows that a character was not in it.

15 dwd: int - This is a binomial variable to show whether or not a character is in the a Dance With Dragons. A 1 shows a character being in that novel while a 0 shows that a character was not in it.

16 age: int – This variable shows the character’s age.

17 death.ratio: int - This variable represents a grouping within the Allegiance field to show which grouping of allegiances have the lowest/highest ratio of characters being killed.

18 deadornot: int – This binomial variable shows whether or not the character is dead or not. A 1 shows a character as being dead while a 0 means the character is still alive.

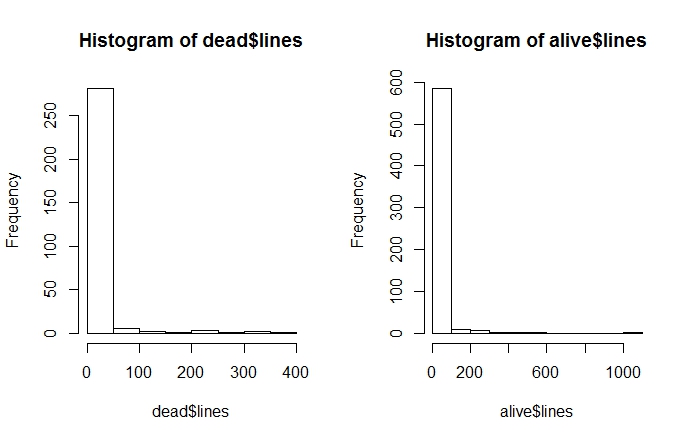
19 affiliation: int – This variable, where available, shows how many alliances to other kingdoms a character has.

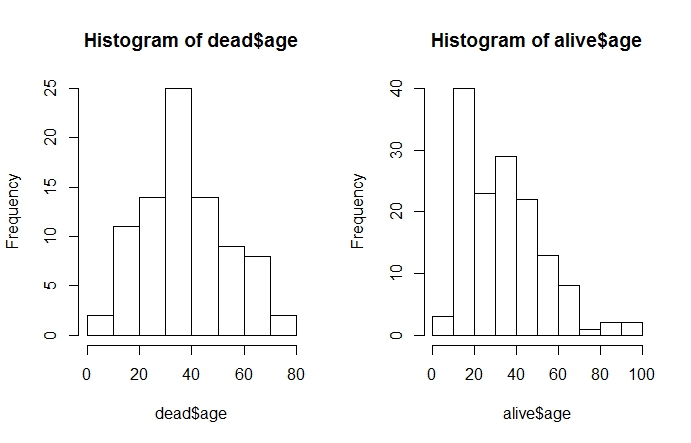
**A new variable “length” is added to present how many chapters each character being alive.**

**Our Method:**

We have learned that in general older males who are slaves are killed off more often than young, noble females. Looking at the gender coefficient we see that males have nearly a 14% higher chance of being killed off than females. Using the summary statistics in R we found that whether or not the character is in the book “A Dance with Dragons” is the top indicator of whether or not they will live going forward. This variable was found to be statistically significant. By just having a character be in this book it actually decreased the chances of being killed by about 37%.

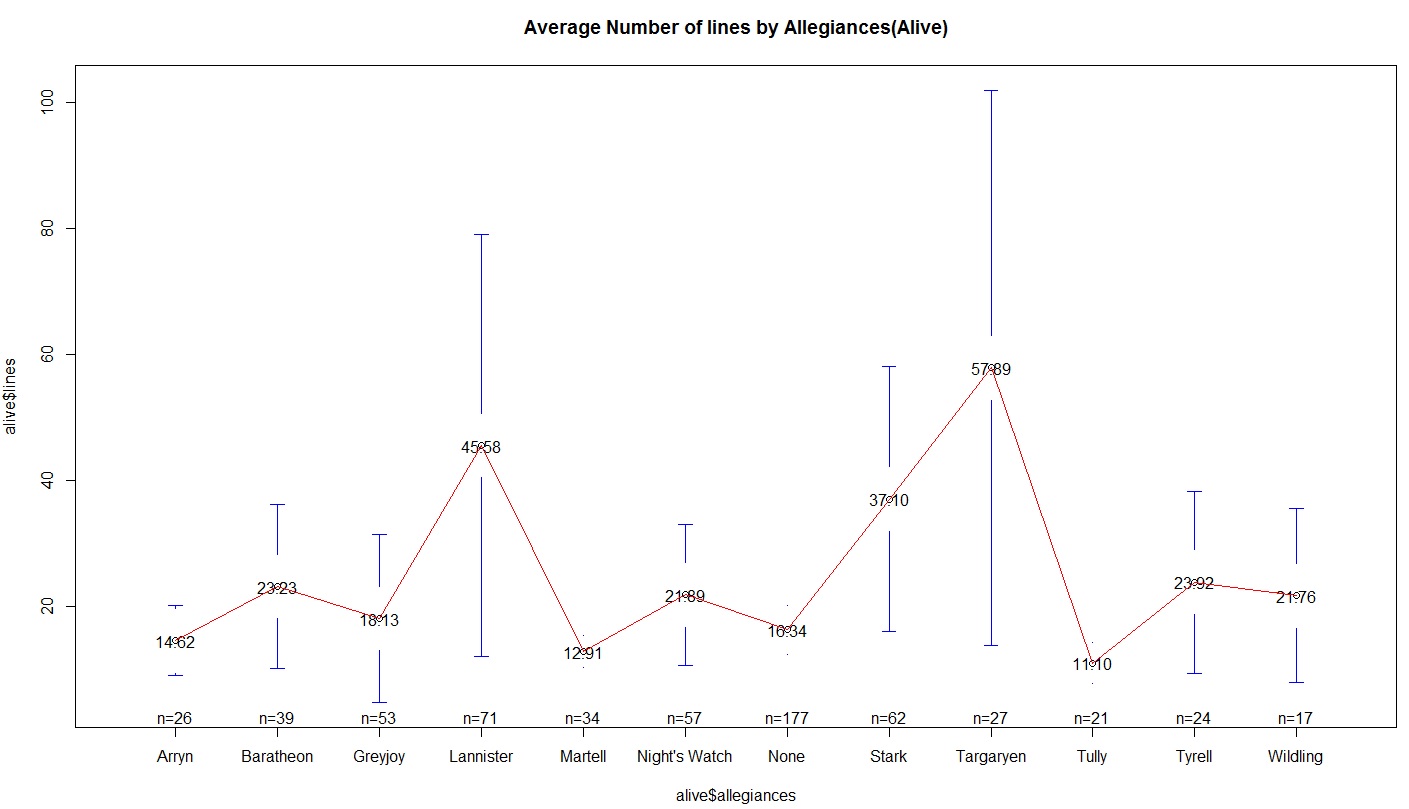
To predict if Jon Snow will be killed or not we used a multiple linear regression model. We analyzed the data by performing Exploratory Data Analysis. The goal of using EDA is to try to see if any trends can be spotted right away. In addition to this we wanted to see which variables are correlated with each other. Doing this allows us to reduce our model to only the necessary variables, effectively eliminating the multicollinearity that may exist otherwise. Within R we are able to do this by looking at the correlation, PCA, rpart, etc. but since our data has so many NA values which cannot be omitted or replaced, we cannot see much from running the correlation functions.

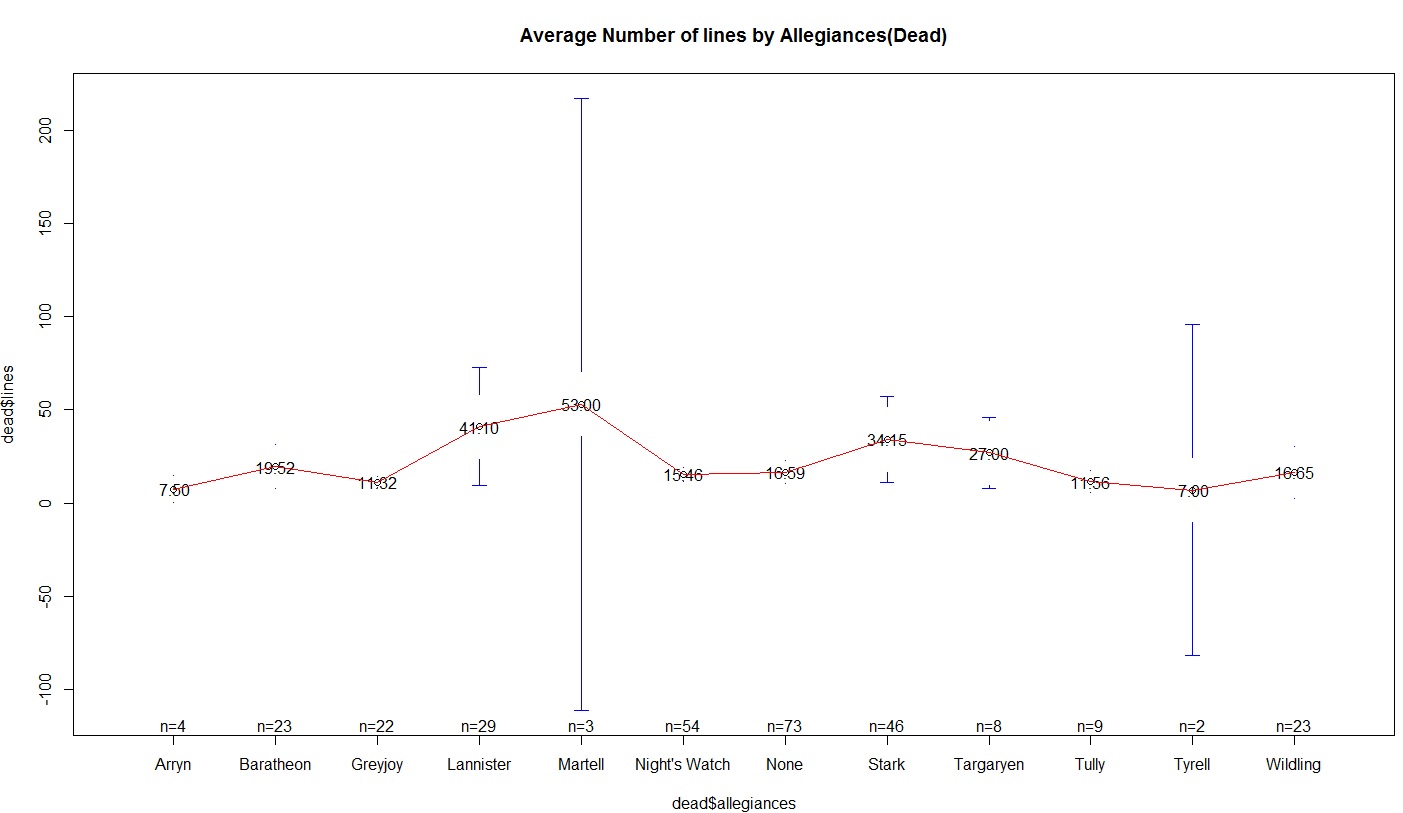
**Using plot and histograms to see relations:****Characters die more often the fewer times they appear.**



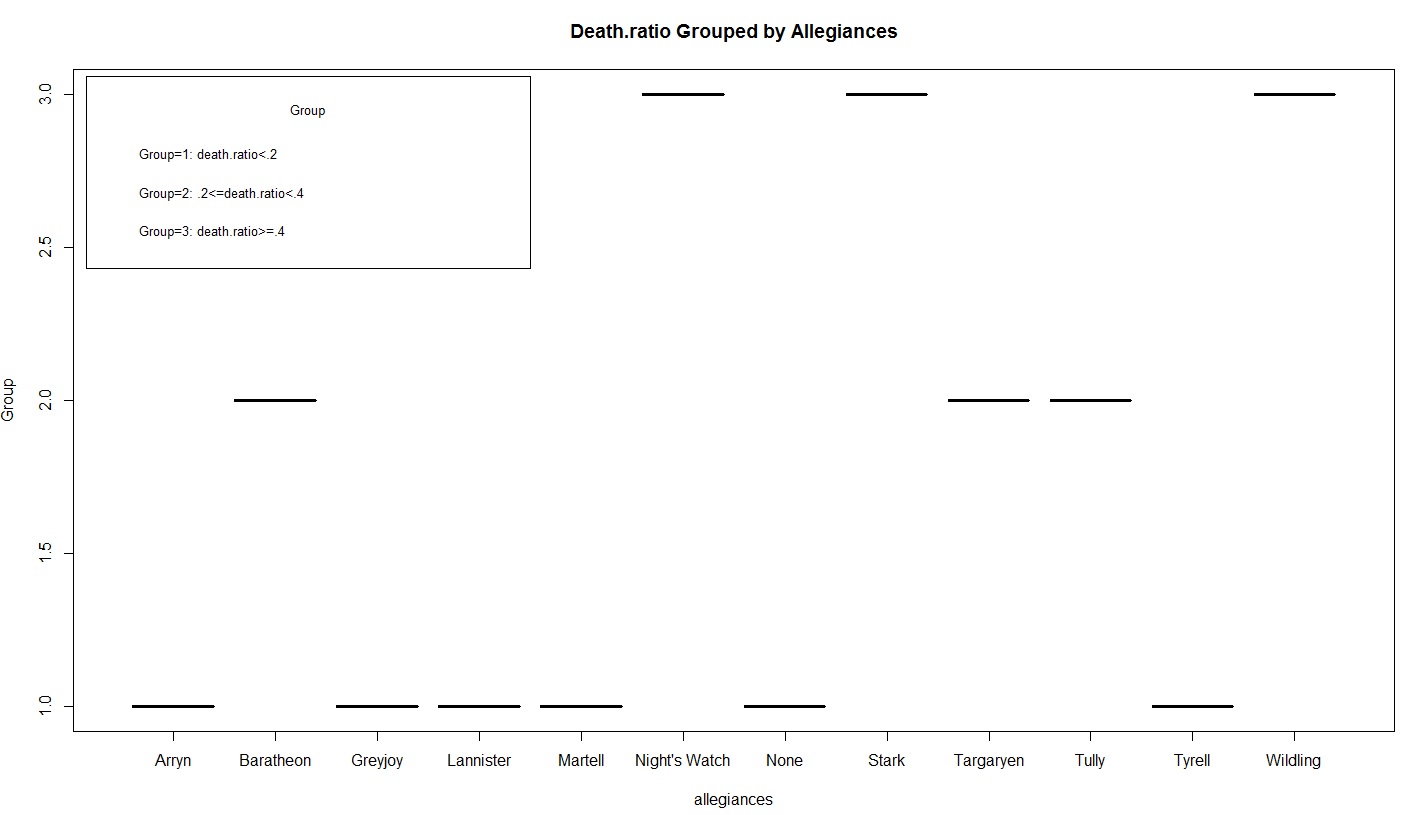
**People who are dead: between 20-40**

**People who are alive: between 10-50**



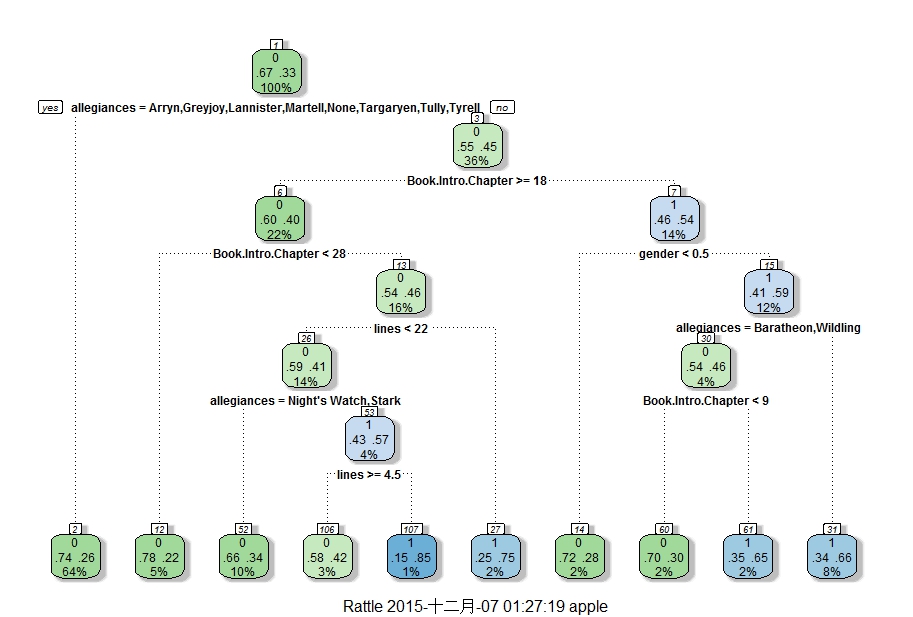


The two plots above show the average number of lines by allegiances. In the same allegiance, characters have more lines if they are alive compared to dead characters.

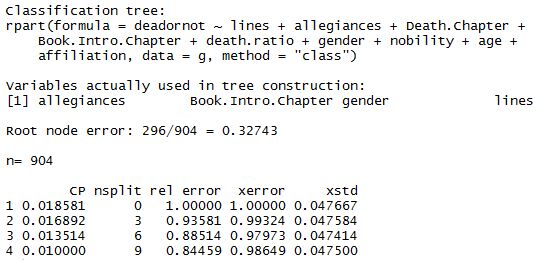


Twelve allegiances are cut into three groups based on the death.ratio. In Group 1, it contains allegiances whose death.ratio is less than 20%. Group 2 includes allegiances whose death.ratio is no smaller than 20% but also less than 40%. Group 3 contains allegiances that have a death.ratio of at least 40%.

From this analysis we found that the characters who are loyal to Night’s Watch, Stark and Wildling have a higher probability of being killed.



From the decision tree, the first split is what is considered to be the most useful split. So we first look into if a character belongs to one of the families (Arryn, Greyjoy, Lannister, Martell, None, Targaryen, Tully, Tyrell) or not. If yes, we go down to the left. 64% of the population resides in node 2. 74% of these characters are alive, while 26% die. If we go down to the right and look into node 3, this indicates 36% of the population does not belong to any of the allegiances above. At the same time, 55% of these characters are alive, while 45% are dead. We can easily find out the second most important split is “Book.Intro.Chapter”. Though this method may not be perfect at selecting which variables are useful to use in a model, it is still very useful to see how characters are split into binomial decisions/variables. Although the technique is limited due to the dataset we have, we will still take the output of the decision tree into consideration. The predictions used in the decision tree will be used in our model.

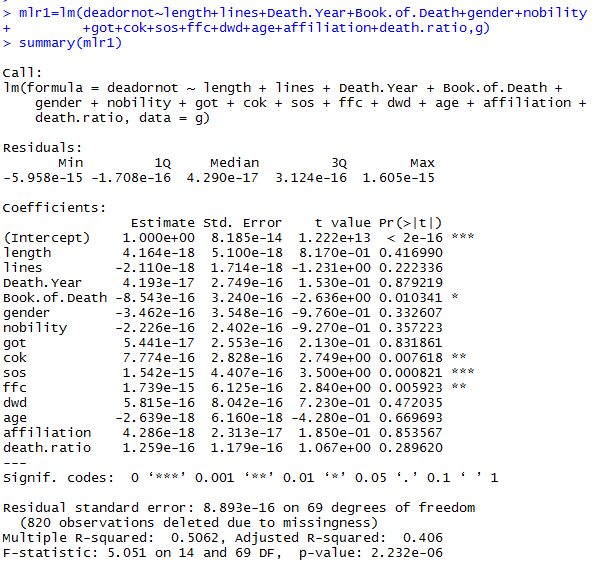


Our misclassification error is .32743 \* .98649 = .323 in 9-fold cross-validation, which is a little bit high.

**Models:**

Below is our first model. It includes the data of five books. In the output, it shows the third, fourth and fifth books are significant, as well as the predictor “Book.Of.Death”.

However, these five books have only two outcomes: 0 and 1. We want to see whether the model can be better if we run models without five books.



AIC: 30

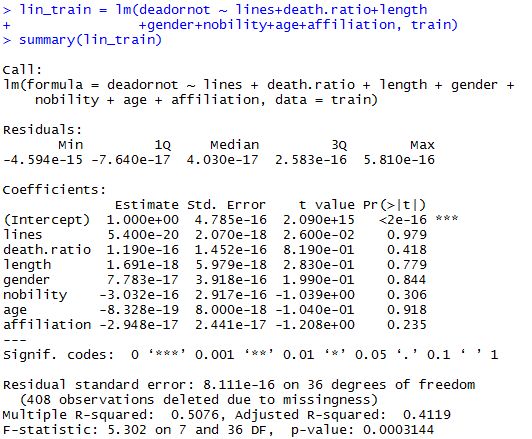
Adjusted R-squared: .406

R-squared: .5062

After modifying several models, we picked out two models.

lin\_train = lm(deadornot ~ lines+death.ratio+length+gender+nobility+age+affiliation, train)

In our first model, we add a new variable “length” to tell how many chapters a character being alive. We take the difference of Book.Intro.Chapter and Death.Chapter.



AIC: 16

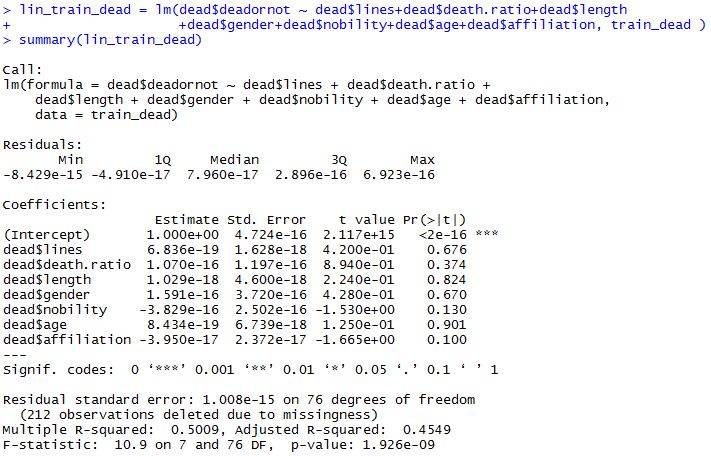
Adjusted R-squared: .4119

R-squared: .5076

Although the R-squared and Adjusted R-squared values are not too bad, the P-values are all very high, showing none of them are significant.

We also built another model focusing on dead characters only. In the following model, according to the plots we made, we assume that the length and lines are exponentially related to our response.

lin\_train\_dead = lm(dead$deadornot ~ exp(dead$lines)+dead$death.ratio+exp(dead$length)+dead$gender+dead$nobility+exp(dead$age)+dead$affiliation, train\_dead)



AIC: 16

Adjusted R-squared: .4549

R-squared: .5009

The Adjusted R-squared increased a little bit, though still not ideal according to the parameters.

**Interpreting the Results:**

While going through the process of choosing a model we found that several variables could be thrown out along the way as they did not do a good job of predicting the response y. We found this by discarding the variable with the worst p-value, R-squared, Adjusted R-squared and AIC, compared to others. Once we did this we would rerun a multiple linear regression model and noticed a higher Adjusted R-Squared value after eliminating one of the variables. This confirmed that we should be eliminating that variable. We already know that Jon Snow is going to stay on the screen in the next season due to the poster exposed recently. In our models, the high intercept and low coefficients show that Jon Snow, holding all else equal, will be killed in the future.

The top predictors in our model to show who will be killed during the television series are whether or not the character is in the books a Feast of Crows, and a Dance with Dragons, and also the gender of that character.

The 5 characters most likely to be killed during the upcoming Game of Thrones are: (Based on the model with no NA values)

1 – Dagmer

2 – Jacks

3 – Jommy

4 – Shagga

5 – Vylarr

**What we learned:**

Both of our models are not very good, however, these are the best two among all models. We try to think about the reasons for having poor models. Because our dataset is relatively small and at the beginning of our project, we were limited to the ways of reducing or picking out the important variables due to NA values. Because of this we could not use the Mean Squared Errors to see how good our model is as well. These made our predictions less accurate.

When we were dealing with the dataset, we replaced NA values of variable “lines” by numbers less than 25. Since the smallest number of the top 100 popular characters is 25, we assume the rest of the characters have lines fewer than 25 in the TV show. This makes sense somehow but after we plot the “lines”, we found that the majority of the numbers are less than 25 which makes the plot less real and mistakes are more likely. We also consider that the range of the outcomes impact a lot. Some variables have values ranges from 0 to hundreds but others have 0 and 1 outcomes. It may be better to standardize it like taking standard deviation when analyzing.

Furthermore, it is very difficult, given our dataset, to accurately predict who will be killed during the upcoming show. This can be seen by looking at the Adjusted R-Squared value of our second model, which is .4549. This is a relatively low Adjusted R-Squared value. This is because the variables we have data on are not the best predictors for who will be killed during the show. Another reason is that our best variables we are using to predict with are binomial. Since they are binomial, if we have a large dataset we will have many lines that have the exact same features on those variables.

There are several other external factors which will influence a character’s death. For example, what does the film crew think about a role, the mood of the author George R.R. Martin, the contract of the character or the perspective from audiences on the character etc.

Although the model may not be perfect for predicting a television show’s outcome we both have learned quite a bit from this project and are looking forward to watching the show to see if our predictions turn out to be true.