AI for Auburn Baseball #2 (Models) Blue

Cycle 3 Report

Authors:

Trent Gavron (tmg0046)

Justin Whisonant (jgw0035)

Austin Beville (acb0157)

Lauren Lyons (lnl0017)

Comp 4710 Senior Design Team #10

Department of Computer Science and Software Engineering Samuel Ginn College of Engineering,

Auburn University

Table of Contents

1. System Metaphor (JW)

Our system will be used to generate reports for the Auburn baseball team of the optimal defensive positioning for each player in the opposing team's lineup. Our predictive model will display the best defensive positioning by determining which of our pitchers is pitching and how the current batter has performed in previous at-bats against pitchers that have similar metrics (vertical/horizontal break, velocity, spin rate, etc.), similar pitching styles (left vs. right handed, overhand vs. sidearm vs. submarine, etc.), and similar pitch types or arsenals (four-seam fastball, sinker, curveball, slider, etc.). After the filtering process, this information will be displayed on a model of a baseball field, which will show the percentage of ground balls hit to each section of the infield and a heat map of where the flyballs landed in the outfield.

2. Cycle Intent (JW)

Our plan for Cycle 3 is going to be to have a finalized product for Coach Teaford and the baseball team to have access to through the web application that the database team has been building. Our models and visualizations are nearly done, as the only real unfinished aspect of them is our outfield predictions and visualizations. We also might try to get our models to work with specific batters on opposing teams. The beginning of this cycle's goal will be to have both the infield and outfield models up and running, and once that is complete, our goal will be to work with the database team to connect our models to their website. During this process, we will have to make sure the website is easily usable and clean with our models on it, and we will have to make sure that they work seamlessly on the website, just like they do on our local machines. Lastly, if necessary, we will help the database team with any components of the website they might need, as there is potential that we finish our models before the website is completely ready to go.

3. User Stories (TG, JW, LL, AB)

Optimal Defensive Positioning
Summary: As an Auburn Baseball Coach, I want to predict where the batter is going to hit the ball so I am able to position my players in the optimal defensive locations.
Description: The Machine Learning Model will effectively predict where the other team's batter will hit the ball based off of the type of pitch the Auburn pitcher throws and its location. Our first output will be a sheet for each batter of an opposing team which displays how we expect them to hit ground balls in the infield based on the pitch type and right/left handed pitcher. Our next output will take an input of our pitcher and a pitch type and return the expected probabilities of infield ground balls for both left and right handed batters.
Planned Hours: 20
Actual Hours: 45
Coder Names: Trent, Lauren, Justin, Austin
Tester Names: Trent, Lauren, Justin, Austin
Reviewer Names: Trent, Lauren, Justin, Austin
Story Status: In Progress

Individual Matchup Defensive Positioning
Summary: As an Auburn Baseball Coach, I want to know how the individual match-ups will affect how our defense should position themselves.

Description: The model will allow coaches to select specific match-ups (pitcher vs batter) to see how each individual player performs instead of using generic data from across all games. This will be trained more specifically based on pitcher statistics and batter's hit history.
Planned Hours: 18
Actual Hours: 10
Coder Names: Trent, Justin
Tester Names: Trent, Justin
Reviewer Names: Austin, Lauren
Story Status: In Planning

Pitcher/Pitch Dependent Defensive Positioning
Summary: As an Auburn pitcher, I want to know how the pitches I throw affect/determine where the opposing batter will hit the ball.
Description: The model will show how different pitch types and locations affect where the ball lands. And be a good tool to show pitchers how the models are predicting the hitting of the ball. This will have outputs for both Left and Right handed batters and show how the pitch types will most likely be hit for that pitcher's average pitch statistics.
Planned Hours: 14
Actual Hours: 15

Coder Names: Trent, Lauren, Justin, Austin
Tester Names: Trent, Lauren, Justin, Austin
Reviewer Names: Trent, Lauren, Justin, Austin
Story Status: In Planning

Infielder Ball In Play Knowledge
Summary: As an Auburn infielder, I want to know where ground balls are most likely to go / if they are likely to be hit at all.
Description: The model will give the likelihood of hitting a ground ball and which ‘slice’ of the infield the ball will be hit in.
Planned Hours: 20
Actual Hours: 25
Coder Names: Trent, Lauren
Tester Names: Trent, Lauren
Reviewer Names: Austin, Justin
Story Status: In Progress

Outfielder Ball in Play Knowledge
Summary: As an Auburn outfielder, I want to know where fly balls and line drives that travel over 150 ft are most likely to go / if they are likely to be hit at all.
Description: The model will give the likelihood of hitting a fly ball or line drive and which ‘slice’ of the outfield the ball will land in.
Planned Hours: 12
Actual Hours: 12
Coder Names: Austin, Justin
Tester Names: Austin, Justin
Reviewer Names: Trent, Lauren
Story Status: In Planning

Creating an Easily Readable Output
Summary: As an Auburn Baseball Coach, I need to know the exact positions that the model is suggesting be shifted to adjust to the ball’s predicted path.
Description: The model will present a graph of the baseball diamond with the previously mentioned ‘slices’ but will also give a description in words, something along the lines of “SHIFT LEFT w/ Deep 2B”. This is essentially taking the models we have trained and generated to give real-world recommendations.
Planned Hours: 18

Actual Hours: 9
Coder Names: Trent, Lauren, Justin, Austin
Tester Names: Trent, Lauren, Justin, Austin
Reviewer Names: Trent, Lauren, Justin, Austin
Story Status: In Planning

Coach’s Optimal Positioning of Fielders
Summary: As an Auburn Baseball Coach, I need to know what areas on the field the ball is most likely to land in.
Description: The model will present a graph of the baseball diamond overlaid with a heat map alongside the previously mentioned ‘slices’ that will show the most likely areas on the field for the ball to land. This is an additional visualization tool we will create to give real-world recommendations.
Planned Hours: 18
Actual Hours: 10
Coder Names: Trent, Lauren, Justin, Austin
Tester Names: Trent, Lauren, Justin, Austin
Reviewer Names: Trent, Lauren, Justin, Austin
Story Status: In Planning

Easily Printable and Accessible Data Representation for Coaches
Summary: As an Auburn Baseball Coach, electronic devices are not allowed in the dugout, which means I cannot easily have access to the model during games. I need the data to be presented in a concise, readable format that can be printed and placed into a binder.
Description: The software will take all the data from the model (graphs, shift descriptions, and numeric data) and compile it into a small form factor that can be easily printed without wasting space on the page or bloating the page with unnecessary information.
Planned Hours: 18
Actual Hours: 26
Coder Names: Trent, Lauren, Justin, Austin
Tester Names: Trent, Lauren, Justin, Austin
Reviewer Names: Trent, Lauren, Justin, Austin
Story Status: In Planning

Filtering to Get the Most Relevant Data
Summary: As an Auburn Baseball Coach, I need to be able to filter the data down to what is most relevant for the current game / upcoming game.
Description: The data fed into the model will be selected from the current filter, meaning coaches would be able to change the dataset to just one team, one pitcher, or even just one type of pitch from said pitcher.
Planned Hours: 10

Actual Hours: 8
Coder Names: Trent, Lauren
Tester Names: Trent, Lauren
Reviewer Names: Austin, Justin
Story Status: In Progress

Model Training Transparency and Modifiability
Summary: Add the ability to explain how our models are working and calculating their probabilities
Description: The will show the process by which a model reaches its conclusions. No part of the learning process will be treated as a ‘black box’, instead, each aspect must be transparent and easily tweakable.
Planned Hours: 6-8
Actual Hours:
Coder Names: Trent, Lauren, Justin, Austin
Tester Names: Trent, Lauren, Justin, Austin
Reviewer Names: Trent, Lauren, Justin, Austin
Story Status: In Planning

4. Design Documentation (LL & AB)

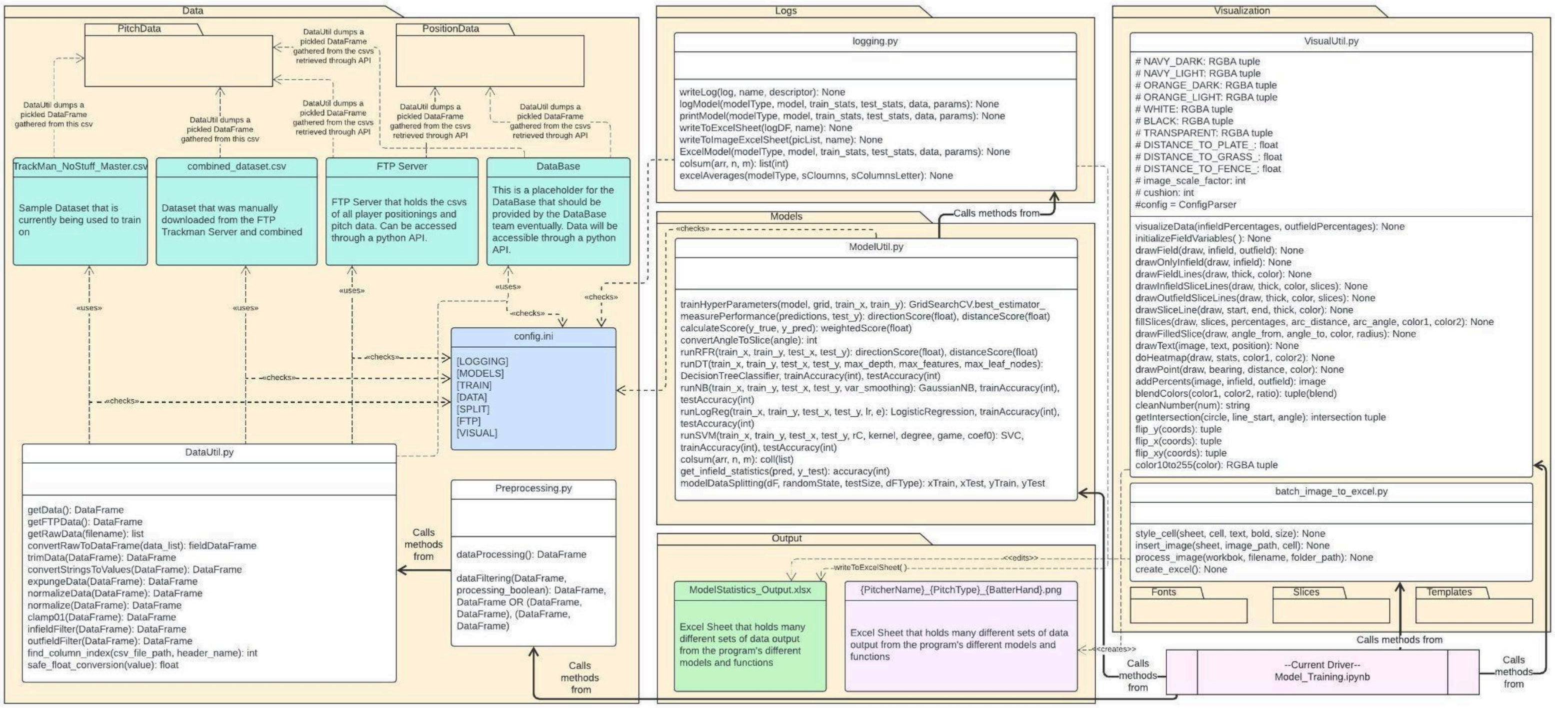


Figure 1: (UML Diagram of Source Code)

Our Machine Learning model mainly runs Python scripts using the Python Machine-Learning Library scikit-learn. The version we use of scikit-learn is 1.4.0 on Python version 3.12.1. You can install and check the list of dependencies on its website at https://scikit-learn.org/stable/install.html. The Database platform is yet to be determined, but it will be Auburn University-approved. The Python scripts will be run on an Auburn server and use an API to request the data from the Database's platform. Another package used is Pandas(2.2.0).

For this set of scripts, we use separate models for the Infield and Outfield, since they require different sets of data and will give different sets of output. Then we will also use separate models for Pitcher-centered output models and Batter-centered output models. We may also use two different models for the slice calculations and pinpoint calculation for the outfield. We use a supervised algorithm since we have data that has been processed and labeled. The scikit Machine Learning Model that results in the highest accuracy for the Infield is a Decision Tree Classifier. We have also tested Support Vector Machine

Classification, Logistic Regression Classification, Naive Bayes Classification, and Random Forest Regression for this Infield Model. We currently average over 30 run outputs from the Decision Tree Classifier, Logistic Regression Classifier, and Naive Bayes Classifier to get our final output. Then we get the prediction probability based on each Pitcher's Fall Averages in order to produce an image output. We are planning to use a Logistic Regression Model to fit our Outfield Model. The validation for using the model we will use for the Outfield models will be based on the tests we will run with each ML model. Ideal outcomes will produce the best accuracy out of all of the models. We may not be able to choose our final model or models, however, until we run them through the large datasets of all SEC games that we may be allowed to use. We currently do use a Naive Bayes Classifier, Logistic Regression Classifier, and a Support Vector Machine to classify where the balls are most likely to be hit in the outfield. There is currently 15 sections in the outfield, with 5 slices across and 3 levels deep.

There are many ways for us to retrieve the data currently. The end goal is to only use the Database's API when the application is completely working. At the current moment, we can either use a sample Trackman Dataset, a manually combined CSV, or a Dataset retrieved from a TrackMan Server using the FTP API. No matter the retrieval method, we filter down our given dataset to only acknowledge when the column PitchCall is defined as 'InPlay'. This is the only actionable data given to us. When a pitch is called 'InPlay', the result will be in column TaggedHit as a 'LineDrive', 'GroundBall', 'FlyBall', 'Bunt', or 'Popup'. We are only interested in 'GroundBall's for the Infield model. For the Outfield model, we are interested in 'FlyBall's and 'LineDrive's with a Distance field larger than 150 ft. The distance from home plate to second base is about 128 feet. So, due to our Sponsor's suggestion and this measurement, we use 150 feet as the cutoff for the interest point of 'LineDrive's. We ignore 'Popup's as there is no needed shift in the defense for these. We ignore 'LineDrive's with a Distance smaller than 150 ft because defenders must be in the right place at the right time for those hits, so shifting the defenders will not assist in their ability to defend against these. We ignore 'Bunt's in the dataset, but we do give an indication of each batter's ability to bunt. There are also some rows of the data that are defined as 'NA' for the Direction or Bearing, and those will also be filtered out. Along with any blanks or unrefined rows, as they will be removed.

We use the Direction field when calculating Infield data because of Sponsor suggestions and it is defined in the documentation as “The horizontal angle formed by the intersection of the y-axis and the ball’s path (in the x-direction) as it leaves the bat”. This means we can use this field to determine which of the five sections the ball travels into and we should not have to worry about any irregular forces on the ball since it is traveling along the ground. Therefore, it should generally travel straight in the direction it was hit. We use the Bearing field when calculating Outfield data because of Sponsor suggestions and it is defined in the documentation as “The horizontal angle formed by the intersection of the y-axis and a straight line drawn from the plate to the landing spot of the ball”. So, this helps mitigate the need to acknowledge any non-parabolic tendencies of the ball’s flight path.

For Infield data, we were filtering to only accept ‘Medium’ or ‘High’ confidence of the HitLaunchConfidence to ensure that we were using quality data. However, some bad quality data was still showing up, so we removed that filter. Now we assume that the confidence for launch angle cannot be the same as confidence for the direction. Instead, we filter the Direction field through the range -45 to 45 degrees. For Outfield data, we were also filtering to only accept ‘Medium’ or ‘High’ confidence of the HitLandingConfidence to ensure that we were using quality data. However, we removed that filter. The range of Direction should only fall in between -45 and 45 degrees, as home plate to second base is considered to be the y-axis and 0 degrees. To provide images to help visualize the axes for the baseball field, approved images from Auburn University are required since this will be an Auburn product. The infield and outfield will be divided into fifths due to our sponsor’s suggestion, resulting in 18-degree sections. Section 1 ranges from angle [-45, -27). Section 2 ranges from angle [-27, -9). Section 3 ranges from angle [-9, 9]. Section 4 ranges from angle (9, 27]. Section 5 ranges from angle (27 to 45]. For the Outfield, there are two avenues of visualization that we are going to test. We will have slices just like the Infield, but we will also split it into shallow Outfield and deep Outfield. So, there will be 10 slices that show the percentage of the probability that the ball will land there. The percentages will be determined by a model similar to the Infield’s Model. There will be a heatmap that consists of every datapoint that is relevant to each situation. The heat map does not require any machine learning. It is just visualization. The second avenue will be added onto the previous visualization outputs. The model will output a pinpoint of the single point of where it determines the ball should land. Then it will have an outline of a circle where the radius is based on the size of the cluster near it. There will be a certain density of hits needed for the circle's radius to be determined to be a certain size. We are hoping that this will both provide information on where the pitch is expected to be hit and the past results.

When it comes to visualizing the data, we have to generate an image that clearly shows where the ball will go. The current plan for this is slices in the infield and a heat map of the outfield. Since the coaches liked the design from the last cycle, we started future proofing the way these images are generated. Before, the field was a premade image and the slices were locked at 5 for both infield and outfield, and there was no way to show just the infield or just the outfield odds. This was extremely limiting and made developing the heatmap visuals difficult. Now, the visualization draws the field from scratch using pycairo and can be modified to show just the infield, outfield, or both with any number of slices desired or instead of slices to show a heatmap of projected landing locations for the outfield. For the slices, the model will output its confidence in each field slice and send them into our visualization class. This takes the percentages and scales them so that the most likely slice appears as the darkest color and the least likely slice is the lightest color (besides white). The infield has been set to Auburn Orange and the outfield to Auburn Navy. The values of the shades between darkest and lightest come from Auburn's official color scheme and gradients that are also in use by the database team. Each slice is independent of the others and can be changed to any color by using a custom recolor function. To get a clean gradient between color values, a blend function was also made that linearly interpolates between the red, green, blue, and alpha values of a color to create a gradient. Finally, the percentages given by the model are drawn onto the image into specified positions of the image to align with their respective slices. The end results are shown below (using randomly generated values for testing):

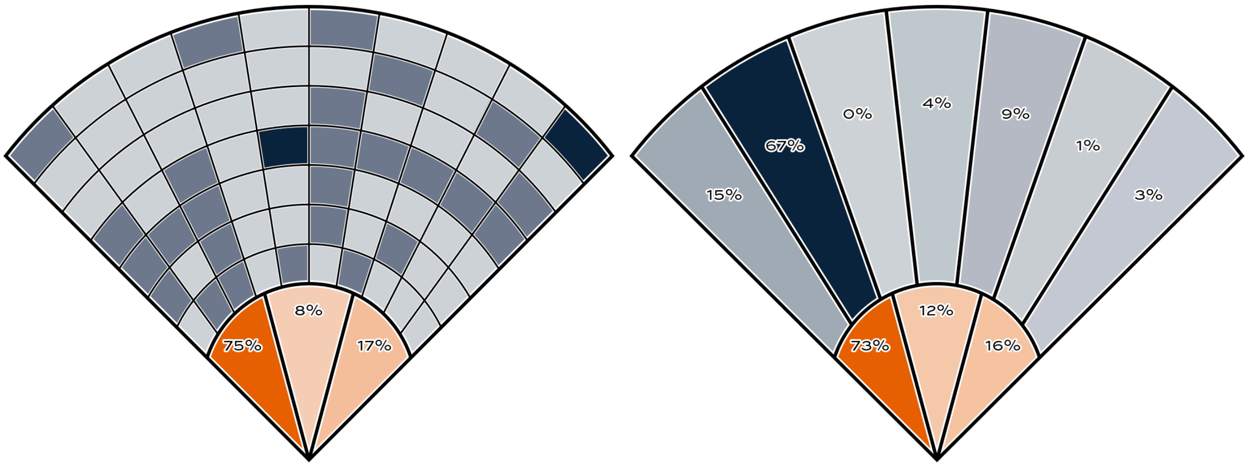


Figure 2: Data Visualization Examples - Heatmap vs Slices

The heat map visualization has been completely finished. It no longer plots points but instead will divide the field into a curved grid that darkens depending on how many predictions are within the cells area. This method allows for full customization of the grid size both horizontally and vertically. With this being completed, the coaches have complete control over how the output should look, whether they want the outfield to be presented as a heatmap or slices, dense or sparse, it is all possible. Our team is particularly proud of how the visualization for the data came out aesthetically as well as how well it manages to convey the information. It has that Auburn flare while looking both professional and stylish. The coaches agreed and really liked the way the outfield heatmap turned out.

The next thing to add would be an option to disable the text and just rely on a scale / legend to describe the percentages or likelihood of the ball going into a certain cell / slice. This would make high-density slices possible without crowding the image with text, as the coaches would only need to see the color of the slice to know the value of it. As it stands right now, the coaches don't seem like they will be needing more than 5 slices in the infield, and the outfield can have many more slices without text overlap, so it should be alright.

Additions to our design and functionality have been an Excel Output sheet and a config file. The config file holds seven different sections: 'LOGGING', 'MODEL', 'TRAIN', 'DATA', 'SPLIT', 'FTP', 'VISUAL'. Each of these hold specific variables that can be easily edited. Concerning logging, we can produce outputs through log files, printed debug and output statements, or our Excel Sheet outputs. For our machine learning models, we can toggle on/off our Decision Tree Classifier, Naive Bayes Classifier, Logistic Regression Classifier, Support Vector Machine Classifier, and the Random Forest Regression Classifier. We can specify how many times to run the models and which columns to filter on. We can tell the system which form of data retrieval we would like to use. We have the ability to split the data into different types of training sets. For FTP data retrieval, we have metadata associated that can help specify what data is wanted. We also can control the types of visualizations that are output. For the Excel Outputs, it is put into a file called 'ModelStatistics_Output.xlsx'. The available sheets that we can create are the Infield and Outfield correlation matrices, a sheet for each model type's runs, an average of each model type's output, and the different requested image output models that are listed below.

Documentation for the testing data can be found at https://trackmanbaseball.zendesk.com/hc/en-us/articles/5089413493787-V3-FAQs-Radar-Measurement-Glossary-Of-Terms

After data is filtered to the above specifications, these are the model specifications that we are using. These specifications are easily edited in the config file.

Outfield Model

Input Data Columns: "PitcherThrows", "BatterSide", "TaggedPitchType", "RelSpeed", "InducedVertBreak", "HorzBreak", "RelHeight", "RelSide", "SpinAxis", "SpinRate", "VertApprAngle", "HorzApprAngle", and "Extension"

Output Data Columns: Expected fly ball location (direction and distance)

Infield Model

Input Data Columns: "PitcherThrows", "BatterSide", "TaggedPitchType", "RelSpeed", "InducedVertBreak", "HorzBreak", "RelHeight", "RelSide", "SpinAxis", "SpinRate", "VertApprAngle", "HorzApprAngle", and "Extension"

Output Data Columns: FieldSlices (5 intervals of 18 degrees using the direction metric)

Different Output Models

Basic Hitting Prediction Model

Description: This will be a basic Machine Learning Model implemented to test the accuracy and metrics that will be used in the more specific models. This model will be used as a baseline to get familiar with and see which model types work best for the data. This is the most basic All Hitters vs All PitchTypes. There is no specification for certain pitchers, batters, or teams.

Example: An example of this model would be inputting a pitch's statistics and seeing where the model would expect the ball to be hit in play.

Status: Started

Overall Pitchers vs Hitters

Description: This will be the data given from either just the current season or from the relevant (data from past seasons for hitters that are playing this current season) history of one team vs Auburn. It will show the most likely place to hit for any opposing batter against any Auburn pitcher.

Example: An example of this would be the entirety of the LSU team's in-play hits across all of Auburn's pitchers for every past game for Auburn vs LSU that season.

Status: Not Started

Hitter vs Pitcher

Description: This is the individual matchup between each Auburn pitcher and each opposing batter.

Example: An example of this would be the likelihood of where the ball is hit when Auburn pitcher A faces Tennessee batter C regardless of any other data.

Status: Not started

Hitter vs Pitcher & Pitch Type

Description: This is the matchup between each Auburn pitcher's specific pitch type and each opposing batter.

Example: An example of this would be the likelihood of where the ball is hit when Auburn pitcher A throws a fastball when facing Clemson batter B regardless of any other data.

Status: Not started

<u>Hitter vs Pitch Type</u>

Description: This shows how the opposing batter performs against each pitch type.

Example: An example of this would be Georgia's batter against a curveball, no matter the pitcher.

Status: Not started

<u>Hitter vs Pitch Style</u>

Description: This shows how the opposing batter performs against each pitch style.

Example: An example of this would be Alabama's batter A against a sidearm pitcher, no matter the pitcher.

Status: Not started

<u>Pitcher vs LeftHitter</u>

Description: This will show generally how hitters who bat on the left-hand side perform against a specific Auburn pitcher.

Example: An example of this is Auburn pitcher A's resulting hit percentages when throwing against a batter on the left-hand side.

Status: Not started

<u>Pitcher vs RightHitter</u>

Description: This will show generally how hitters who bat on the right-hand side perform against a specific Auburn pitcher.

Example: An example of this is Auburn pitcher B's resulting hit percentages when throwing against a batter on the right-hand side.

Status: Not started

There may also be a need to distinguish between the hitter's tendencies for more situations than these because their reactions to the ball may deviate due to inning, late game play, pressure situations, or location. There can also be a distinction where we use only pure Auburn data and when we find pitchers that are similar to our own and use them as extra data points. These distinctions could be provided at a later date or at the Sponsor's request.

5. Lessons Learned (AB)

For our team, one of the bigger challenges comes from being able to properly filter the data on such a massive scale. All information that we feed into the model has to be correct or else it risks sabotaging the learning process for the model. When working with test data, monitoring the inputs is not as difficult, but when the team moves on to using the full API with millions of data points, ensuring our filtering system works is going to be of paramount importance.

Another challenge is going to be implementing pitcher vs batter matchups. Limiting the data to just one pitcher vs one batter will drastically reduce the amount of data the model has to absorb, which could likely lead to less accurate results. Figuring out a way to implement this without providing too little information will be a tricky balancing act.

Another challenge would be creating the heat map. Having the model predict which slice of the field the ball will likely land in is much simpler than predicting the exact spot on the field that the ball will land on. With the metrics the team has to feed into the model, it seems like the best approach right now would be to isolate everything to the moment the ball is struck and then perform a kinematic analysis where the ball is treated as a projectile. The reason this is tricky is because batting is inherently reactive while pitching is active. Pitchers are able to control exactly how the ball arrives into the zone and in what position it crosses the zone, and batters have to react to this and attempt to get a hit, causing them to have less control over where the ball ends up going. The exit velocity and launch angle entirely determine the distance, but the direction can likely be predicted depending on where the pitch crosses home plate, again, since the batter has less control over this.

Another lesson our team has learned is how important source control is. When different team members were working on different implementations of a system (one for infield and one for outfield) they started clashing with each other because the code was saved over. The team has since made a plan to prevent this from happening and allow everyone to work in a separate branch and then merge once it's completed. This way, nobody will lose their work and nothing will lose its functionality.

The team also learned how important it is to future-proof what we are working on. It may be easier to code a quick solution, but there’s no modularity to that. Sometimes it is better to take the extra time and effort to create something that works now, and can be easily modified to work in multiple ways later. Our visualization is where we learned this. It worked for what we needed, but was too rigid to use repeatedly in the ways we wanted to use it. After taking the time to restructure the framework for it, the visualization can become any shape and draw anything the team might need it to.

The final lesson our team learned this semester was the importance of communication with other teams. Getting everything set up with the database team turned out to be a difficult task. We didn’t exactly know what they were trying to do fully, but by working and talking with them, we were able to figure out what was wrong and solve the issues that arose when combining our work.

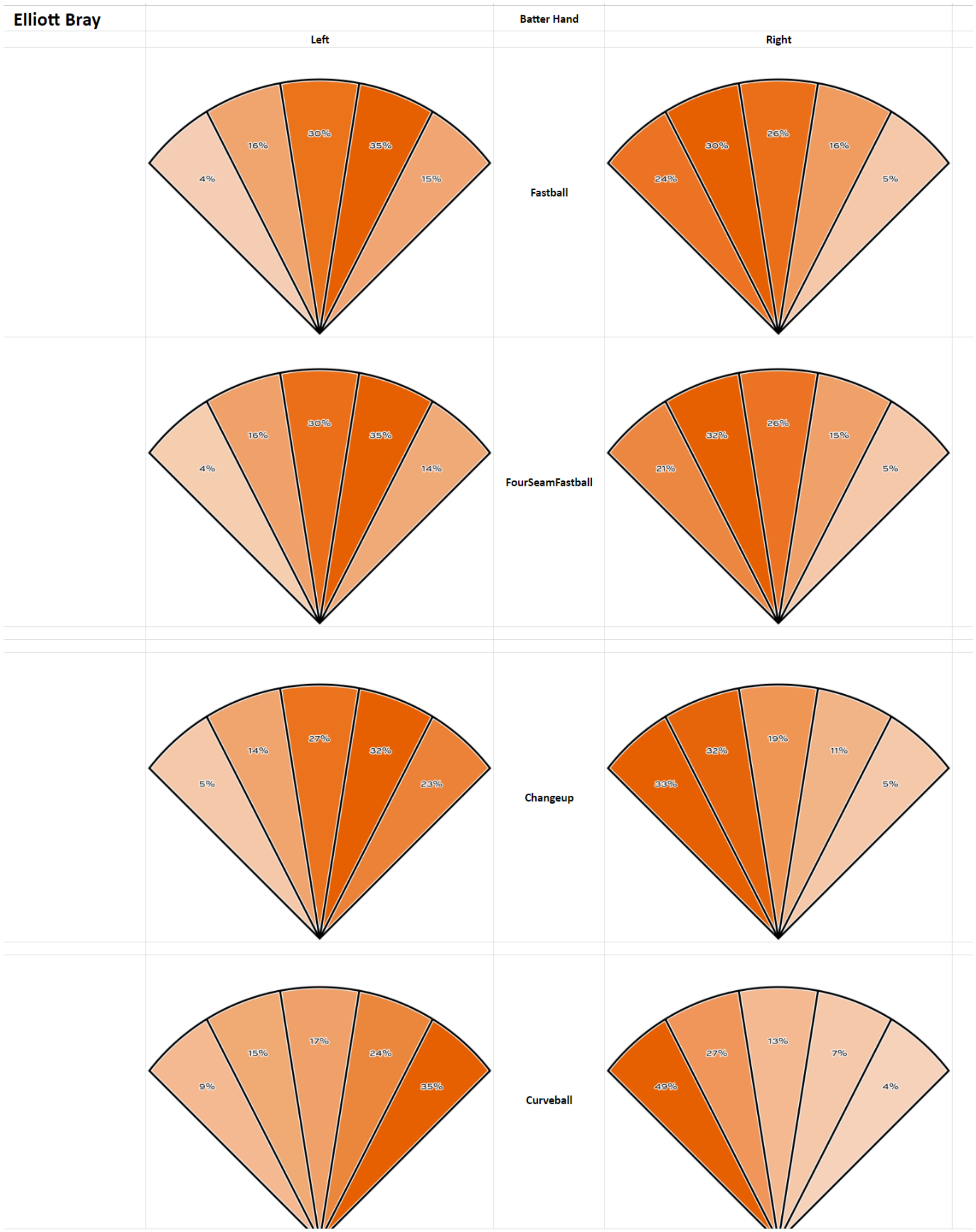
6. Test Results (TG)

30 Run Averages:

- Decision Tree
  - Training:
    - Accuracy: 0.3542
    - Average Error: 0.9477
    - F1 (micro): 0.3542
    - F1 (macro): 0.3307
    - F1 (weighted): 0.3676
    - AUC (macro): 0.9848
    - AUC (weighted): 0.9799
  - Testing:
    - Accuracy: 0.3171
    - Average Error: 0.9957
    - F1 (micro): 0.3171
    - F1 (macro): 0.2929
    - F1 (weighted): 0.3314
    - AUC (macro): 0.9847
    - AUC (weighted): 0.9799
- Naive Bayes
  - Training:
    - Accuracy: 0.2965
    - Average Error: 1.0752
    - F1 (micro): 0.2965
    - F1 (macro): 0.2887

        - F1 (weighted): 0.3013
        - AUC (macro): 0.9799
        - AUC (weighted): 0.9765
    - Testing:
        - Accuracy: 0.2958
        - Average Error: 1.0739
        - F1 (micro): 0.2958
        - F1 (macro): 0.2884
        - F1 (weighted): 0.3006
        - AUC (macro): 0.9799
        - AUC (weighted): 0.9767
- Logistic Regression
    - Training:
        - Accuracy: 0.3301
        - Average Error: 0.9538
        - F1 (micro): 0.3301
        - F1 (macro): 0.2646
        - F1 (weighted): 0.3789
        - AUC (macro): 0.9754
        - AUC (weighted): 0.9633
    - Testing:
        - Accuracy: 0.3288
        - Average Error: 0.9534
        - F1 (micro): 0.3288
        - F1 (macro): 0.2632
        - F1 (weighted): 0.3780
        - AUC (macro): 0.9755
        - AUC (weighted): 0.9634

Example Visualization Output Using our Pitcher Averages:



7. Management Plan (TG)

- Task Assignments:
  - Created and tested outfield models for pitcher average predictions: Lauren
    - Complete
  - Outfield Visualization work: Austin
    - Complete
  - Worked with Database team to get visualizations running on their front-end: Austin
    - Complete
  - Connected to Database to pull information (specifically updated pitcher averages for model input): Trent
    - Complete
  - Created main.py file which is run weekly on the database server and re-trains the pitcher averages models, re-predicts the probabilities for every pitcher's averages in the database (90k+ datapoints), and then writes those predictions to the database: Trent
    - Complete (and tested on server)
    - also created other files needed to maintain the produce such as README and requirements, along with shell scripts to install the requirements
  - Connected to Database for pulling of all data for training models: Trent
    - In Progress
    - Stopped due to lack of historical data in database significantly decreasing training set size

- Development Schedule
  - Week of April 21st-27th:
    - Finish up work on code for the semester

- ensure the server runs everything correctly and displays the predicted pitcher averages correctly
- clean up codebase with refactor and document more thoroughly in the code

▪ Write user and developer docs

- Document how to use and develop the product
- Document what further iterations can be made to improve and add on to the product

8. Memoranda (JW)

Date: 3-25-2024 **Hours:** 1.5
- Discussed our plans for Cycle 3
- Further developed our outfield model and its visualizations
- Talked with database team about our plans for integration on their website and what we needed to do to make the process as smooth as possible
- Talked about the way their website will be setup and potential changes to our model's visualizations

Date: 4-1-2024 **Hours:** 2
- Officially got our outfield visualizations to be both dynamic and functional
- Gained access to the database team's PostgreSQL database and Docker containers
- Attempted to connect to those, but the servers were not working during the meeting
- Discussed plans with the database team about the functionality of the website and how our models would be re-trained and re-ran on new data

Date: 4-8-2024 **Hours:** 1.5
- Got a base model of predictive models up and running on the website with dummy values
- Got pages for each pitcher and each school on the website
- Discussed what our next steps should be to get our real models officially on the pages

Date: 4-9-2024 **Hours:** 2
- Met with the database team members that are in charge of the database itself
- Got them updated on our exact outputs and the methods for integration that would be best
- During this meeting and throughout the next day, we officially got our models up and running on their web application

Date: 4-15-2024 **Hours:** 1.5

- Met with database team and officially got our real models on their webpage
- Tested the functionality of the website, and it seemed to be completely working
- Ensured that the product was acceptable with both Adam and Coach Teaford

9. Source Code

- GitHub Link: https://github.com/tmgavron/shifting_model

10. Presentation Slides

- Presentation Link: Cycle 3 Presentation

11. Previous Cycle Docs and Grade Sheets

- Architectural Spike Report: Architectural Spike Report.docx
- Architectural Spike Presentation: Architecture Spike Presentation
- Cycle 1 Report: Cycle 1 Report.docx
- Cycle 1 Presentation: Cycle 1 Presentation
- Cycle 2 Report: Cycle 2 Report.docx
- Cycle 2 Presentation: Cycle 2 Presentation

12. Sponsor's Approval

Adam and Coach Teaford have reviewed the report and approved of our plans.