Final Year Project Report

Basketball analytics using Machine Learning and Spatiotemporal data from the NBA

Cleon Grant

A report submitted in part fulfilment of the degree of

**BSc (Hons) in Computer Science**

**Supervisor:** Jamie Ward

Department of Computer Science

Goldsmiths University of London

May 13, 2019

Table of Contents

[Final Year Project Report 1](#_Toc8990546)

[1](#_Toc8990547)

[Basketball analytics using Machine Learning and Spatiotemporal data from the NBA 1](#_Toc8990548)

[Cleon Grant 1](#_Toc8990549)

[1. Abstract 3](#_Toc8990550)

[*2.* *Introduction* 4](#_Toc8990551)

[2.1 Glossary 5](#_Toc8990552)

[2.2.1 Relevant Actions 6](#_Toc8990553)

[2.2.2 Traditional Positions 7](#_Toc8990554)

[2.2.3 Screens 8](#_Toc8990555)

[2.2.4 Plays 8](#_Toc8990556)

[2.2.5 Other Terms 9](#_Toc8990557)

[3. Literature Survey 9](#_Toc8990558)

[4. Specification 11](#_Toc8990559)

[4.2 Success Criteria 11](#_Toc8990560)

[2.1 Obtaining the Data 12](#_Toc8990561)

[5. Design and Implementation 15](#_Toc8990562)

[5.1 Loading the dataset. 18](#_Toc8990563)

[5.2 Visualising the dataset 19](#_Toc8990564)

[5.3 Detecting Possession 19](#_Toc8990565)

[5.3.1 Results 19](#_Toc8990566)

[5.4 Detecting a shot 20](#_Toc8990567)

[5.4.1 Results 21](#_Toc8990568)

[5.5 Detecting a pass 21](#_Toc8990569)

[5.6 Detecting dribble 22](#_Toc8990570)

[5.7 Creating Hidden Markov Model 22](#_Toc8990571)

[5.8 Creating visuals based on features 22](#_Toc8990572)

[6. Success Criteria Evaluation 23](#_Toc8990573)

[7. Discussion and Future Work 31](#_Toc8990574)

[8. Bibliography 35](#_Toc8990575)

[9. Appendices 35](#_Toc8990576)

1. Abstract (Half a page or less mainly to help a librarian decide how to categorise your project)

*2. Introduction*

Throughout the history of the National Basketball Association (NBA), players, coaches and fans alike have always had their favourites. Until the 1946/47 season, all parties had to rely on their basketball IQ and a keen eye to separate the skilled players. Today, most people rely on post-game statistics to discern which player is on top of the food chain. These stats only shed little light on how a player performs throughout a match and is not the most accurate form of valuing players; this is especially true for the coaches and managers of these players.

A good and active example of this in the NBA today would be Russell Westbrook, who is famous for consistently having double digits in the three vital stats – Points, Assists and Rebounds. However, when we analyse the games he gets his ‘triple-doubles’, his teammates actually allow him to take all the shots, collect all the rebounds, and only focus on getting ten assists. On paper, this gives the illusion that Westbrook is a triple threat, which may not translate onto the court.

The purpose of this project is to provide more advanced and in-depth analysis of a basketball match. This can ultimately be achieved by combining the time and space features of spatiotemporal data provided by the NBA to detect events as they happen and generate statistics. This data will tell us where a player is at any given time and what they are doing at that time and position.

Coaches are especially interested in this type of information as their eyes cannot be on everyone at once. They have to worry about their players before they can worry about the ones the club have an interest in. It provides a wealth of features such as behaviour analysis/ situation analysis and even probability to succeed after specific actions. It can also produce visual displays such as current paths the ball may follow, which could show how a team may play their offence or a players' tendencies. Advanced Statistics can change the way we glance at NBA games.

There have been many technological solutions towards this in the NBA over recent years. From implementing and automating replays to now introducing multi-camera configurations to track player movement and ball. This technology is the cornerstone of this project. STATS [1] have paired up with SportsVU [1] and the NBA to set up six camera configurations in team's home-courts, which can capture data at 25 frames per second. Through this, it is very apparent that this sort of data and information for teams is appealing to coaches as well as the league.

The goal of this project is to show the potential of spatiotemporal data provided by SportsVU and explore the types of analysis it can infer using Machine Learning techniques. This can be achieved through event detection by looking at the different features provided by the dataset from SportVU.

## 

## 2.1 Glossary

This section will be used to outline some of the technical keywords that will be used throughout the rest of the report that is required to understand some of the assumptions of this project.

A typical NBA basketball match will have two teams of twelve, five active players on the court and seven on the bench. Each team is trying to score in the opponent's basket. While a team is in control of the ball, it is called having possession, and that team would be on offence as their next objective would typically be to attack the basket. The other team would be considered the defensive team. Matches are 48 minutes long and are split into four, twelve minute quarters. The game clock gets stopped during interruptions such as time outs, fouls, the ball going out of bounds and so on. Players have the potential to score 1, 2 or 3 points upon the basket entering the hoop. Three points are awarded for shooting beyond the arc, two for anywhere within the arc and one point for points scored on awarded foul shots. By the end of the 48 minutes, if the score is tied, five minutes of overtime is played, and play will continue until there is a team with more points.

### 2.2.1 Relevant Actions

Basketball is a very complex game and can combine many of actions and movements below to create new actions and movements. However, some necessary actions that are relevant to the preceding sections will are outlined below.

#### Dribbling

* When a player has possession of the ball and moves to another position on the court while bouncing the basketball is considered dribbling. Some may use the term driving, although typically used when a player is attacking the basket and entails that player is heading towards the basket.

#### Shooting

* A shot is when a player attempts to score. There are many different types of ‘shot’ in basketball. A shot is one of the more significant stats in basketball.
  + **Layups:** Where the player drives to the basket and releases the ball as close to the rim/backboard as possible in motion. This is worth two points. (See Appendix A)
  + **Close-Range Shots:** Where the player is relatively stationary compared to the layup and released the ball about 3-5 meters from the basket. This is worth 2 points (See Appendix A)
  + **Long-Range Shots (AKA deep twos):** This is similar to the close-range shot; however, it is about 5-7 meters depending on where you are on the arc. This is usually considered a harder shot as it is still worth 2 points, but you are further from the basket. (See Appendix A)
  + **Three-point Shot:** This is any shot outside of the arc, which ranges from 6.7m to 7.24m This is worth 3 points (See Appendix A)
  + **Free-Throw:** A shot from the free-throw line inside the arc. This is precisely 4.6m away from the basket and is marked by a line inside a circle within the arc. This is always worth 1 point, however, depending on the foul, 1-3 foul shots can be awarded. (See Appendix A

#### Block

* When an offensive player attempts a shot, and a defensive player touches the ball during the upwards motion of the ball, then the defensive player has blocked the shot. This is another primary stat.

#### Rebound

* When someone misses a shot, and the ball touches the backboard or rim, the defensive team have the opportunity to gain possession and go on the offence. The player who grabs the ball at this moment is said to have caught the rebound. This is one of the 5 vital stats.

#### Pass

* A pass is made when a player throws the ball to another teammate. The pass is complete when that player gains possession of the ball. (See Appendix A).

#### Assist

* This is a pass that is transformed into a scored shot. For example, the offensive player is driving towards the basket and sees their teammate with no defenders near them. The player with the ball might pass the ball to the teammate who then proceeds to score. It is said that the player that made the pass then got an assist. This is another primary stat used by the NBA.

#### Steal

* When a defensive player takes the ball from the offensive player. There are rules around how a player might go about doing this (see Fouls below). Usually, after a steal, a fast break happens where players all cut back to the opposite basket at full speed to stop the quick bucket.

#### Foul

* There are plenty of rules across the NBA; how many steps you can take while you are not dribbling; what constitutes to a blocked shot; or goal-tending and the amount of contact that two players can have [2]. All fouls awarded either; a turnover (change of possession); some foul shots (see Free Throw in this section) or a mixture of both. As Basketball is typically officiated, there are commonly three officials to monitor the game, where any three can make calls on a play.

### 2.2.2 Traditional Positions

Traditionally, there are three types of a player split between 5 different positions on the court.

* Guards – Usually the smallest on the team. Very fast with high basketball IQ. The first line of defence after a turnover.
  + Point Guard (PG) – The ball handler and person who organises and facilitates the team on the court. Commonly known as the 1 guard.
  + Shooting-Guard (SG) – A person tasked with scoring most of the points. They tend also to be able to have a good long shot. Also known as the 2 guard. (3-point shooter/2-point shooter)
* Forwards – The taller players who are more physical than the guards, they are in charge of getting rebounds and committing to defence.
  + Small-Forward (SF)– Great for inside and outside plays. First inside option and should be able to shoot outside the arc if need be.
  + Power-Forward (PF)– Very physical as they play closer to the basket where some of the rules change. Good with mid and long shots inside the arc. Known as the big, or 4.
* Center (C) – The most significant and most physical player on the court. Ideally, their role is to protect the rim as and to grab rebounds for the team. Also known as the big, or 5.

### 2.2.3 Screens

Screens are very prominent in the modern game of basketball. These are usually set by the inside players, who are made to stand in a very quiet position in order to block the defender from following the ball handler. This allows the ball handler to take advantage of a situation without a defender. This is the ignition of one of the most basic and famous plays in basketball; The Pick and Roll. This is where the screener, then turns towards the basket and runs alongside the ball handler, hoping to receive the ball for a simple basket. The screener also has to option of going to the 3-point line; this is called popping, changing the name to Pick and Pop. (See Appendix A)

### 2.2.4 Plays

The term 'Play' explains movements in a basketball game that constitutes a basket being scored while the ball is live. (While the clock is running) This can be done through a form of offensive tactic, pick and rolls, fast breaks, post-ups and many other situations. (See Appendix A)

### 2.2.5 Other Terms

Spatiotemporal data – A dataset that contains both space and time information. In regards to this, it will hold any players’ location at any given time.

Advanced Statistic – Any statistic not currently provided by NBA Stats in their official statistics nor their advanced statistics [3]. Some examples of current advanced statistics are below.

* **Player Efficiency Rating** – A statistic that measures a plyers per-minute productivity. Its an algorithm that rewards positive contributions a player makes to his team and penalises the negative one in a point value system. It is then adjusted to the pace of the game and time on the court for better comparisons to other players.
* **Win Shares** – This is similar to Player Efficiency Rating, only it estimates a player’s contribution to their teams win total. This essentially values players based on wins, splitting them in to offensive and defensive win shares.
* **True Shooting Percentage** - Typically, evaluating shooting efficiency is difficult as traditional field goal percentage values 2 pointers and 3 pointers equally, even though 3 pointers are far more valuable. It also doesn’t take Free Throws into account, which can very easily turn the tide of a game. True Shooting Percentage weights free throw attempts and uses total points scored rather than just field goals scored.

3. Literature Survey

The goal of this project is to provide further insight for coaches into players and in game situations through the use of spatiotemporal data and provide advanced statistics. There is a growing field of research that combines Machine Learning and Deep Learning with Computer Vision to relay positive results. There have been a large number of projects that are including spatiotemporal analysis on different sports such as football, tennis and even American football.

There have been many projects of people using different types of data as well as different methods to obtain their results. Currently, within your standard NBA team, it is typical for a team to have multiple personnel to help coaches and agents with decision making, with very little use of the software available. There are a couple of projects in the works with large teams such as Second Spectrum who are aiming to find a centralised solution [4].

Overall, during background research, several projects have come up can be grouped into the following four categories: (Explain examples individually when you add them in )

* Visualisations– This usually involves a project that aims to visualise some data from an NBA dataset. Such visualisations commonly come in the form of heatmaps [5] (Using colours to show player tendencies on the court) or even the typical trajectory of a shot.

There were some fundamental issues with these reports. They did not exactly achieve much analysis of the game and focused more on what could be inferred visually from users, rather than automatically by the program.

* Event Detection – This tends to feature programs that use some unsupervised machine learning or deep learning to detect when a specific type of action or play was happening in a basketball match. [10] [11]

There were instances of projects that used deep learning algorithms to detect some events such as RNN. Deep learning solutions are a plausible route to take to solve this problem; however, it can be costly to execute on lots of data. It also is much more complex to understand a deep neural network than incorporate an unsupervised machine learning algorithm. As time and computing power is a constraint, it does not work well for this project.

* Spatiotemporal Analysis – Typical reports looked at how players moved around the court and how other players’ movements and actions affected their decisions. [6] [7] [8] [9]

These reports were best as they gave an insight on how to understand the data as the authors had come to their conclusions on the data themselves. However, the problem with these projects is the fact that they were outdated.

* Critiquing NBA traditions – There were plenty of reports and articles that looked at features of modern basketball that were not precisely applicable compared to when the standards came about many years ago. One report looked at the positions on the court, critiquing whether we should continue to use PG/SG/SF/PF/C as the five positions as many players are combinations of 2 or more positions which lead to new types of players. [12]

These projects were the most impressive considering I have personally been playing basketball most of my life, and it does challenge the way you think about the game, with some concrete evidence to support its claims. However, this research area was far from the main focus and attention of this project but is considerable research for future work.

The final aim of this project was to be able to achieve a wholesome system that could accomplish all of the above research areas. Therefore, the research, as mentioned earlier, is to be used as the criterion to check against this project. The final aim for this project was to be able to achieve a wholesome system that could accomplish all of the above research areas. And thus, I have decided to use these areas as a base criterion to check against my project.

4. Specification

In this portion of the report, a definition of the ideal solution to the problem mentioned in the previous sections will be discussed. In terms of creating a complete solution for coaches, an ideal system would have to include live visualisation of the data to use as a reference point to see what the computer sees. Such reference points will help the coach validate what the system outputs during the analysis. It would also include some event detection based on the spatiotemporal data. Event detection means that it would use the movement data of the players and ball to interpret some events during the game. The next step is to use the events detected in the data to infer whether a player is going to switch between events by training a machine learning model to pick out trends in matches and create likelihoods for each transition. From this point, the ideal system would also be able to do a similar thing for offensive and defensive plays, which extends to more complicated plays; detect the event of a defensive setup, and look at the transition probabilities between them to make some predictions for future games.

There is much scope for this project, following this iterative process of: Visualisation > Detection > Analyse > Probability prediction; as this cycle can be applied to multiple areas of basketball. Coaches will be able to analyse how a player behaved during a game to certain events as well as predict how other players will behave to those same events by using that information. It would also have the potential to then advise the coach on a players’ performance based on the analysis provided by the system based on those events, whether it be a specific play to how successful a play is running through a player.

## 4.1 Success Criteria

Based on the limitations mentioned above, the success criteria below will be used to aid in evaluating the system and judging the success of the project as a whole.

1. Gather suitable experimental data. This would be a database that would hold the X and Y positions of the players and the ball throughout a whole match. The frame rate of the data would not be the biggest cause for concern, as long as it was more than once a second. This would allow the gathering of different types of plays and observe different instances of events.
2. Visualise the data. Get an understanding as a developer what was happening in the dataset. This step served the dual purpose of showing the meaning behind specific numbers and their results in action as well as supporting the visualisation portion of the project. A successful visualisation should emulate the look and pace of a real basketball game so that any given coach or knowledgeable person can figure out what is happening in the visualisation at any given time.
3. Create an algorithm/machine learning model detecting one basic action in basketball. The necessary actions mentioned in section 2.2.1 outline some of the actions I was aiming to detect. As expected, the data in the majority of datasets are unlabelled, meaning the number of defined actions that are detectable by the algorithms should be at least 3. Some actions are more straightforward to detect than others. Also, some actions do not give much insight, such as knowing who is dribbling at what time.
4. Extract at least one advanced statistic for each action and create a visual display using it.
5. Create an interface that can select specific players on the court and view all the above information at that time.
6. Improve the accuracy of the model to be more accurate. Must at least be better than baseline.

As this is an academic project, there are portions of this criteria that would be missing if it were to be implemented in the real world such as increasing win rate of a team or improving player value due to new play style after gaining information from the advanced analytics.

## Obtaining the Data

For this project to work, It would first need to have access to NBA spatial-temporal data. When starting this project, a large portion of the time was invested in looking at live camera solutions. There is one app in particular that shone above the others called HomeCourt [13], that was able to help community players keep track of their stats. All the user required was to place their phone down pointing towards to hoop, and they could begin taking their shots. It would be able to detect where on the court the shot was taken and provide some stats like how many shots were made and missed.

It was decided not to pursue a video solution was due to the complexity and time constraint of including computer vision and machine learning with this project; however, it has the potential for future work.

The data that was used for this project came from an open source R project from Github that was using one match from SportVU data [14]. The decision to use this instead of SportVU's full dataset is due to the fact that the datasets they offered at the SportVU website hold 8 million entries. For some rapid development, it would be easier to use a dataset that held fewer data. This entry from the Github was a single match from the same dataset, which meant that this could be used to save time. By doing it this way, the code and models can be applied to any part of the large dataset as it will all be structured the same.

Permission for use of SportVU's dataset was requested for this project in which they were happy to comply as long as they were referenced and had access to the report and code (See Appendix B). They were also happy to provide play-by-play data from the same website; however, it did not get around to being used.

Many other datasets could have selected. A few people were providing their self-made databases which may or may not have been generated automatically. There were also some companies selling licenses to accurately collected data solely used for Deep Learning and Machine Learning, however, said licenses were definitely out of budget [15]. If this project were to continue and got funding, these datasets are ideal and would be implemented instead.

For this project, the amount of data I was receiving for the cost and effort it took to obtain was good enough to use and fit the project scope well.

The success criteria will be the means to measure just how closely this project follows what it set out to do. For this reason, each item in the criteria will be used as a general test case to ensure that the program ends up looking as intended. The information will be put into the following table:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Criteria Number | Summary | Steps | Expected Result | Actual Result | Success? If not, why (Y/N) |
| 1 | Gather suitable experimental data and load it into the program. | >Find a Database with locations of player and ball>convert to python data type for later access | Database gets loaded and can be accessed by a variable. Even have different important columns and subcolumns as separate variables. |  |  |
| 2 | Be able to visualise match given related SportsVU data. Get an understanding as a developer what was happening in the dataset. | >Get players represented as points on graph>Visually separate by team and colour ball>Animate movement> Add image of the court | Have 11 moving dots on the screen. 1 colour for the ball, 1 colour for the 5 teammates and another colour for the other 5 team mates. They should all move independently and according to the data on the background of a basketball court. |  |  |
| 3 | Create some machine learning models and algorithms to detect at least 3 basic basketball actions. | Pick most effortless action to define>Look at the type of data relating to it>Consider possible learning algorithms> Split data> Get some Labels>Test > Repeat | The system should be able to correctly identify at least 3 basic actions fairly accurately ( greater than 50% accuracy) |  |  |
| 4 | Extract at least one advanced statistic for each action and create a visual display using it. | Pick out one action detectable by the program > consider how knowing when and where an action happens can benefit a coach> Show it on a graph against time or other players. | Each action should be able to provide some increased insight into the game happening to the user compared to average box scores and information on NBA stats website. |  |  |
| 5 | Create an interface that can select specific players on the court and view all the above information at that time. | Have the visualisation in a small window>Make the players clickable to load up current statistics about them>Compare stats with other players. Display stats from the game and current stats. | The user should be able to cleanly navigate through the program with sectioned areas made by using Matplotlib’s subplot functions to make small plots inside the main plot. This creates a type of GUI interface for the user to navigate. |  |  |
| 6 | Improve the accuracy of the model to be mostly. Must at least be better than baseline. | Look for edge cases within the way I have detected events within the game to decrease the true positive/negative results | Have a higher success rate with at least one or more detected basketball actions. |  |  |

5. Design

## 5.1 Requirement Analysis

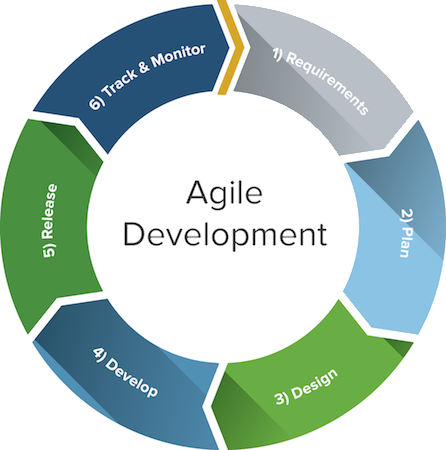
To further validate the use of this project, I wanted to get a better understanding of my end users. I attempted to get into contact with some US professional coaches, however I did not hear a response back. Therefore, I resorted to contacting some local coaches from the UK. This was done in hopes to emulate the mind set of a professional NBA coach. I had full consent to the use of this information for the purposes of this project. The answers for the questionnaire were answered digitally for easy analysis. There were 7 closed questions. The results of this can be seen in the appendix (See Appendix C).

### 5.1.1 Results Summary

* The mean years coaching rounded up was 7.5.
* On average, coaches scored the importance of game statistics at 8.4/10
* They also thought that the post game status reflected the match shown by a result of 7.9/10
* However, they were only slightly likely to check advanced statistics if available with an average response of 3.5/5.
* 90% of coaches have looked into statistics of their opposing team or players before a big match
* 100% of coaches surveyed think that these types on insights will be beneficial to their coaching careers.

## 5.2 Planning

In order to successfully structure this project, an agile development style process was taken up. This follows a wheel of development processes that software engineer and systems analysts follow to implement or maintain a system. An example of the cycle mentioned above that was followed in this is shown below.



In order to plan this project, a critical path was established. Critical paths hold milestones that are required to produce a minimal version of the project. Following this path also gives a logical follow to the project. Creation of a Gantt Chart was the next step (See Appendix D). This allowed the addition of some timescales to selected milestones as well as the addition of smaller tasks that could make made up those milestones. Gantt charts also provide an overview of the entire project and its timescales. Following the critical path would result in an effective minimal viable product. The critical path of the program was as follows:

1. Retrieve some input data
2. Visualise the data
3. Classify one aspect of the game
4. Create some analysis for that detected feature

The critical path was effective because I could also use it as a workflow, to complete multiple modularised sections of this program. As the weeks went on, I wrote up some weekly logs on what was accomplished on what date as well as what the plan going forward was.

## 5.3 Design

I did have an original design in mind for the software. This was briefly outlined within the Preliminary report and can be seen in the Appendix. As I will mention later, there was been no advances on a design as it was never implemented due to constraints in time. However, if this project were to continue in the future, there is enough content and room for a full blown system with UI. (See Appendix D)

## 6. Implementation

## 6.1 Justification of Programming Language

The programming language I chose to develop in is Python 3.x. This reason for this is because it has plenty of available libraries that specialise in data processing and Machine Learning algorithms. It also meant that I could use PyCharm, (Integrated Development Environment) with special add-ons to make data science tasks easier like code cells and live dataset visuals. In order to obtain these packages, I had to create a Conda environment which held more Python packages for me to use such as Numpy, Sci-kit learn, Pandas and Matplotlib.

This project was completed using a MacBook Pro early 2015, 8GB Ram, 2.7GHz i5 Intel Processor. Thus, is expected to work on anything of a similar specification or higher.

## 6.2 Loading the dataset.

The dataset provided is a JSON file. Within python, this can be directly translated into a dictionary using the JSON library. The problem with this method is dictionary searching for keys are very slow as dictionaries in python are equivalent to hash tables. Hash tables have a worst-case time complexity of O(n).

Once a dictionary contains the full dataset, It can be converted into a Pandas ‘DataFrame’. Pandas is a popular python library that offers a robust data structure called DataFrames that make manipulation and analysis easier. Pandas allow for SQL type operations on the dataset as well as allow the application of functions on multiple rows of data in excellent time.

The dataset was split up into several layers. ‘Events’, ‘Moments’ and ‘Positional Data’. (See Appendix E)

* Events hold the name of the two teams playing, the names of all the players, their positions (see Traditional Positions 2.2.2) jersey numbers, game date, event ID and moments.
  + EventID was good with providing a broader understanding of whether the data flows on from the last entry or not, as the numbers were not always concurrent. Event ID revealed that there was missing positional data, which would show on a visualisation.
* Moments are all the instances the SportVU camera captures from the basketball game. This dictionary held what Quarter the game was in, the absolute time, game time left, whether the shot clock has ended and positional data. There was one column that always held ‘Null' in every entry making it difficult to discern its representation. Resources online were also not able to shed light on this mystery column. Therefore, it is assumed to be a placeholder that may get used in the future. 'Absolute time' would have to be converted as it would be in Unix time; however, there are modules for that in python.
* The positional data column was itself its own dictionary. This would capture the 10 active players on the court as well as the ball. The columns would be the ‘team ID’, ‘playerID’, ‘x position’, ‘y position’ and ‘ball diameter’. The teamID and playerID of the ball was always -1. The ball diameter for the 10 players was always NaN (See Appendix E).

## To make everything easy to access and work with it is ideal to make all these portions of data into variables to access them.

## 5.2 Visualising the dataset

The best way to see everything that is happening on the court is through a top-down view of the courts, with the players being represented as dots. Something along the lines of the image below. [insert picture]

## In order to achieve this, the python library Matplotlib is required. This is a 2D plotting library that gives a Matlab-like interface. A scatter graph over a picture of a basketball court is used to display and represent the ball and players on the court. The ball is made smaller and any player with the same ID the same colour. Initially, this was animated using multiple nested for loops to read the right data from the dictionaries and animate them. Even though this managed to animate the data by making the scatter points move, this caused the animation to be very choppy and slow. Fortunately, Matplotlib has an animation function that can smoothly animate arrays, dictionaries and even Pandas DataFrames. The logic and flow of the code do change; however, as initialisation of all the items in the animation is first required. Then have the animate function update their values as the animation goes on.

## 5.3 Detecting Possession

The first thing worked on was detecting possession of the ball at any given time. The reason possession was chosen first was it was one of the more comfortable things to receive results for. Secondly, once the player with the ball is known, advanced metrics can start to be inferred. Furthermore, it can also aid the detection of some of the necessary actions such as detecting passes or increase the accuracy of whether an event was a shot or a block instead as possession should not change during a shot, but will for a block.

As this dataset is unlabelled, the machine learning algorithms I can use are limited. I decided to just use KNN (a supervised learning algorithm) with K = 1. When K = 1, the algorithm doesn’t require labels as the correct label/answer would be the nearest data point. As the ball moves around along the X and Y axis, its nearest data point (player) changes constantly. If I can return the nearest 1 neighbours to the ball, I can infer that neighbour is the player in possession. 1NN is statistically defined as



Where Classifier (C) assigns the label Y as the closest data point to X in the feature space.

### 5.3.1 Results

This performed mostly well as a base case. As mentioned earlier, this data is unlabelled. Therefore, in order to measure the performance of this basic form of detection, I decided to pick out a 3 random windows of data that were 10 seconds long and print out the output. I then passed that portion of data into my animation and observed how many times possession positivity changed. I would then note how many false positives and how many false negatives it got. (Show some results)

## 5.4 Detecting a shot

The detection of a shot within the game was very difficult to overcome as the data is index in relation to time. As a shot isn’t a single instance of time, but over a period of time, it was hard to quantify and figure out where to place my outputs in the dataset or how to input the data.

In the end, I decided to use a sliding window, which is a Pandas method that allows you to have a variable hold a defined window size of data, that will automatically move through your Dataframe. This works much better for this as a shot needs to be observed over a period of time and not for one instance.

The column of data that was observed was the ball diameter. The reason for this was because the SportVU cameras were placed where the box score was – over the center of the court. This meant that if the ball went up into the air, the diameter of the ball would increase, and if the ball went down, the diameter would decrease. Based on this, I would be able to infer whether the ball was going up and down for long enough periods.

The final problem I had to overcome was how I was going to pass 5 data points at a time into an unsupervised machine learning algorithm. So I decided to find the mean of the rolling window, so the average height of that window. This gave me a rolling mean over the dataset. This allowed me to see the general direction of the ball very quickly which helped classification. I decided to make my sliding window cover the equvilent of 2 seconds in real time. This is because it takes players on average 0.8 seconds to release their shot ( show evidence). Therefore, 2 second windows will be able to catch the moments before and after the release of the ball while capturing the change of diameter well.

I created 2 models for this task. One model was using Mean Shift. Mean shift is an unsupervised algorithm that doesn’t require any parameters and focuses on locating the maxima of a mode seeking algorithm to find clusters. In short, it aims to find dense areas in data. For this, I used Sklrearn’s MeanShift function which only requires a bandwidth to work. Bandwidth describes how dense the areas have to be to be taken into consideration. Sklearn also provided an estimate\_bandwidth function that can estimate the bandwidth for any data given. This algorithm was great as it didn’t require any tuning, however it was very, very slow with large amounts of data. When used on the entire dataset, it would take hours to create labels for the windows. Through using Mean Shift, I realised it was only ever giving my labels between 2 and 4 labels. This gave me the idea to use KMeans clustering with K being 4 as it was the highest number of clusters given by MeanShift.

Time Complexity

Mean Shift K Mean

** 

k = Number of Clusters

n = Number of points to be clustered

T = Number of iterations until convergence

As Mean Shift increases at a faster rate with time compared to K means, I have decided that it may be better to use K means. Meanshift is also very sesative to the selection of bandwidth (h). Similar to learning rate, a large h can decrease convergence time, but might result in merged clusters. This means inputs may be clustered incorrectly even if they’re distinct.

KMeans clustering is a type of unsupervised learning algorithm that aims to find K groups in the data where K is predefined. It works by assigning each data point to on of K groups based on similarity. As I knew that I wanted my K to be 4, split my data into training and test and validation and created my model. I then use the centroids of the 4 clusters as benchmarks for analysis as I know that if a window goes from a high-medium (cluster 3) to a high value (cluster 4), then ball is rising and is likely going to be a shot.

### 5.4.1 Results

First thing to mention was the amount of time it took for computation. K-Means is considerably faster when using a larger portion of the dataset. One thing I noticed was the clusters were mostly identical between Mean Shift and K-=Means with K = 4. In order to judge the success of this algorithm were to look at the actual box score for this game as it will tell you how many shots the team took in total. As this is not currently team specific, by adding the two values together, you can see the total value of shots and compare it to the amount of shots counted by the model. (Do this) The error would be the distance between these two numbers. This is not true error as it doesn’t do this between each shot but does this against the final amount of shots. I also found that the model was able to easily cluster the sliding windows on any part of my validation data.

## 5.5 Detecting a pass

For this is section, the use of a learning algorithm isn’t required in my opinion as pass can simply be defined as when possession changes from one player to another of the same team. As I can detect possession per player and use the dataset to see what team they’re on, I can use possession to infer passes. This does mean however, that the accuracy of a pass is mutually dependant on the accuracy of possession. In my program, I simply check if the possession has changed, if it has changed, is it someone on the same team, if so, then it is a successful pass, however, if it qualifies as a shot (which may go over many players of the same team) then it does not qualify as a pass. I will mention in later sections how to improve this accuracy and reduce this dependency.

## 5.6 Detecting dribble

For detecting dribbling, idea is that if you aren’t shooting or passing, and you are moving with the ball, while in possession, then you are dribbling the ball. Therefore, I have decided to infer the following information using a Hidden Markov Model. A Hidden Markov Model (HMM) is a probabilistic model with some state (x) that can change over time. You cannot observe this state but you can observe something correlated to the state (y). The model will need to specify the probability that maps y to x as well as take into consideration the previous states and give a probability that x is happening.

## 5.7 Creating Hidden Markov Model

In order to come up with the probabilities of each state transition, I would need to look at the occurance of them within my data. If I know that a player shoots 40% of the time after their dribble

In relation to dribbling, I know that I’m not shooting, and I’m not passing and my Y would be the changing coordinates of the player and ball. I would also know that my possession is with this same player, thus the probability would be high.

## 5.8 Creating visuals based on features

As I’ve been able to find out information in relation to time on the court, I am able to create some visuals and statistics that can be really useful to coaches and players alike. One thing that might be useful is showing an animation of how the ball travels on each possession. This can very easily show a coach how their team is playing, and if they’re correctly executing a play properly. This also outlines how an enemy team may usually play very quickly. This is done by converting and smoothing all the points into a line, and picking windows of data to view how the ball travels in that time.

I can also create a graph showing who they play through the most by simply counting how many seconds the player holds the ball for. This could mean that a team uses a player as a main form of offense, or that this player holds the ball too long compared to the rest of the team when there maybe be a better player to let hold the ball.

In regards to shooting, a heat map can be generated of where players tend to shoot against a team. This provides positional advantages to coaches as they can see where other teams make most of their shots in a game and exposes the wholes in the opponents defence. It can also help expose weaknesses in their own defence.

I can also create visuals on passes made by players. Using Matplotlibs arrow function, you can see the direction of the ball at any given point by looking at the changing x and y locations of the ball. This can help show the coach how a team moves the ball. This can be really good for knowing who to focus the defence on against another team it will reveal who the ball goes through the most to create the offense. Some teams use the Center player to run an offense considering they’re the biggest and tallest. Whereas some might prefer to run through the Point Guard as they tend to have the highest basketball IQ on the team. Knowing who the ball needs to get to can really give an upper hand.

Lastly, I can produce running stats on all the players as the game goes on such as distance travelled, where they went on the court and even the player they connect and pass to the most.

Flow of program

# 6. Success Criteria Evaluation

For this section, I have split it into success criteria validation and verification. Each sub section will hold a table explaining where these criteria have been met or not and how/why.

* + Validation: does your system fulfil the original requirements?

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Criteia Number | Summary | Steps | Expected Result | Actual Result | Success? If not, why (Y/N) |
| 1 | Gather suitable experimental data and load it into the program. | >Find a Database with locations of player and ball>convert to python data type for later access | Database gets loaded and can be accessed by a variable. Even have different important columns and sub columns as separate variables. | JSON type databases can be loaded into the program, this would be a single match that (proving it has the same column headers) can be loaded into the Pandas DataFrame for quick access later in the program. | Y |
| 2 | Be able to visualise match given related SportsVU data. Get an understanding as a developer what was happening in the dataset. | >Get players represented as points on graph>Visually separate by team and colour ball>Animate movement> Add image of court | Have 11 moving dots on the screen. 1 colour for the ball, 1 colour for the 5 team mates and another colour for the other 5 team mates. They should all move independently and according to the data on the background of a basketball court. | Currently, I have achieved the expected result. I have decided to create my own court using Matplotlib as it can always be the correct size as I’ve had problems previously with using internet pictures. | Y |
| 3 | Create some machine learning models and algorithms to detect at least 3 basic basketball actions. | Pick easiest action to define>Look at the type of data relating to it>Consider possible learning algorithms> Split data> Get some Labels>Test > Repeat | The system should be able to correctly identify at least 3 basic actions fairly accurately ( greater than 50% accuracy) | The program gives an output every time an action is detected. As mentioned earlier, unsupervised machine learning does not have a means to test itself against automatically and testing was done manually by me. | Y |
| 4 | Extract at least one advanced statistic for each action and create a visual display using it. | Pick out one action detectable by the program > consider how knowing when and were an action happens can benefit a coach> Show it on a graph against time or other players. | Each action should be able to provide some increased insight into the game happening to the user compared to normal boxed score and information on NBA stats website. | I have selected some specific insights that can be shown based on the things I can detect within the program. (Reference) | Y |
| 5 | Create an interface so you can select specific players on the court and view all the above information at that time. | Have the visualisation in a small window>Make the players clickable to load up active statistics about them>Compare stats with other players. Display stats from game and current stats. | The user should be able to cleanly navigate through the program with sectioned areas made by using Matplotlib’s subplot functions to make small plots inside the main plot. This creates a type of GUI interfece for the user to navigate. | This has not been explored. | N  Reason: Due to time constraints, it was the least important feature as this is purely academic and UI serves no purpose to the functionality of the program. |
| 6 | Improve accuracy of model to be mostly. Must at least be better than baseline. | Look for edge cases within the way I have detected events within the game to decrease the true positive/negative results | Have a higher success rate with at least one or more detected basketball actions. | Whilst programming, some edge cases were taken into account and accounted for such as detecting dribbling. All models perform better than baseline. | Y |

Based on the table above, I can say that my program mostly fulfilled the original requirements. This is due to most of the success criteria being completed (5/6).

Test cases for models

Possession:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test Number | Test case | Expected Result | Actual Result | Success? (Y/N)  If no, why? |
| 1 | Does it show possession of current team? | The program should be able to output the current offensive team | You are able to see what team is in possession through it’s team ID. This can be cross referenced with external data or the teams IDs at the top level of the dataset. | Y |
| 2 | Does it show possession of current player? | The program should also be able to tell the user what player is in possession. | You are able to find out the player ID of the player nearest to the ball. | Y |
| 3 | Does it time how long a player has the ball for? | The program should be able to time how long someone has the ball for and store that in against that players ID. | While a player has new possession of the ball, the current absolute time is taken and stored. When possession changes, the absolute time is taken again and the difference is added to that players’ time. |  |
| 4 | Is the time accurate? | The time recorded against each player should be close enough to actual time and be reflective of the amount of possession in the game. | This is accurate because I’m using the absolute time from the dataset to start and finish the elapsed time. This means that the calculated timing of possession are as accurate as the capture rate of the SportsVU camera. The only time it would not be accurate is through the way I detect possession. This is because if a pass is made, that goes past the opposition, then that opposing player will have possession time added, even though they were never in possession. This is the biggest flaw, however, there are a few things that can be done to improve this accuracy that will be mentioned in later sections. | Y |
| 5 | Is it better than 50% accuracy? |  |  |  |
|  |  |  |  |  |

Shooting:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test Number | Test case | Expected Result | Actual Result | Success? (Y/N)  If no, why? |
| 1 | Does it detect when a shot has been taken? | There should be some sort of indication that a shot has been detected. | There is an output in the console currently when there is a shot. However, this can be easily change to something more useful. | Y |
| 2 | Does it show who has taken the shot? | When the shot is taken, the person who took the shot should also be identified. | The player is output in the string that gets shown in the console. This is just the player ID and team ID of that player. This can easily be translated to full names. | Y |
| 3 | Does it give the time of when the shot was released? | When the shot is released, it should return some information about the time it was shot. (Game time, shot clock time left, absolute time) | N/A | N  Reason:  I haven’t added the code as I wanted to focus on getting more events detectable which will allow for more information to be inferred. However, the addition of this case is very simple as there are columns for each moment that hold information about time such as: Game time, shot clock time left and absolute time. |
| 4 | Does the total number of shots match up to the box statistics for the match for total shots in the game (including layups, 2 point shots and 3 point shots) |  |  |  |
| 5 | Is it better than 50% accuracy? |  |  |  |
|  |  |  |  |  |

Dribbling:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test Number | Test case | Expected Result | Actual Result | Success? (Y/N)  If no, why? |
|  | Does the program tell the user who is dribbling the ball? | This should return the player ID of the person dribbling with the ball. | This will return the player ID of the person dribbling, as for a player to be dribbling, they’re automatically in possession of the ball. | Y |
|  | Can you find out how long they’ve been dribbling for | This should be able to give timing relative to the game time or absolute time. | I am able to return the amount of time someone has been dribbling for by counting how the player was stationary for when they get possession. I then subtract this number from the running possession time for that player. This would be the amount of time they’re dribbling for. | Y |
|  | Can you find out where they’ve been dribbling the most? | Be able to keep track of the areas where the players have been dribbling and show some visual i.e a heat map | Currently, I use a smoothed connected line to demonstrate the path of the ball. This can be changed to become a heat map by making the lines represent colours | Y |
|  | Find out player with highest dribble time? |  |  |  |
|  | Is the time accurate? |  |  |  |
|  | Is it better than 50% accuracy? |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

Passing:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test Number | Test case | Expected Result | Actual Result | Success? (Y/N)  If no, why? |
|  | Can you detect when a pass was made? | Have an output show when a pass has been made by a player. | Currently, I have a string printing out in the console when a pass gets made. | Y |
|  | Can you detect who it made the pass and who it was to? | Return the passer and receiver of the last known pass. | I am able to get this information by finding who the new player in possession is. However, it’s not been implemented. | N  Reason: Time constraints, it hasn’t been at the top of my priority list. |
|  | Can you find out who has been passed to the most? | Similar to possession, find out who has been passed to the most by counting how many received passed a player has. | N/A | N  Reason: Time constraints, not being able to complete the last test case means I haven’t been able to complete this case. |
|  | Is it better than 50%? |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

* + Verification: does it function correctly according to the specification?

# Discussion and Future Work

This chapter is aimed t revisiting the projects aims and objectives and evaluates the success through observing the methodology used to complete it’s tasks. It will also discuss further proespects for future work and the type of knowledge gained upon completing this project.

Aims of Project

Visualise a game using Matplotlib

Have possession detected using 1NN

Detecting shooting using KMeans Clustering

Detect a pass through the detection of Possession

Detect a dribble through HMM.

Throughout the report, I have achieve some aspect of these goals, although not necessarily in the best way. In this section, I will critic parts of my project that can be improved for much better results.

7.1 Visualise a game/ The dataset

This was the most consistent, as matplotlib had a lot of documentation regarding the animation module. The biggest problem wouldn’t so much the visualisation, but the data. This is because the animation will instantly reflect any missing pieces of data. This is does happen with datasets where no information is provided as to why data is missing. This causes a problem because the actual entry wouldn’t be null, it just wouldn’t exist at all. For example, the event ID mentioned in the beginning of the report usually is sequential. There are some moments where the are not continuous. This makes the animation look very choppy and unprofessional. As the end user would be a professional coach, this should be as accurate and professional looking as possible.

Furthermore, the dataset I ended up training my models on is of a 2015 game. Training on old data isn’t a good idea as basketball is constantly evolving and the style of play changes. This means the features are also going to be different. With the project in its current form, the year does not matter. On the other hand, the more that gets detected (such as whole offensive plays) the more the data has to be relevant.

Despite these things, the dataset was very rich with information that can easily be used to detect different pieces of interesting information. It was also very easy to obtain and visualise. Based of this, I would say the visualisation and chosen database for this project was successful.

7.2 Possession

In order to detect possession of a player, I return the nearest player from each team. As mentioned in previous sections, this is a quick but very inaccurate way to do this. This is because if player A passes the ball to player B with defender A in between, defender A will have ‘possession’ as the ball travels past them during the pass. This is also true for long distance shots and loose balls (When no one has possession due to an interception or missed shot). Possession is the foundation of detection within this program and unfortunately affects many of the other calculations such as dribbling and passing. I would say that even though I achieved my goal noted within the success criteria regarding possession, achieving better than 50% accuracy isn’t great for something that plays a fundamental role in detecting events. If I had more time on this project, I would use passing to help support possession detection. I would combine the trajectory of the ball with the passer and receiver locations to discern if the pass intended for that player. If the receiver has the ball, then the time that would of normally gone to a nearby defender, would instead go to the 2 teammates that completed the pass.

7.3 Passing

Continuing on from section 7.2, my passing is based off the accuracy of possession. Having the accuracy for possession lower than preferred, naturally makes the passing accuracy low. As aim was to have a foundation for detecting events, I would say this is successful as it’s accuracy solely depends on possession. If the possession can exclude nearby defenders and people underneath the basket, then the passing would also be much stronger.

7.4 Shooting

Shooting in basketball is a very broad term in basketball. Many people imagine a 3 point shot or even a 2 point jumper when they think of shooting, however, as mentioned in section 2.2.1, it can also include close range shots and layups. This makes the amount of shots relatively high. At first, it looked incorrect until you start counting the amount of shots there are in a game. For this reason, I believe this was successful. One thing that can improve this feature is to separate the different types of shot. This would be pretty difficult considering that I’m using 4 centroids to figure out if those 2 seconds of ball movement are high or not. The higher centroids are considered to be shot. The lower centroids are ignored. One possible method would be to increase the value of K. Another possible method to improving this detection would be to figure out the diameter of the ball when its in line with the rim. This might be able to be done by finding the mean of the centroids. Anything above this height should be considered a shot, in which case you look at the movement of the player in possession as well. If they’re stationary, and behind the arc it will be considered a 3 pointer. If they’re in motion under the basket, it would be a layup.

Dribbling

Overall, I believe the biggest shortfall of this project is the possession accuracy as it has become a very important factor of the detection of other things. This makes this inapplicable to the real world as you’ll end up with inaccurate detections. On the other hand, this is an excellent starting point for a very expansive system that has great potential to provide some strong and accurate analysis for a real head coach. Despite this, most of the aims that were set were achieved and it is just a matter of improving on those features.

Future Improvements

In its current state, the program would not be fit for professional use (As the target user is NBA coaches) as it does have a considerably low accuracy for possession (Which ended up being the foundation for other detection methods) . I have complied a short list on the most important things that would need to be worked on in order for it to be more ready for professional usage.

* Improve possession accuracy.
* Use player trajectory to define if a shot is good or bad
* Use more data from more recent matches to fit the data to.
* Consider Deep Learning for classification. (RNN)

Knowledge Gained

After all the hard work on this project, I really got a feel for Machine Learning and the challenges that it brings. This is due to the process of finding real world data, interpreting it and trying to help an algorithm make sense of it. I managed to learn about plenty of unsupervised learning methods such as Mean Shift and Bayesian Models such as the Hidden Markov Model. I also took a small dive into Semi-Supervised Machine Learning which holds potential for future improvements. It was a steep learning curve also as I have little experience in Sklearn, Numpy, Pandas and Matplotlib, let alone combining them all together. There were also functionalities within these libraries that weren’t well documented or used often, which took hours of extensive research and trial and error to overcome. Completing this project forced me to use features of PyCharm that I didn’t even know existed (E.G Scientific Mode, Cell Mode etc) This project also helped me fine tune my agile development methodology as well as my planning skills. I have definitely gained more soft skills as well as technical skills throughout this journey.

Project Management

Through the use of several project plans, Gantt charts and weekly logs, the project ran pretty fluently. Even though the project had a lot of scope, it was felt very manageable through the use of agile development methodology, as everything was its own module that can be tweaked and fine-tuned.

Conclusion

The aim of the project is to provide a system that can produce head coaches advanced statstics and insights through Machine Learning on spatio-temporal data. Looking at the project as a whole, it was a successful project as it provided a strong foundation for great detection and stastics to be provided. Over the 6 months, I managed to achieve most of what I set out to do, only missing out on accurate readings due to time constraints. Currently, there is only a single project I’ve seen that aims to combine all these insights into one place. This is a team of programmers who have been working on this for years. I believe I have done relatively well considering I’ve had much less time as well as less minds working on the project. If I had the resources available, I would be able to increase the amount of things detectable such as being able to detect whole offensive and defensive basketball plays and work on the accuracy of these detections. Overall, it has been a remarkable experience to learn about Basketball and the NBA in much more depth. It also was also great for self-learning.

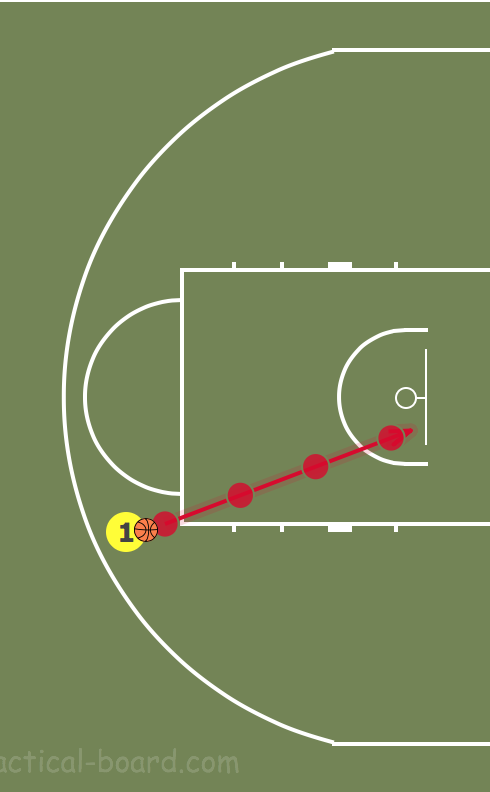
* + How successful or unsuccessful was your work?
  + How much of what you originally set out to achieve did you achieve?
  + How did it compare to what others have done?
  + How could it be extended if you had had more time?

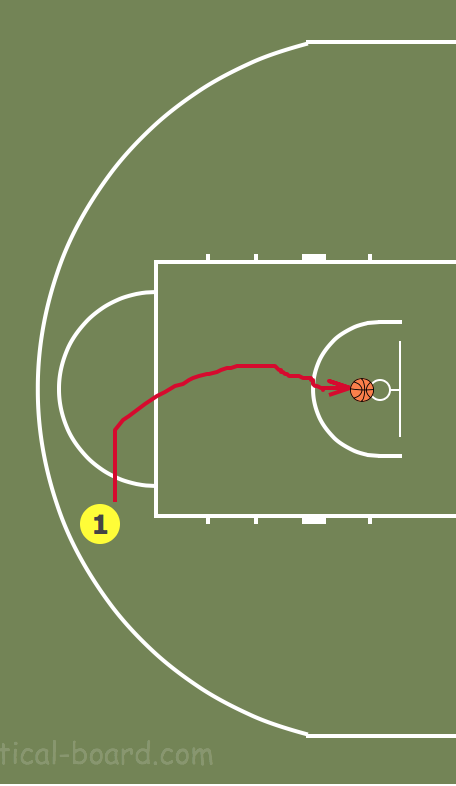
These questions should be dealt with at a more general level than in the Testing and Evaluation chapter. It should be possible for a reader to get a reasonable idea of what you have done from reading the Introduction and Conclusion.

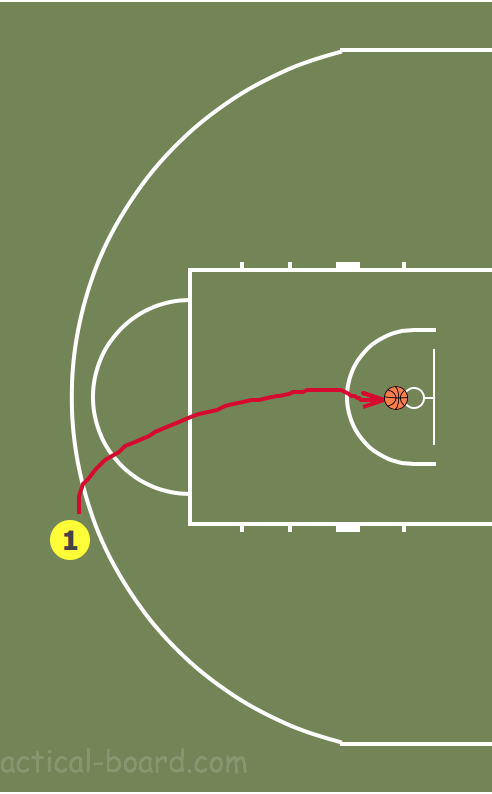
1. Bibliography (All the books, papers, websites that you looked at in order to complete your project.)

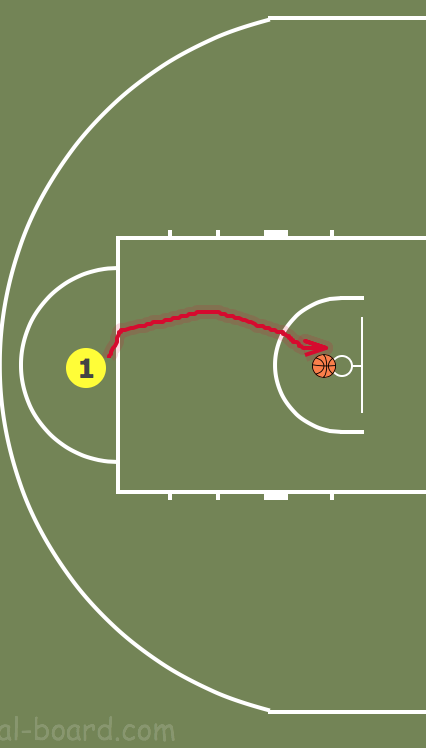
https://www.stats.com

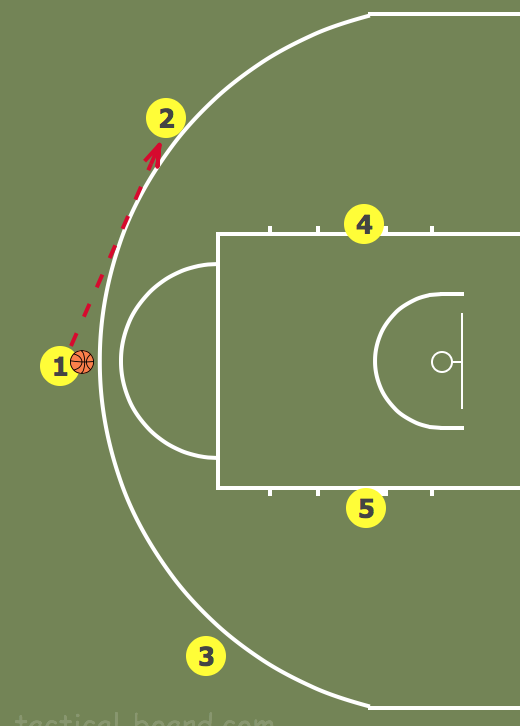
1. Appendices containing:
   1. Program Code: **in the electronic version only**.
   2. Any other relevant material such as database dumps, screenshots, examples of input data, glossaries, graphs
   3. **Final report only (not the draft report):**the electronic copy should include your original proposal, revised proposal if any, PPR and all your weekly logs.

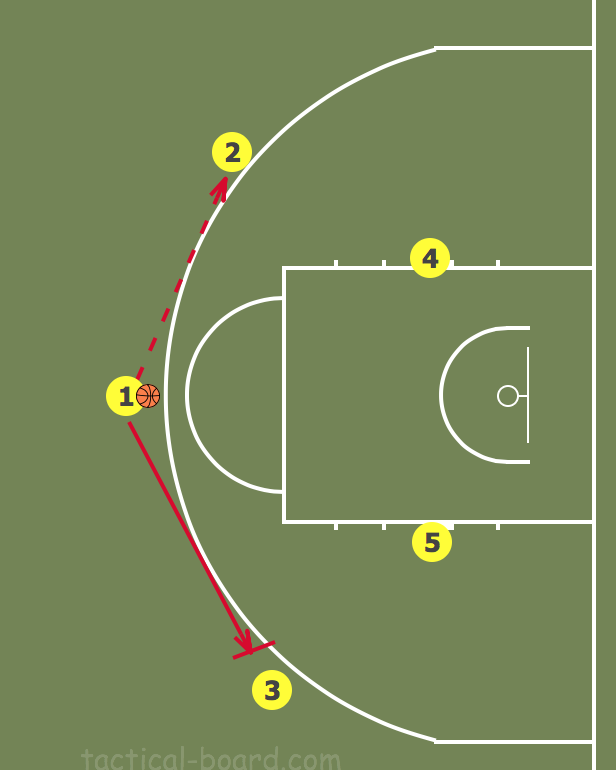


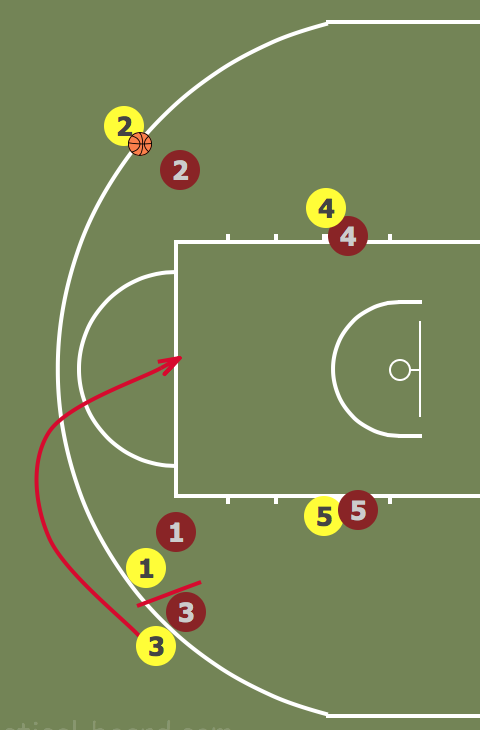


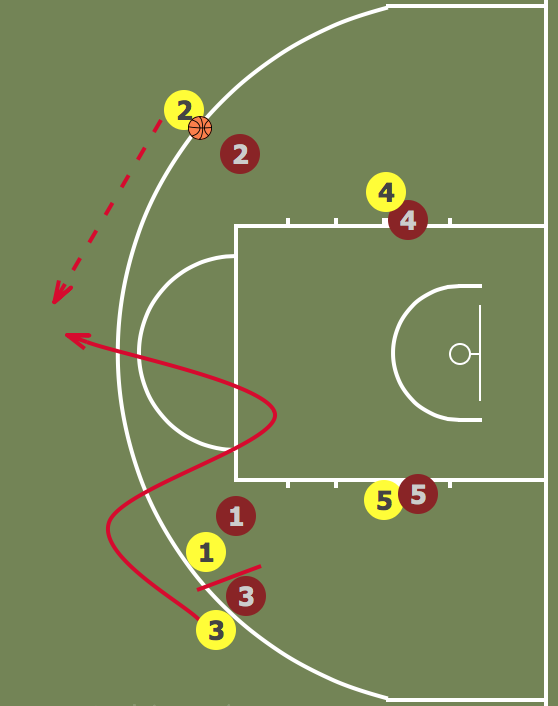




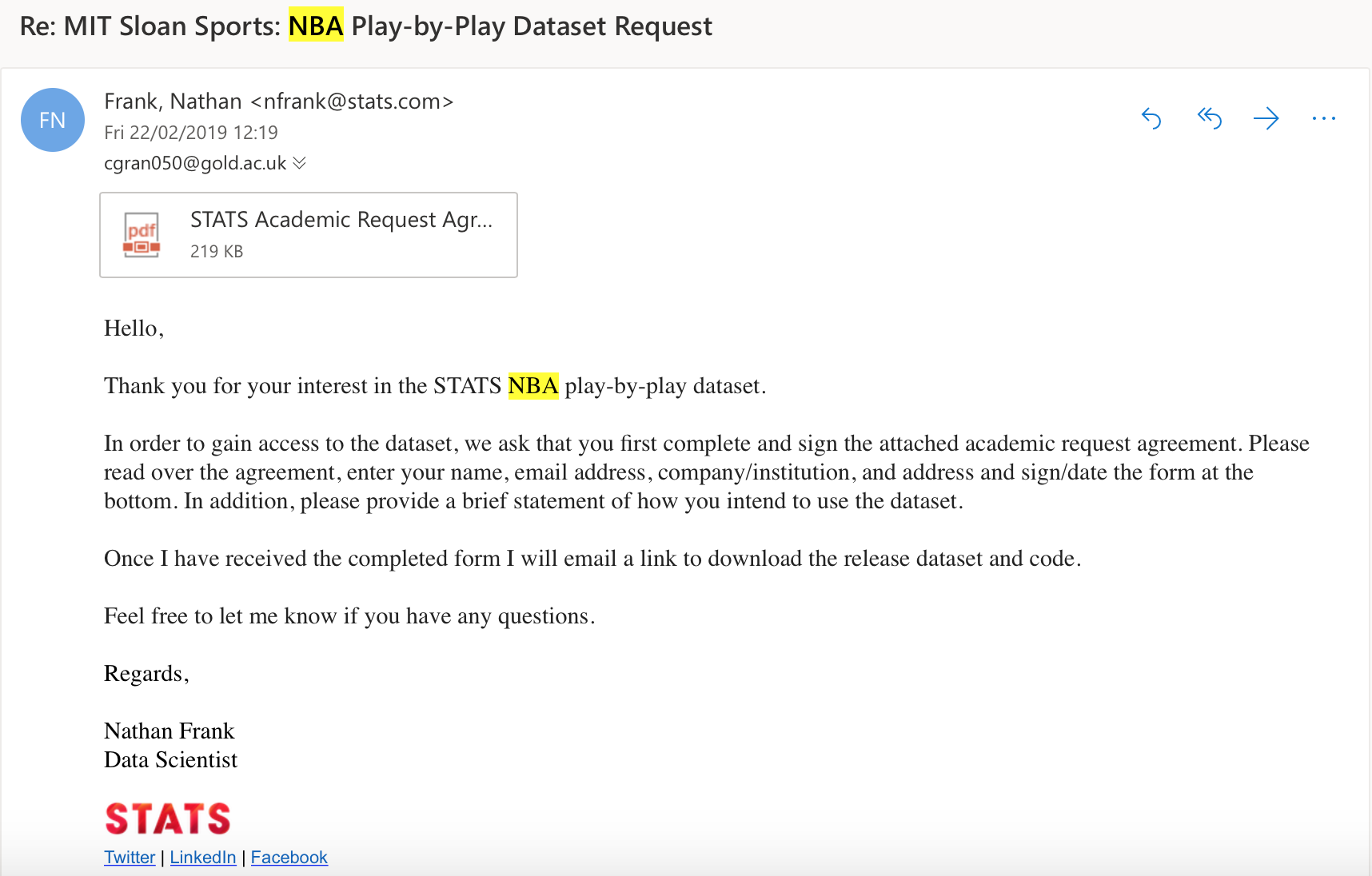


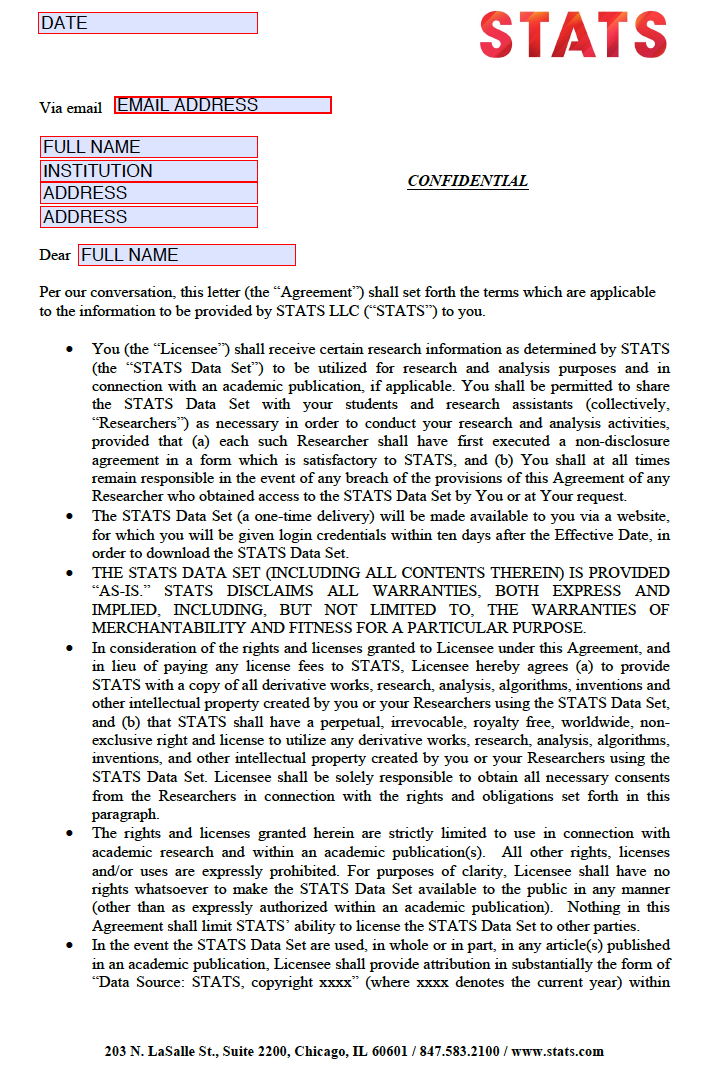


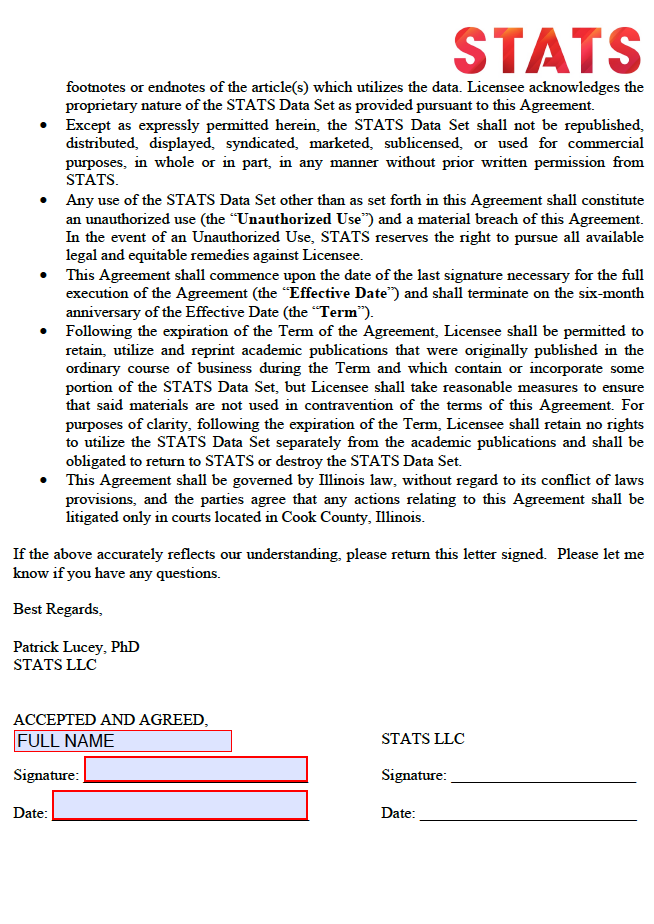




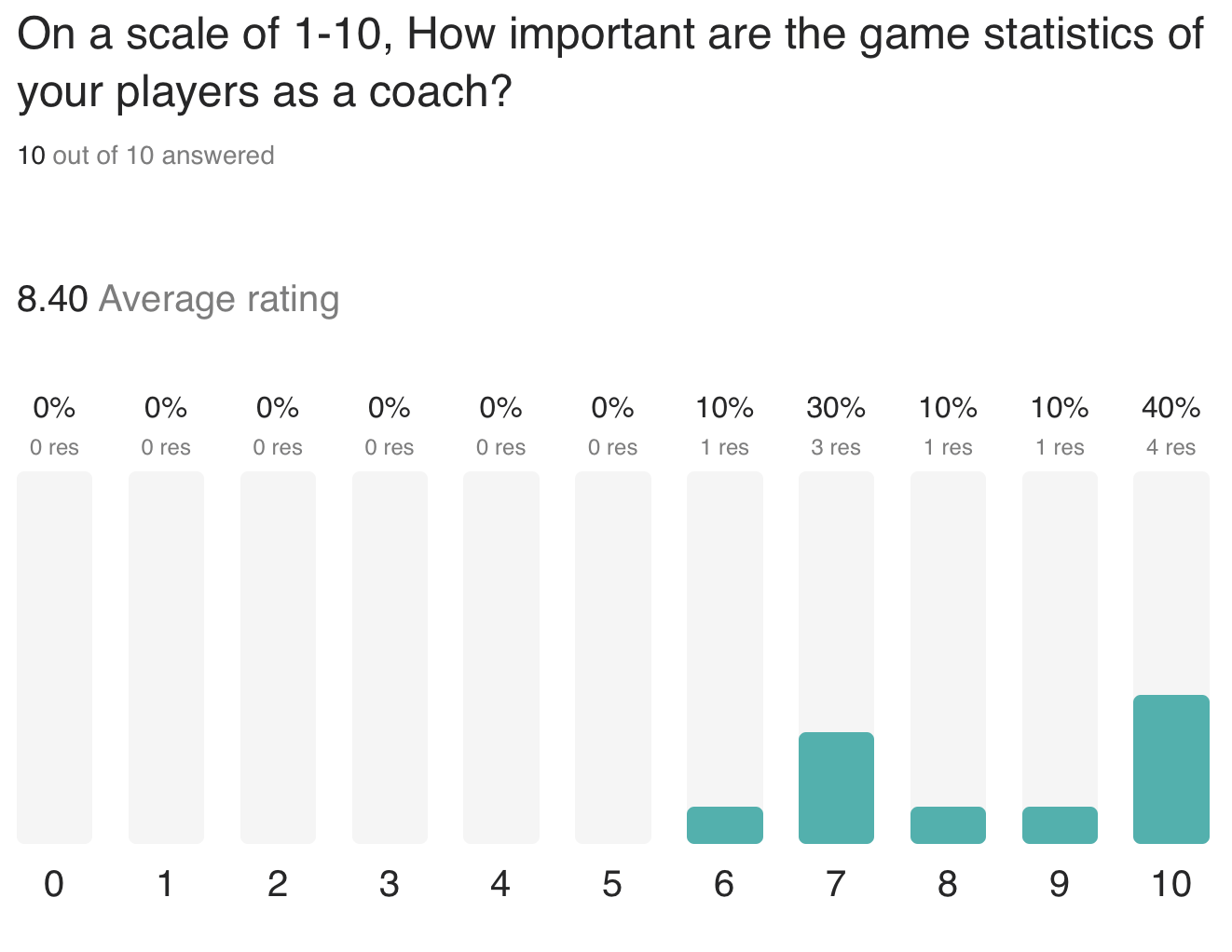
APPENDIX B

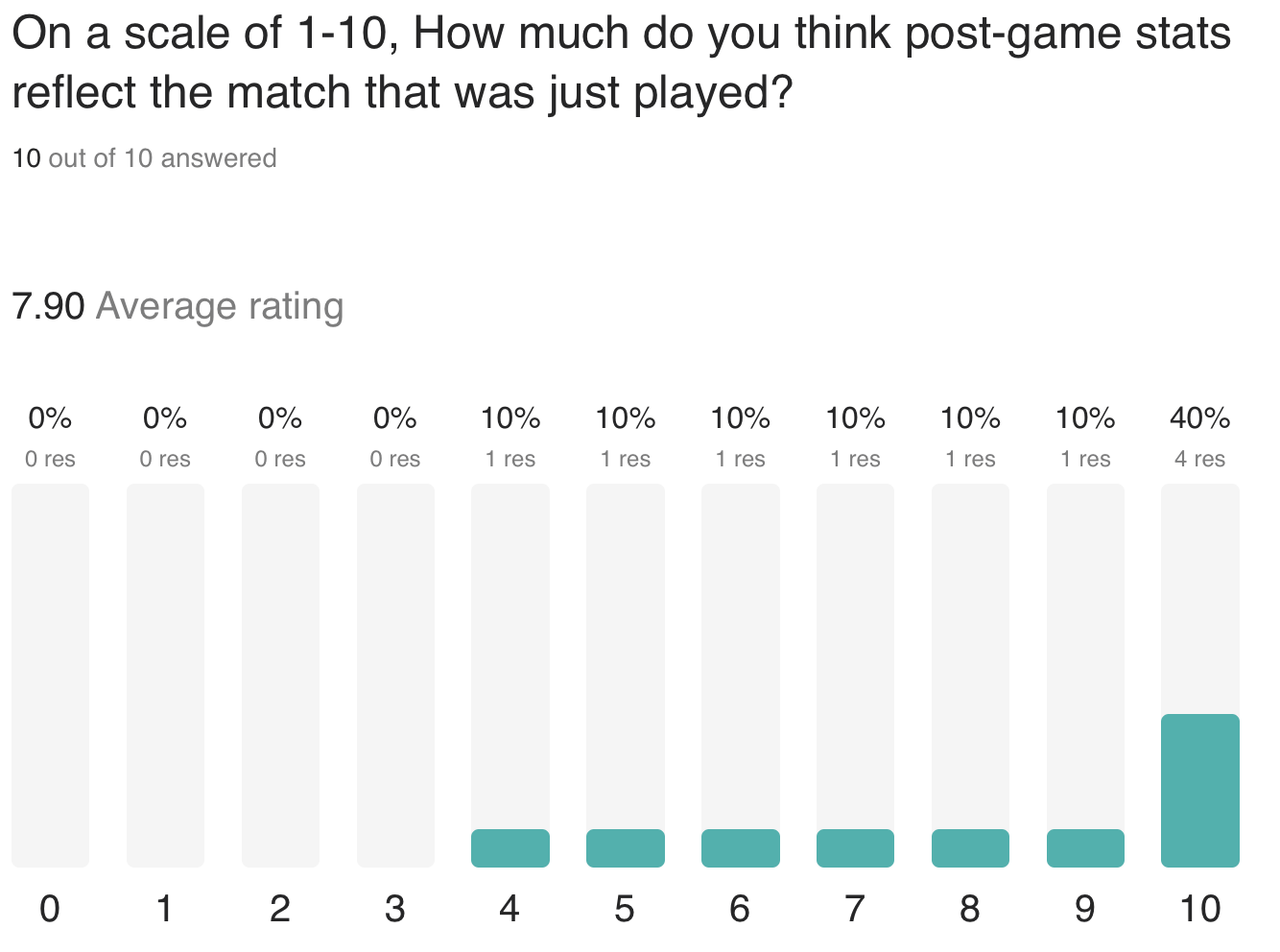


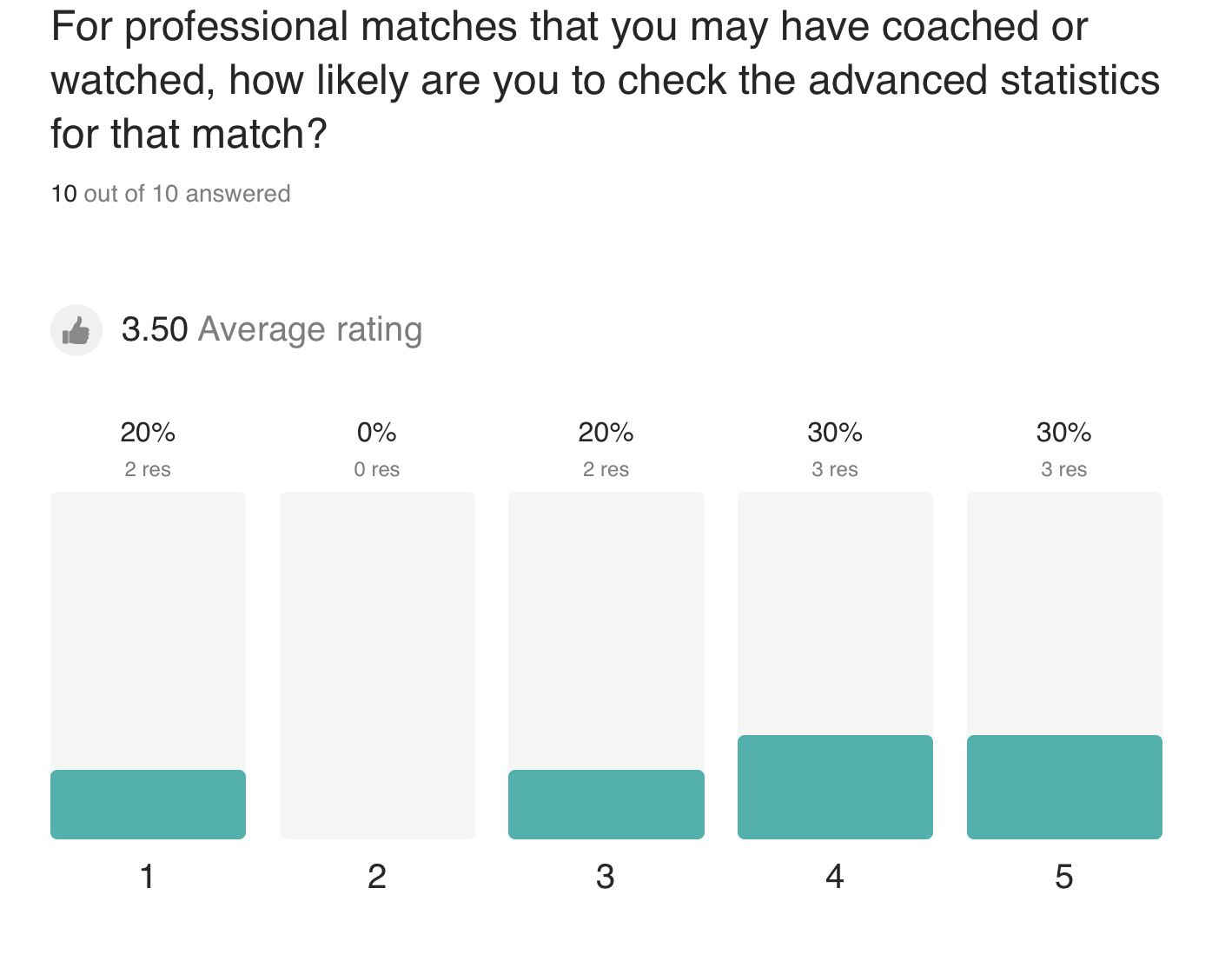


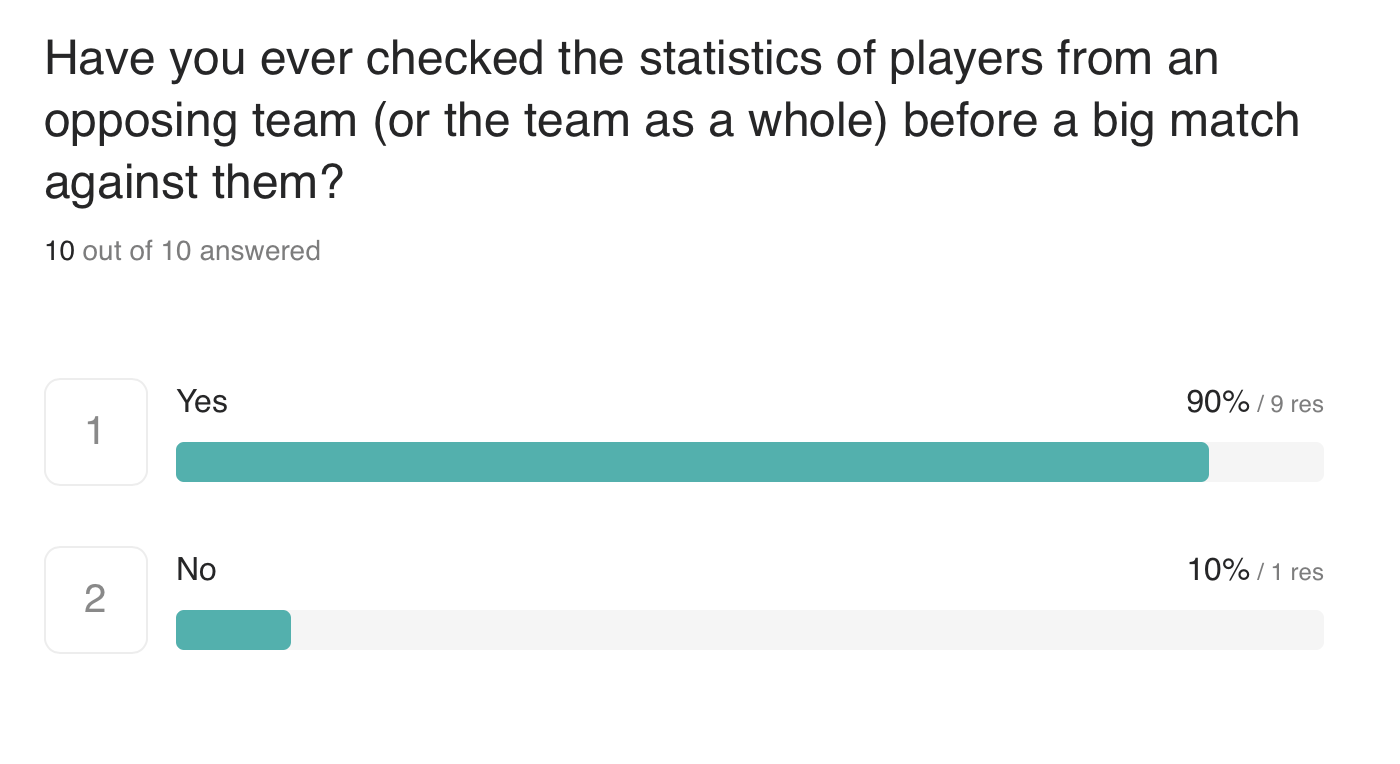


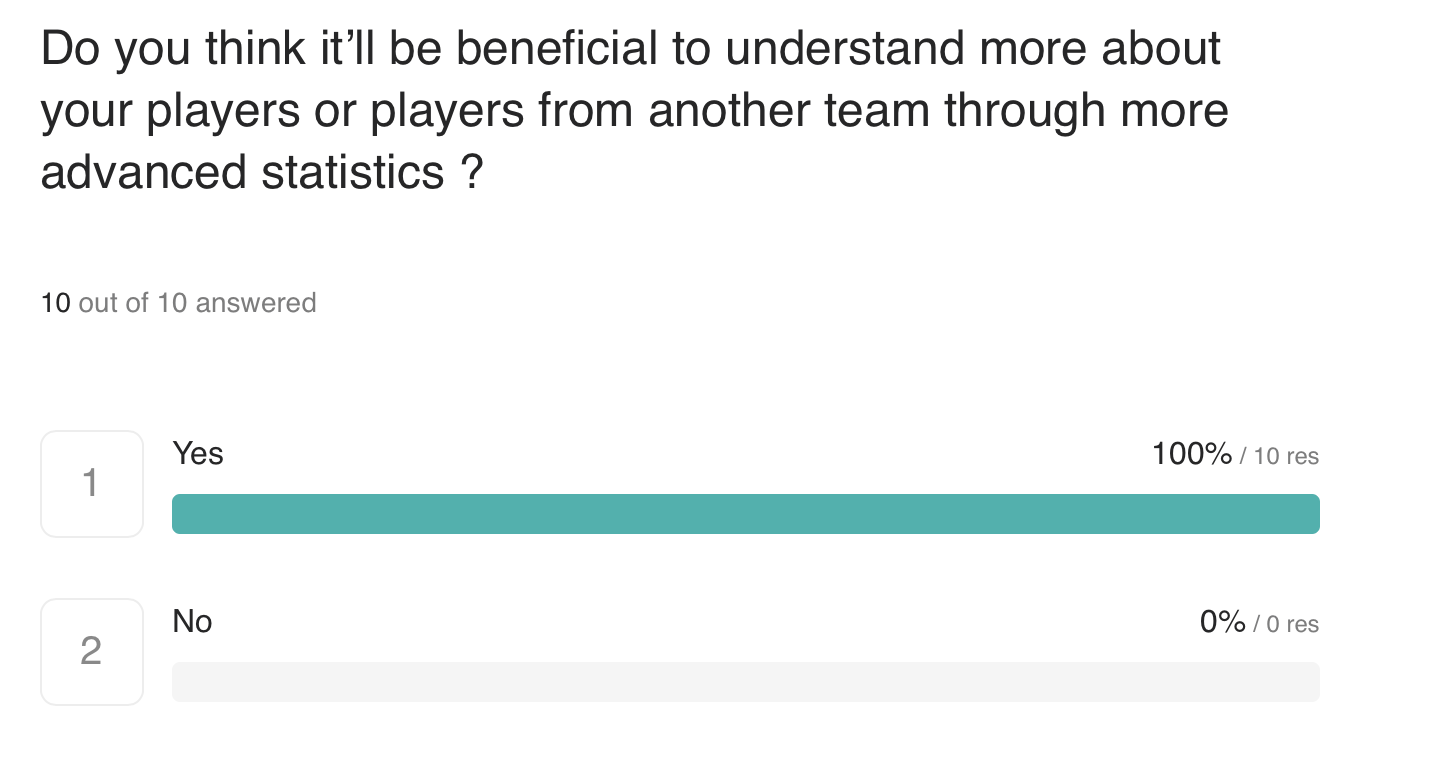
Appendix C



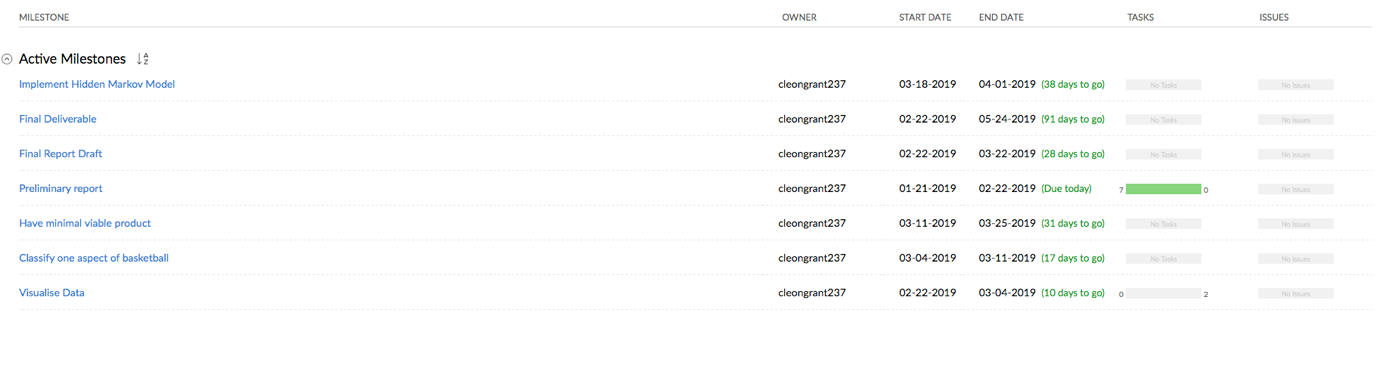


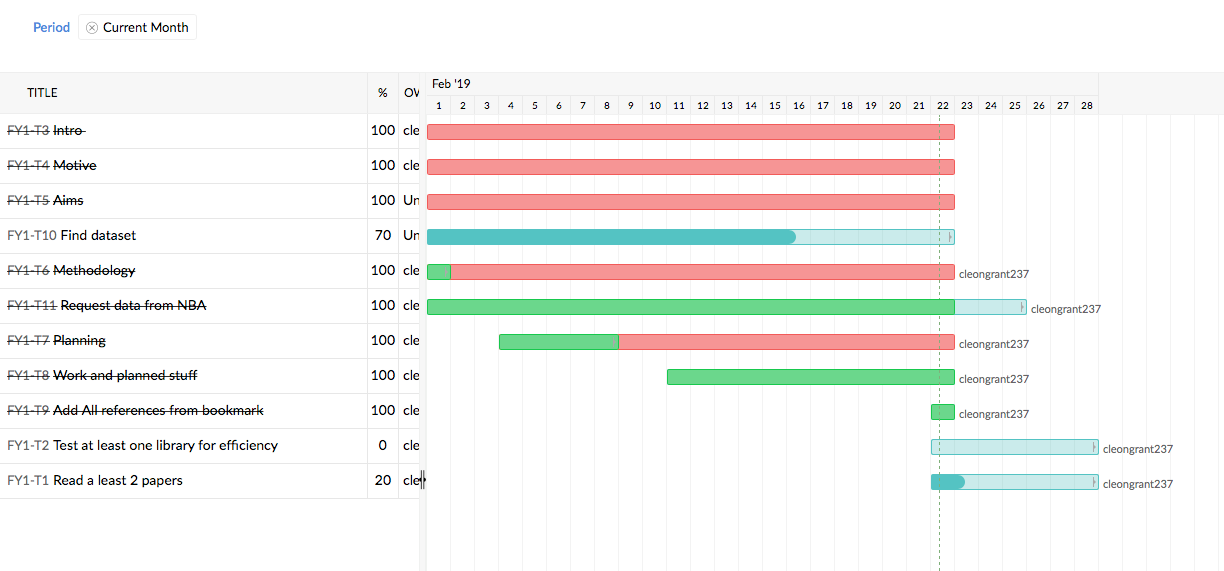


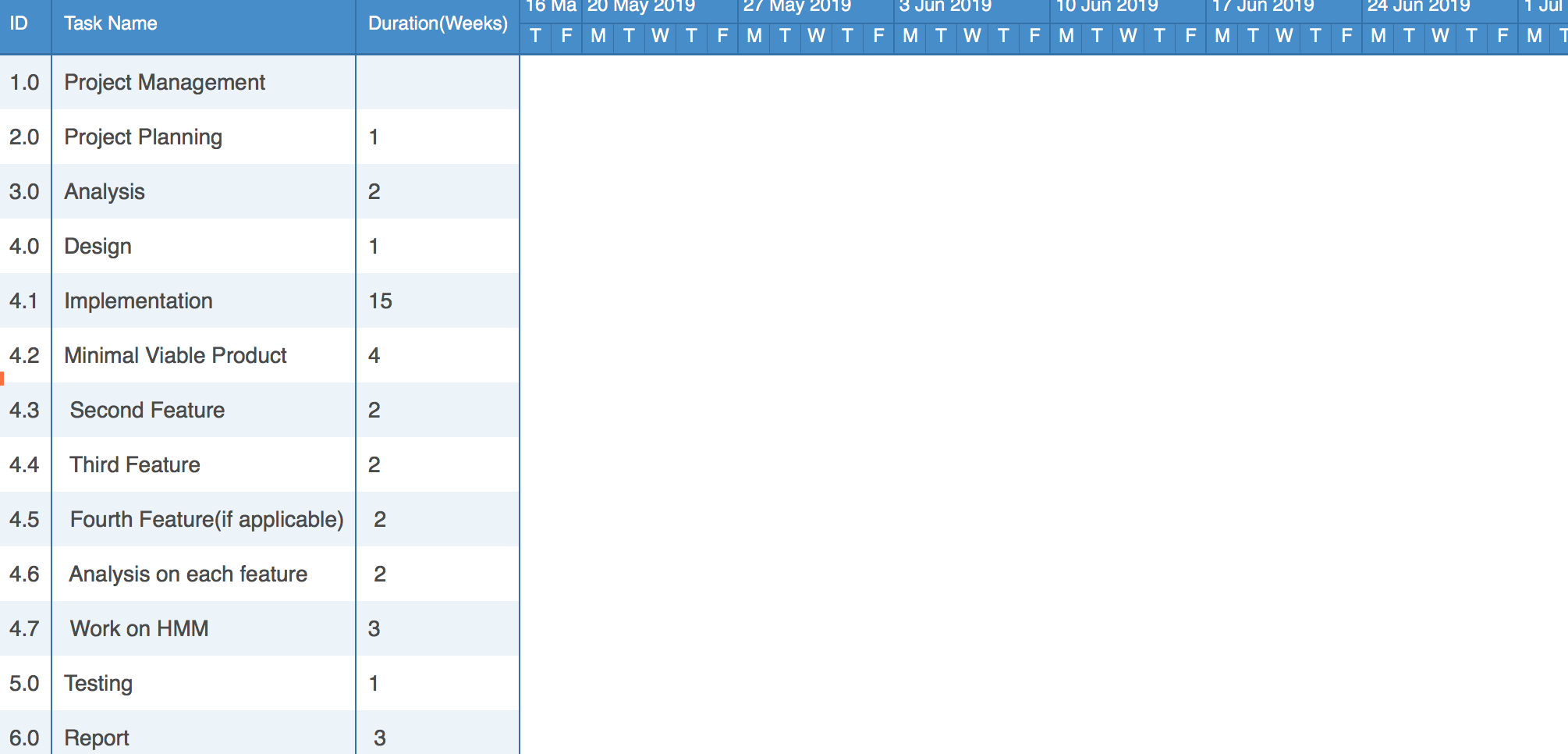




Appendix D







Feb 19

Mar 19

Apr 19

May 19

Jun 19

