NHL Structural Model

NHL

```r
#loading required libraries
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.2     v readr     2.1.4
## v forcats   1.0.0     v stringr   1.5.0
## v ggplot2   3.4.2     v tibble    3.2.1
## v lubridate 1.9.2     v tidyr     1.3.0
## v purrr     1.0.1
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```r
library(RCurl)
```

```
##
## Attaching package: 'RCurl'
##
## The following object is masked from 'package:tidyr':
##
##     complete
```

```r
library(ggplot2)
```

```r
#the data
library(RCurl)
link <- getURL("https://raw.githubusercontent.com/M-ttM/Basketball/master/gamedata.csv")
nhl <- read.csv(text = link)
head(nhl)
```

```
##         Date             Visitor G                Home G.1  X  Att.  LOG Notes
## 1 2017-10-04      Calgary Flames 0     Edmonton Oilers   3    18347 2:34    NA
## 2 2017-10-04     St. Louis Blues 5 Pittsburgh Penguins   4 OT 18652 2:38    NA
## 3 2017-10-04 Philadelphia Flyers 5     San Jose Sharks   3    17562 2:27    NA
## 4 2017-10-04 Toronto Maple Leafs 7       Winnipeg Jets   2    15321 2:33    NA
## 5 2017-10-05     Arizona Coyotes 4       Anaheim Ducks   5    17174 2:38    NA
## 6 2017-10-05 Nashville Predators 3       Boston Bruins   4    17565 2:39    NA
```

WEIBULL DISTRIBUTION

```r
install.packages("fitdistrplus")
```

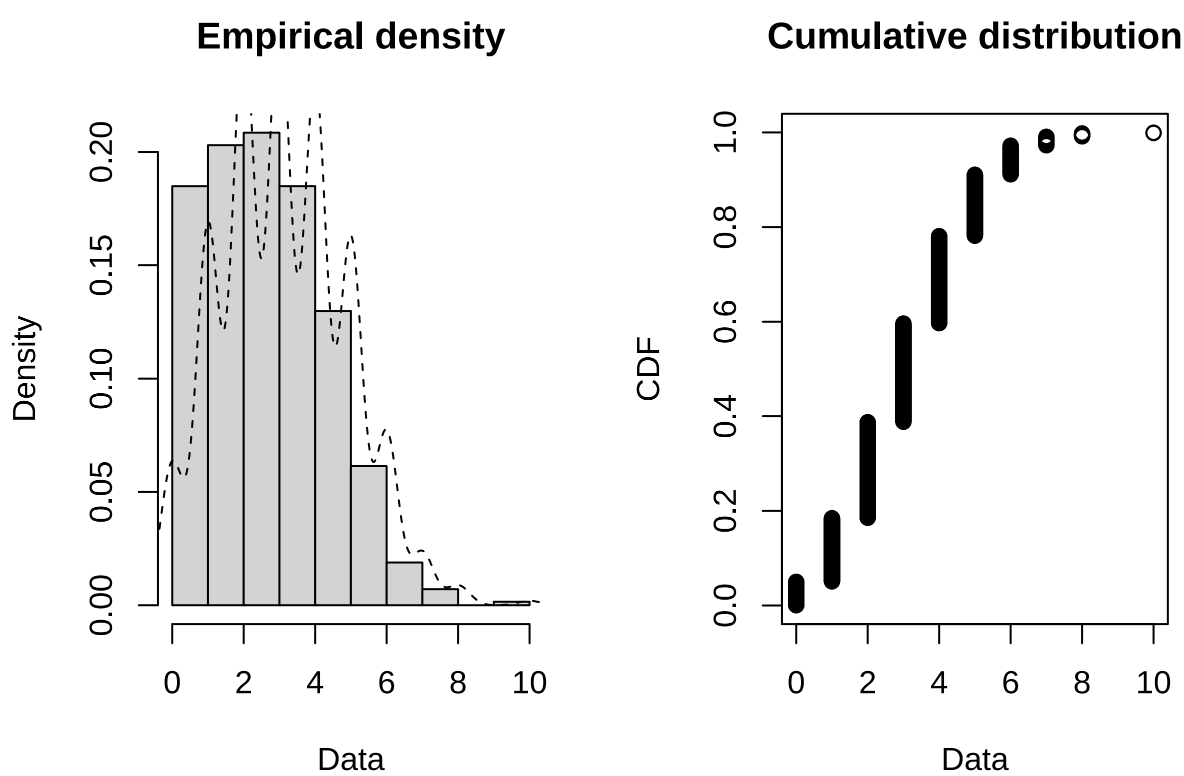
```
## Installing package into '/opt/r'
## (as 'lib' is unspecified)
```

```r
library(fitdistrplus)
```

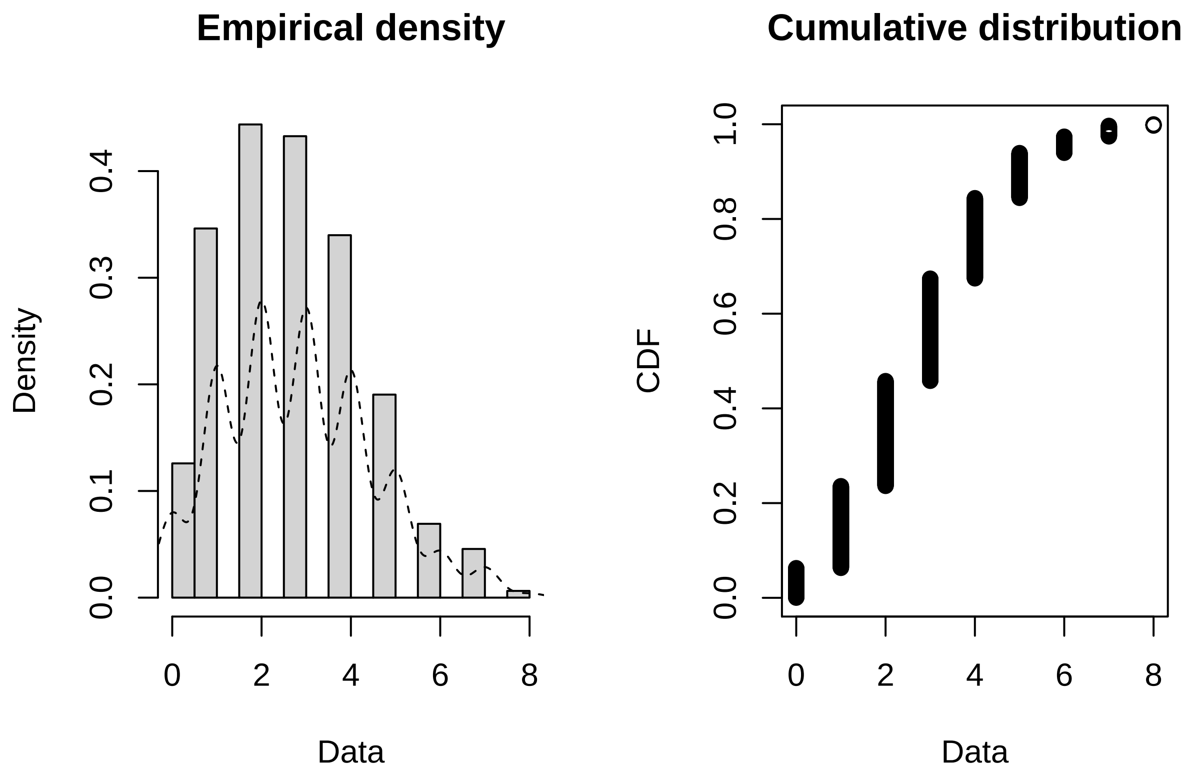
```
## Loading required package: MASS

##
## Attaching package: 'MASS'

## The following object is masked from 'package:dplyr':
##
##     select

## Loading required package: survival
```

```r
nhl <- nhl %>% mutate(GHadj = G.1+.5, GAadj = G+.5)
```

```r
plotdist(nhl$G.1, histo = TRUE, demp = TRUE)
```




```r
plotdist(nhl$G, histo = TRUE, demp = TRUE)
```



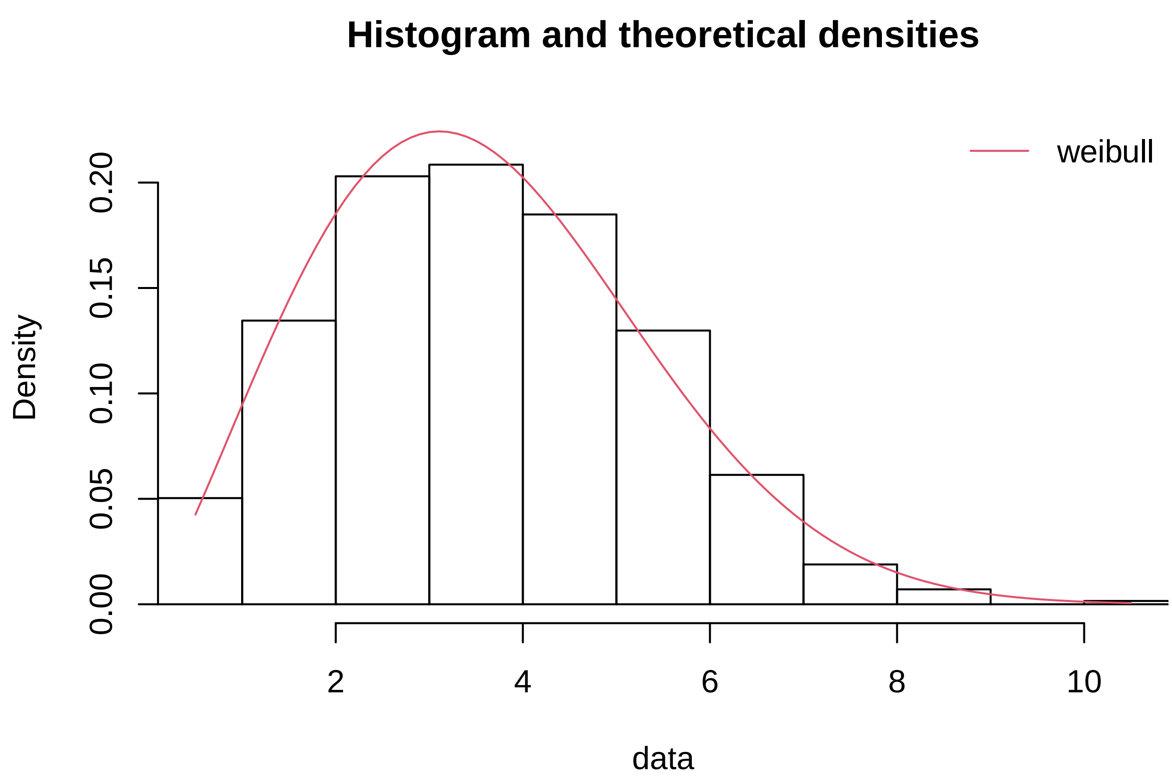

```r
fit.GH <- fitdist(nhl$GHadj, "weibull")
summary(fit.GH)
```

```
## Fitting of the distribution ' weibull ' by maximum likelihood
## Parameters :
##       estimate Std. Error
## shape 2.203528 0.04907325
## scale 4.089189 0.05467170
## Loglikelihood:  -2470.345   AIC:  4944.689   BIC:  4954.985
## Correlation matrix:
##          shape    scale
## shape 1.000000 0.305757
## scale 0.305757 1.000000
```

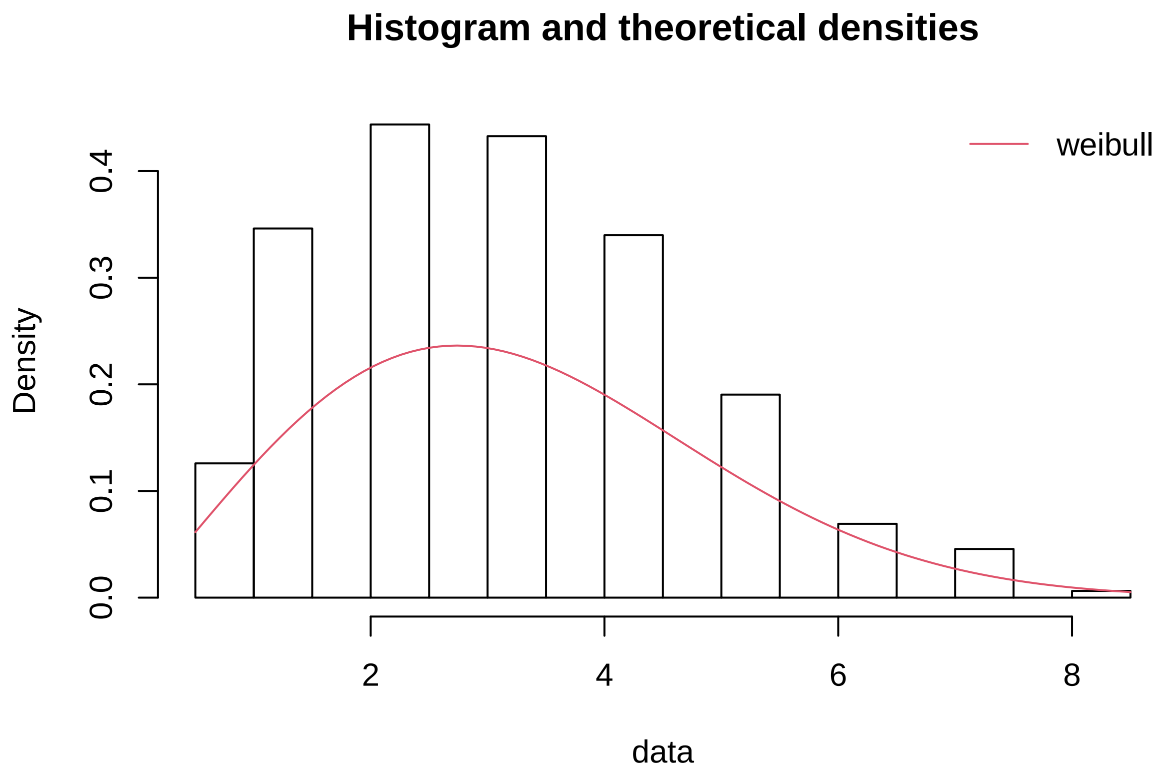
```r
fit.GA <- fitdist(nhl$GAadj, "weibull")
summary(fit.GA)
```

```
## Fitting of the distribution ' weibull ' by maximum likelihood
## Parameters :
##       estimate Std. Error
## shape 2.090796 0.04641639
## scale 3.738554 0.05272833
## Loglikelihood:  -2401.036   AIC:  4806.073   BIC:  4816.368
## Correlation matrix:
##           shape     scale
## shape 1.0000000 0.3083788
## scale 0.3083788 1.0000000
```

```
denscomp(fit.GH)
```




```
denscomp(fit.GA)
```



WINS PER SEASON PER EXTRA GOAL PER GAME

WP = GF^(a) / [GF^(a) + GA^(a)]

slope = (a * GF^(a-1) * GA^(a)) / (GF^(a) + GA$^{(a))}$(2)

a = shape = 2.2

```
count <- nhl %>%
  group_by(Home) %>%
  summarize(GF = sum(G.1), GA = sum(G))
```

```
a = 2.2
count <- count %>%
  mutate(WP = (GF^(a)/(GF^(a) + GA^(a)))) %>%
  arrange(desc(WP))
```

```
count <- count %>%
  mutate(Slope = (a * GF^(a-1) * GA^(a)) / (GF^(a) + GA^(a))^(2))
```

```
count <- count %>%
  mutate(xWpS = Slope * 82)

print(count)
```

```
## # A tibble: 31 x 6
##    Home                          GF    GA    WP   Slope  xWpS
##    <chr>                      <int> <int> <dbl>   <dbl> <dbl>
```

```
##   1 Winnipeg Jets          159   101 0.731 0.00272 0.223
##   2 Minnesota Wild         137    90 0.716 0.00327 0.268
##   3 Colorado Avalanche     146    98 0.706 0.00313 0.256
##   4 Vegas Golden Knights   147   103 0.686 0.00322 0.264
##   5 Boston Bruins          148   104 0.685 0.00321 0.263
##   6 Pittsburgh Penguins    151   110 0.668 0.00323 0.265
##   7 Washington Capitals    138   103 0.656 0.00360 0.295
##   8 Nashville Predators    143   107 0.654 0.00348 0.285
##   9 Toronto Maple Leafs    144   109 0.649 0.00348 0.286
##  10 Dallas Stars           128    99 0.638 0.00397 0.326
## # i 21 more rows
```

PER GAME PREDICTION

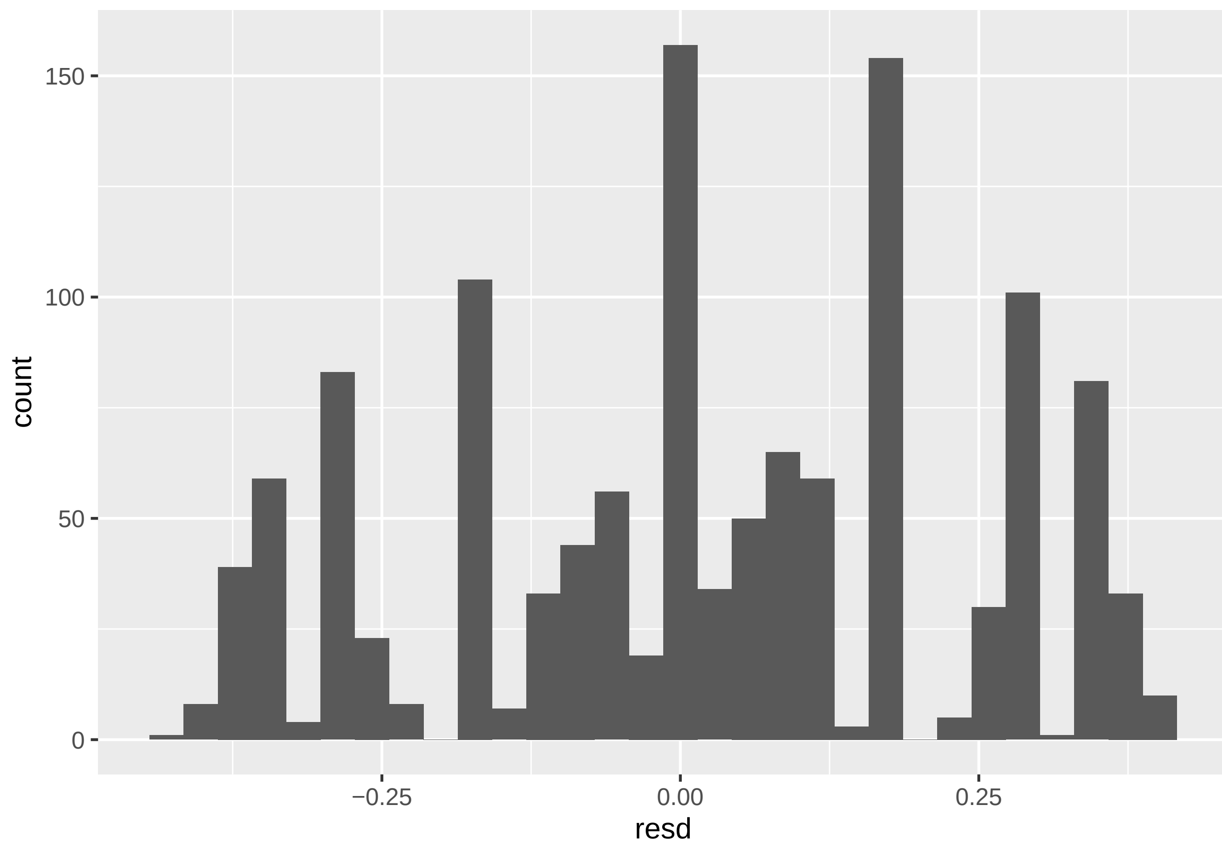
```
a = 2.2
nhl <- nhl %>%
  mutate(WP = (G.1^(a)/(G.1^(a) + G^(a))),
         diff = G.1 - G,
         Win = ifelse(diff>0, 1, 0)) %>%
  arrange(desc(WP))
print(head(nhl))
```

```
##         Date               Visitor G                  Home G.1 X  Att.  LOG Notes
## 1 2017-10-04      Calgary Flames 0       Edmonton Oilers   3   18347 2:34    NA
## 2 2017-10-05 Philadelphia Flyers 0     Los Angeles Kings   2   18230 2:37    NA
## 3 2017-10-06  New York Islanders 0 Columbus Blue Jackets   5   18595 2:25    NA
## 4 2017-10-07 Nashville Predators 0   Pittsburgh Penguins   4   18645 2:38    NA
## 5 2017-10-08  Montreal Canadiens 0      New York Rangers   2   18006 2:29    NA
## 6 2017-10-24   Detroit Red Wings 0        Buffalo Sabres   1   16882 2:32    NA
##   GHadj GAadj WP diff Win
## 1   3.5   0.5  1    3   1
## 2   2.5   0.5  1    2   1
## 3   5.5   0.5  1    5   1
## 4   4.5   0.5  1    4   1
## 5   2.5   0.5  1    2   1
## 6   1.5   0.5  1    1   1
```

```
nhl <- nhl %>%
  mutate(resd = Win - WP)
```

```
ggplot(nhl, aes(resd)) +
  geom_histogram()
```

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



From the plot above, we can understand the predictive ability of the Pythagorean Wins Model. It predicts around 150 estimates with the correct win probability, this means that through goals scored and conceded in the game, 150 times the model predicts the correct probability as the result of the game. If we divide the plot in half from the 0 point, we can see that it is asymmetric, with the positive residuals having a higher probability, this implies that it predicts a higher win probability than the actual result more times than it does a lower one. There are clear modes in the distribution - 0, -.2, .2, -.3, .3, and so on. It is to be noted that everything on the left of 1 had a positive probability but a value for 0 which implies either a tie or a loss. The probability associated with such a game is never predicted higher than 50%. For wins, it ranges above the 50% point (depending on goals scored and conceded).

LOGISTIC REGRESSION

A logistic regression with two variables - Goals For and Goals Against will be used to predict the winning probability of a team. It predicts the odds for a success, in this context, a win given the variables. Therefore, intuitively, considering similar level of teams, will have the same coefficient affecting the probability of winning a game as a goal being scored must have the same but opposite implication on the chances of a team winning. Otherwise, goals scored are valued higher than goals conceded. Therefore it will just use the difference in the two values to increase the odds in relation to the coefficient value as per its slope. In a pythagorean model, there is a scale and shape which help in understanding and modelling such situations better than logistic regression.

NHL PREDICTION

```
#the data
library(RCurl)
```

```
link <- getURL("https://raw.githubusercontent.com/M-ttM/Basketball/master/allrecords2.csv")
nhl1 <- read.csv(text = link)
head(nhl1)
```

```
##   X     Team GF_mid GA_mid OTL_mid W_mid L_mid GP_mid  GF  GA OTL  W  L
## 1 1  Anaheim    144    142       9    25    17     51 235 216  13 44 25
## 2 2  Arizona    118    172       9    12    29     50 208 256  12 29 41
## 3 3   Boston    157    119       8    29    11     48 270 214  12 50 20
## 4 4  Buffalo    114    163       9    14    26     49 199 280  12 25 45
## 5 5  Calgary    137    135       8    25    16     49 218 248  10 37 35
## 6 6 Carolina    137    154       8    22    19     49 228 256  11 36 35
```

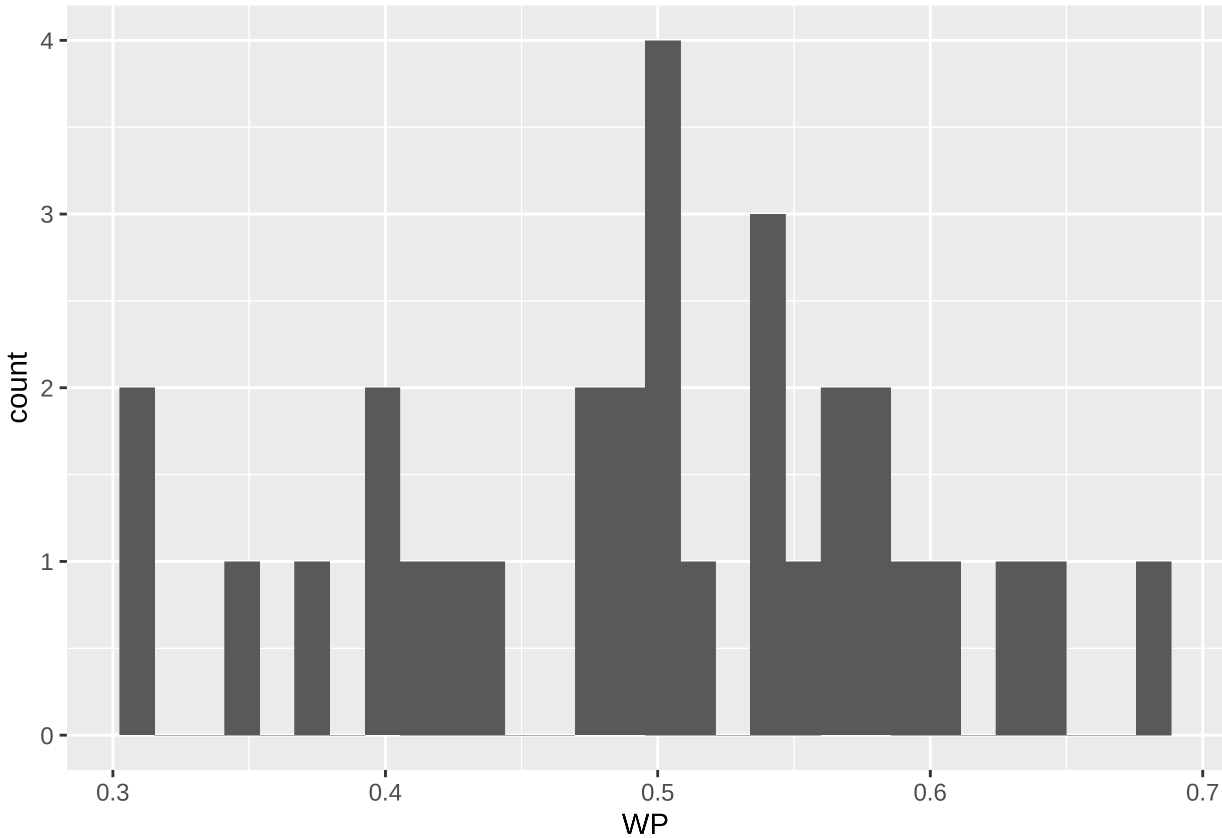
second half WP using first half pf/pa and Weibull distribution estimate.

```
a = 2.2
nhl1 <- nhl1 %>%
  mutate(WP = (GF_mid^(a)/(GF_mid^(a) + GA_mid^(a))),
         W_pred = W_mid + (82-GP_mid)*WP) %>%
  arrange(desc(WP))

print(head(nhl1))
```

```
##    X      Team GF_mid GA_mid OTL_mid W_mid L_mid GP_mid  GF  GA OTL  W  L
## 1 26 Tampa Bay    175    125       3    34    12     49 296 236   5 54 23
## 2  3    Boston    157    119       8    29    11     48 270 214  12 50 20
## 3 29     Vegas    164    128       4    32    12     48 272 228   7 51 24
## 4 31  Winnipeg    164    136       8    29    13     50 277 218  10 52 20
## 5 17 Nashville    145    123       7    29    11     47 267 211  11 53 18
## 6 10    Dallas    155    134       4    28    18     50 235 225   8 42 32
##          WP   W_pred
## 1 0.6770481 56.34259
## 2 0.6478653 51.02742
## 3 0.6330304 53.52303
## 4 0.6015351 48.24912
## 5 0.5895266 49.63343
## 6 0.5793944 46.54062
```

```
ggplot(nhl1, aes(WP)) +
  geom_histogram()
```

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



second half WP for each team by running a regression using the first half wins/pf/pa data, and using the predicted values.

```
fit.1 <- lm(W ~ GF_mid + GA_mid, data = nhl1)
summary(fit.1)
```

```
##
## Call:
## lm(formula = W ~ GF_mid + GA_mid, data = nhl1)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -10.7825  -2.8040  -0.3477   3.1952  10.0873
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 44.11887   14.86803   2.967 0.006089 **
## GF_mid       0.26093    0.06148   4.244 0.000218 ***
## GA_mid      -0.28259    0.06138  -4.604 8.18e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.647 on 28 degrees of freedom
## Multiple R-squared:  0.7027, Adjusted R-squared:  0.6815
## F-statistic:  33.1 on 2 and 28 DF,  p-value: 4.207e-08
```

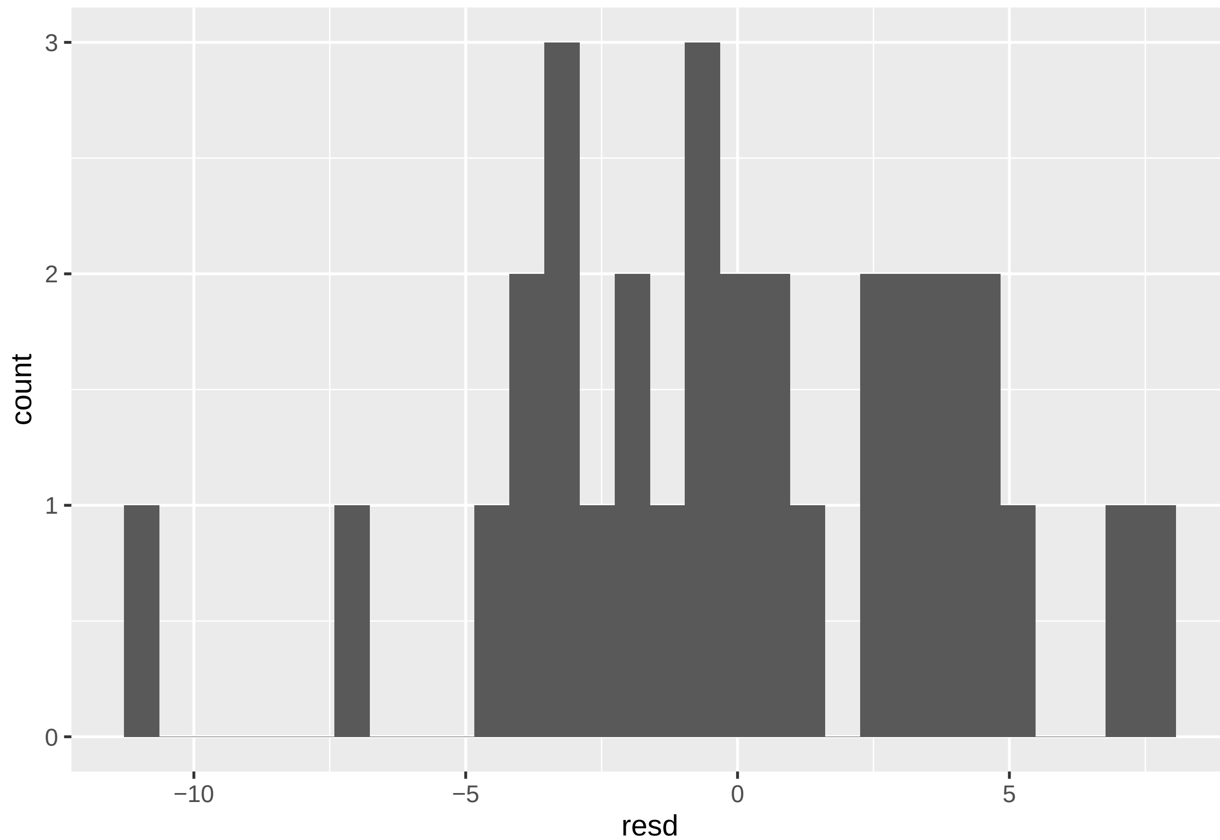
Which predictions does better at predicting the second half of the season?

```
nhl1 <- nhl1 %>%
  mutate(resd = W_pred - W)
```

```
nhl1$resd_l <- resid(fit.1)
```

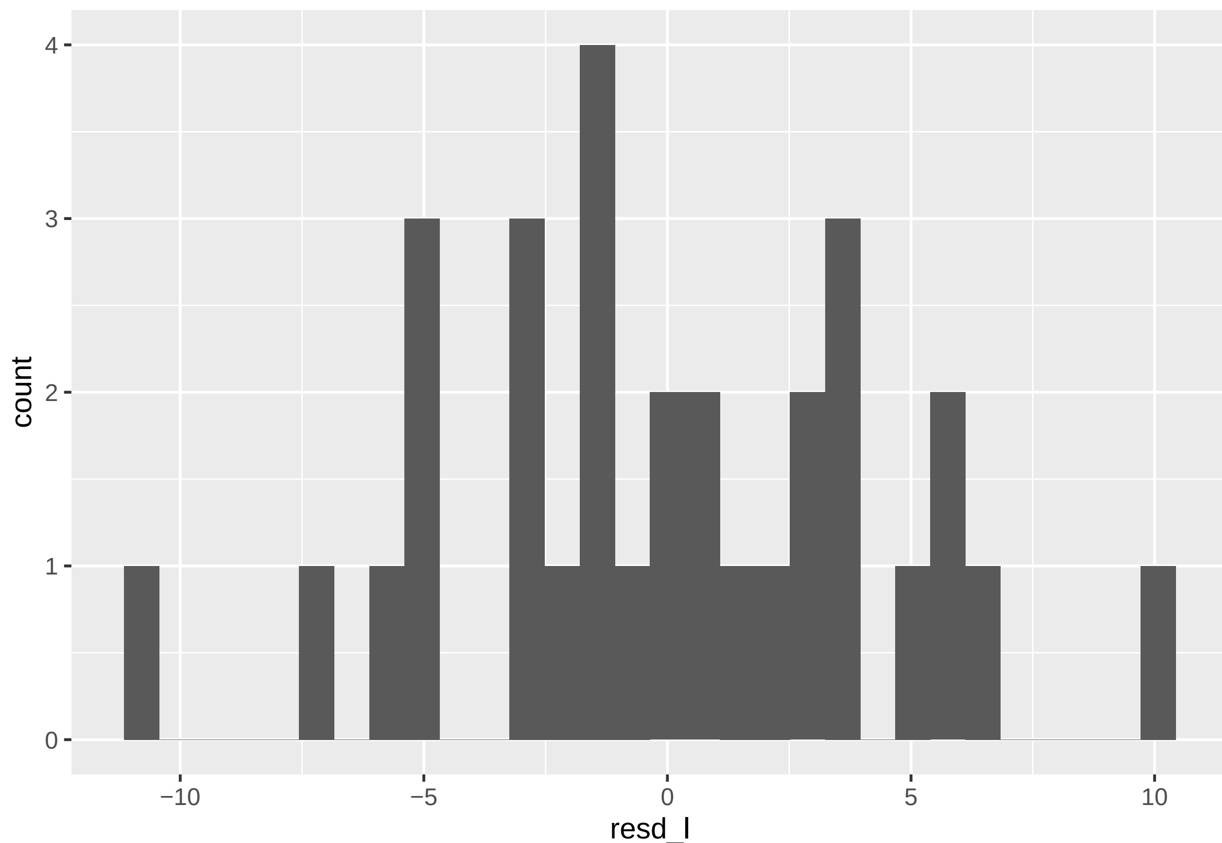
```
ggplot(nhl1, aes(resd)) +
  geom_histogram()
```

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




```
ggplot(nhl1, aes(resd_l)) +
  geom_histogram()
```

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



The Pythagorean Win Model is better at prediction with higher values at 0 and more of the residuals centered around 0 and within the 5 win error range. FOr the regressions, the residuals are more disperesed and less percentage is in the -5 to 5 error range. It also has more outliers in the residuals distribution.