S429 Individual Project

Kristina Herard

2025-11-04

```
## Warning: package ’car’ was built under R version 4.3.3
```

```
## Loading required package: carData
```

```
## Warning: package ’caret’ was built under R version 4.3.3
```

```
## Warning: package ’xfun’ was built under R version 4.3.3
```

```
##
## Attaching package: ’xfun’
```

```
## The following object is masked from ’package:base’:
##
##     attr
```

```
## Warning: package ’tidyr’ was built under R version 4.3.3
```

```r
library(readr)
```

```
## Warning: package ’readr’ was built under R version 4.3.3
```

```r
bluejays <- read_csv("C:/Users/krist/Downloads/bluejays.csv")
```

```
## Rows: 4236 Columns: 118
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (16): pitch_type, player_name, events, description, des, game_type, sta...
## dbl  (86): release_speed, release_pos_x, release_pos_z, batter, pitcher, zon...
## lgl  (15): spin_dir, spin_rate_deprecated, break_angle_deprecated, break_len...
## date  (1): game_date
##
## 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.
```

```r
attach(bluejays)
bluejays[1:5,]
```

```
## # A tibble: 5 x 118
##   pitch_type game_date  release_speed release_pos_x release_pos_z player_name
##   <chr>      <date>             <dbl>         <dbl>         <dbl> <chr>
## 1 SI         2022-10-01          95.3         -1.96          5.49 Chapman, Matt
## 2 SL         2022-10-01          85.2         -1.48          5.57 Guerrero Jr.,~
## 3 SI         2022-10-01          95           -2.07          5.4  Bichette, Bo
## 4 FF         2022-10-01          96.5         -1.83          5.65 Springer, Geo~
## 5 SI         2022-10-01          96.8         -1.99          5.39 Merrifield, W~
## # i 112 more variables: batter <dbl>, pitcher <dbl>, events <chr>,
## #   description <chr>, spin_dir <lgl>, spin_rate_deprecated <lgl>,
## #   break_angle_deprecated <lgl>, break_length_deprecated <lgl>, zone <dbl>,
## #   des <chr>, game_type <chr>, stand <chr>, p_throws <chr>, home_team <chr>,
## #   away_team <chr>, type <chr>, hit_location <dbl>, bb_type <chr>,
## #   balls <dbl>, strikes <dbl>, game_year <dbl>, pfx_x <dbl>, pfx_z <dbl>,
## #   plate_x <dbl>, plate_z <dbl>, on_3b <dbl>, on_2b <dbl>, on_1b <dbl>, ...
```

```r
names(bluejays)
```

```
##   [1] "pitch_type"
##   [2] "game_date"
##   [3] "release_speed"
##   [4] "release_pos_x"
##   [5] "release_pos_z"
##   [6] "player_name"
##   [7] "batter"
##   [8] "pitcher"
##   [9] "events"
##  [10] "description"
##  [11] "spin_dir"
##  [12] "spin_rate_deprecated"
##  [13] "break_angle_deprecated"
##  [14] "break_length_deprecated"
##  [15] "zone"
##  [16] "des"
##  [17] "game_type"
##  [18] "stand"
##  [19] "p_throws"
##  [20] "home_team"
##  [21] "away_team"
##  [22] "type"
##  [23] "hit_location"
##  [24] "bb_type"
##  [25] "balls"
##  [26] "strikes"
##  [27] "game_year"
##  [28] "pfx_x"
##  [29] "pfx_z"
##  [30] "plate_x"
##  [31] "plate_z"
##  [32] "on_3b"
##  [33] "on_2b"
##  [34] "on_1b"
##  [35] "outs_when_up"
##  [36] "inning"
## [37] "inning_topbot"
## [38] "hc_x"
## [39] "hc_y"
## [40] "tfs_deprecated"
## [41] "tfs_zulu_deprecated"
## [42] "umpire"
## [43] "sv_id"
## [44] "vx0"
## [45] "vy0"
## [46] "vz0"
## [47] "ax"
## [48] "ay"
## [49] "az"
## [50] "sz_top"
## [51] "sz_bot"
## [52] "hit_distance_sc"
## [53] "launch_speed"
## [54] "launch_angle"
## [55] "effective_speed"
## [56] "release_spin_rate"
## [57] "release_extension"
## [58] "game_pk"
## [59] "fielder_2"
## [60] "fielder_3"
## [61] "fielder_4"
## [62] "fielder_5"
## [63] "fielder_6"
## [64] "fielder_7"
## [65] "fielder_8"
## [66] "fielder_9"
## [67] "release_pos_y"
## [68] "estimated_ba_using_speedangle"
## [69] "estimated_woba_using_speedangle"
## [70] "woba_value"
## [71] "woba_denom"
## [72] "babip_value"
## [73] "iso_value"
## [74] "launch_speed_angle"
## [75] "at_bat_number"
## [76] "pitch_number"
## [77] "pitch_name"
## [78] "home_score"
## [79] "away_score"
## [80] "bat_score"
## [81] "fld_score"
## [82] "post_away_score"
## [83] "post_home_score"
## [84] "post_bat_score"
## [85] "post_fld_score"
## [86] "if_fielding_alignment"
## [87] "of_fielding_alignment"
## [88] "spin_axis"
## [89] "delta_home_win_exp"
## [90] "delta_run_exp"
##  [91] "bat_speed"
##  [92] "swing_length"
##  [93] "estimated_slg_using_speedangle"
##  [94] "delta_pitcher_run_exp"
##  [95] "hyper_speed"
##  [96] "home_score_diff"
##  [97] "bat_score_diff"
##  [98] "home_win_exp"
##  [99] "bat_win_exp"
## [100] "age_pit_legacy"
## [101] "age_bat_legacy"
## [102] "age_pit"
## [103] "age_bat"
## [104] "n_thruorder_pitcher"
## [105] "n_priorpa_thisgame_player_at_bat"
## [106] "pitcher_days_since_prev_game"
## [107] "batter_days_since_prev_game"
## [108] "pitcher_days_until_next_game"
## [109] "batter_days_until_next_game"
## [110] "api_break_z_with_gravity"
## [111] "api_break_x_arm"
## [112] "api_break_x_batter_in"
## [113] "arm_angle"
## [114] "attack_angle"
## [115] "attack_direction"
## [116] "swing_path_tilt"
## [117] "intercept_ball_minus_batter_pos_x_inches"
## [118] "intercept_ball_minus_batter_pos_y_inches"
```

```r
sum(is.na(bluejays))
```

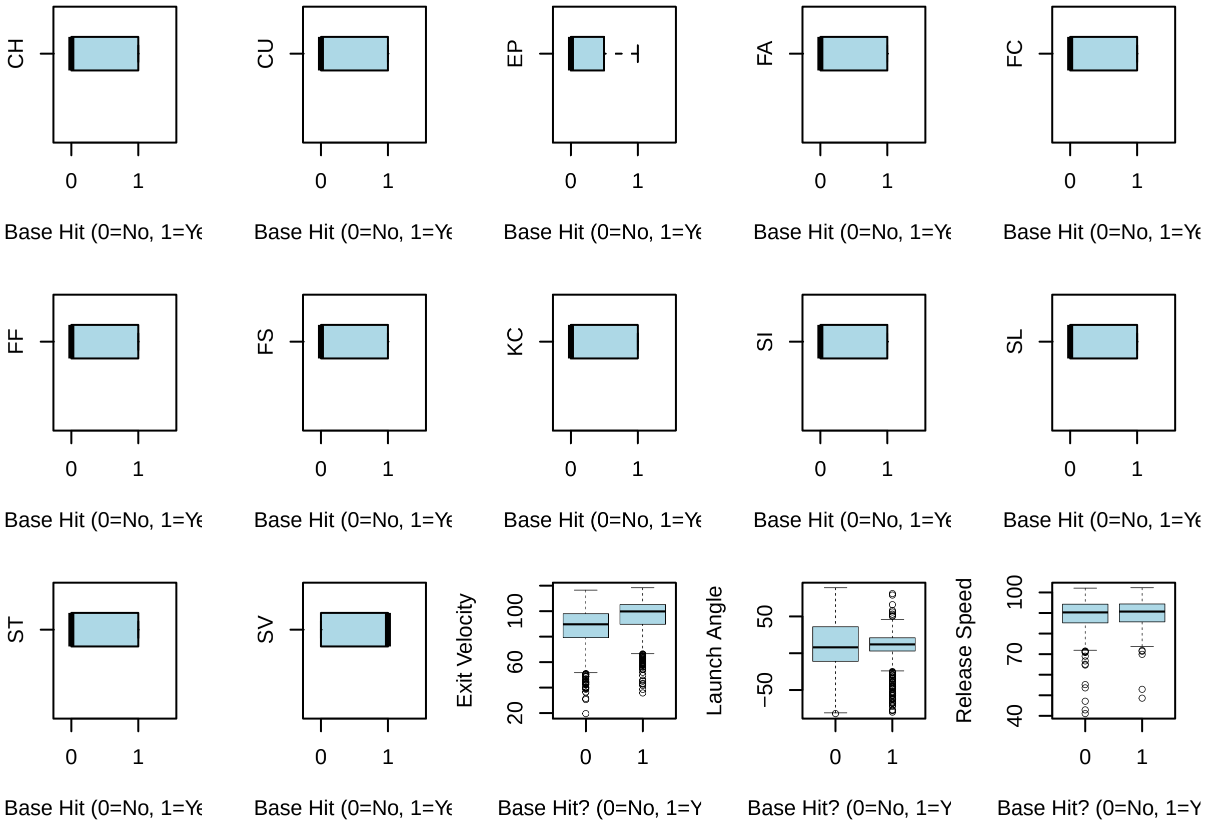
```
## [1] 74498
```

```r
bluejays$hit <- ifelse(bluejays$events %in% c("single","double","triple","home_run"), 1, 0)

length(unique(bluejays$pitch_type))
```

```
## [1] 12
```

Boxplots including pitch type, exit velocity, launch angle, release speed and whether they will result in a base hit or not.



Interpretation:

From the box plots of pitch types, we can see that the pitches of changeup, curveball, fastball, cut fastball, four-seam fastball, split-finger fastball, knuckle curve, sinker, slider, and sweeper all have one box for "Base Hit = 0", with low variability. This means that these pitches rarely resulted in base hits in this dataset, and the pitch type performs consistently while used (as shown from limited spread of the box).

The eephus pitch boxplot shows an even smaller box, with a longer upper tail. This means thatmost of the data is very close together, meaning low variability just as the other pitches do. However, the long upper tail suggests that there are a few data points that are much higher than the main cluster. This data is right skewed, and could indicate rare pitches with unusually high speed, or extreme launch angles. This could also indicate the few times that this pitch was hit with an unusually high exit velocity.

Finally, the slurve pitch also has low variability, however, the median line is closer to "Base Hit = 1", meaning while performance is similarily consistent across outcomes, the slurve pitch is slightly more likely to result in hits at more favorable values.

The median exit velocity for hits is clearly higher than for non-hits. The entire box for hits is shifted upward, meaning most batted balls that become hits leave the bat faster. oOutliers on the low end for both groups show a few slow exit velocities, but hits are generally clustered higher. In conclusion, higher exit velocity increses liklihood of a base hit, harder hit balls are more likely to get through the defense.

The median launch angle for hits is positive and around 10-20 degrees, while for non-hits, it is centered around 0 or slightly negative. The spread for hits is narrower, meaning successful hits tend to ocur within a more optimal range of angles. Non-hits have a wider range of launch angles, including many negative ones (grounders) or very high ones (pop-ups). In conclusion, balls hit at moderate upward angles are more likely to result in hits, while very low or very high angles usually lead to outs.

Like exit velocity, release speed is higher on average for hits. The median and upper quartile are both shifted upwards for base hits. The distributions are fairly tight, suggesting consistent release speeds among hit events. In conclusion, pitchers or batted balls with higher release speed correlate with hits, likely reflecting stronger contact or better swing mechanics.

Run the Logistic Regression:

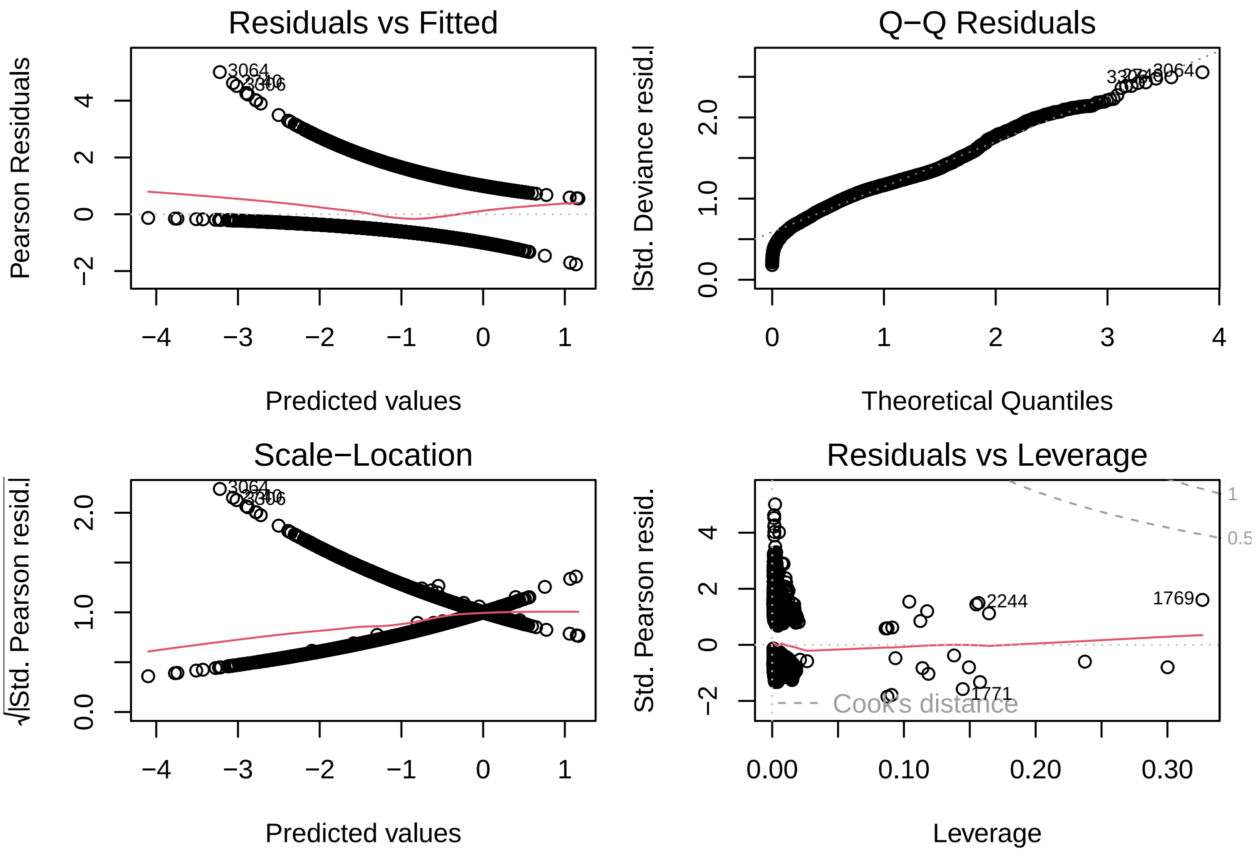
```
m1 <- glm(bluejays$hit~launch_speed+launch_angle+release_speed+pitch_type, data = bluejays, family = bi
summary(m1)
```

```
##
## Call:
## glm(formula = bluejays$hit ~ launch_speed + launch_angle + release_speed +
##     pitch_type, family = binomial, data = bluejays)
##
## Coefficients:
##                Estimate Std. Error z value Pr(>|z|)
## (Intercept)   -5.874568   0.998131  -5.886 3.97e-09 ***
## launch_speed   0.050211   0.002889  17.381  < 2e-16 ***
## launch_angle  -0.004481   0.001356  -3.305  0.00095 ***
## release_speed  0.007976   0.011305   0.706  0.48047
## pitch_typeCU   0.046027   0.206579   0.223  0.82369
## pitch_typeEP   0.158739   1.268367   0.125  0.90040
## pitch_typeFA   0.738768   0.829791   0.890  0.37330
## pitch_typeFC   0.003001   0.158613   0.019  0.98490
## pitch_typeFF  -0.155477   0.160506  -0.969  0.33271
## pitch_typeFS  -0.131818   0.285388  -0.462  0.64416
## pitch_typeKC   0.126188   0.265538   0.475  0.63463
## pitch_typeSI  -0.179558   0.163108  -1.101  0.27096
## pitch_typeSL   0.004490   0.129857   0.035  0.97242
## pitch_typeST  -0.248903   0.196146  -1.269  0.20445
## pitch_typeSV   1.039721   0.694411   1.497  0.13432
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 5389.1  on 4219  degrees of freedom
## Residual deviance: 5011.6  on 4205  degrees of freedom
##   (16 observations deleted due to missingness)
## AIC: 5041.6
##
## Number of Fisher Scoring iterations: 4
```

From the p-values in the regression, it can be seen that none of the pitch types and release speed are not actually significant in terms of hit probability, however, launch angle and launch speed have very small p-values, showing that those predictors are significant when determining the probability of getting a base hit.

Plot the regression

```
par(mfrow = c(2,2), mar=c(4,4,2,1))
plot(m1)
```



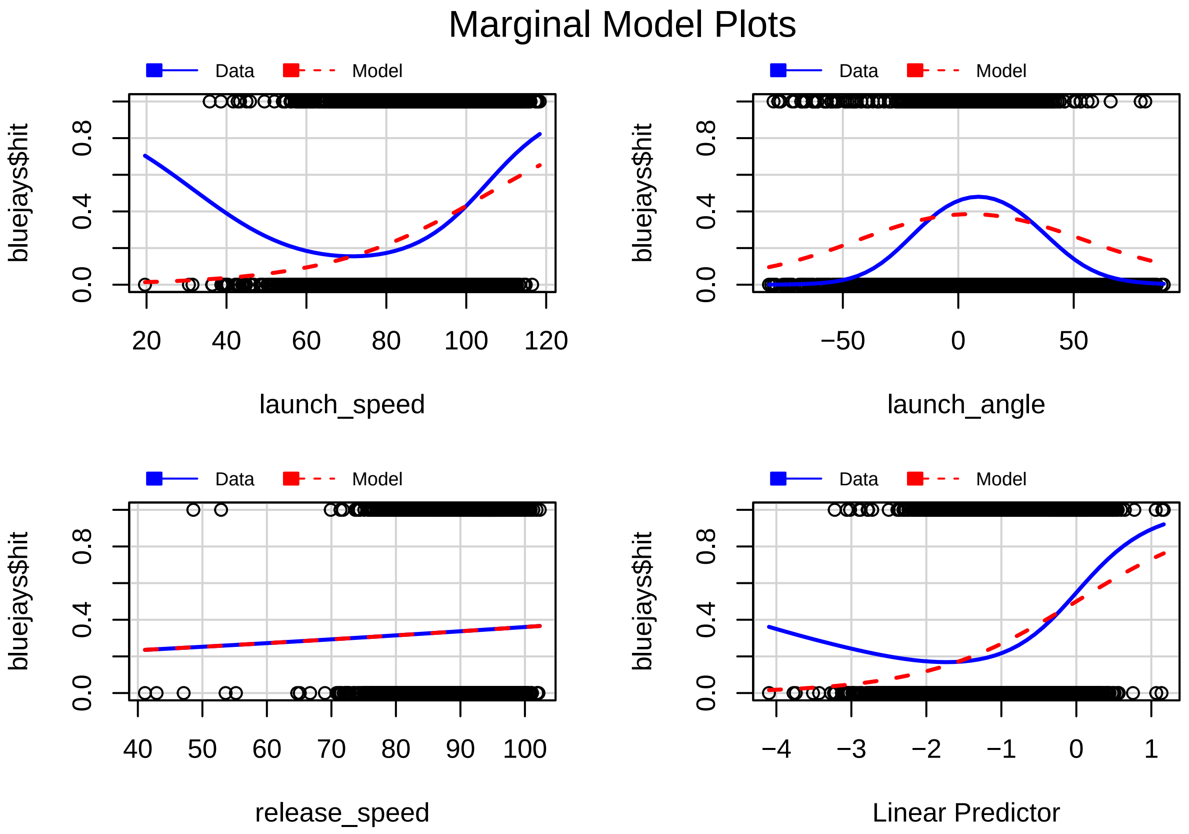
The residuals vs fitted plot shows that there is a clear curved pattern in the graph, and the residuals are not randomly scattered. This suggests non-linearity, meaning this model may not capture the relationship between the predictors and the log-odds of a base hit.

The Q-Q plot shows that the residuals deviate upwards ar both tails. this tells us that the models residuals have heavier tails than expected, siggesting again that this model may have outliers or that it is not a good fit for the data.

The scale-location plot shows a distinct pattern, not a flat plot. This indicates the variance is not constant, implying predictions for extreme fitted probabilities may be less reliable.

The residuals vs, leverage plot shows that most points have low leverage and small residuals, which is good. While there are a few potentially influential points in the plot, none appear to exceed Cook's distance of 1, which tells us they are not extreme outliers. marginal model plots

```
mmps(m1, terms = ~ launch_speed + launch_angle + release_speed)
```



In the first plot, We can see a steep increase in the curve at higher launch speeds. Because the blue and red lines are in no way similar, this relationship can be described at positively nonlinear, meaning higher launch speed substantially increases the probability of a hit, but the effect is not perfectly linear.

In the second plot, a clear bell shaped pattern exists, also suggesting a non-linear relationship between launch angle and a base hit. From the plot, it can be seen that a base hit is most likely to appear at a moderate launch angle, which matches real world intuition.

In the third plot, both blue and red lines are almost flat across the range of release speeds. This implies that there is no meaningful relationship between the pitchers release speed and whether the ball becomes a base hit. This conclusion is consistent with the insignificant p-value in the regression output.

This plot seems to fit the data the best out of all the plots. The blue curve shows the expected s-shaped logistic relationship between predicted log-odds, and hit probability.

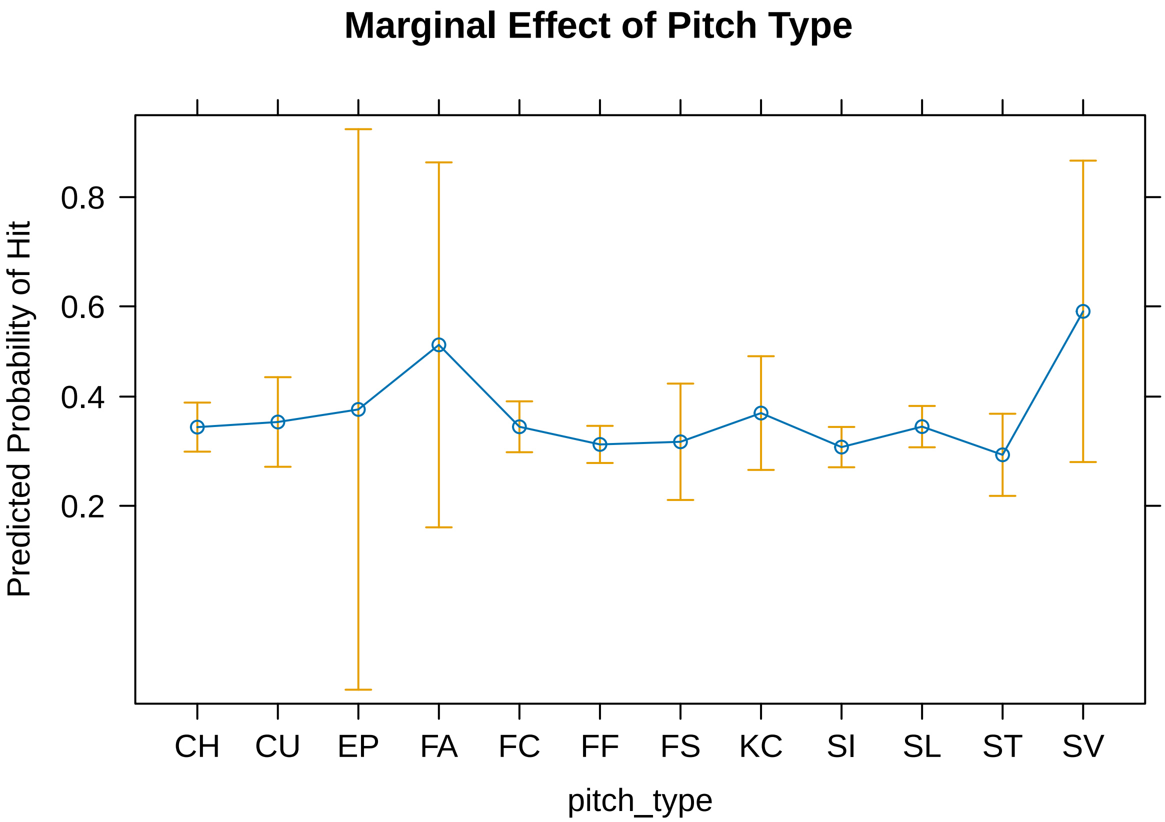
```
library(effects)
```

```
## Warning in check_dep_version(): ABI version mismatch:
## lme4 was built with Matrix ABI version 1
## Current Matrix ABI version is 0
## Please re-install lme4 from source or restore original 'Matrix' package
```

```
## Use the command
##     lattice::trellis.par.set(effectsTheme())
##   to customize lattice options for effects plots.
## See ?effectsTheme for details.
```

```
plot(effect("pitch_type", m1), main = "Marginal Effect of Pitch Type", ylab = "Predicted Probability of
```

```
## Warning: Using ‘$‘ in model formulas can produce unexpected results. Specify your
##   model using the ‘data‘ argument instead.
##   Try: hit ~ launch_speed +
##   launch_angle + release_speed + pitch_type, data =
```



This plot shows the predicted probabilities vary only modestly across pitch types. Most types yield probabilities between 0.3 and 0.4, suggesting pitch type has limited effect on hit likelihood after accounting for other factors such as launch speed and angle. This is consistent with the p-values we obtained in the regression earlier. FA (fastball) and SV (sweeper) show slightly higher average hit probabilities, but their error bars are large. EP (eephus) has an extremely wide confidence interval, likely because it occurs very rarely in this data.

Because almost all error bars overlap, none of these differences are statistically significant at the 95% level.

```
backwardAIC <- step(m1, direction = "backward")
```

```
## Start:  AIC=5041.58
## bluejays$hit ~ launch_speed + launch_angle + release_speed +
##     pitch_type
##
##                 Df Deviance    AIC
## - pitch_type    11   5019.6 5027.6
## - release_speed  1   5012.1 5040.1
## <none>               5011.6 5041.6
## - launch_angle   1   5022.5 5050.5
## - launch_speed   1   5372.4 5400.4
##
## Step:  AIC=5027.6
## bluejays$hit ~ launch_speed + launch_angle + release_speed
##
##                  Df Deviance    AIC
## - release_speed  1   5019.8 5025.8
## <none>               5019.6 5027.6
## - launch_angle   1   5032.4 5038.4
## - launch_speed   1   5382.2 5388.2
##
## Step:  AIC=5025.8
## bluejays$hit ~ launch_speed + launch_angle
##
##                 Df Deviance    AIC
## <none>              5019.8 5025.8
## - launch_angle  1   5032.4 5036.4
## - launch_speed  1   5385.4 5389.4
```

```r
n <- length(bluejays$hit)
backBIC <- step(m1, direction = "backward", k=log(n))
```

```
## Start:  AIC=5136.85
## bluejays$hit ~ launch_speed + launch_angle + release_speed +
##     pitch_type
##
##                  Df Deviance    AIC
## - pitch_type     11   5019.6 5053.0
## - release_speed  1   5012.1 5129.0
## <none>               5011.6 5136.9
## - launch_angle   1   5022.5 5139.5
## - launch_speed   1   5372.4 5489.3
##
## Step:  AIC=5053.01
## bluejays$hit ~ launch_speed + launch_angle + release_speed
##
##                  Df Deviance    AIC
## - release_speed  1   5019.8 5044.9
## <none>               5019.6 5053.0
## - launch_angle   1   5032.4 5057.4
## - launch_speed   1   5382.2 5407.2
##
## Step:  AIC=5044.86
## bluejays$hit ~ launch_speed + launch_angle
##
##                 Df Deviance    AIC
## <none>              5019.8 5044.9
## - launch_angle  1   5032.4 5049.1
## - launch_speed  1   5385.4 5402.1
```

From the AIC selection process, the predictors included in the best model are only launch speed and launch angle. This makes sense from our regression output earlier, as these are the only 2 predictors with significant p-values. Based on this result, we should remove the predictors of pitch type and release speed.

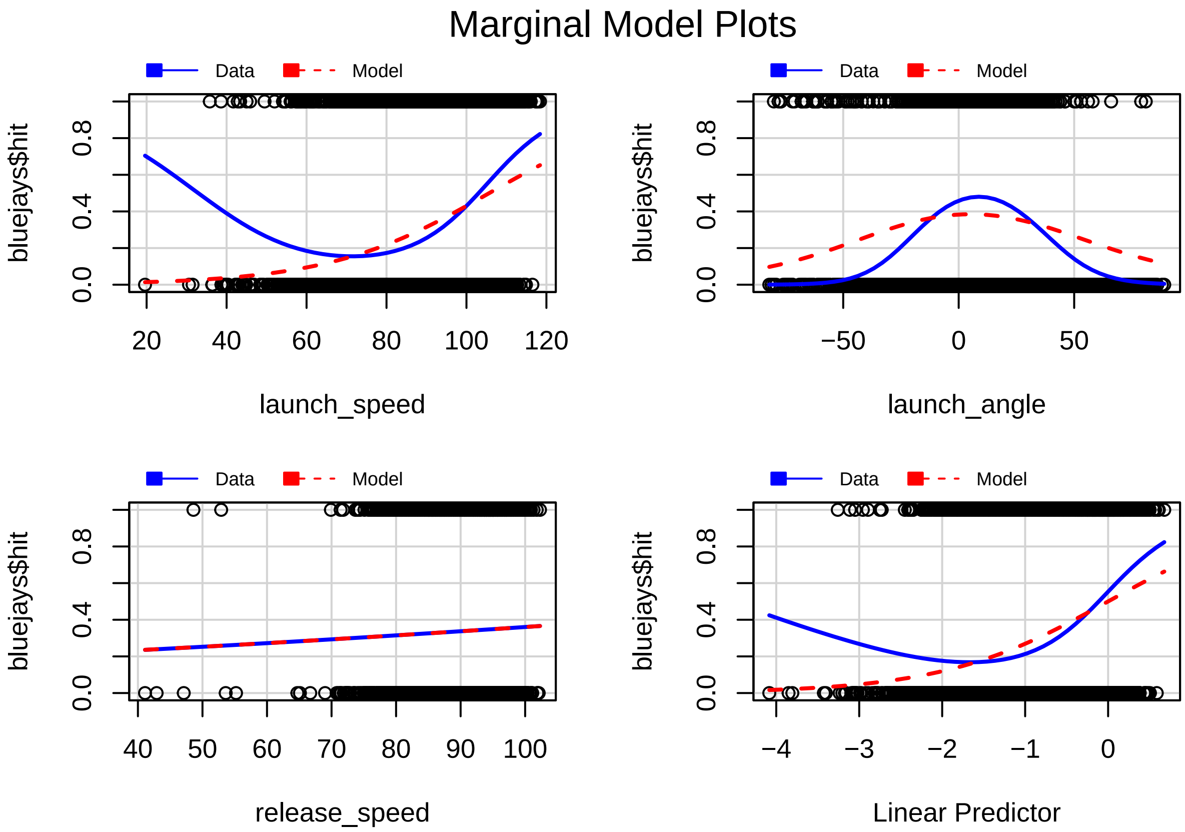
creating model 2 without pitch type

```
m2 <- glm(bluejays$hit~launch_speed+launch_angle+release_speed, data = bluejays, family = binomial)
summary(m2)
```

```
##
## Call:
## glm(formula = bluejays$hit ~ launch_speed + launch_angle + release_speed,
##     family = binomial, data = bluejays)
##
## Coefficients:
##                Estimate Std. Error z value Pr(>|z|)
## (Intercept)   -5.010140   0.549226  -9.122  < 2e-16 ***
## launch_speed   0.050121   0.002878  17.417  < 2e-16 ***
## launch_angle  -0.004693   0.001318  -3.561  0.00037 ***
## release_speed -0.002530   0.005675  -0.446  0.65576
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 5389.1  on 4219  degrees of freedom
## Residual deviance: 5019.6  on 4216  degrees of freedom
##   (16 observations deleted due to missingness)
## AIC: 5027.6
##
## Number of Fisher Scoring iterations: 4
```

From the regression of model 2, we can pull the same conclusion as model 1, and say that the pitchers release speed is statistically insignificant in predicting whether a batter will get a base hit or not. marginal model plots of m2

```
mmps(m2)
```



These plots the same as those of model 1 which makes sense since they include the same predictors.

creating model 3 without release speed

```
m3 <- glm(bluejays$hit~launch_speed+launch_angle, data = bluejays, family = binomial)
summary(m3)
```

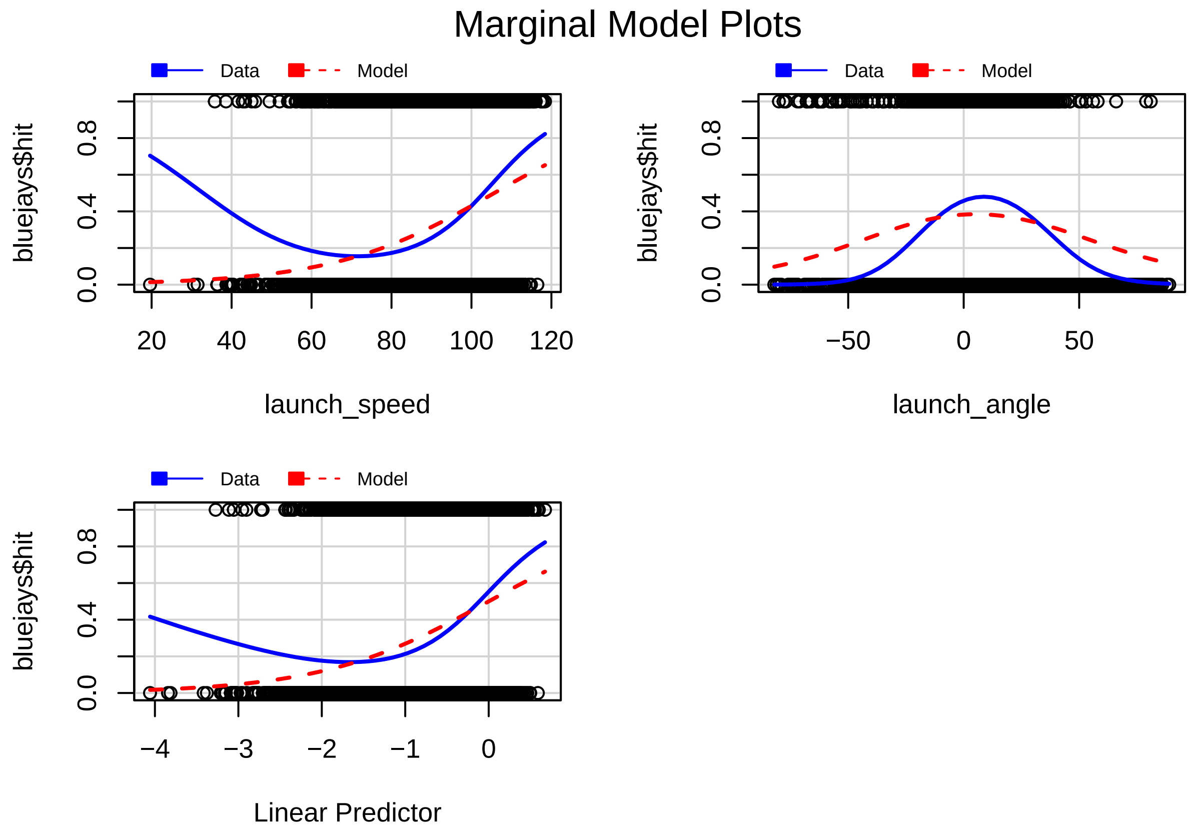
```
##
## Call:
## glm(formula = bluejays$hit ~ launch_speed + launch_angle, family = binomial,
##     data = bluejays)
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -5.223877   0.268919 -19.425  < 2e-16 ***
## launch_speed  0.049976   0.002858  17.487  < 2e-16 ***
## launch_angle -0.004646   0.001313  -3.538 0.000404 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 5389.1  on 4219  degrees of freedom
## Residual deviance: 5019.8  on 4217  degrees of freedom
##   (16 observations deleted due to missingness)
## AIC: 5025.8
##
```

```
## Number of Fisher Scoring iterations: 4
```

Model m3 now only includes predictors of statistical significance.

marginal model plot of m3

```
mmps(m3)
```

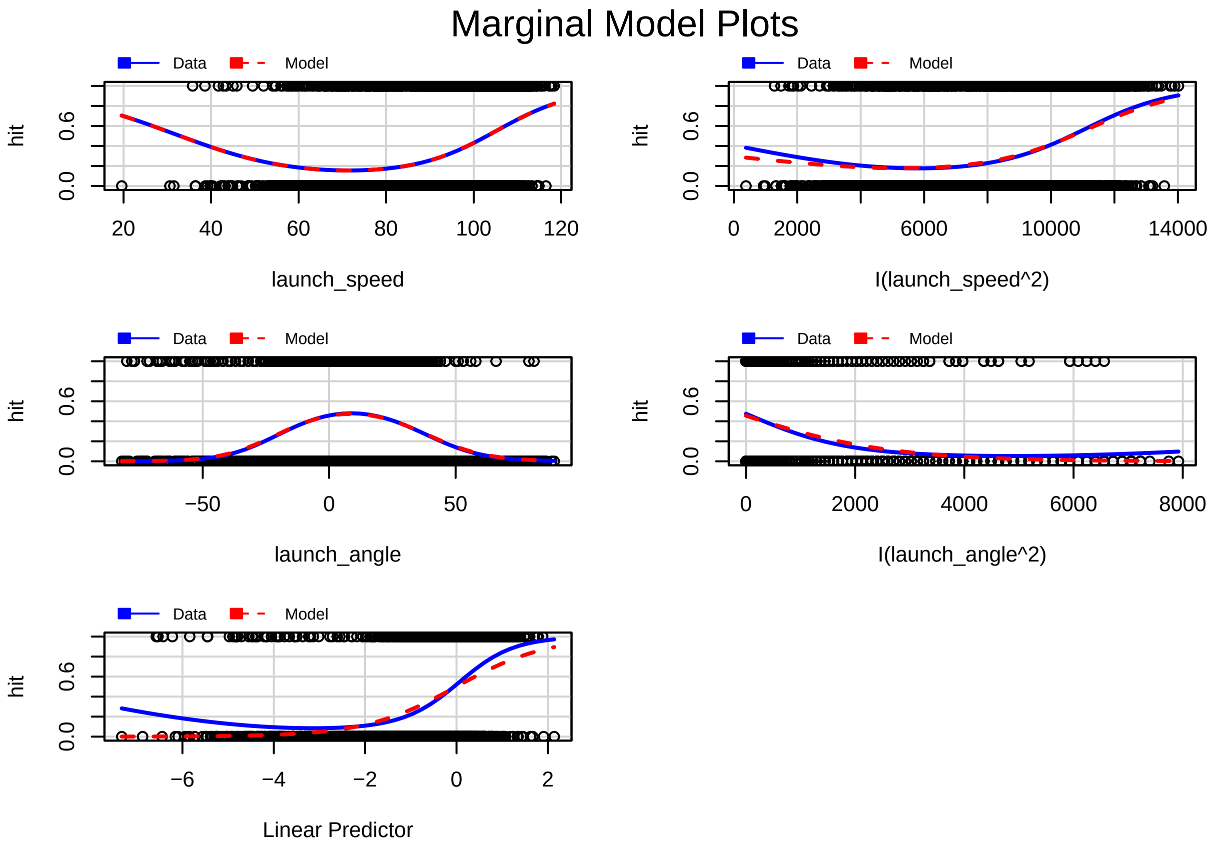


```
bluejays2 <- data.frame(
     hit = bluejays$hit,
     launch_speed = bluejays$launch_speed,
     launch_angle = bluejays$launch_angle
)
bluejays2 <- na.omit(bluejays2)

m4 <- glm(hit ~ launch_speed + I(launch_speed^2) + launch_angle + I(launch_angle^2), data = bluejays2,
summary(m4)
```

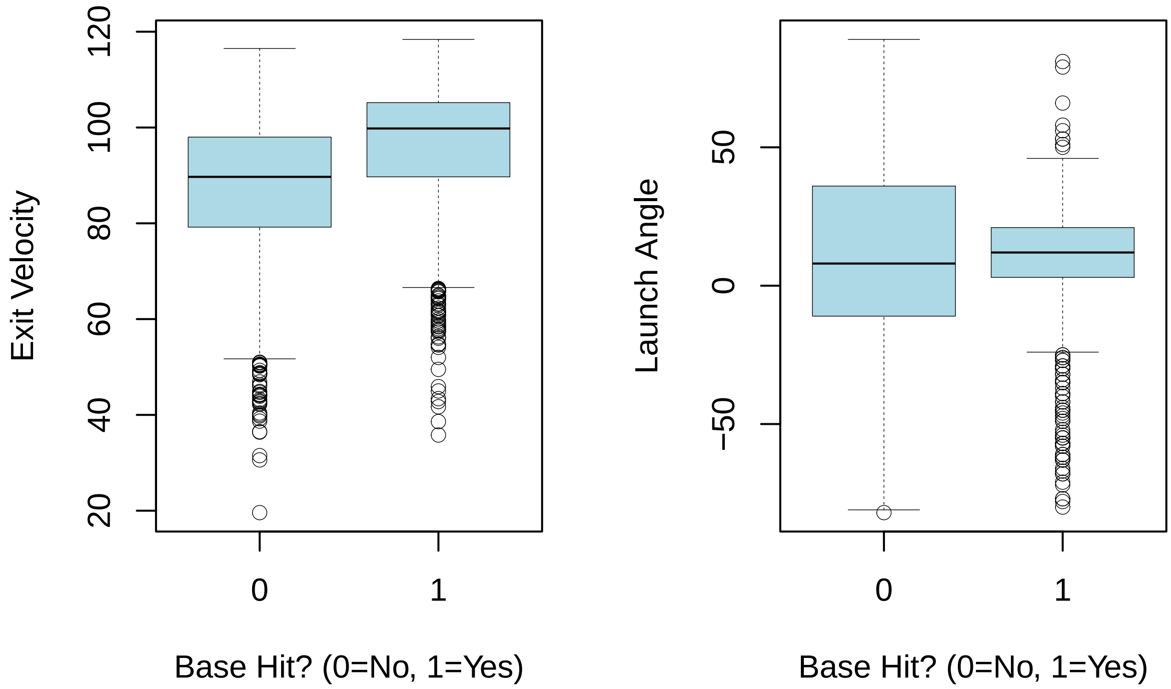
```
##
## Call:
## glm(formula = hit ~ launch_speed + I(launch_speed^2) + launch_angle +
##     I(launch_angle^2), family = binomial, data = bluejays2)
##
## Coefficients:
##                       Estimate Std. Error z value Pr(>|z|)
## (Intercept)        8.041e+00  1.063e+00   7.568 3.80e-14 ***
## launch_speed      -2.376e-01  2.494e-02  -9.529  < 2e-16 ***
## I(launch_speed^2)  1.565e-03  1.446e-04  10.820  < 2e-16 ***
## launch_angle       1.483e-02  2.174e-03   6.821 9.05e-12 ***
## I(launch_angle^2) -7.824e-04  6.168e-05 -12.685  < 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: 5389.1  on 4219  degrees of freedom
## Residual deviance: 4660.6  on 4215  degrees of freedom
## AIC: 4670.6
##
## Number of Fisher Scoring iterations: 5
```

```r
mmps(m4)
```



data fits model better after transformation

```r
par(mfrow = c(1, 2))
boxplot(launch_speed ~ hit, ylab = "Exit Velocity", data = bluejays2, col = "lightblue", xlab = "Base Hi
boxplot(launch_angle ~ hit, ylab = "Launch Angle", col = "lightblue", data = bluejays2, xlab = "Base Hi
```



exit velocity is left skewed, meaning the higher the exit velocity, the more likely a base hit will occur.

launch angle is very slightly left skewed, meaning when the launch angle is between 0 and 25, a base hit is more likely to occur.

```
vif(m3)
```

```
## launch_speed launch_angle
##     1.005488     1.005488
```

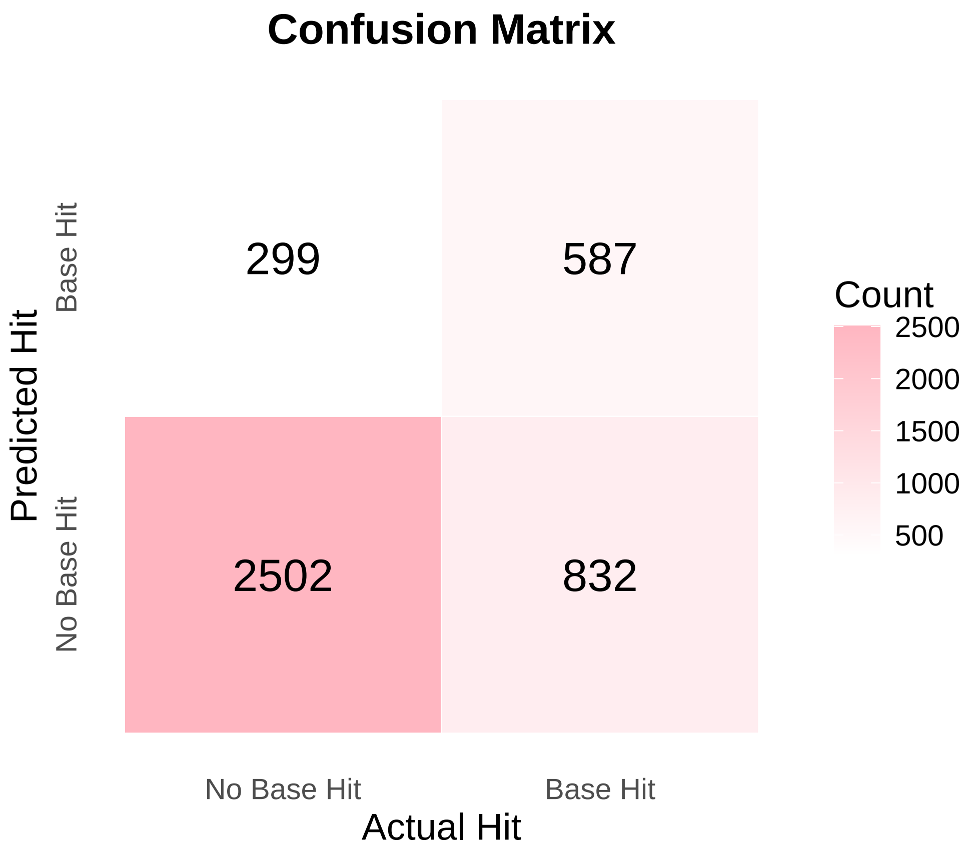
no collinearity between predictors which is good.

```
probabilities <- predict(m4, type = "response")
predicted_hit <- ifelse(probabilities > 0.5, 1, 0)
conf_matrix <- confusionMatrix(
  factor(predicted_hit),
  factor(bluejays2$hit),
  positive = "1"
)
print(conf_matrix)
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction    0    1
##          0 2502  832
##          1  299  587
##
##                Accuracy : 0.732
##                  95% CI : (0.7184, 0.7453)
##     No Information Rate : 0.6637
##     P-Value [Acc > NIR] : < 2.2e-16
##
##                   Kappa : 0.3383
##
##  Mcnemar’s Test P-Value : < 2.2e-16
##
##             Sensitivity : 0.4137
##             Specificity : 0.8933
##          Pos Pred Value : 0.6625
##          Neg Pred Value : 0.7504
##              Prevalence : 0.3363
##          Detection Rate : 0.1391
##    Detection Prevalence : 0.2100
##       Balanced Accuracy : 0.6535
##
##        ’Positive’ Class : 1
##
```

```r
cm <- conf_matrix
con_table <- cm$table
conf_df <- as.data.frame(con_table)
colnames(conf_df) <- c("Predicted Hit", "Actual Hit", "Count")
library(grid)

ggplot(conf_df, aes(x = `Actual Hit`, y = `Predicted Hit`, fill = Count)) +
  geom_tile(col = "white") +
  geom_text(aes(label = Count), col = "black", size = 6) +
  scale_fill_gradient(low = "white", high = "lightpink") +
  scale_x_discrete(labels= c("No Base Hit", "Base Hit")) +
  scale_y_discrete(labels = c("No Base Hit", "Base Hit")) +
  theme_minimal() +
  labs(title = "Confusion Matrix", x= "Actual Hit", y = "Predicted Hit") +
  theme(
    text = element_text(size = 14),
    axis.ticks = element_blank(),
    axis.ticks.length = unit(0, "cm"),
    panel.grid = element_blank(),
    plot.title = element_text(hjust = 0.5, face = "bold", size = 16),
    axis.text.y = element_text(angle = 90, hjust = 0.5)
  ) +
  coord_fixed()
```



```
TP <- cm$table[2,2]
FP <- cm$table[1,2]
FN <- cm$table[2,1]

precision <- TP / (TP + FP)

recall <- TP / (TP + FN)

F1 <- 2*(precision * recall) / (precision + recall)
F1
```

```
## [1] 0.5093275
```

```
precision
```

```
## [1] 0.4136716
```

```
recall
```

```
## [1] 0.6625282
```

when this model predicts a hit, it is only correct 41% of the time. This means the model is noy reliable in predicting actual hits. A "hit" in baseball is relatively rare (most plate appearances are outs). So predicting hits is difficult — and your model is producing too many false positive hit predictions.

of the actual hits in this dataset, this model correctly predicted 62% of them. A hit is rare and hard to predict.

An F1 of ~0.51 means:

The model finds many of the true hits (66% recall)

But its hit predictions are often wrong (41% precision)

Overall, the model is somewhat useful, but not highly reliable

This is typical for logistic regression on baseball "hit/no hit".

anova between models

```
anova(m1, m2, test="Chisq")
```

```
## Analysis of Deviance Table
##
## Model 1: bluejays$hit ~ launch_speed + launch_angle + release_speed +
##     pitch_type
## Model 2: bluejays$hit ~ launch_speed + launch_angle + release_speed
##   Resid. Df Resid. Dev  Df Deviance Pr(>Chi)
## 1      4205     5011.6
## 2      4216     5019.6 -11  -8.0204   0.7115
```

From ANOVA, it can be seen that the p value of the analysis of deviance between models m1 and m2 is greater than 0.05, meaning that pitch type does not significantly improve the model, and these 2 models predict the same.

```
anova(m2, m3, test="Chisq")
```

```
## Analysis of Deviance Table
##
## Model 1: bluejays$hit ~ launch_speed + launch_angle + release_speed
## Model 2: bluejays$hit ~ launch_speed + launch_angle
##   Resid. Df Resid. Dev Df Deviance Pr(>Chi)
## 1      4216     5019.6
## 2      4217     5019.8 -1 -0.19844    0.656
```

Since the p-value of this test is also greater than 0.05, we can say that the release speed has no statistical significance in the model.

In conclusion, from the ANOVA tests, using a model with extra predictors such as release speed or pitch type will have no significance over using a model with only launch speed and launch angle. This means we should use model m3, as it holds only statistically significant precitors, and it has the least model complexity of the three.

```
coef(m3)
```

```
##  (Intercept) launch_speed launch_angle
## -5.223876531  0.049976287 -0.004645533
```

```
mean(launch_speed, na.rm = TRUE)
```

```
## [1] 90.25379
```

```
mean(launch_angle, na.rm = TRUE)
```

```
## [1] 11.21567
```

```
x <- c(1, mean(launch_speed, na.rm = TRUE), mean(launch_angle, na.rm = TRUE))
logodds <- sum(coef(m3)*x)
probability <- exp(sum(coef(m3)*x))/(1+exp(sum(coef(m3)*x)))

logodds
```

```
## [1] -0.7654299
```

```
probability
```

```
## [1] 0.3174685
```