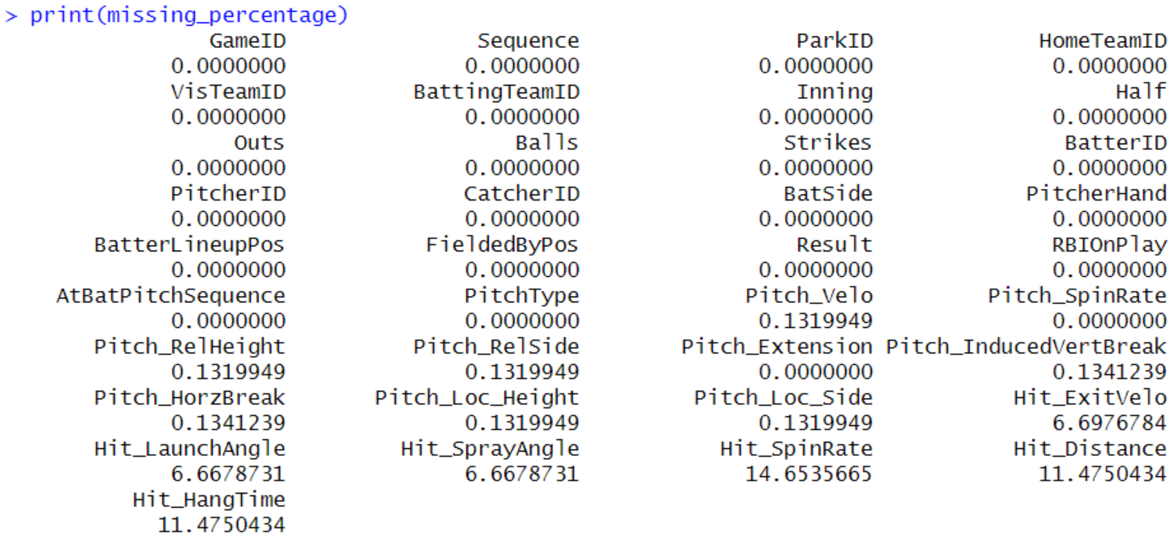
Which batters deserved the most home runs?

Home runs are not regular outcomes of hits and it takes excellent batters to achieve this. Like in the dataset, only 2969 of the 93943 hits achieved homeruns. Building an expected homerun model for minor league baseball batters provides a good way of evaluating batters, as it incorporates batted parameters and helps to identify over-and-underachieving batters, which can help us make informed scouting decisions in a very competitive system of trades, free agency and the draft. Below, I outline the processes I took with some key takeaways from the model.

Data Processing

With two data frames given, I merged both on two key variables, GameID and Sequence using the inner join method. This method only kept rows with the same GameID and Sequence values from the two datasets which ensures data consistency. While I had a single dataset to work with, there was however a problem of **missing values**. For instance, a row with "Result" variable of a 'InPlay-Out' could come with missing data for hit exit velocity, hit launch angle or hit spray angle variables. In fact, Spin Rate had a 14.7% of total observation missing with Hanging Time and Hit Distance having 11.5% respectively.

Figure 1: *Missing Value Percentage Per Variable*



```
> print(missing_percentage)
                GameID               Sequence                 ParkID             HomeTeamID
             0.0000000              0.0000000              0.0000000              0.0000000
             VisTeamID          BattingTeamID                 Inning                   Half
             0.0000000              0.0000000              0.0000000              0.0000000
                  Outs                  Balls                Strikes               BatterID
             0.0000000              0.0000000              0.0000000              0.0000000
             PitcherID              CatcherID                BatSide            PitcherHand
             0.0000000              0.0000000              0.0000000              0.0000000
       BatterLineupPos           FieldedByPos                 Result              RBIOnPlay
             0.0000000              0.0000000              0.0000000              0.0000000
    AtBatPitchSequence              PitchType             Pitch_Velo         Pitch_SpinRate
             0.0000000              0.0000000              0.1319949              0.0000000
       Pitch_RelHeight          Pitch_RelSide        Pitch_Extension Pitch_InducedVertBreak
             0.1319949              0.1319949              0.0000000              0.1341239
       Pitch_HorzBreak       Pitch_Loc_Height         Pitch_Loc_Side           Hit_ExitVelo
             0.1341239              0.1319949              0.1319949              6.6976784
       Hit_LaunchAngle         Hit_SprayAngle           Hit_SpinRate           Hit_Distance
             6.6678731              6.6678731             14.6535665             11.4750434
          Hit_HangTime
            11.4750434
```

Due to the cause of missingness unknown, it was dangerous to immediately delete missing observations. Hence, I employed **imputation** techniques using the mice package in R. Due to computational costs, I only imputed the six batting parameters. Lasso and Predictive Mean Matching (PMM) were used, with Result variable and the batting variables being used as the prediction matrix for imputation. As it can be seen in the Density Plots of Figure 1 and 2 below, the imputation types were good fits for some variables and not very good for others. Lasso's imputations were similar to the original variables in Hit Launch Angle, Hit Spray Angle and Hit Spin Rate while PMM's imputations were more aligned with Hit Distance and Hit Hang Time. While Hit Exit Velocity was close to call, PMM looked slightly a better fit than lasso. In the modelling process, I used imputed values from one of lasso and PMM whose distribution was more aligned with the original data.

Figure 2: A Density Plot of Lasso Imputation. **(***Red: Imputed Values, Blue: Original Values***)**

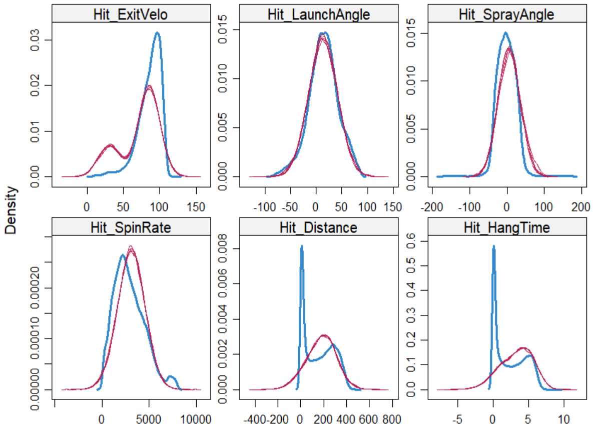
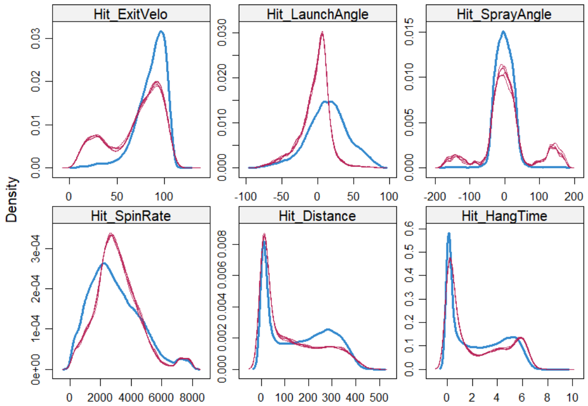


Figure 3: A Density Plot of PMM Imputation. **(***Red: Imputed Values, Blue: Original Values***)**



Model Building:

What factors determine home run probability? Only a few, from a physics point of view: Fly ball distance, spray angle, distance and height of the fence at that spray angle. The latter are dependent on the particular park and I do not have that available in the data. So, it primarily depends on spray angle and fly ball distance, which measures how far a fly ball travels from the point of contact with the bat to where it lands or becomes a home run. Alan Nathan (2020) has shown that fly ball distance depends on exit velocity, launch angle, batted ball, spin, batted ball spin axis and drag. However, not all these variables are available in the data. This brought about my decision to include Exit Velocity, Launch Angle, Spray Angle and Hanging Time.

You’ll realize that even though six batting parameters were given in the data, I chose to use four of them for the model. Spray Angle and Hanging Time are “surrogates” and by using them, the model implicitly accounts for the effects of Spin Rate without having to include it. Also, batted ball distance was excluded as a predictor since the model has components that determine distance (Exit Velocity and Launch Angle) and so including “Hit_Distance” would have resulted to double counting.

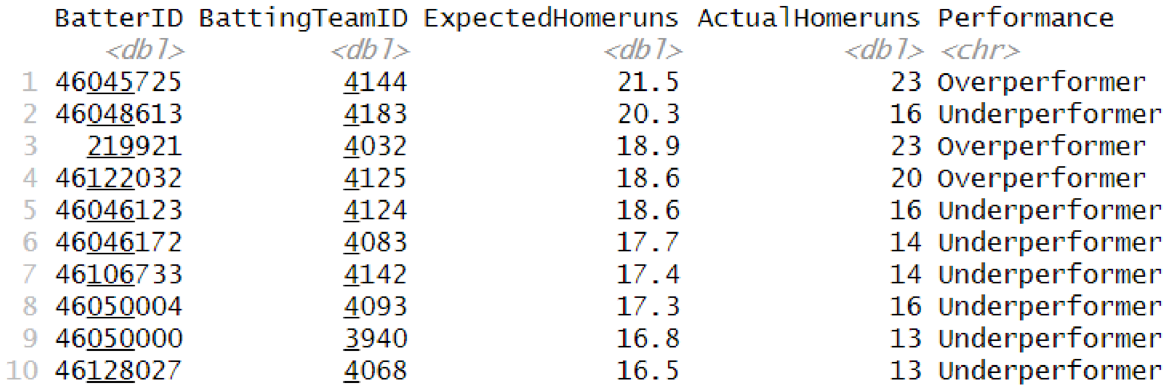
To determine expected homeruns for each batter, a generalized additive model (GAM) was fitted with the logit as a link function. GAM is used since it is a nonparametric way of characterizing the dependence of some variable, such as home run probability, to the factors that determine it. It is not easily possible to parametrize the dependence of home run probability on the various factors with a simple formula. The physics is more complicated and GAM allows for modeling the probability of a home run as a function of predictors, while capturing potential nonlinear relationships between the predictors and the response variable. The predicted probabilities will be summed up by each batter, and the batters with the highest scores would be the ones who deserved the most homeruns. Several models were built, with Table 1 results showing that Model 1 was best for the desired expected homerun model, having the smallest AIC and the highest deviance explained percentage. AIC measures the relative quality of a statistical model for a given dataset and deviance explained shows the model's explanatory power and how much better it performs compared to a baseline model. The results of model1 can be seen in the leaderboard in Figure 4, which answers the question, “which player deserves to hit the most homeruns?”.

Table 1*: Results from GAMs*

Model	**Explanation**	**Akaike Information Criterion (AIC)**	**Bayesian Information Criterion (BIC)**	**Deviance Explained (%)**
Model 1	Multivariate smooth across all predictors	7958.160	8664.003	70.4
Model 2	Separate smooth terms for each predictor	8015.060	8307.508	69.8

Model 3	Interaction of exit velocity and launch angle	7982.863	8319.886	70
Model 4	Focused on exit velocity and launch angle	11316.622	11562.879	57.3
Model 5	Independent effects of exit velocity & angle	11386.712	11552.218	56.9
Model 6	Two interaction pairs. Balances key interactions with inclusion of all factors	8065.233	8568.155	69.8

Figure 4: Results of model1



```
     BatterID BattingTeamID ExpectedHomeruns ActualHomeruns Performance
        <dbl>         <dbl>            <dbl>          <dbl> <chr>
 1   46045725          4144             21.5             23 Overperformer
 2   46048613          4183             20.3             16 Underperformer
 3     219921          4032             18.9             23 Overperformer
 4   46122032          4125             18.6             20 Overperformer
 5   46046123          4124             18.6             16 Underperformer
 6   46046172          4083             17.7             14 Underperformer
 7   46106733          4142             17.4             14 Underperformer
 8   46050004          4093             17.3             16 Underperformer
 9   46050000          3940             16.8             13 Underperformer
10   46128027          4068             16.5             13 Underperformer
```

Other Interesting Takeaways

Optimal Launch Angle for hitting a home run

By finding the optimal launch angle, batters can successfully maximize their chances of hitting a homerun and scouting can be geared towards batters who hit at the given angle. Figure 5A indicates the optimal launch angle for hitting home runs is around 30° and so teams hoping to increase their homeruns must focus on achieving launch angles within the 25°–35° range while maintaining a high velocity. Figure 5B also shows the optimal launch angle varies slightly by pitch type received and pitch type in most cases have little influence on homerun probability with approximately a little over 0.2 being the highest homerun probability achieved for any pitch type.

***Figure 5A**: **Left**: Smoothed home run probability as a function of launch angle for batted balls hit for all velocities in the data. **Right**: Smooth home run probability as a function of launch angle for batted balls hit between 100 and 105 mph.*

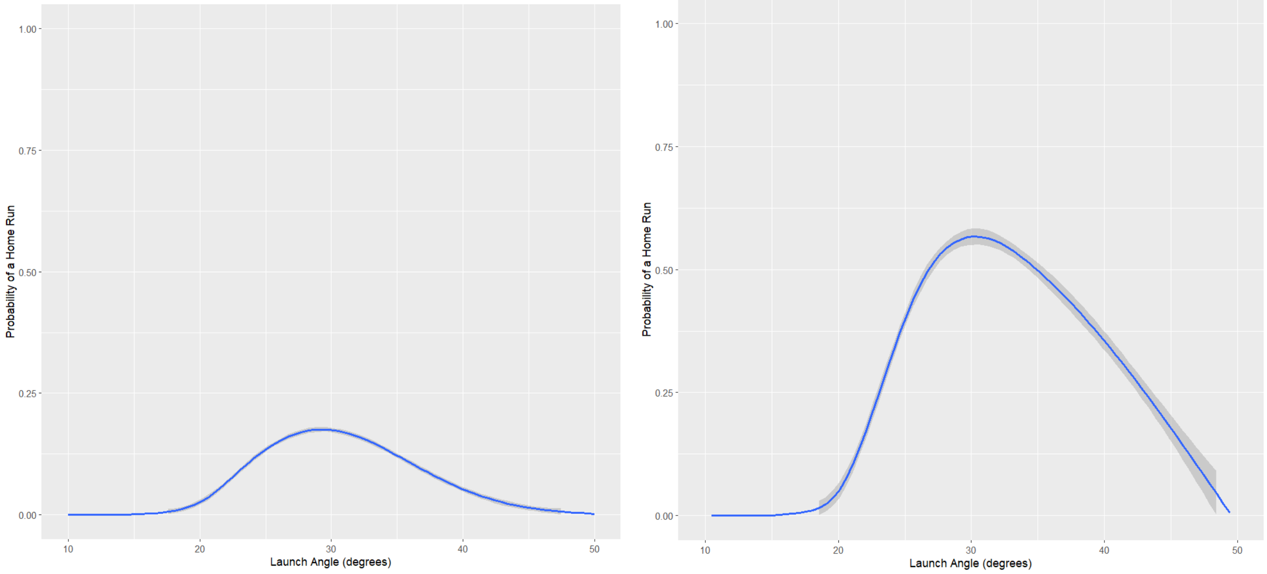
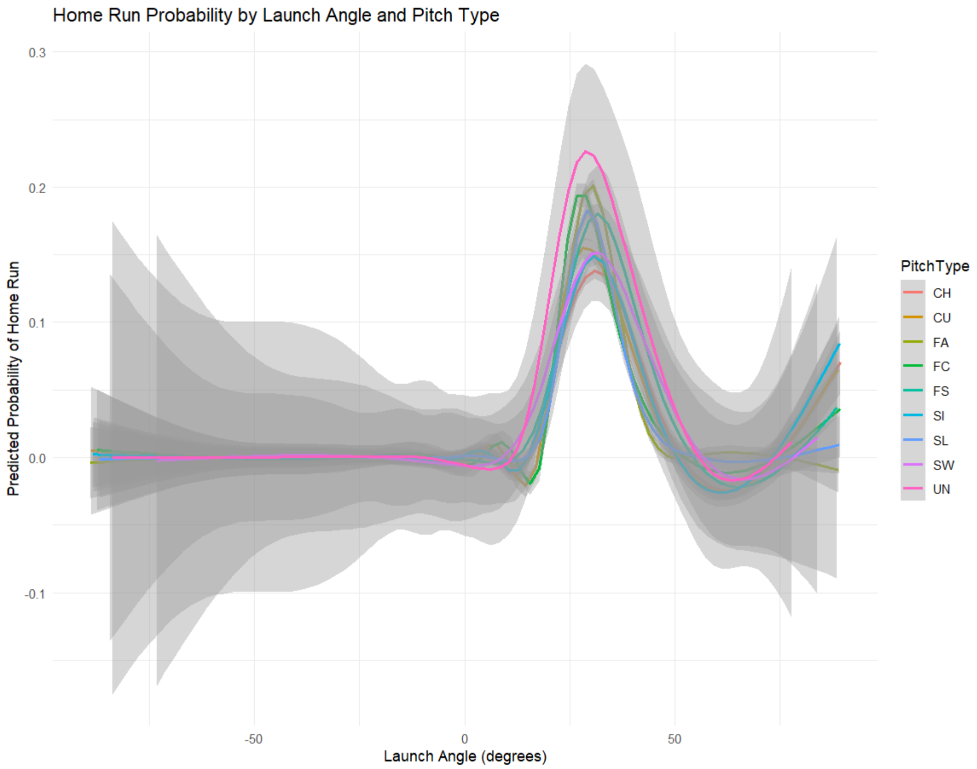


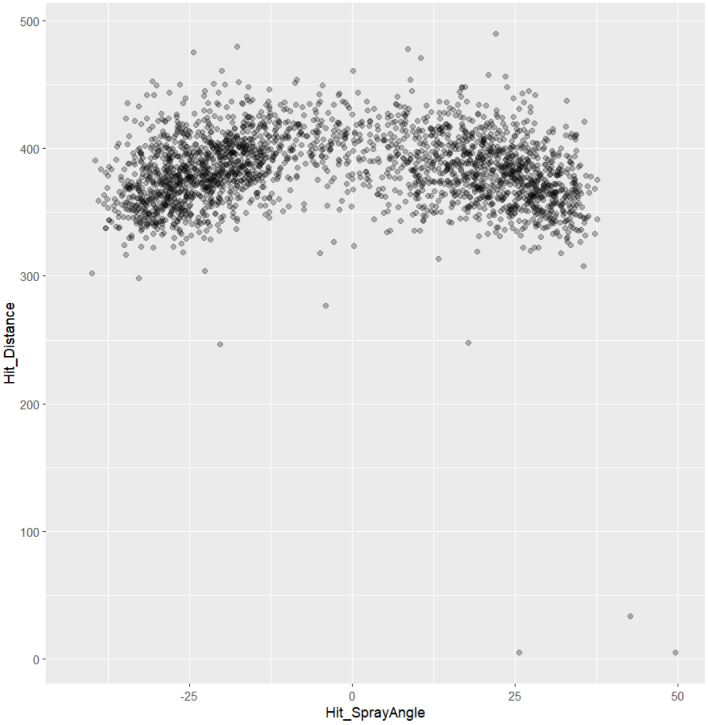
Figure 5B: Home Run Probability by Launch Angle and Pitch Type



Spray angle effects

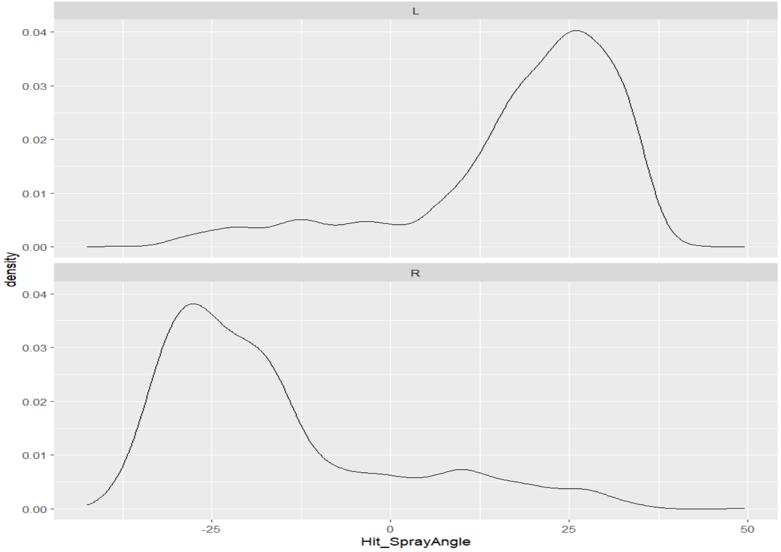
The data (excluding imputed values) confirms that most home runs happen at spray angles between -25° and +25°, with batted ball distances normally between 350 and 450 feet. However, there are three curious outliers, as can be seen in Figure 6: Outlier with almost no distance (Hit Distance ≈ 0 feet, Spray Angle ≈ +25°) which could be an inside-the-park homerun, outlier at ≈300 Feet (Spray Angle ≈ 0°) which could be as a result of short fence, wind-assisted or unique park factor and outlier at ≈300 Feet (Spray Angle ≈ +50°) which could be as a result of short foul pole fence.

Figure 6: *Scatter plot of distance travelled and spray angle for home runs hit*



Additionally, it can be seen from Figure 7 that while left-handed batters mostly prefer to use the opposite field (positive spray angles) to achieve homeruns, right-handed batters prefer to use the pull-side (negative spray angles) to achieve homeruns. Interestingly, both left- and right-handed batters achieve spray angles of approximately +25° and -25° respectively.

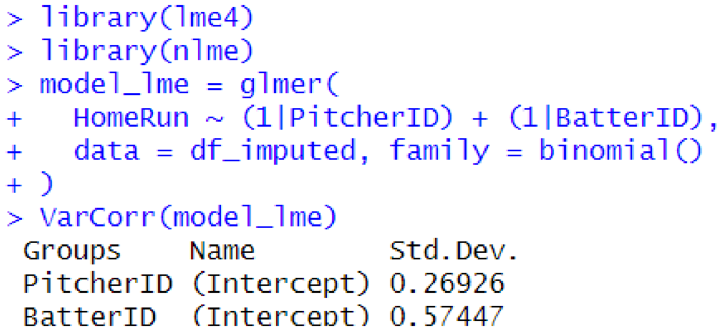
Figure 7: *Density estimates of spray angle of home runs hit by left(L) and right(R)-handed batters.*



Who determines homeruns? Batters or Pitchers?

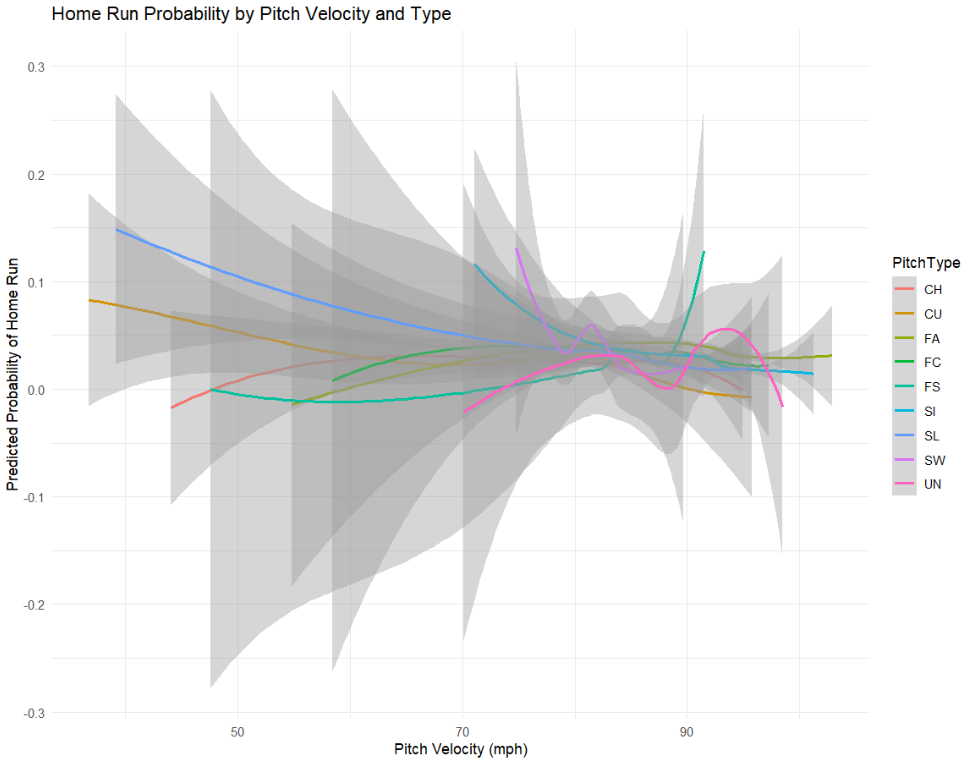
A linear mixed effects model, as shown in *Figure 8A* having batters and pitchers as random effects and logit as a link function for probability of homerun estimates the standard deviations of the random effects as 0.27 and 0.57 for pitcher and batter respectively, indicating that the variation in homeruns is due mainly by hitting and not pitching. Additional evidence is provided in *Figure 8B* where homerun probabilities doesn't seem to increase for any given pitch velocity for different pitch types.

Figure 8A: *A logistic mixed effects model with batter and pitcher as random effects on homerun probability*



```
> library(lme4)
> library(nlme)
> model_lme = glmer(
+   HomeRun ~ (1|PitcherID) + (1|BatterID),
+   data = df_imputed, family = binomial()
+ )
> VarCorr(model_lme)
 Groups    Name        Std.Dev.
 PitcherID (Intercept) 0.26926
 BatterID  (Intercept) 0.57447
```

Figure 8B*: Homerun probability by pitch velocity and pitch type*



Additional Information I wish I had

- Matchday temperature effects: To test the hypothesis whether weather influences homerun rates and if it does, by how much?
- Distances to the fences and drag variables: As Alan Nathan (2020) has shown, these variables are great determinants of homeruns and having it in the data set would improve the model's efficiency.
- Reason for missingness: When the reason for missing values is known, an appropriate imputation method can be matched with it and imputation can get better.
- More Specifics: The scouting director being more specific about what the leaderboard would be used for can add more context to the analysis.

References

Nathan, A. (2020). *Contributions to Variation in Fly Ball Distances: A Followup*. University of Illinois at Urbana-Champaign. Retrieved from http://baseball.physics.illinois.edu/FlyBallDistanceFollowup.pdf