Final Project

Binary Logistic Regression Analysis White Sox

David Xu & Andy Chu

March 14, 2022

```
cubs <- read_csv("C:/Users/David/Downloads/cubs.csv")
```

```
## Rows: 162 Columns: 8
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## dbl (8): WL, R, RA, Hits, TB, SO, HomeAway, DayNight
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
#View(cubs)
```

Introduction

Our project goal is to be able to predict how well the Chicago Cubs and Chicago White Sox should perform during the 2021 season. Our logistic regression model will focus on predicting either of the Chicago teams winning a game, based on their game statistic performance during the season. To predict the response variable of the team winning a game,we plan to use what we think were the most suitable predictors, including runs scored, runs allowed, hits per game, errors per game, base on balls per game, if the game was played home or away, and if the game was during day or night. We will include all the 162 game statistics of both teams during the 2021 baseball season. Our data set will be collected from the MLB official website and the Baseball-Reference website.

Simple Logistic Regression

a. The predictor we chose was Runs Against. Because it showed the best improvement in decrease of residual deviance.

```
modR <- glm(WL ~ R, family = "binomial", data = cubs)
modRA <- glm(WL ~ RA, family = "binomial", data = cubs)
modHits <- glm(WL ~ Hits, family = "binomial", data = cubs)
modTB <- glm(WL ~ TB, family = "binomial", data = cubs)
modSO <- glm(WL ~ SO, family = "binomial", data = cubs)
modHA <- glm(WL ~ HomeAway, family = "binomial", data = cubs)
modDN <- glm(WL ~ DayNight, family = "binomial", data = cubs)
#summary(modR)
summary(modRA)
```

```
##
## Call:
## glm(formula = WL ~ RA, family = "binomial", data = cubs)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -1.90726  -0.53379  -0.02547   0.59497   2.70622
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)   3.4010     0.5733   5.933 2.98e-09 ***
## RA           -0.8796     0.1414  -6.219 4.99e-10 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 222.10  on 161  degrees of freedom
## Residual deviance: 120.17  on 160  degrees of freedom
## AIC: 124.17
##
## Number of Fisher Scoring iterations: 6
```

```
#summary(modHits)
#summary(modTB)
#summary(modHA)
#summary(modDN)
```

b. The slope, measuring the change in log(odds) for every unit change in Runs against, is -0.8796.

The estimated odds ratio below tells us that the odds of winning a game increase by 0.4149489 for every additional runs against. From the regression output above, we see that the p-value is really small, therefore, we reject the null hypothesis of change in log(odds) being zero.

```
exp(-0.8796)
```

```
## [1] 0.4149489
```

c. The 95% CI for the slope is from -1.187431 to -0.6291236.

Also, the 95 % confidence interval for the odds ratio is between 0.3050038 and 0.5330588. It suggests that we are 95% confident that the odds of winning increase by a factor of between 0.3050038 and 0.530588 with an additional run against.

```
confint(modRA)
```

```
## Waiting for profiling to be done...
```

```
##                 2.5 %     97.5 %
## (Intercept)  2.369680  4.6336012
## RA          -1.187431 -0.6291236
```

```
exp(confint(modRA))
```

```
## Waiting for profiling to be done...
```

```
##                  2.5 %      97.5 %
## (Intercept) 10.6939699 102.8839062
## RA           0.3050038   0.5330588
```

d. Fitted model:

$$\widehat{log(Odds)} = 3.4010 - 0.8796(RA)$$

We decided to plug in the following values for runs against as we found them to be good intervals for predicting the probability to win: 0, 3, 7, 10

$$\widehat{log(Odds)} = 3.4010 - 0.8796(0) = 3.4010$$
$$\frac{\pi}{1-\pi} = e^{3.4010} = 29.99408$$
$$\pi = \frac{e^{3.4010}}{1+e^{3.4010}} = 0.9677358$$

$$\widehat{log(Odds)} = 3.4010 - 0.8796 * 3 = 0.7622$$
$$\frac{\pi}{1-\pi} = e^{0.7622} = 2.142986$$
$$\pi = \frac{e^{0.7622}}{1+e^{0.7622}} = 0.6818312$$

$$\widehat{log(Odds)} = 3.4010 - 0.8796 * 7 = -2.7562$$
$$\frac{\pi}{1-\pi} = e^{-2.7562} = 0.06353273$$
$$\pi = \frac{e^{-2.7562}}{1+e^{-2.7562}} = 0.05973745$$

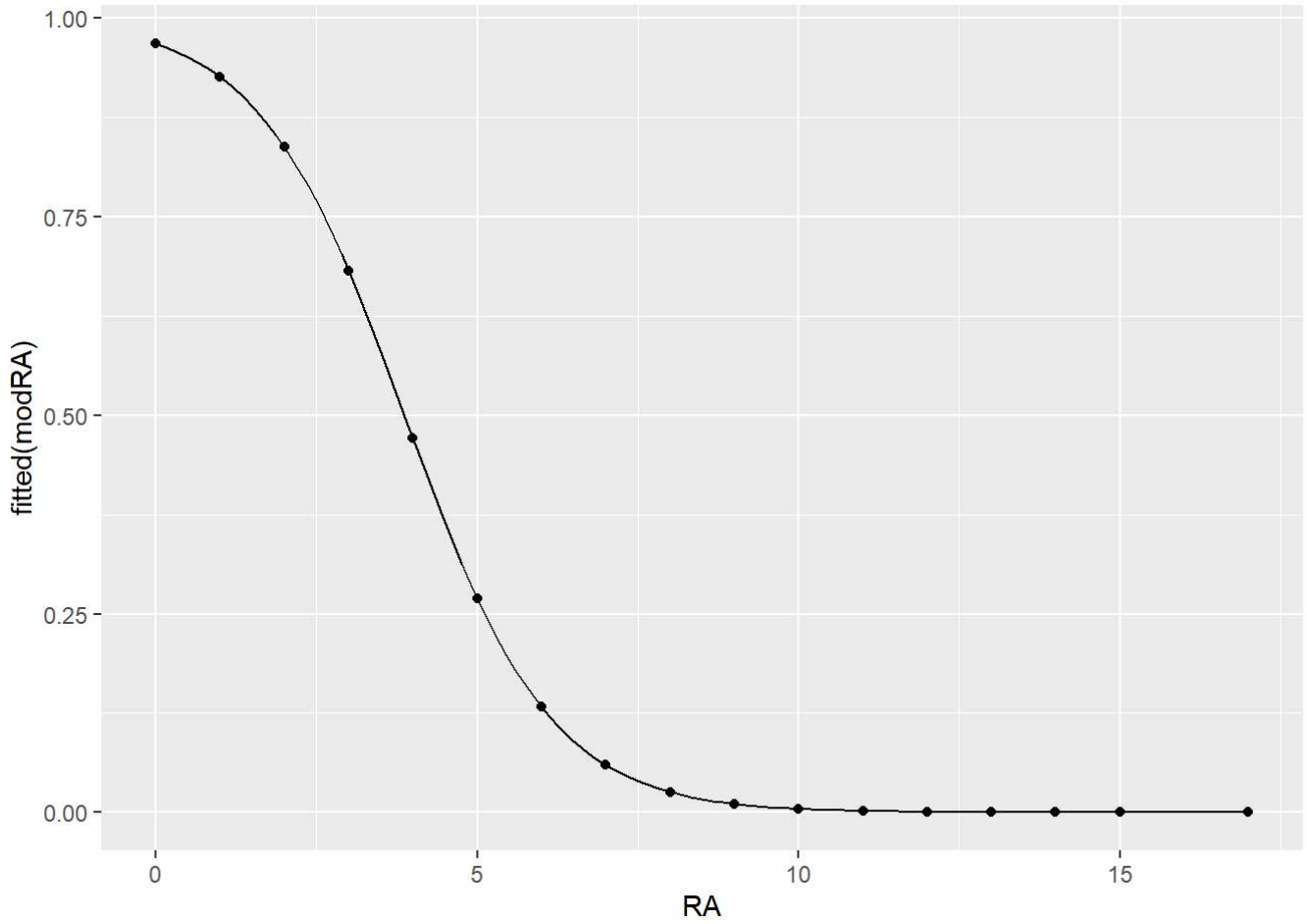
$$\widehat{log(Odds)} = 3.4010 - 0.8796 * 10 = -5.395$$
$$\frac{\pi}{1-\pi} = e^{-5.395} = 0.00453922$$
$$\pi = \frac{e^{-5.395}}{1+e^{-5.395}} = 0.004518709$$

Comment:

We see that the winning probability decreases from 96.77% to 0.45% as more runs are scored by the opponents. Interestingly, the winning probability is only 96.77% for the Cubs even though

e. From the graph we can see that the probability of winning a game increases drastically from 0 Runs against to 9 Runs against. If Cubs has more than 9 runs allowed, the probability of winning is nearly zero.

```
MODR_fun<- makeFun(modRA)
gf_point(fitted(modRA)~RA, data = cubs)%>%
  gf_fun(MODR_fun(RA)~RA, color = "Black")
```



f. The output below gives us the G-statistic of 101.94 with a small P-value. Therefore, we can conclude that the model is useful in predicting the response variable, whether the Cubs will win their games.

```
anova(modRA, test ="Chisq")
```

```
## Analysis of Deviance Table
##
## Model: binomial, link: logit
##
## Response: WL
##
## Terms added sequentially (first to last)
##
##
##      Df Deviance Resid. Df Resid. Dev  Pr(>Chi)
## NULL                   161     222.10
## RA    1   101.94       160     120.17 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Multiple Logistic Regression

a.

The multiple regression model we decided to be the best was the RA + TB + HomeAway. This model gives us one of the smallest residual deviance and the p-values for the predictor variables were all small to indicate all 3 variables to be significant. We did use only RA + TB, and it also gave a significant response.

```
modRA
```

```
##
## Call:  glm(formula = WL ~ RA, family = "binomial", data = cubs)
##
## Coefficients:
## (Intercept)           RA
##      3.4010      -0.8796
##
## Degrees of Freedom: 161 Total (i.e. Null);  160 Residual
## Null Deviance:       222.1
## Residual Deviance: 120.2     AIC: 124.2
```

```
mod1 <- glm(WL~RA+TB, family = "binomial", data = cubs)
mod2 <- glm(WL~RA+TB+HomeAway, family="binomial", data = cubs)
mod3 <- glm(WL~RA+Hits+TB+HomeAway+DayNight,family="binomial", data = cubs)


anova(modRA, mod1, mod2, mod3, test ="Chisq")
```

```
## Analysis of Deviance Table
##
## Model 1: WL ~ RA
## Model 2: WL ~ RA + TB
## Model 3: WL ~ RA + TB + HomeAway
## Model 4: WL ~ RA + Hits + TB + HomeAway + DayNight
##   Resid. Df Resid. Dev Df Deviance  Pr(>Chi)
## 1       160    120.166
## 2       159     80.244  1   39.922 2.643e-10 ***
## 3       158     72.352  1    7.892  0.004967 **
## 4       156     72.227  2    0.125  0.939423
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
summary(mod2)
```

```
##
## Call:
## glm(formula = WL ~ RA + TB + HomeAway, family = "binomial", data = cubs)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.3390  -0.1881  -0.0044   0.2250   3.5125
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -0.45223    0.84948  -0.532   0.5945
## RA          -1.40129    0.26029  -5.384 7.30e-08 ***
## TB           0.38474    0.08494   4.529 5.91e-06 ***
## HomeAway     1.80041    0.70347   2.559   0.0105 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 222.104  on 161  degrees of freedom
## Residual deviance:  72.352  on 158  degrees of freedom
## AIC: 80.352
##
## Number of Fisher Scoring iterations: 7
```

b.

Fitted model:

$$\widehat{log(Odds)} = -0.45223 - 1.40129RA + 0.38474TB + 1.80041HomeAway$$

We are interested to predict the probability of winning a game if the game is played away with 7 runs against and 7 total bases. We are curious about this case since if there are only 7 total bases, it is not optimistic to win the game if the opponent has 5 runs, unless the Cubs has a grand slam for instance and we all know it barely happens.

$$\widehat{log(Odds)} = -0.45223 - 1.40129 * 7 + 0.38474 * 7 + 1.80041 * 0$$

$$\widehat{log(Odds)} = -7.56808$$

$$\frac{\pi}{1-\pi} = e^{-7.56808} = 0.0005166835$$

$$\pi = \frac{0.0005166835}{1 + 0.0005166835} = 0.0005164167$$

And we see the probability of winning the game under this case is very small as expected.

c.

Statistical and practical Significance

```
exp(-1.40129)
```

```
## [1] 0.2462791
```

```
exp(0.38474)
```

```
## [1] 1.469232
```

```
exp(1.80041)
```

```
## [1] 6.052128
```

```
sd(cubs$RA)
```

```
## [1] 3.945127
```

```
exp(-1.40129*sd(cubs$RA))
```

```
## [1] 0.003972867
```

```
sd(cubs$TB)
```

```
## [1] 6.448346
```

```
exp(0.38474*sd(cubs$TB))
```

```
## [1] 11.95245
```

Comment:

We can see that the odds ratio for Runs Against(0.2462791), Total Base(1.469232) and Home Away(6.052128). To indicate the significance we multiplied the odds ratio by the standard deviation to find the whether the odds ratio had a significant change. Showing a significant change indicates the predictor variable being significant in our model. In our case there is significant change for every predictor.

Prediction Table

a. and b.

```
cubs2 <- cubs%>%
  mutate(win_prob = fitted(mod2)) %>%
  mutate(success = ifelse(win_prob > 0.5 & WL == 1 | win_prob < 0.5 & WL == 0, "success", "failu
re"))

#View(cubs2)
```

c.

```
tally(~WL+success, data = cubs2)
```

```
##    success
## WL  failure success
##   0       8      83
##   1       8      63
```

d. Success Rate, in terms of losses/wins/overall

```
63/71
```

```
## [1] 0.8873239
```

```
83/91
```

```
## [1] 0.9120879
```

Comment:

The success rate of predicting Cubs wins using our model is 88.73%

The success rate of predicting Cubs loosing using our model is 91.2%

Conclusion

To conclude the Cubs results our Simple regression model and Multi regression model are both were useful. From the prediction table indicated how useful our model actually is in predicting the Cubs games winning or loosing during the season. To improve on our model we could have included more statistic about pitchers. Because we found out that Strike outs was not very useful to our model. Due to Covid-19 the audience capacity could not be 100% or otherwise it would have been a potential predictor variable to be used for our model. It allows people who are curious about the future Cubs winning a game they could use this model to predict their chances of winning.