Map501_F429147

By Keunwoo Kim

```
library("tidyverse")
library("magrittr")
library("here")
library("janitor")
library("gridExtra")
library("readxl")
library("Lahman")
library("viridis")
library("lindia")
library("lme4")
library("caret")
library("pROC")
library("car")
library("dplyr")
library("nnet")
```

```
#q1a.
head(Managers)
```

```
##    playerID yearID teamID lgID inseason  G  W  L rank plyrMgr
## 1 wrighha01   1871    BS1   NA        1 31 20 10    3       Y
## 2  woodji01   1871    CH1   NA        1 28 19  9    2       Y
## 3 paborch01   1871    CL1   NA        1 29 10 19    8       Y
## 4 lennobi01   1871    FW1   NA        1 14  5  9    8       Y
## 5 deaneha01   1871    FW1   NA        2  5  2  3    8       Y
## 6 fergubo01   1871    NY2   NA        1 33 16 17    5       Y
```

```
df_managers <- Managers %>%
  mutate(win_pct = W/(W + L)) %>%
  select(playerID, teamID, yearID, lgID, plyrMgr, win_pct)
```

```
#q1b.
#b1.
df_teams <- Teams %>%
  select(yearID, teamID, DivWin, CS)
#b2.
man_teams <- merge(df_managers, df_teams, by = c("yearID", "teamID")) %>%
  select(-lgID)
#b3, 4.
awards_man <- merge(man_teams, AwardsShareManagers, by = c("yearID", "playerID")) %>%
  mutate(sqr_point_pct = sqrt(pointsWon/pointsMax))
#b5.
summary(awards_man$teamID)
```

```
##      NYA      ATL      HOU      LAN      BOS      CLE      OAK      SLN      SFN      NYN 
##       32       26       26       25       23       22       22       22       21       20 
##      MIN      TOR      TEX      SDN      DET      CHA      CHN      SEA      BAL      PHI 
##       18       18       17       16       15       14       14       13       12       12 
##      PIT      TBA      CIN      MIL      KCA      MON      ARI      COL      LAA      WAS 
##       12       12       11       11       10       10        8        8        7        7 
##      ANA      CAL      FLO      ML4      MIA      ALT      BFN      BFP      BL1      BL2 
##        6        6        6        5        4        0        0        0        0        0 
##      BL3      BL4      BLA      BLF      BLN      BLU      BR1      BR2      BR3      BR4 
##        0        0        0        0        0        0        0        0        0        0 
##      BRF      BRO      BRP      BS1      BS2      BSN      BSP      BSU      BUF      CH1 
##        0        0        0        0        0        0        0        0        0        0 
##      CH2      CHF      CHP      CHU      CL1      CL2      CL3      CL4      CL5      CL6 
##        0        0        0        0        0        0        0        0        0        0 
##      CLP      CN1      CN2      CN3      CNU      DTN      ELI      FW1      HAR      HR1 
##        0        0        0        0        0        0        0        0        0        0 
##      IN1      IN2      IN3      IND      KC1      KC2      KCF      KCN      KCU      KEO 
##        0        0        0        0        0        0        0        0        0        0 
##      LS1      LS2      LS3      MID      ML1      ML2      ML3      MLA      MLU (Other) 
##        0        0        0        0        0        0        0        0        0        0 
```

```
awards_man <- awards_man %>%
  drop_na() %>%
  mutate(teamID = droplevels((teamID)))
head(awards_man)
```

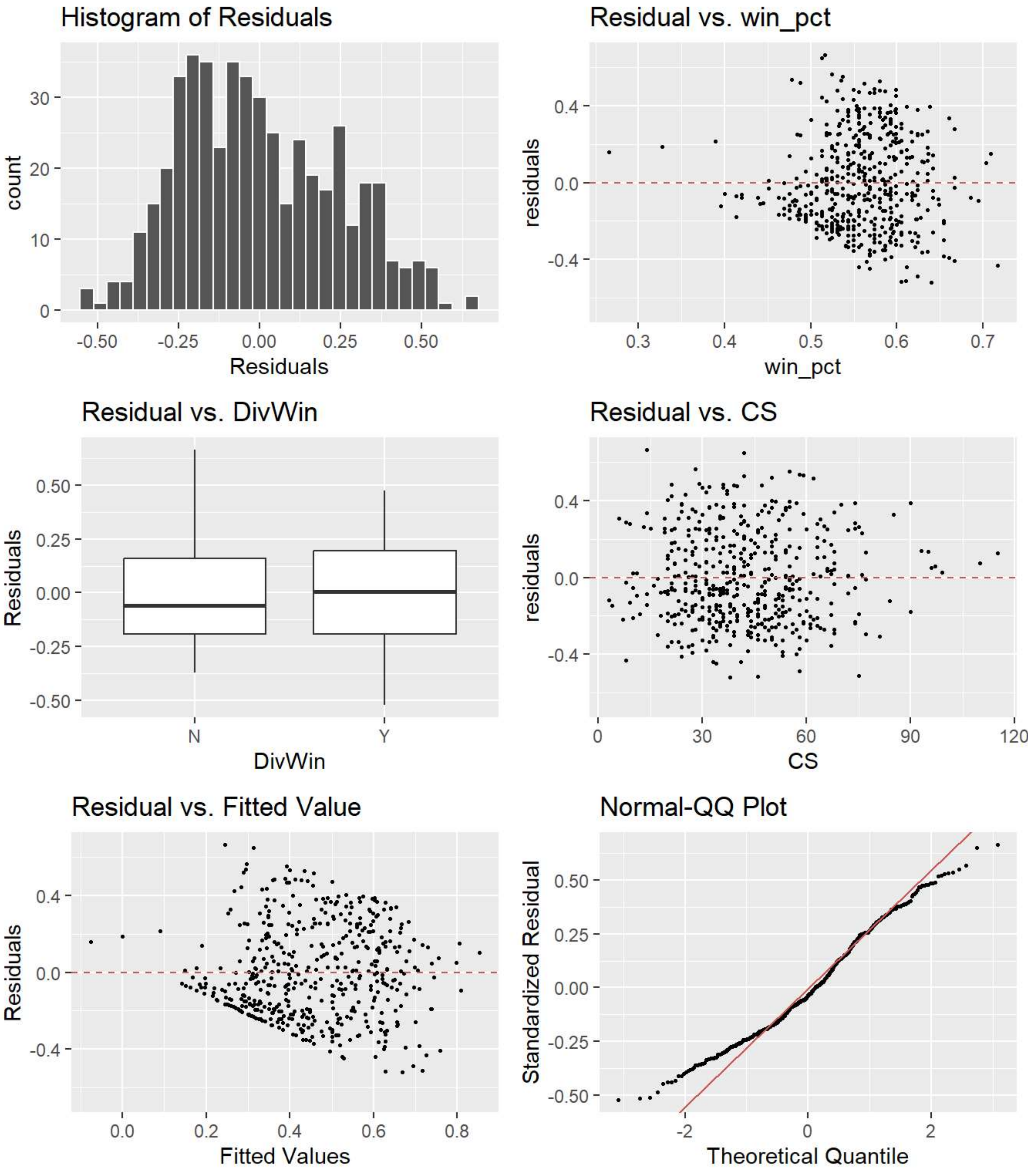
```
##   yearID  playerID teamID plyrMgr   win_pct DivWin CS                   awardID
## 1   1983 altobjo01    BAL       N 0.6049383      Y 33 BBWAA Manager of the Year
## 2   1983   coxbo01    TOR       N 0.5493827      N 72 BBWAA Manager of the Year
## 3   1983 larusto01    CHA       N 0.6111111      Y 50 BBWAA Manager of the Year
## 4   1983 lasorto01    LAN       N 0.5617284      Y 76 BBWAA Manager of the Year
## 5   1983 lillibo01    HOU       N 0.5246914      N 95 BBWAA Manager of the Year
## 6   1983 owenspa99    PHI       N 0.6103896      Y 75 BBWAA Manager of the Year
##   lgID pointsWon pointsMax votesFirst sqr_point_pct
## 1   AL         7        28          7     0.5000000
## 2   AL         4        28          4     0.3779645
## 3   AL        17        28         17     0.7791937
## 4   NL        10        24         10     0.6454972
## 5   NL         9        24          9     0.6123724
## 6   NL         1        24          1     0.2041241
```

```
#q1c.
spp_mod <- lm(sqr_point_pct ~ win_pct + DivWin + CS, data = awards_man)
summary(spp_mod)
```

```
##
## Call:
## lm(formula = sqr_point_pct ~ win_pct + DivWin + CS, data = awards_man)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.52222 -0.19156 -0.03509  0.18024  0.66480
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.7270095  0.1409706  -5.157 3.67e-07 ***
## win_pct      1.8065034  0.2518290   7.174 2.77e-12 ***
## DivWinY      0.1365885  0.0266360   5.128 4.25e-07 ***
## CS           0.0027189  0.0006249   4.351 1.66e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2401 on 482 degrees of freedom
## Multiple R-squared:  0.2709, Adjusted R-squared:  0.2664
## F-statistic: 59.71 on 3 and 482 DF,  p-value: < 2.2e-16
```

#q1c. According to the coefficient’s p-value, the Intercept and win_pct are meaningful as each P-value is much lower than 0.05. While the DivWin and CS’s p-values are too high to interpret as the two coefficients are meaningful. This interpretation also aligns with the adjusted r-squared. The adjusted r-squared shows whether the dependent variable is meaningful and could be trusted if the result is closer to 1. In terms of this model, the adjusted r-squared is tiny by 0.001263, which overlaps with the dependent variable DivWin and CS’s p-value. The form of fitted model is sqr_point_pct = 0.41 + 0.11*win_pct* + *0*DivWin - 0*CS

```
#q1d.
spp_mod_plots <- spp_mod %>%
  gg_diagnose(plot.all = FALSE)

plot_all(spp_mod_plots[1:6])
```



#q1d. The Residuals vs. Fitted graph shows how the fitted values distribute the residuals by the randomly distributed scatter and the shape of the red line. There should be a random distribution if the independent and dependent variables are in a linear relationship. In addition, the red line, which is the average of the residuals, should not show patterns such as a curve or U shape. In the current model's Residuals vs. Fitted graph, the scatters do not show certain patterns and are distributed randomly around the red line. Therefore, it could be interpreted as the linearity and homoscedasticity assumptions are fulfilled. Normality could be examined using the histogram and the normal QQ plot. According to the Histogram, the asymmetry is seen in the graph, which shows a downtrend from -3 to 6. Likewise, the normal QQ plot shows a heavy-tailed distribution near -3 and +3. As the scatters deviate from the diagonal line, the Normality is not fulfilled. In the Residual vs win_pct and Residual vs CS plots, the scatters are distributed with no patterns, and the red line is in a straight line from the y-axis 0. This implies that the mean average of the residual is 0. The Residual vs DivWin plot shows similar Interquartile Ranges between each category, and the median is near 0. Therefore, it is interpreted that the three independent variables fulfil the independence.

```
#q1e.
predict(spp_mod, newdata = data.frame(win_pct = 0.8, DivWin = "Y", CS = 8))
```

```
##         1
## 0.8765326
```

```
print(0.5013271^2)
```

```
## [1] 0.2513289
```

#q1e. The result of the code first code is 0.5013271. sqr_point_pct is in square root. Therefore, to interpret there, the result should be squared. According to the model prediction, the team earned approximately 25.13% of the pointsMax.

```
#q1f.
confint(spp_mod)
```

```
##                    2.5 %       97.5 %
## (Intercept) -1.004002402 -0.450016666
## win_pct      1.311685209  2.301321669
## DivWinY      0.084251412  0.188925564
## CS           0.001490967  0.003946747
```

#q1f. The intercept and win_pct's confidence interval are narrow, representing that both values are statistically meaningful. On the other hand, the DivWin and CS have a wide range and contain 0 in between the range. It could be interpreted as the variables do not have statistical meaning. This interpretation also explains DivWin and CS's p-value greater than 0.05 from the previous founding.

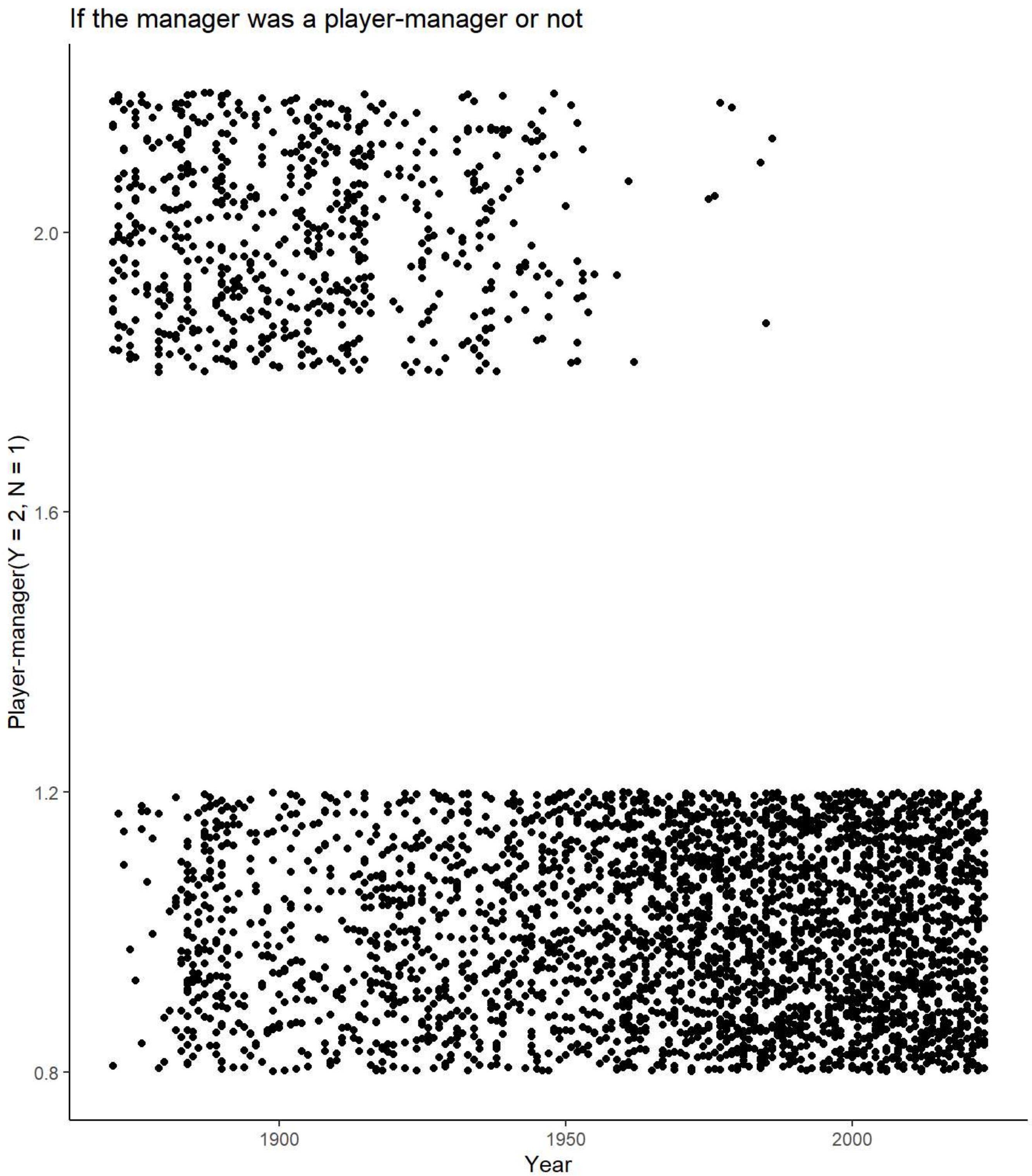
```
#q2a.
str(df_managers)
```

```
## 'data.frame':    3749 obs. of  6 variables:
##  $ playerID: chr  "wrighha01" "woodji01" "paborch01" "lennobi01" ...
##  $ teamID  : Factor w/ 149 levels "ALT","ANA","ARI",..: 24 31 39 56 56 90 97 111 136 136
...
##  $ yearID  : int  1871 1871 1871 1871 1871 1871 1871 1871 1871 1871 ...
##  $ lgID    : Factor w/ 7 levels "AA","AL","FL",..: 4 4 4 4 4 4 4 4 4 4 ...
##  $ plyrMgr : Factor w/ 2 levels "N","Y": 2 2 2 2 2 2 2 2 2 2 ...
##  $ win_pct : num  0.667 0.679 0.345 0.357 0.4 ...
```

```
summary(df_managers$plyrMgr)
```

```
##    N    Y
## 3104  645
```

```
ggplot(df_managers, aes(x = yearID, y = as.numeric(plyrMgr))) +
  geom_jitter(height = 0.2, width = 0) +
  labs(x = "Year", y = "Player-manager(Y = 2, N = 1)"
       ) +
  ggtitle("If the manager was a player-manager or not")+
  theme_classic()
```



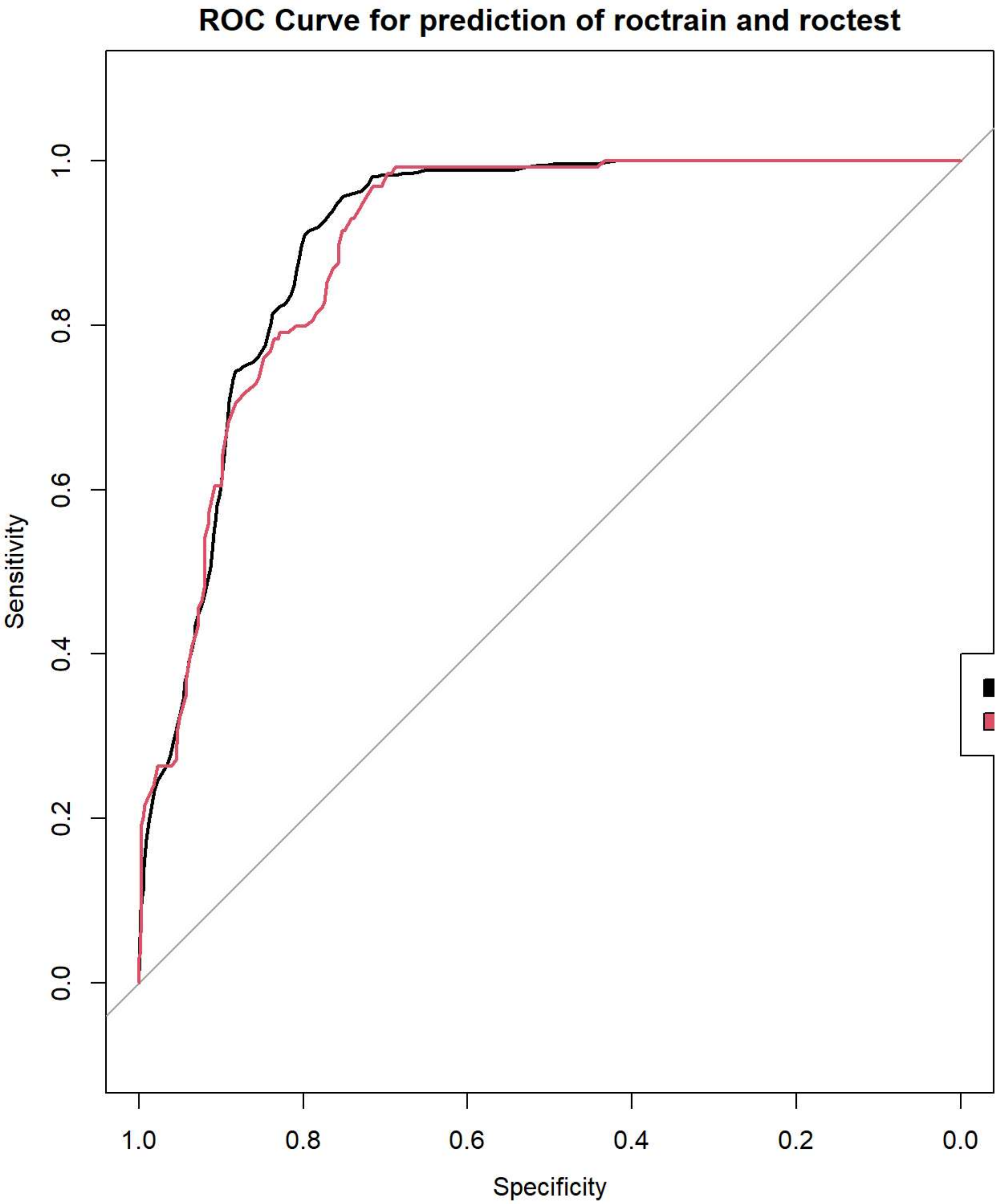
#q2a. According to the graph, in the early 1900s, the player-manager was more common, and the player with a player-manager role ratio was higher than the ones that did not. However, as time passed, a decreasing trend in the player-manager role was observed, and approximately from the 1980s, the role disappeared.

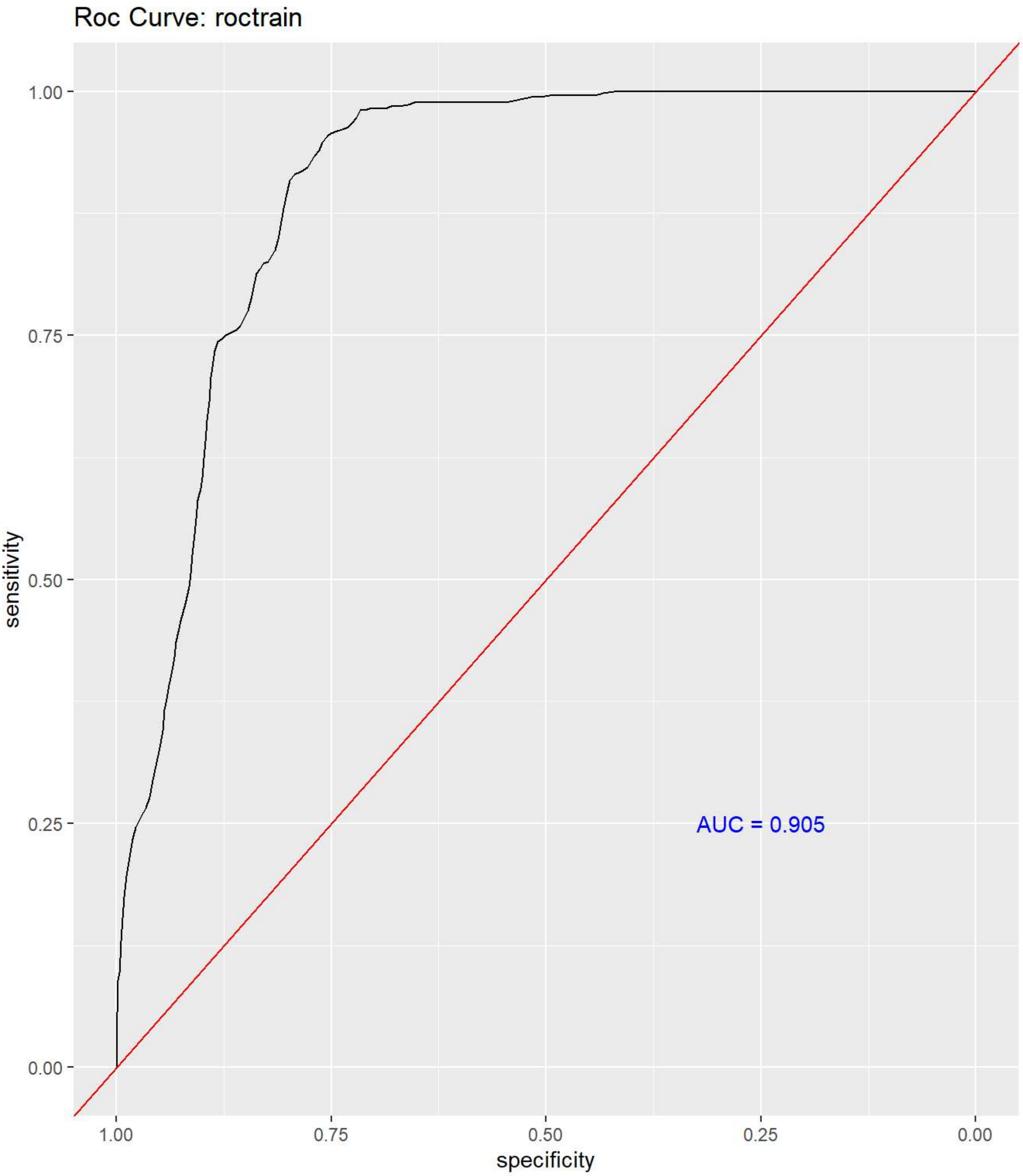
```
#q2b.
glm1 <- glm(plyrMgr~yearID, family = "binomial", data = df_managers)
summary (glm1)
```

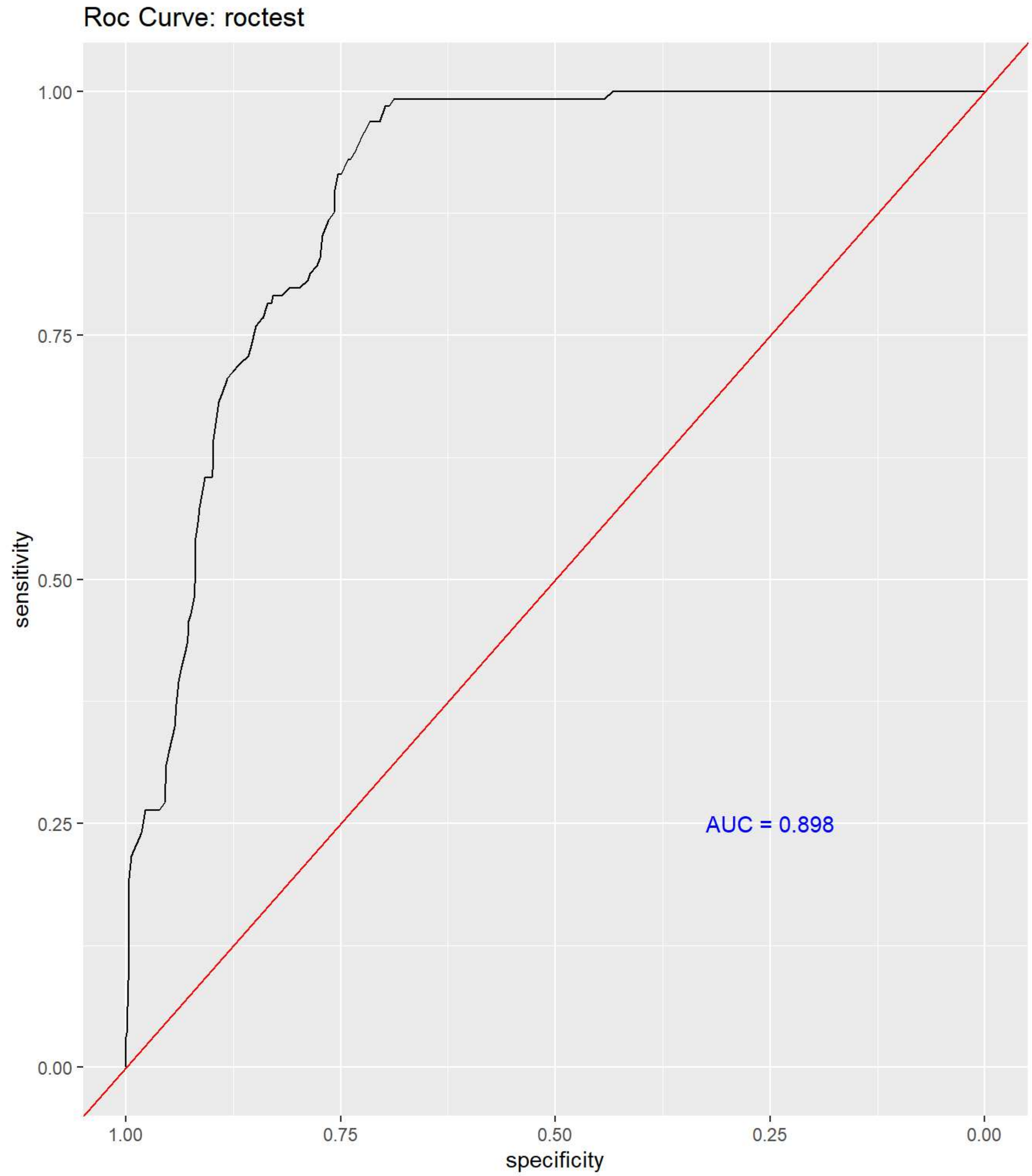
```
##
## Call:
## glm(formula = plyrMgr ~ yearID, family = "binomial", data = df_managers)
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) 88.604237   3.412071   25.97   <2e-16 ***
## yearID      -0.046611   0.001779  -26.19   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 3442.4  on 3748  degrees of freedom
## Residual deviance: 2127.7  on 3747  degrees of freedom
## AIC: 2131.7
##
## Number of Fisher Scoring iterations: 6
```

#q2b. According to the model summary, the p-value is significantly lower than the usual threshold of 0.05. Furthermore, the Residual Deviance has significantly decreased from 3442.4 (Null Deviance) to 2127.7 (Residual Deviance) by adding yearID as an independent variable. Considering the p-value and decrease of the residual, the yearID is statistically meaningful in predicting the plyrMgr. Form of fitted model is ln(p/1-p) = 88.60 - 0.047*yearID

```
#q2c.
set.seed(123)
train <- c(df_managers$plyrMgr) %>%
 createDataPartition(p = 0.8, list = FALSE)
df_managers.train <- df_managers[train,]
df_managers.test <- df_managers[-train,]


train_glm1 <- glm(plyrMgr~yearID, family = "binomial", data = df_managers.train)


predicttrain <- predict(train_glm1, newdata = df_managers.train, type = "response")
predicttest <- predict(train_glm1,  newdata = df_managers.test, type = "response")

roctrain <- roc(response = df_managers.train$plyrMgr, predictor = predicttrain, plot = TRUE,
main = "ROC Curve for prediction of roctrain and roctest", auc = TRUE)
roctest <- roc(response = df_managers.test$plyrMgr, predictor = predicttest, plot = TRUE, auc
= TRUE, add =TRUE, col = 2)
legend(0, 0.4, legend = c("training", "testing"), fill = 1:2)

ggroc(roctrain, legacy.axes = FALSE) +
  geom_abline(aes(intercept = 1, slope = 1), colour = "red") +
  labs(title = "Roc Curve: roctrain", x = "specificity", Y = "Sensitivity" ) +
  annotate(geom = "text", x = 0.25, y = 0.25, label = paste("AUC =", round(auc(roctrain),
3)), colour = "blue")
ggroc(roctest, legacy.axes = FALSE) +
  geom_abline(aes(intercept = 1, slope = 1), colour = "red") +
  labs(title = "Roc Curve: roctest", x = "specificity", Y = "Sensitivity" ) +
  annotate(geom = "text", x = 0.25, y = 0.25, label = paste("AUC =", round(auc(roctest), 3)),
colour = "blue")
```


ROC Curve for prediction of roctrain and roctest
Sensitivity
Specificity
1.0
0.8
0.6
0.4
0.2
0.0


Roc Curve: roctrain
1.00
0.75
0.50
0.25
0.00
sensitivity
1.00
0.75
0.50
0.25
0.00
specificity
AUC = 0.905



#q2c. The AUC is examined between 0 from 1. The higher AUC could be interpreted as the model showing better predictive power. Regarding the ROC curves, if the Curve breaks away from the diagonal line (random guessing) and is shaped in a curve, the model contains an adequate balance between Sensitivity and Specificity. The ROC plot for each roctrain and roctest scored AUC score each by 0.905 and 0.898. The difference in the AUC score is significantly smaller by 0.007. This means that both models showed coherently high predictive power. Furthermore, the two ROC curves are in similar shape accroding to the ROC Curve for prediction of roctrain and roctest plot. In conclusion, the model is adequate for actual application, and there is no worry of overfitting.

```
#q2d.
cutoff <- coords(roctrain, "best", best.method = "youden")

multi_df_managers <- multinom(plyrMgr~yearID, data = df_managers)
```

```
## # weights:  3 (2 variable)
## initial  value 2598.608780
## iter  10 value 1066.949689
## final  value 1063.830106
## converged
```

```
set.seed(123)
train1 <- c(df_managers$plyrMgr) %>%
 createDataPartition(p = 0.8, list = FALSE)
df_managers.train <- df_managers[train1,]
df_managers.test <- df_managers[-train1,]

train_mod1 <- multinom(plyrMgr~yearID, data = df_managers.train)
```

```
## # weights:  3 (2 variable)
## initial  value 2079.441542
## iter  10 value 848.098216
## iter  20 value 846.571139
## final  value 846.571128
## converged
```

```
predicttrain1 <- predict(train_mod1, newdata = df_managers.train, type = "class")
predicttest1 <- predict(train_mod1, newdata = df_managers.test, type = "class")

T1 <- table(predicttrain1, df_managers.train$plyrMgr)
T2 <- table(predicttest1, df_managers.test$plyrMgr)
T1
```

```
##
## predicttrain1    N    Y
##             N 2293  277
##             Y  191  239
```

```
T2
```

```
##
## predicttest1   N   Y
##            N 576  73
##            Y  44  56
```

```
sstrain <- T1[1, 1] / (T1[1, 1] + T1[2, 1]) +
  T1[2, 2] / (T1[1, 2] + T1[2, 2])
sstest <- T2[1, 1] / (T2[1, 1] + T2[2, 1])+
  T2[2, 2] / (T2[1, 2] + T2[2, 2])
sstrain
```

```
## [1] 1.386286
```

```
sstest
```

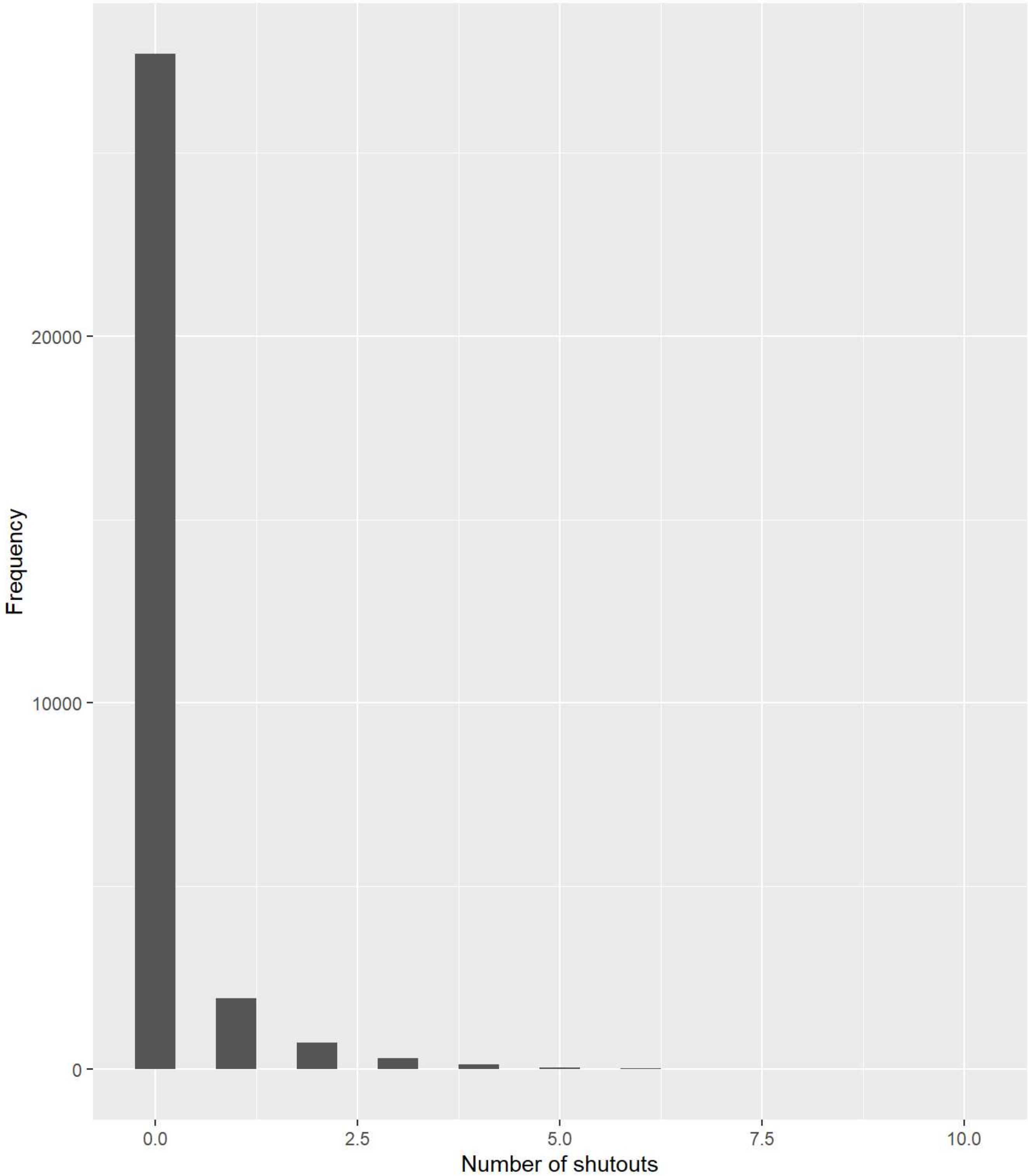
```
## [1] 1.363141
```

#q2d. The sum of sensitivity for train and test data makes no odds. Therefore, there is a weak risk of overfitting.

```
#q3a.
df_pitchers <- Pitching %>%
  filter(IPouts > 1) %>%
  mutate(innings = IPouts/3) %>%
  drop_na()

df_people <- People %>%
  select(playerID, weight, height, throws) %>%
  drop_na()

df_pitchers <- merge(df_pitchers, df_people, by  = "playerID")

head(df_pitchers)
```

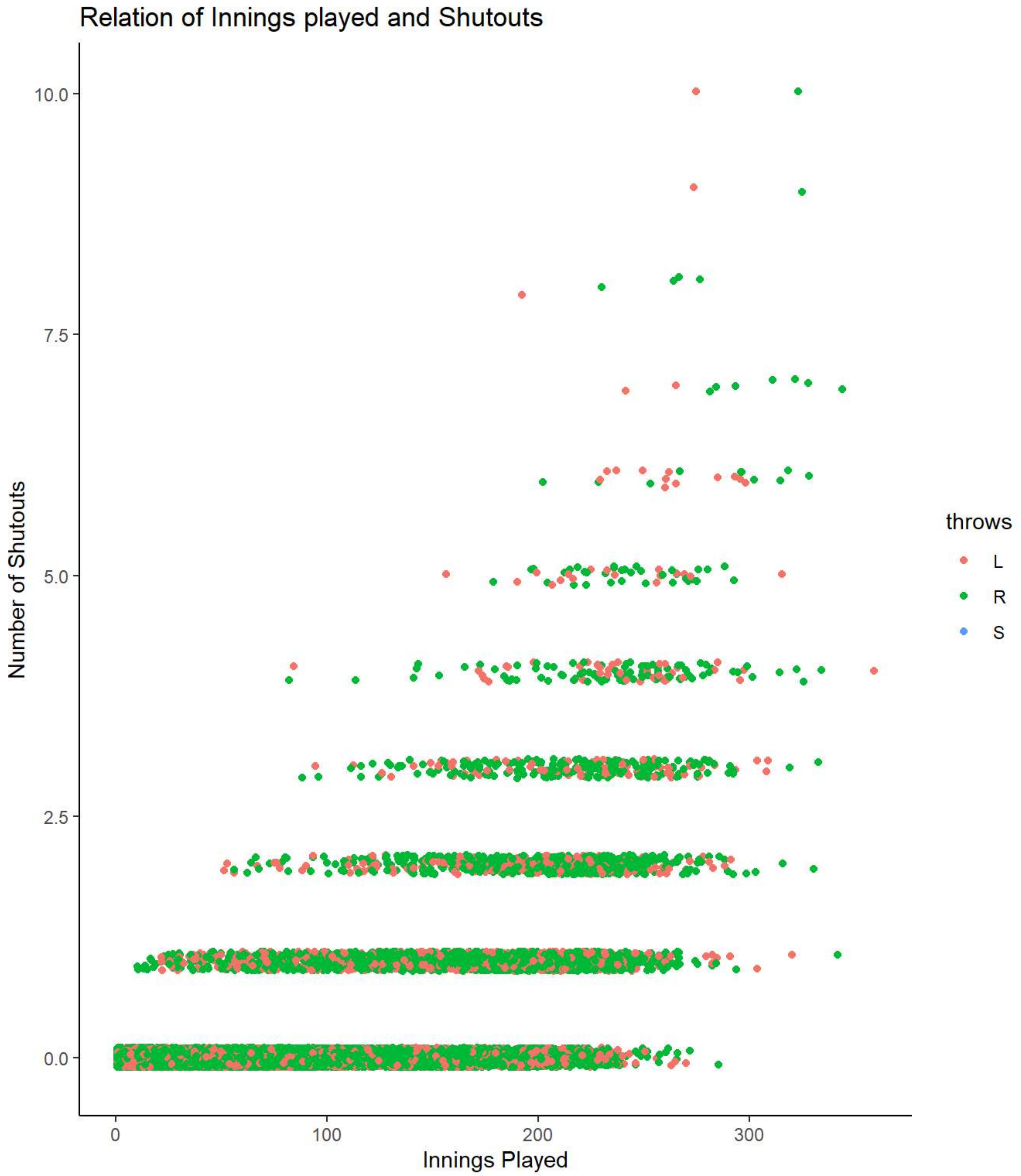
```
##    playerID yearID stint teamID lgID W L  G GS CG SHO SV IPouts   H ER HR BB SO
## 1 aardsda01   2004     1    SFN   NL 1 0 11  0  0   0  0     32 20  8  1 10  5
## 2 aardsda01   2006     1    CHN   NL 3 0 45  0  0   0  0    159 41 24  9 28 49
## 3 aardsda01   2007     1    CHA   AL 2 1 25  0  0   0  0     97 39 23  4 17 36
## 4 aardsda01   2008     1    BOS   AL 4 2 47  0  0   0  0    146 49 30  4 35 49
## 5 aardsda01   2009     1    SEA   AL 3 6 73  0  0   0 38    214 49 20  4 34 80
## 6 aardsda01   2010     1    SEA   AL 0 6 53  0  0   0 31    149 33 19  5 25 49
##   BAOpp  ERA IBB WP HBP BK BFP GF  R SH SF GIDP  innings weight height throws
## 1 0.417 6.75   0  0   2  0  61  5  8  0  1    1 10.66667    215     75      R
## 2 0.214 4.08   0  1   1  0 225  9 25  1  3    2 53.00000    215     75      R
## 3 0.300 6.40   3  2   1  0 151  7 24  2  1    1 32.33333    215     75      R
## 4 0.268 5.55   2  3   5  0 228  7 32  3  2    4 48.66667    215     75      R
## 5 0.190 2.52   3  2   0  0 296 53 23  2  1    2 71.33333    215     75      R
## 6 0.198 3.44   5  2   2  0 202 43 19  7  1    5 49.66667    215     75      R
```

```
#q3b.
df_pitchers %>%
  ggplot(aes(SHO)) +
  geom_histogram(binwidth = 0.5) +
  labs(
    x = "Number of shutouts", y = "Frequency",
    titel = "shutouts by Pitchers"
  )
```



#q3b. The Poisson Regression is an adequate model as the shutouts occur during specific periods, and at the same time, the data is count-data bigger than 0.

```
#q3c.
ggplot(df_pitchers, aes(x = innings, y = SHO, colour = throws)) +
  geom_jitter(height = 0.1,) +
  labs(x = "Innings Played",  y = "Number of Shutouts", title = "Relation of Innings played a
nd Shutouts") +
  theme_classic()
```



#q3c. As the Innings Played increased, the Number of Shutouts also grew. This trend in the graph shows a positive relationship between the variables. However, not all the pitchers showed an increase in shutouts in direct proportion to the innings. For instance, personal and team abilities could have affected the result. According to the differences in the colour by type of Throws, it was spotted that most pitchers are right-handed, and switching pitchers is rare.

```
#q3d.
poisson_mod1 <- glm(SHO ~ innings + weight + height + throws, family = "poisson", data = df_p
itchers)
anova(poisson_mod1)
```

```
## Analysis of Deviance Table
##
## Model: poisson, link: log
##
## Response: SHO
##
## Terms added sequentially (first to last)
##
##
##         Df Deviance Resid. Df Resid. Dev  Pr(>Chi)
## NULL                    30873      26075
## innings  1  15315.9     30872      10759 < 2.2e-16 ***
## weight   1     85.7     30871      10673 < 2.2e-16 ***
## height   1     57.3     30870      10616 3.685e-14 ***
## throws   2     11.9     30868      10604  0.002642 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

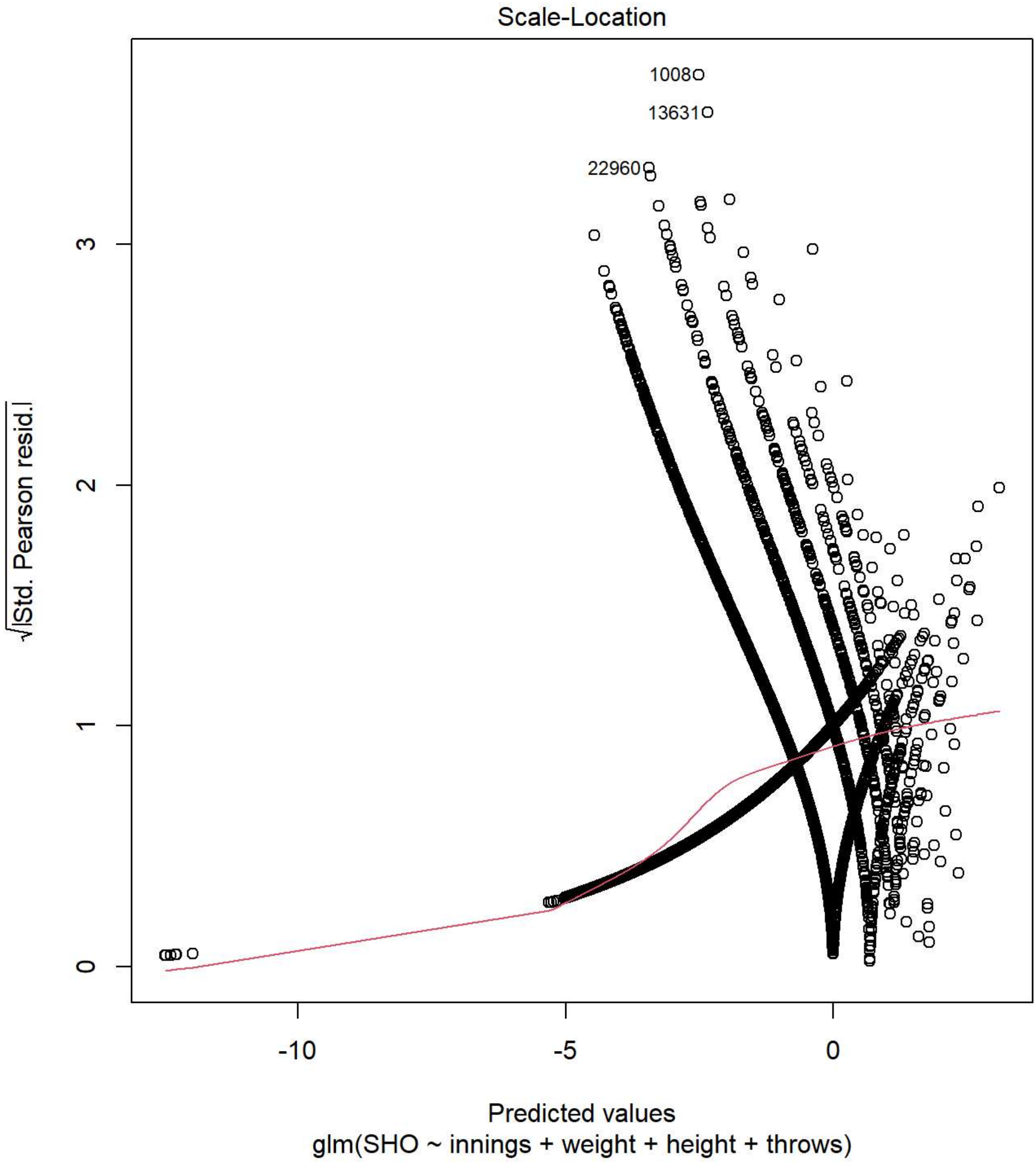
#q3d. The analysis of variance provides us with the deviances and p-value. By interpreting the p-value, it is possible to tell if the difference between the deviance of the null model and the actual models' deviance is significant. For instance, the innings deviance is 15315.9, and the p-value is <2.2e – 16. Under the null hypothesis, the p-value indicates the likelihood of getting that deviance by chance. Therefore, as the p-value is smaller, it signifies that the variable is more significant. Weight: The deviance is 85.7, and the p-value is < 2.2e-16. Under the null hypothesis, the chance of getting the deviance (85.7) is smaller than 2.2e-16. Height: The deviance is 57.3, and the p-value is 3.685e-14. Under the null hypothesis, the chance of getting the deviance (57.3) is 3.685e-14. Throws: The deviance is 11.9, and the p-value is 0.002642. Under the null hypothesis, the chance of getting the deviance (11.9) is 0.002642. The four p-values for the variables are smaller than 0.05. Therefore, the null hypothesis is not accepted, and the variables are significant. However, the decrease in the deviance of height and throws is comparatively low, which could be seen as less significant than innings and weight.

```
#q3e.
poisson_mod2 <- glmer(SH ~ innings + weight + height + throws + (1 | teamID), family = "poiss
on", data = df_pitchers)
summary(poisson_mod2)
```

```
## Generalized linear mixed model fit by maximum likelihood (Laplace
##   Approximation) [glmerMod]
##  Family: poisson  ( log )
## Formula: SH ~ innings + weight + height + throws + (1 | teamID)
##    Data: df_pitchers
##
##      AIC      BIC   logLik deviance df.resid
## 110155.9 110214.2 -55070.9 110141.9    30867
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -4.2171 -0.9684 -0.3551  0.6053  9.0082
##
## Random effects:
##  Groups Name        Variance Std.Dev.
##  teamID (Intercept) 0.06698  0.2588
## Number of obs: 30874, groups:  teamID, 35
##
## Fixed effects:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept) -5.220e-01  1.444e-01  -3.615 0.000301 ***
## innings      1.042e-02  4.419e-05 235.729  < 2e-16 ***
## weight      -5.061e-03  2.042e-04 -24.792  < 2e-16 ***
## height       1.971e-02  2.080e-03   9.475  < 2e-16 ***
## throwsR     -8.349e-02  8.070e-03 -10.347  < 2e-16 ***
## throwsS     -1.856e+00  1.003e+00  -1.850 0.064364 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##         (Intr) innngs weight height thrwsR
## innings  0.047
## weight   0.256  0.104
## height  -0.919 -0.107 -0.508
## throwsR  0.068  0.006 -0.022 -0.094
## throwsS -0.004  0.005  0.001  0.003  0.005
## optimizer (Nelder_Mead) convergence code: 0 (OK)
## Model is nearly unidentifiable: very large eigenvalue
##  - Rescale variables?
## Model is nearly unidentifiable: large eigenvalue ratio
##  - Rescale variables?
```

#q3e. The variance of Team ID makes it possible to find the by applying square root to standard deviation. According to the code result, the standard deviation is below 0.5 by 0.2588, indicating that each team's expected shutouts do not differ much. If a particular team has more shutouts, it could increase the variance and improve the random effect. However, in this case, the variability of the team ID's random effect will be less critical. Therefore, the team ID is not a vital predictor. The form of fitted model is log(μ) = − 0.52 + 0.01*innings − 0.005* weight + 0.02 * height − 0.35 * throwsR − 0.19 * throwsS

```
#q3f.
plot(poisson_mod1,  which = 3)
```



#q3f. The graph is used to evaluate homoscedasticity. To meet homoscedasticity, the red line (mean of the residuals) should be straight, and the residuals should be randomly distributed. However, the scatters on the graph follow certain curves rather than random distribution. In addition, the red line has a precise curved shape. Furthermore, increasing the expected value increases the residuals' distribution. These unusual patterns violate homoscedasticity.

```
#q3g.
summary(poisson_mod1)
```

```
##
## Call:
## glm(formula = SHO ~ innings + weight + height + throws, family = "poisson",
##     data = df_pitchers)
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) -7.062e+00  5.158e-01 -13.692  < 2e-16 ***
## innings      2.052e-02  1.862e-04 110.181  < 2e-16 ***
## weight      -9.470e-03  8.155e-04 -11.613  < 2e-16 ***
## height       6.247e-02  7.892e-03   7.915 2.47e-15 ***
## throwsR     -1.022e-01  2.970e-02  -3.439 0.000583 ***
## throwsS     -8.229e+00  1.155e+02  -0.071 0.943219
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 26075  on 30873  degrees of freedom
## Residual deviance: 10604  on 30868  degrees of freedom
## AIC: 18022
##
## Number of Fisher Scoring iterations: 10
```

```
exp(-0.1022)
```

```
## [1] 0.902849
```

```
exp(-0.00947)
```

```
## [1] 0.9905747
```

```
exp(0.06247)
```

```
## [1] 1.064463
```

#q3g. According to the summary of the poisson_mod1, the baseline of the throws is throws L, and the exponential value of the throws R is around 90% of the expected shutout chance of throws L. Therefore, left-handed pitchers are expected to pitch 10% more times of shutouts. In terms of weight, the coefficient is negative, and the exponential value is around 0.99. Consequently, the expected chance of shutouts decreases by 1% for every unit increase. On the other hand, the height coefficient is positive, and the exponential value is around 1.06. In conclusion, around 6% of the shutout chance increases for each height unit.