Statistical Modelling of American Football Injuries

OÉ Gaillimh
NUI Galway

Author: Trevor Kilgannon

Final Year Project

National University of Ireland Galway

Supervisor: Dr Andrew Simpkin

April 2020

I hereby certify that this material, which I now submit for assessment on the programme of study leading to the award of degree is entirely my own work and had not been taken from the work of others save and to the extent that such work has been cited and acknowledged within the text of my work.

Signed: ID No:

Date:

Acknowledgements

I would like to acknowledge my supervisor Dr Andrew Simpkin for his help, support and guidance throughout the completion of my dissertation.

Contents

1. Introduction

The billion-dollar sports industry [1] has experienced a rapid evolution since the explosion of technology which has given people access to information at the touch of a button. Scientists and coaches can now track a large amount of what their athletes do during preparation, training and recovery. From tracking their training load, to monitoring their mood or mental state as well as their nutrition. Organizations want to ensure that their athletes are in prime physical and mental condition to perform at their best on game day. Small details like these might seem insignificant for individual players however when added together over a full squad it can have a massive impact on a team's results. It is these small gains that can be the difference between success and failure.

Nowadays athletes crave information from medical professionals and coaches. They want their game broken down and analyzed which, will give them the best possible chances of establishing themselves as elite athletes. GPS trackers such as STAT Sports Apex Athlete Device, Kitman Labs Performance and Health Analytics, Orreco's Bio-Marker Analysis and NeuroForce1's Bio-Strap are a great examples of the level of technology now available which can track athletes performance levels in training and in matches. This gives coaches real time data and insights, allowing them to tailor training sessions, monitor in-game performance, assess player load and highlight areas for improvement [2].

One must be aware that while gathering information is easy, interpreting it proves much more difficult. Dr. Fergus Connolly, sport scientist and author of the bestselling book "Game Changer - The Art of Sports Science" [3], has said "The big data concept has many flaws that many miss." By studying groups and not individuals we tend to average out the data for people within that group when in fact the results for individuals may vary significantly [4]. But what can be done is adding colour to the picture by not just giving a single answer but by providing a few possible answers that will hopefully help us make more informed decisions.

For my Final Year Project (FYP) I was keen to combine my passion for statistics, computational methods and sport. After sustaining a serious knee injury in 2019 playing Gaelic Football, I was curious as to how elite sports teams monitor and manage training load to minimize the number of both long and short-term injuries to their athletes, which can be financially costly [5]. Training load can be measured by the amount of energy expended during a workout and is a useful way of comparing shorter rigorous training sessions with longer sessions that are not as demanding [6]. Every sports organization measures training load in different ways using different indicators. and very few, if any, publicly disclose this information.

This study is based on a data sample obtained from a National Collegiate Athletic Association (NCAA) American football team. Every athlete that plays NCAA football in college has a five-year window of eligibility to complete a typical four-year degree so long as they maintain certain academic standards and other requirements set out by the NCAA [7]. Based on this we know that the playing personnel shouldn't change greatly year on year but there should be a steady number of new athletes each year. While final year footballers move on with their careers along with a couple of possible dropouts

It is extremely beneficial for the management and backroom team to get players into the football program in their first year of college so that they can start tracking their training and performance. This would allow them to determine how different sessions and distances covered affects their training load and susceptibility to injury for example. By collecting this data early on it will give the management team a better picture of the athletic makeup of individual players. A mathematical model can then be made specific to the player and the teams needs allowing management to tailor players individual training programs.

This is important as in this study we are not looking for a one size fits all model to suit everyone in the team. The pitfalls of studying groups rather than individuals may result in increased risk of injury due to not accounting for the variation in total distance and training load between players. Additionally, given the high prevalence of concussion in American football, it is even more critical to have specific, accurate data to aid injury prevention. Particularly at college level in which young footballer's brains are still developing. This will pay off in the future for the coaches as it gives them a greater chance of keeping their players fit and on the pitch in their later years in college, when they are fulfilling their potential. Furthermore, a college does not want to be churning out injury plagued footballers in the same way that the NFL has a history of doing in which the average career lasts two-and-a-half years for all positions [8]. Negative press like this might deter new players joining the college.

As mentioned this study is based on a data sample obtained from a National Collegiate Athletic Association (NCAA) American football team which contained GPS tracker training load data were collected over an 89-day period which included training sessions and games. Injury data were also recorded each day. As a first step this data set was reviewed to insure that it was consistent and useable. No corrupt data points were noted. After this the first aim was to examine the training load data in detail by using contrasting methods that can be used to model the trends in training load over time. The second aim of this project is to model the injuries sustained by the players and investigate the relationship between training load and these injuries. With the overarching goal of the project to determine which statistical models best explained the relationships between measured variables and injuries, to improve overall performance through injury prevention.

I will use the R programming language to analyze the data, as it has a large number of packages and libraries and readily usable tests available. I aim to communicate my results in a clear, concise and visual manner so that they may be appreciated by those without a background in statistics. Chapter 2 provides a thorough data summary. An explanation of the statistical approaches used is presented in Chapter 3. Chapter 4 contains the results of modelling the training load variable overs over time are provided. Finally, in Chapter 6 I will summarise my project and explore possible directions for future research.

2. Data Description and Visualization

2.1 Data Description

As previously mentioned, this is a study of a sample of a data set I obtained on an NCAA American football team with a panel of 23 male players. The data were collected over an 89-day observational period. This longitudinal data contains eleven observations of each player which were gathered on different days during the observational period. Using GPS trackers, management were able to collect total distance (TD) in metres, player load (PL), number of injuries, player position and time.

Total distance can also reflect a player's position on the pitch. For example, a wide receivers job involves sprinting to create space and making themselves available for a pass, so they tend to cover more ground than most when playing. In comparison nose guards are defensive linemen whose job it is to maintain a defensive line in the middle of the pitch at the line of scrimmage, as a result they cover far less distance. Measuring and monitoring variables like total distance and player load is a very useful method of injury prevention. It allows management to tailor sessions and acts as a guide for knowing the optimum time to rest individual players based on the data they're seeing.

It would be expected player positions and types of injuries sustained are closely linked. Each position in American football have very different roles and expectations. For example, it might be expected that a field goal kicker might suffer more hip flexor and groin related strains due to their very one-dimensional role which involves coming off the bench a couple times a game to kick goals or take kick-offs. In contrast a quarterback puts far more demand on their upper body making them more susceptible to rotator cuff and bicep injuries. Being the focal point of every offensive play their workload far outweighs that of a field goal kicker so they will have slightly different approaches to warming up. It is important to note however that quarterbacks also have their own periods of inaction when their team is defending so in this sense, just like field goal kickers, they must manage themselves carefully during this time.

2.2 Data Visualization

Data visualization is a very important tool in statistics as it gives us a chance to recognize correlations and patterns in a large array of data points. By using various plots and graphs, we can bring the data to life. Humans are very good at recognizing patterns and so by presenting data visually we give ourselves an opportunity to spot patterns. It is also a good way of establishing any significant outliers or abnormalities within the data. In my analysis of the data I used the ggplot2 library [9] as it can create impactful, elegant and complex plots. I was able to plot the variables to determine any relationships between the measured variables and the response variable of interest. This will help give a sense of what are the most suitable methods for modelling these different relationships.

```
# created 2x2 table with mean and sd TDkm for injured and uninjured players,
presented in easy to read kable table
dat %>% group_by(injury) %>% summarise(mean=mean(TDkm), sd=sd(TDkm)) %>% muta
te_if(is.numeric, round, 2) %>% kable()

# created 2x2 table with mean and sd PL for injured and uninjured players, pr
esented in easy to read kable table
dat %>% group_by(injury) %>% summarise(mean=mean(PL), sd=sd(PL)) %>% mutate_i
f(is.numeric, round, 2) %>% kable()
```

Code Chunk 1: Summary of TD and PL for injured and uninjured players.

injury	mean	sd
No	3.7	1.5
Yes	4.2	1.8

Table 1.1: Summary of TD for injury.

injury	mean	sd
No	425.31	167.54
Yes	490.03	194.99

Table 1.2: Summary of PL for injury.

```
# summary of player positions for number of players ever_injured/injured/unin
jured, presented in easy to read kable table

# n ranges from 1-1039 observations (11 observations per player)

dat %>% group_by(position, ever_injured_cat, injury) %>% summarise(n = n(),
mean_PL=mean(PL), sd_PL=sd(PL), mean_TDkm=mean(TDkm), sd_TDkm=sd(TDkm)) %>% m
utate_if(is.numeric, round, 2) %>% kable()
```

Code Chunk 2: Summary of player positions.

Position	Ever Injured	Injury	n	Mean PL	SD PL	Mean TD	SD TD
Backs, WRs & Secondary	No	No	141	370.30	124.92	3.67km	1.30km
Backs, WRs & Secondary	Yes	No	293	473.75	191.87	4.28km	1.76km
Backs, WRs & Secondary	Yes	Yes	12	552.75	186.31	5.08km	1.81km
Linebacker & Tight End	No	No	43	436.40	140.22	3.99km	1.25km
Linebacker & Tight End	Yes	No	129	449.81	177.06	3.97km	1.44km
Linebacker & Tight End	Yes	Yes	4	322.25	250.92	2.86km	2.35km
Linemen	No	No	48	405.69	134.18	2.91km	0.86km
Linemen	Yes	No	356	399.65	152.48	3.21km	1.23km
Linemen	Yes	Yes	13	483.77	166.72	3.80km	1.28km

Table 2: Summary of Player Positions.

In table 1.1, the ever injured variable represents if a player was injured at any stage during the 89-day period. Injury however, represents if a player was injured at particular time during the 89-days. We can see that the average total distance covered in training and games by players who did not sustain injury during the observational period is ~3.7km, with a standard deviation of ~1.5km. On the other hand, players who incurred injuries during this period tended to cover more ground with an average total distance of ~4.2km and a standard deviation of ~1.8km. While an extra 0.5km may not seem remarkable at first glance, when put into context, it carries far more weight.

American football is very much a “stop-start” game composed of many phases of play. Every one of these phases is fast, explosive and chaotic containing many movements and tackles. No time is wasted as the referee controls the game clock, pausing it after any incomplete pass or any play that goes out of bounds. This results in players being far more involved and switched on for nearly every second of play. Players spend far less time lightly jogging and walking around clocking up meters as is the case in soccer and GAA where the clock is continuously running. Essentially when you see an American football player cover 0.5km more than another player, it is likely that they are exerting themselves significantly more. There is a high probability that this 0.5km comprises high intensity, explosive movements which put their muscles and joints under massive pressure. An example being a typical wide receiver, leaving them more susceptible to injury as a result of fatigue etc.

The variance reflects the other positions in which certain players cover far less ground, so would be less susceptible to leg muscle injuries as a result of sprinting. For example, the quarterback position is more upper-body-dominant and such a player would cover relatively little ground; therefore, total distance would not be an accurate predictor of injury.

Interestingly all the injuries occurred within a 12-day period, between day 38 and day 49, which would have been at the end of the preseason training schedule. This is a correlation that I would like to investigate further as preseason schedules can be quite demanding between intensive training, games and travelling. I suspect that the training load may have been overly burdensome on the young athletes bodies and eventually took its toll at the end of this period, as seen in the accumulation of injuries recorded. It would be interesting to see if these players reinjured at any point after this and if the injury free players remained so during the 89 days. The priority of preseason training should be adequate preparation of players whilst avoiding injuries, so as to provide a good basis for the season ahead.

```
# group trajectories for Load and TD by injury
ggplot(dat, aes(x = time, y = TDkm, fill = injury)) + geom_point(aes(color =
injury)) +
  geom_smooth(aes(color = injury),se = FALSE)

ggplot(dat, aes(x = time, y = PL, fill = injury)) + geom_point(aes(color = in
jury)) +
  geom_smooth(aes(color = injury),se = FALSE)
```

Code Chunk 3: Group trajectories for TD and PL by injury.

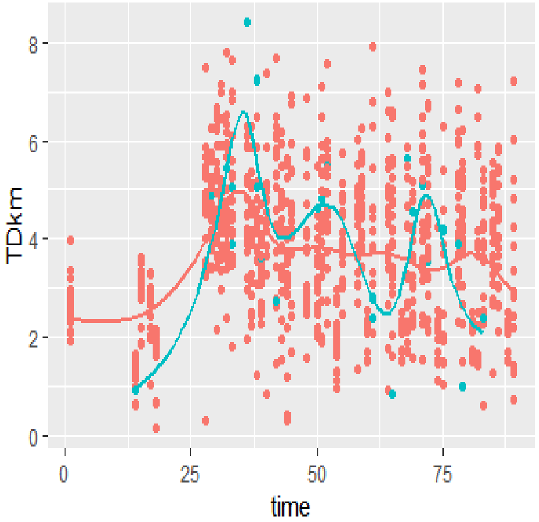


Figure 3.1: Group trajectories for TD.

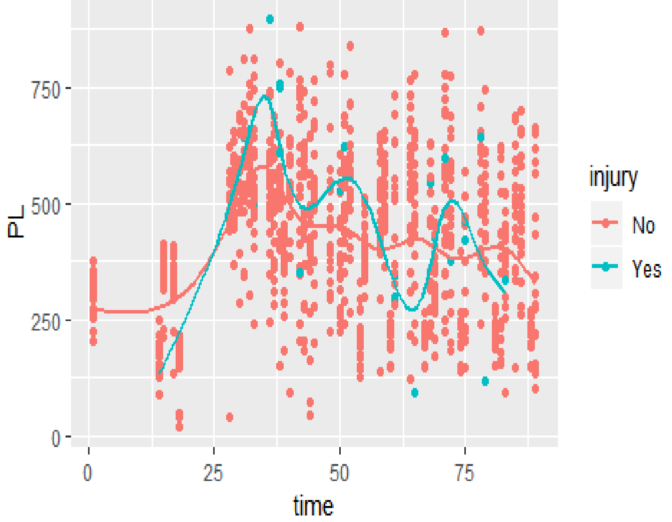
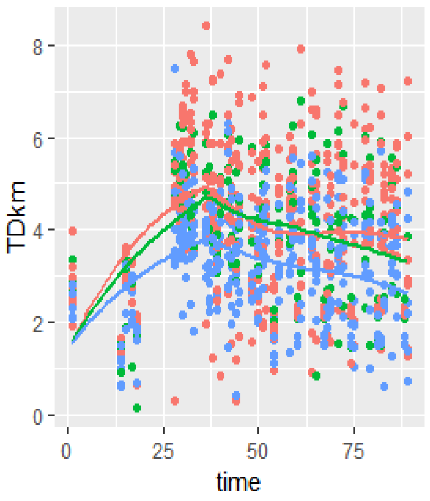


Figure 3.2: Group trajectories for TD.

```
# group trajectories for Load and TD by position
ggplot(dat, aes(x = time, y = TDkm, fill = position)) + geom_point(aes(color
= position)) + geom_smooth(aes(color = position),se = FALSE)

ggplot(dat, aes(x = time, y = PL, fill = position)) + geom_point(aes(color =
position)) +
  geom_smooth(aes(color = position),se = FALSE)
```

Code Chunk 4: Group trajectories for TD and PL by position.



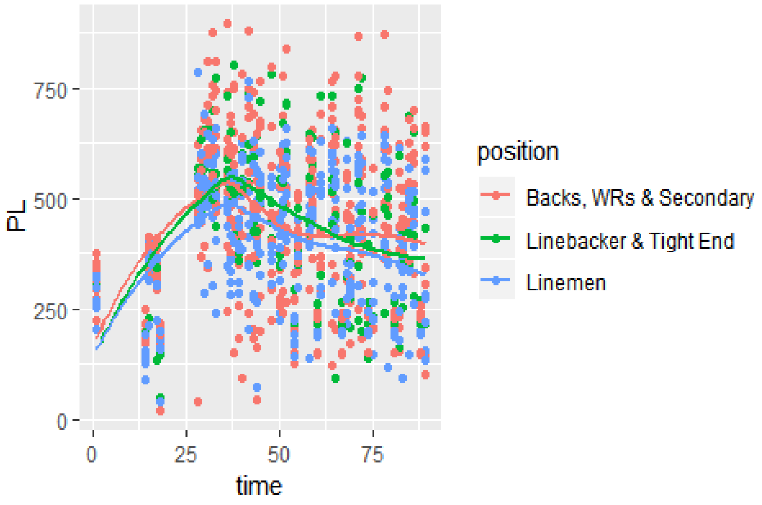


Figure 4.1: Group trajectory for TD.

Figure 4.2: Group trajectory for PL.

In the Figure 4.1 with group trajectories TD against time, we can see that players who suffered an injury were covering slightly less ground in the first ~15 days than the players who did not get injured. However, in the next ~20 days, the total distance covered increased and surpassed that of the uninjured players. Around day 35, the squad covered the most distance with a rather gruelling session, in which the players who sustained an injury covered an average of ~4.6km while the players who did not get injured covered an average of ~4.3km. It was also around this period that there was a peak in the number of injuries sustained possibly as a result of the tough preseason schedule taking a toll. It is unsurprising that the more ground players covered, the more injuries the squad accumulated. From day 35 to day 89, both the injured and uninjured players' trajectories had a very gradual decline, as the average total distance covered tailed off and settled at ~3.5km for those who picked up injuries. While it gradually decreased and settled at ~3.1km for players who remained injury-free.

In Figure 4.2 with group trajectories PL against time we can see a very similar trend to that of the ggplot 4.1 with TD against time. In much the same way as TD, PL is slightly less in the first ~15 days for players who experienced injury. But again, over the next ~20 days players who suffered injuries were under increased PL in comparison to the players who remained injury free. Around day 35, there was peak in the number of injuries sustained, which coincided with a peak in PL. Again, from day 35 to day 89, the PL trajectories for players who incurred injuries as well as those who stayed injury-free decreased gradually.

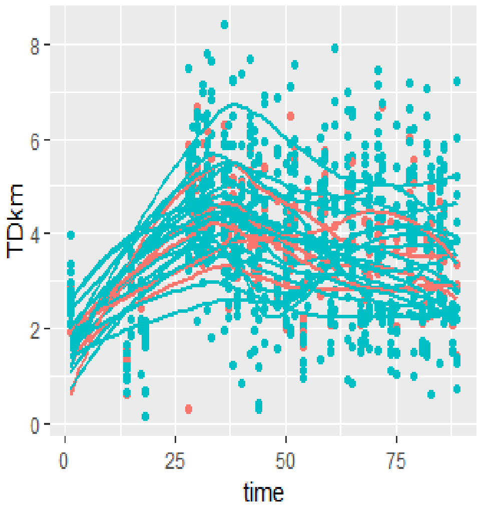
As it is sensitive information, I am not privy as to what units PL is measured in and what it specifically represents for this team. I can, however, conclude that there is a strong correlation between PL and TD from analysing the ggplots. There appears to be a direct relationship between PL and players' physical exertion.

The 89-day observational period is taken from the beginning of July to end of September – so the squad were in training camp for days 1-50 and in season from day-50 to day-89. We know preseason can be a taxing time for players' bodies after a long off season, so it is possible that the periodisation concept is used here in the same way many teams utilize it in order to better manage their players workload. Periodisation is the structured planning of athletic or physical training [10]. The goal of periodisation in sport is to plan a training program so as to reach optimum performance for the most important competition or period of the year. By implementing this, it should also have helped reduce the likelihood of injuries.

```r
# individual trajectories for Load and TD by injury
ggplot(dat, aes(x = time, y = TDkm, group = id, color = ever_injured_cat)) +
geom_point() + geom_smooth(se = FALSE)

ggplot(dat, aes(x = time, y = PL, group = id, color = ever_injured_cat)) + ge
om_point() + geom_smooth(se = FALSE)
```

Code Chunk 5: Individual trajectories for TD and PL by injury.



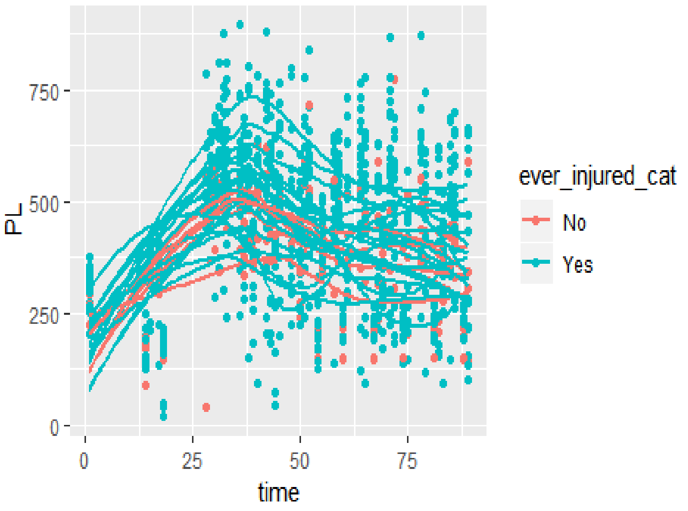


Figure 5.1: Individual trajectories for TD. Figure 5.2: Individual trajectories for PL.

In Figure 5.1 with individual trajectories for TD over time we can see that the greatest distance covered by any player during a session was ~6.8km by Player 18, who also suffered an injury on day 38, we can see that this player is a bit of an outlier in comparison to the rest of the squad. The greatest distance covered by an uninjured player was ~5.75km between day 30 and day 35, by managing their training load well they managed to stay injury free. Towards the bottom of the plot, we can see an injured player with a low TD which gradually increases over time, insinuating to me that they are rehabbing from an injury and are gradually incorporating themselves back into training. Aside from this, the individual trajectories for players are well-bunched and mixed around the middle of the plot.

We will need to delve deeper to figure out potential causes for these players getting injured e.g. certain positions they play in. One interesting player trajectory can be seen around the middle. Their TD peak (~4.0km) occurs much earlier in the observational period compared to the rest of the squad. However, on further inspection, we can see they have a sharp drop in TD which reaches a plateau around day 46. This would suggest to me that they may have picked up a minor injury that required a reduced participation in training and some rest. From day 46 to day 89, their TD gradually increases as they are eased back into training and their TD appears to level off at ~4.0km. This gives me a sense that this player is playing a position with a specific role.

In Figure 5.2 with individual trajectories for PL over time we can see that again PL is a strong reflection of TD. We can see many of the same trajectories with Player 18 who peaked for both TD and PL around day 38. We see a similar trajectory for PL relating to that player that we suspect is coming back from injury near the bottom of the plot.

```
# boxplot of Load and TD by injury
ggplot(data = dat, aes(x = injury, y = TDkm, fill = injury)) + geom_boxplot()

ggplot(data = dat, aes(x = injury, y = PL, fill = injury)) +
  geom_boxplot()
```

Code Chunk 6: Boxplot of TD and PL by injury.

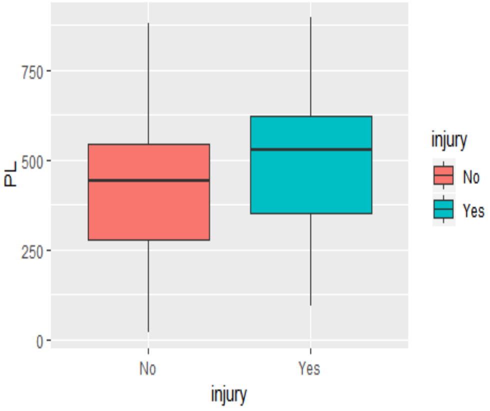


Figure 6.1: Boxplot of TD by injury.

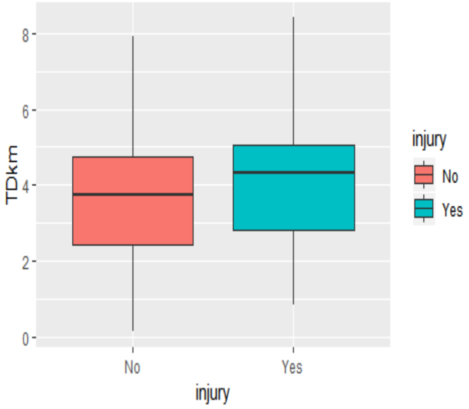


Figure 6.2: Boxplot of PL by injury.

```
# boxplot of Load and TD by position
ggplot(data = dat, aes(x = position, y = TDkm, fill = position)) +
  geom_boxplot()
```

```
ggplot(data = dat, aes(x = position, y = PL, fill = position)) +
  geom_boxplot()
```

Code Chunk 7: Boxplot of TD and PL by position.

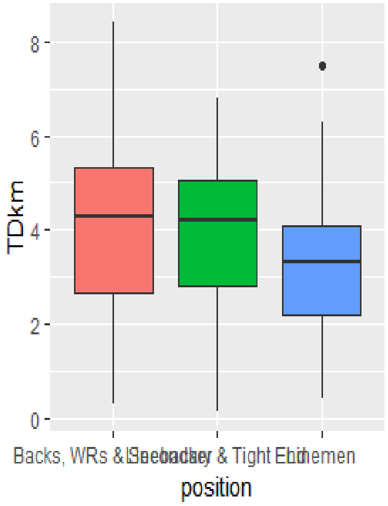


Figure 7.1: Boxplot of TD by position.

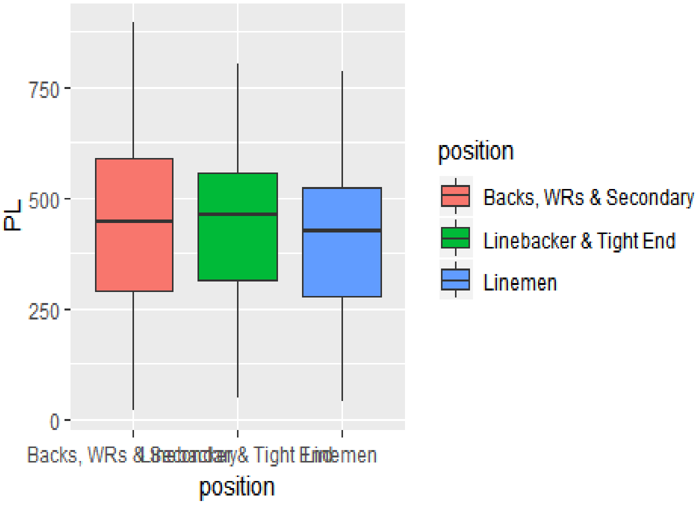


Figure 7.2: Boxplot of PL by position.

In Figure 6.1 with TD against injury we can see that for uninjured players the median is ~3.8km while the interquartile range ranges from ~2.4km to ~4.8km. The first quartile is ~3.0km while the third quartile is ~7.5km. This box plot is normally distributed with a slight skew to the right. We can see that for injured players the median is ~4.0km while the interquartile range ranges from ~2.4km to ~4.9km. The first quartile is ~2.0km while the third quartile is ~8.4km. This box plot is also a little skewed to the right. In Figure 6.2 with PL against injury we can see that again the box plot for uninjured players is normally distributed with a slight skew to the right, once again reflecting the link between TD and PL. For the injured players the median is ~480 while the interquartile range spans from ~290 to ~560, reflecting the fact that many of the players who pick up injuries are on average clocking more metres. The first quartile is ~45 while the third quartile is ~850. Just like the box plot for injured players TD this box plot is a little right skewed also meaning it is positively skewed. As all the box plots are symmetric or positively skewed, the mean is ≠ median. This indicates that the data account for higher frequency of high-valued scores, which in this analysis tells us that there is a strong link between TD and injury.

3. Statistical Methods

3.1 Statistical Modelling

Statistical modelling is a mathematical method that aims to explain the relationship between variables and gives you the option of making predictions based on your results. We will obtain the outcome we are keen to model from the response variable (often on the

Y-axis), by assessing its relationship with the explanatory variables (often on the X-axis). The response variable, which is also known as the dependent variable, is what we want to describe, explain and predict using the explanatory variables. The explanatory variables, which are also known as the independent variables, are what we use to describe, explain and predict the response variable. Model parameters are involved in the mathematical equations that link the dependent variables to the explanatory variables. In order to build a statistical model characteristic analyses of the goodness of fit must be carried out. This will prevent any issues from arising as a result of not adhering to model assumptions. I will discuss the statistical methods that are related to the models in this project [11]. To model training load over time for a group of athletes we will use methods of longitudinal data analysis, in particular linear mixed models with spline and polynomial terms. These will allow us to measure the response variable of interest (injuries) repeatedly through time for multiple subjects (players). For modelling injury, we will use logistic regression with random effects for different players. This will allow us to estimate the probability of the binary outcome (injury) based on the values of the explanatory variables.

3.2 Linear regression

Linear regression is a basic and widely used predictive analysis which mainly operates on continuous data. There are two types:

Simple Linear Regression

Simple linear regression is used to model the relationship between a response variable and only one independent variable. The aim is to model the predicted value of a continuous variable:

$$E(Y_i) = \beta_0 + \beta_1 X_i$$

We assume every Y_i is normally distributed $Y_i \sim \text{Normal}(0, \sigma^2)$ as well as the errors being independent and normally distributed $\epsilon_i \sim \text{Normal}(0, \sigma^2)$. A model for a sample with n observed responses is:

$$Y_i = \beta_0 + \beta_1 X_i$$

with Y_i as the response variable, X_i as the explanatory variable and ϵ_i as the error term. The error term ϵ_i shows all other elements which affect the response variable aside from the independent variables. To estimate the values of the parameters in the equation, β_0 and β_1, the method of least squares is used as it will provide the "best fit" for the data points to some extent. The least squares method calculates the differences between the actual and estimated values of the response variable Y.

$$Q = \sum_{i=1}^{n} (Y_i - \beta_0 - \beta_1 X_i)^2$$

Deviation as a result is reduced by this model.

Multiple Linear Regression

Multiple linear regression models are simple linear regression models that have been expanded to enable more than just one independent variable.

$$Y_i = \beta_0 + \beta_1 X_{i1} + \beta_2 X_{i2} +..+ \beta_1 X_{ip} + \epsilon$$

Linear Regression Assumptions

The relationship between the parameters, response variable and the predictor variable must be linear. The error vector ϵ must have zero mean E=0. Homoscedasicity is a case in which every response variable has equal variance in their errors, no matter what the values of the predictor variables are. When this is not the case, we refer to the situation as heteroscedasticity and we can take it that the errors are uncorrelated $Cov(\epsilon_i, \epsilon_j), i \neq j$. All observations must be independent of each other. When there is little to no independence in the residual values, autocorrelation tends to occur. Y is distributed normally, given any fixed value of X. Autocorrelation can tell us about the variable of interest and possible problems associated with the model of choice. When it is present it can establish if correlation is present between the different values of variables which are based on related aspects.

Model Fit

It is important to establish how well a model fits the data once it is fitted to it. This can be done with the R^2 value which measures goodness of fit. The greater the R^2 value the better. R^2 is explained sum of squares over total sum of squares:

$$R^2 = \sum_{i=1}^{n} (\widehat{Y_i} - \overline{Y_i})^2$$

where 0<R^2<1. In simple linear regression, a t-test is carried out when testing for a strong relationship between the response variable and the predictor variable.

$H_0: \beta_1 = 0, H_a: \beta_1 \neq 0$

$$t_* = \frac{\hat{\beta}_1}{se(\hat{\beta}_1)}$$

reject H_0 when

$$t_* > t_(n-2),1-\frac{\alpha}{2}$$

Likewise, in multiple linear regression separate t-tests can be carried out for every β_i. But an F-test must also be carried out in order to test all the model's credibility.

$$H_0: \beta_1 = \beta_2 =\ldots= \beta_p = 0$$

$$H_a: \beta_i \neq 0$$

Test statistic: $F_* = \frac{MSR}{MSE}$ reject H_0 when $F_* > F_{p,n-p-1}$ To check the credibility of the model we must determine if the assumptions carry. We do this in case error assumptions don't carry, there is a non-linear relationship between X and Y or there are outliers without some predictor variables.

3.3 Polynomial Regression

When the variables in the data set are correlated and their relationship is linear, linear regression works well. However, when the variables are correlated but their relationship does not appear to be linear, a more suitable approach is polynomial regression.

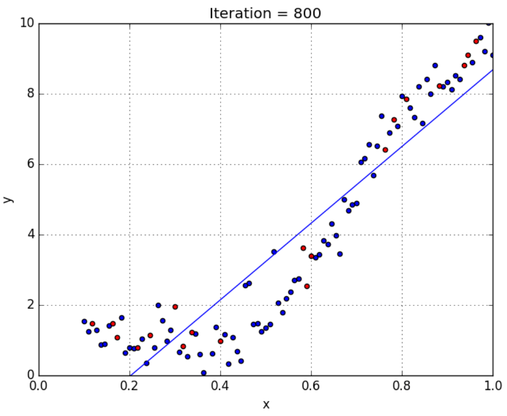


Figure 8.1: Linear regression

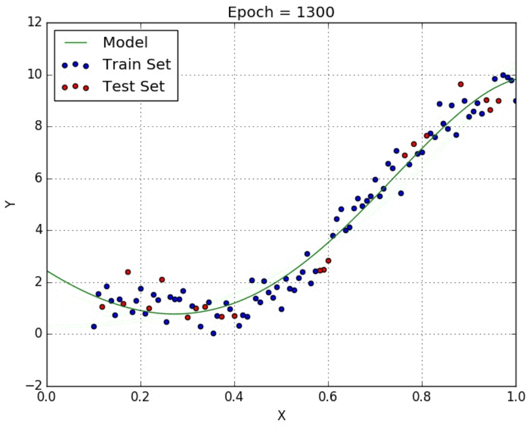


Figure 8.2: Polynomial Regression

As we can see in Figure 8.1, when we use simple linear regression, $y = ax + b$, on the first graph it doesn't fit very well, this an example of under-fitting. Therefore, we increase the complexity of the model in Figure 8.2 by using polynomial regression, $y = ax^3 + bx^3 + cx + d$, on the second graph which fits much better and outlines the points with greater precision. In order to stop over-fitting, which prevents the algorithm from learning the noise in the system and becoming more generalized, more training samples are added [12]. By fitting a polynomial line here, we can achieve a minimum error. Advantages of polynomial regression are it gives the optimum approximation of the relationship that exists between the independent and dependent variable. Polynomial regression can fit a various number of curvatures. However, a disadvantage is that they are easily affected by outliers [13].

3.4 Non-Linear Regression

Transformation

I previously discussed the various assumptions in linear regression and in order to satisfy these, transformations can be used. Transformations to variables or the data are often needed to increase non-linear models' strength and flexibility to gain an understanding of non-linear interactions involving the X_i (input) and Y_i (output) variables. Either of the response or explanatory variables can be transformed in regression, often even both. To achieve homoscedasticity and normality the y variable can be transformed. If transforming the y variable alone isn't enough, then transforming the predictor variable will give linearity. Example non-linear regression model

$$y = ae^{bx}U$$

containing parameters a and b along with multiplicative error term U. By taking the log of both sides we get

$$ln(y) = ln(a) + bx + u$$

with $u = ln(u)$. Popular transformations include the square root, log and reciprocal. Power transformations, such as the Box-Cox, can also be used. Box-Cox transformation is a method that facilitates the transformation of non-linear dependent variables into a normal configuration. It does not, however, assure normality as it does not necessarily check for it. Instead, it checks for the smallest standard deviation. We can add a constant c to the data if the data are not positive, otherwise the Box-Cox transformation will not work [14].

$$y(\lambda) = \sum_{logy}^{\frac{y^y-1}{\lambda}} {}_i f(\lambda_1 = 0)^i f(\lambda_1 \neq 0)$$

$$y(\lambda) = \sum_{log(y+\lambda_2)}^{\frac{(y+\lambda_2)^{\lambda_1}-1}{\lambda}} {}_i f(\lambda_1 = 0)^i f(\lambda_1 \neq 0)$$

Splines

Spline regression is a format for checking non-linearity within predictor variables and representing non-linear functions and interactions between variables. It is an extension of linear models and is a non-parametric regression method. It divides data sets into groups at different intervals. Each of which has its own separate fit, of the time covariate X, by using points or knots to separate the range of X. We can fit regression splines by ordinary least squares when we have the knots selected. However, knot selection involves intricate algorithms which are computationally intensive and time-consuming, such as the model selection criterion Mallows C_p. Hence, we could use penalized regression splines instead and so long as we have plenty of knots, then the location of knots should have little to no effect as the coefficients are shrunk. Also, by reducing C_p, for example, which is just moderately computationally intensive, enables us to select the smoothing parameter which should result in a less variable fit [15].

3.5 Generalized Linear Models

Generalized linear model (GLM) to common linear regression models, such as multiple/simple linear regression and log-linear models, for continuous response variables as well as continuous categorical predictors. GLM's are made up of the following three elements:

Random Component – refers to the exponential family distribution of y_i, the response variable. ϕ is a constant scale parameter in the equation;

$$f(y_i) = exp\frac{\phi}{\wedge}y_i\theta_i - b(\theta_i) + c(y_i, \phi)$$

.

$$\frac{\phi^{a+b}}{2}$$

Systematic Component - the linear predictor η which is produced by the explanatory variable covariates $(X_1, X_2, \ldots X_k)$.

Link Function - relates to the link between the random and systematic components $\eta_i = g(\mu_i)$. Once the link function transforms the response and predictor variables, the generalized linear model expects that a linear relationship exists between them. However, it doesn't assume a linear relationship exists between the two before this. The model also does not need the response variable to be normally distributed.

Assumptions

The data are distributed independently e.g. $Y_1, Y_2, \ldots, Y_n$. Y_i usually follows a Poisson, binomial, normal distribution to name but a few. There is no linear relationship between the dependent and independent variables expected by the generalized linear models. Due to model structure and in some cases overdispersion, variance homogeneity is not always attainable however, this is not necessary for. It is crucial that errors are not normally distributed but that they are independent. Maximum likelihood estimation is preferred to ordinary least squares when estimating parameters. Therefore, big sample approximations are needed. Goodness of fit methods need big samples [16].

Overdispersion

Overdispersion explains the observation that variation is greater than would have been imagined. Overdispersion usually occurs as a result of clumping. If overdispersion is not taken into consideration in the model, it may produce incorrect inferences. Quasi-Poisson models are used to deal with overdispersion when it infringes the model assumption in which $E(Y_i) = \mu_i$ and $Var(Y_i) = \mu_i$. Overdispersion can also occur if crucial predictors are absent or if functionally is specified as linear instead of non-linear, for example. Excessive variance in the data, $Var(Y_i) = \phi\mu_i$, is rectified by the model as it inserts an extra dispersion parameter. As opposed to assuming the dispersion parameter ϕ to be one, the model now enables us to estimate it, allowing us to gauge how much greater the variance is compared to the mean. If the value is more than one, then this is known as overdispersion and if it is less than one, then it is known as under dispersion.

Goodness of fit

Deviance:

$$D = \sum_{i=1}^{n} (y_i ln \frac{y_i}{\hat{\mu}_i} - (y_i - \hat{\mu}_i))$$

Pearson Statistic:

$$P = \sum_{i=1}^{n} \frac{(y_i - \hat{\mu}_i)^2}{\hat{\mu}_i}$$

These two statistics can tell us a great deal about the effectiveness of the model. The data are fitted precisely here as the saturated model has a parameter for each observation. As such, the deviance is double the log-likelihood ratio of the complete model in comparison to the reduced model. If the model doesn't fit correctly then this may be as the deviance is $>> X_{n-p-1}$. Additional predictors or a different model altogether may be required in this case [17].

3.6 Linear mixed models

For n players, we can extend regression using random effects, also known as a linear mixed model (LMM). LMMs may be thought of as an add-on of simple linear models as they take into consideration variation which is explained by the independent variables (fixed effects) and variation which is not explained by the independent variables (random effects). As the model contains a mix of both random and fixed effects, it is referred to as a mixed model. It is the random effects which essentially make-up the error term ϵ. LMMs are commonly used if there is non-independence in the data, like which can appear in hierarchical form. LMMs are a mix of both aggregate and individual data analysis. Using individual regression on each sample results in numerous estimates and a great deal of data. However, it is noisy. On the other hand, the aggregate approach on hierarchical data is less noisy but tends to miss important differences as it averages the samples. Linear mixed models are something of a hybrid version of the two and will be useful when analysing the whole squad of players in my data set.

Random Effects

Fixed and random effects are key parts of mixed models. Parameters that do not vary are known as fixed effects. We could assume that in a population there is a true regression line β which we can obtain an estimate of $\hat{\beta}$. In comparison, parameters which are themselves random variables, are known as random effects. In the following equation: $\beta\ N(\mu, \sigma)$ β is distributed as a random variable where mean is μ and standard deviation is σ. Here we can believe the data to be random variables and the fixed effects to be the parameters. So, on one level both the data and parameters are now random variables however at the largest level they are fixed [18].

$$y = X\beta + Zu + \epsilon$$

with y as a $Nx1$ column vector that has mean $E(y) = X\beta$. X is a Nxp design matrix belonging to the p predictor variables. β is a $px1$ vector of fixed effects regression coefficients. Z is a design matrix of q random effects, u is a $qx1$ unknown vector of random effects. ϵ is a $Nx1$ vector of the residuals, which is a part of y that the model $X\beta + Zu$ does not explain.

In longitudinal data, the observed responses are represented by y_{ij}, where x_{ij} represents fixed effects of explanatory variables, while u_{ij} is a vector for random effects.

$$Y_{ij} = X_{ij}\beta + Zu_i + \epsilon_{ij}$$

Here $i = 1, . . , n, j = 1, . . , n_i$ in which i indicates individuals and j indicates measurements within individuals. A random intercept model enables every individual baseline to differ around the mean β_0 which along with β_1 are the parameters we approximate for the fixed effects. On the other hand, the parameters we approximate are σ_u^2, σ_e^2 which are the variances for the random effects. The random intercept and slope model are

$$Y_{ij} = \beta_0 + X_{ij}\beta_1 + X_{ij}u_{1i} + u_{0i} + \epsilon_{ij}$$

This model enables every individual to differ around the mean. This accompanies a normal distribution that has its own mean and variance. $Y_i\ N(0, G)$ in which G is the covariance matrix of the random effects Y_i. σ_0^2 - random intercept variance. When this is small or equal to zero, it gives similar intercepts for individuals. When it is large, it gives different intercepts for individuals. σ_1^2 - random slope variance. When this is equal to zero, it gives the same slopes for individuals which translates to a parallel regression line. However, when it is large it gives different slopes for individuals. σ_{01} - the covariance between u_{0i} and u_{1i}. When this is equal to zero, no correlation exists between the slope and intercept. When it is positive then high intercepts are linked with high slopes. When it is negative high intercepts are linked with lower slopes.

Prediction with Linear Mixed Models

Best linear prediction is used to estimate the random effects in linear mixed models. If we take $Y = X\beta + \epsilon *$ with $\epsilon *= Uy + \epsilon$ it will result in a linear model with correlated errors, then we will be able to estimate β.

$$\beta^{=}(X^T V^{-1} X)^{-1} X^T V^{-1} Y$$

with $V = Cov(\epsilon *) = yGy^T + R$. We can get $\hat{y} = GZ^T V^{-1}(y - X\hat{\beta})$ by using $\hat{\beta}$. Where

$$V = Cov(\hat{u})$$

$$C = E[(v - E[v])(y - E[y])^T]$$

$$BLP(v) = E[v] + CV^{-}1(y - E[y])$$

We can take our $\hat{\beta}$ and $\hat{y}$ to be the best linear unbiased predictor (BLUP), in which they both rely on $G = Cov(\gamma)$ and $R = Cov(\epsilon)$. BLUP is an important part in linear mixed models in estimating random effects.

Difference between Fixed & Random Effects

It is important to specify if a model predictor is fixed or random when fitting a linear mixed model as both fixed and random effects make up mixed effects models. This is decided based on the aim of the analysis. Fixed effects are where data are collected from every level of the aspect that we are interested in. In this study, time is a fixed effect. Random effects have multiple viable levels, so we are interested in all viable levels. However, we only have a random sample of these levels in the data. In this study, the players are the random effects. The aim of the analysis is not to estimate the effect of the predictor variables on just the players in the sample but to instead estimate the variability associated with the factor "Player ID". The analysis of the data will be contrasting based on whether the factor is classified as a fixed or random effect. As a result, inferences can be misleading if the factor is incorrectly classified. This is likely to happen when there is more than one factor in a study. The common methods for analysing random effects models believe that the random factor has an infinite number of levels. However, these can still work well provided that the overall number of levels of the random factor is 100 times or more the number of levels detected in the data. If the overall number of levels of the random factor is less than 100 times the number of levels detected in the data, then unique methods also known as "finite populations" may be needed. An interaction term which includes a fixed and random factor should be regarded as a random factor [19]. A random factor that is rooted in a random factor should be regarded as random also.

3.7 Logistic regression: Injury modelling

Logistic regression is the most suitable analysis to use if the dependent variable is binary. It is predictive regression and can be used to describe the relationship involving a dependent variable and ordinal or multinomial variables for example.

Assumptions

The dependent variable should be binary by order. If the continuous predictors are transformed to standardized scores and by eliminating scores that are less than -3.29 and bigger than 3.29, then there should not be any outliers in the data. If amongst the independent variables the correlation coefficient is less than 0.9 then it can be said that there is little multicollinearity between the predictors. A multiple linear regression function is recognized as:

$$log_{it}(p) = log\frac{p(y=1)}{1-(p=1)} = \beta_0 + \beta_1 x_{i2} + \beta_2 x_{i2} + .. + \beta_p x_{i2}$$

for $i = 1..n$.

Overfitting

Model fit is a major consideration to consider when picking a model for logistic regression. Variance will increase with every independent variable added, which will be described by the log odds which can be conveyed as R^2. Overfitting occurs if too many variables are continually added, as this will decrease the generalisability of the model past the data to which it is fitted [20]. R^2 are made for binary logistic regression. However, they should be carefully used as they have several computational problems that can cause them to be unnaturally high or low. Goodness of fit tests are a better choice.

3.8 How to compare models, goodness of fit

In statistical modelling, model comparison plays a vital role. It allows us to determine the true relationship that exists between variables by helping us choose a model that fits the data well while not being overly-complex.

Likelihood ratio test

The likelihood ratio test, a hypothesis test, enables us to compare the goodness of fit of two nested models and choose the optimal model. When one model is a unique sample of the other, we refer to them as "nested models". The optimal model is the one which maximizes the likelihood function, $f_n(X-1,\dots,X_n|\theta)$, which is highest the closer it is to the actual value of θ. Log-likelihood functions are used to calculate the likelihood-ratio tests. Usually, simpler models are compared with more complex models to determine if the more complex model is a better fit for the data set, in which case it can be used for ensuing analyses. The hypothesis test contrasts the two models, with H_0 as the smaller model. H_0 is rejected if the test statistics is bigger, meaning the larger model is better suited than the smaller one.

$$H_0: \theta \subset \theta_0$$

$$H_a: \theta \subset \theta_a$$

With $\theta_0 \subset \theta_a$ as subspaces of θ.

$$\lambda(x) = \frac{L(\theta|x): \theta \in \theta_0}{L(\theta|x): \theta \in \theta_a}$$

This test statistic is used to approximate chi-squared random variables. The number of additional parameters between the models is the same value as the degrees of freedom of the test. With this value we are then able to calculate the critical value of the test statistic.

Akaike Information Criterion

Akaike Information Criterion (AIC) is used for comparing statistical models with one another. AIC's scoring grading system ranks models from best to worst. The model with the minimum AIC value is the preferred model. AIC rewards goodness of fit by applying a penalty which is an increasing function of the number of parameters in a model. This penalty is useful as it penalizes overfitting as adding more parameters improves the goodness of fit of a model. We know that AIC will help us choose the best model from a set but what it won't do is tell us about the absolute quality of it. Therefore, we must ensure to not base our outcome on just AIC alone. Carrying out a hypothesis test is beneficial as it will tell us more about the relationship between the variables in the model and our outcome. With $\hat{L}$ the max value of the likelihood function and k the estimated number of parameters, we can say the AIC value for the model is:

$$AIC = -2ln(\hat{L}) + 2k$$

Bayesian information criterion (BIC) is another method which is very similar to AIC and is partly formed on the likelihood function. The model with the smallest BIC is preferred from a limited set of models.

However, the main aim is to produce parsimonious models for the data in in which unnecessary terms are excluded. Note the use of plural in models; it is most unlikely with complex data that a single model will be a clear winner, and it can be most misleading to quote only the best model when several others are very close to it in terms of goodness of fit.

3.9 Cluster Analysis

Cluster analysis involves a range of methods that are used to classify data into relative groups ("clusters") in which they are closely related to the individual elements of the cluster. Normally, the distance between clusters is measured with Euclidean distance or its square. Cluster analysis can be hierarchical or non-hierarchical. The hierarchical method can be distinguished by the evolution of a tree-like structure and involves one of two main procedures, either agglomerative or divisive. The agglomerative approach comprises of variance, linkage and centroid methods. All points begin with their own individual cluster and the two 'nearest' points or 'neighbours' are amalgamated. This step is repeated multiple times until all points have merged into one cluster. The non-hierarchical also known as k-means clustering, is very fast as it only involves computing the distances between points and the centres of groups. The ideal number of clusters is determined initially and then the clusters are allocated the data based on similarities. This is done by the researcher which is not ideal as it can be challenging to determine how many clusters you have. This, along with the fact that the k-means begins with random cluster centres, can lead to varied cluster results on separate iterations of the algorithm. Therefore, the results might not be consistent so other cluster analysis methods may work better such as k-medians. K-medians is not as sensitive to outliers but is far slower for big data in comparison to k-means.

4. Modelling Training Load

I used linear models and linear mixed models to carry out an analysis of the data in R for the outcome variables of interest (distance and load).

4.1 Linear Regression Model

Total Distance

Initially, I used a linear model upon which to base my initial estimation. In this dataset, the predictor variables of most interest are time (in days) and position. In the linear model, we are considering the effect of time on the total distance covered, the results of which can be seen below.

```
lm <- lm(TDkm ~ time, data=dat) # simple linear model considering effect of
time on TDkm
lm1 <- lm(TDkm ~ time + position, data=dat) # multiple linear model
considering effect of time and position on TDkm
lrtest(lm,lm1) # likelihood ratio test comparing simple linear model and
multiple linear model
```

```
## Likelihood ratio test
##
## Model 1: TDkm ~ time
## Model 2: TDkm ~ time + position
##   #Df  LogLik Df  Chisq Pr(>Chisq)
## 1   3 -1900.9
## 2   5 -1856.8  2 88.183  < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
plot(lm1)
```

Code Chunk 8: Linear models for TD and likelihood ratio tests comparing the models.

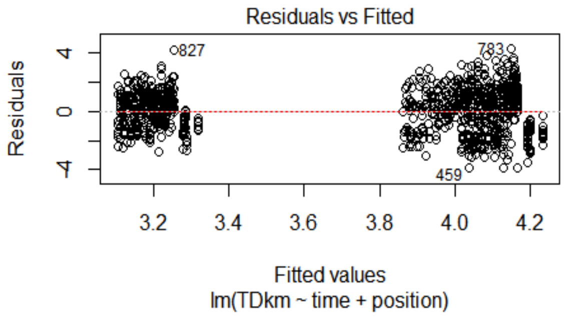


Figure 9.1: Residuals vs Fits plot.

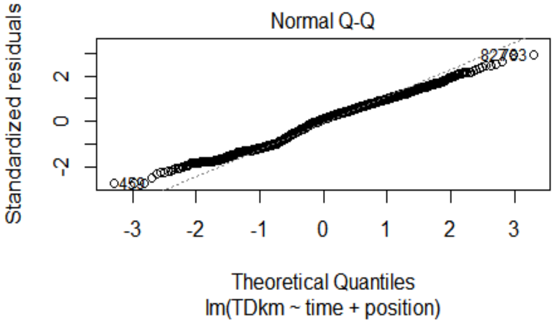


Figure 9.2: Normal Q-Q plot.

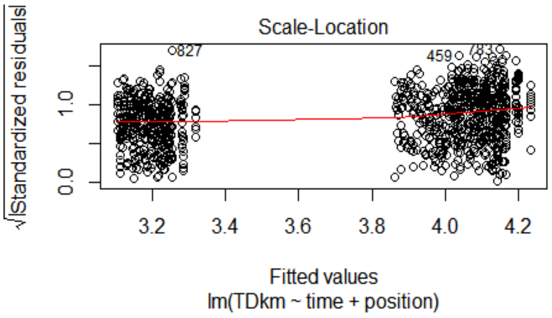


Figure 9.3: Scale-Location plot.

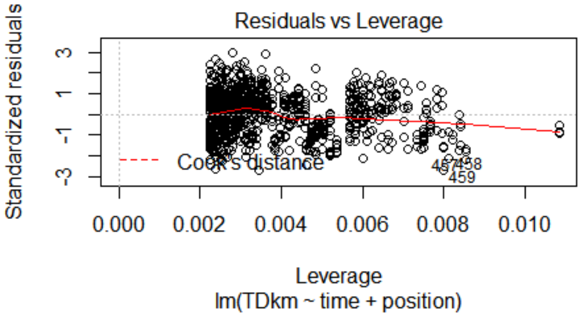


Figure 9.4: Residuals vs Leverage plot.

In Code Chunk 8 the p-value for TD is small, we can conclude that it is significantly affected by time. Based on the log-likelihood scores, we can also tell that position impacts on the total distance covered by a player. This shows that position is a useful variable for the model so we can take lm1 as the better model and use it in place of lm. In Figure 9.2, the Q-Q plot, all the points fall approximately along this reference line. Therefore, we can assume normality as the deviation at the tail ends is not evidence of significant departure. The Scale-Location plot in Figure 9.3 shows some homoscedasticity, indicating constant variance of the residuals. This could be a result of a very demanding training session or game. The R^2 value here is 0.08261 which is minimal so it may insinuate the model does not fit the data well. Usually regression models which fit a data well have a high R^2 value. There can, however, be some exceptions. Some areas of study have a significant amount of unexplained variation which will result in low R^2 values. If we have low R^2 values but still have noteworthy independent variables, we can still determine significant conclusions relating to the relationships between the variables. It should be noted that low R^2 values can cause problems if the aim is to create precise predictions. However, a high R^2 value alone is also insufficient.

Player Load

```
lm2 <- lm(PL ~ time, data=dat) # simple linear model considering effect of ti
me on PL
lm3 <- lm(PL ~ time + position, data=dat) # multiple linear model considering
effect of time and position on PL
lrtest(lm2,lm3) # likelihood ratio test comparing simple linear model and mul
tiple linear model

## Likelihood ratio test
##
## Model 1: PL ~ time
## Model 2: PL ~ time + position
##   #Df  LogLik Df  Chisq Pr(>Chisq)
## 1   3 -6798.6
## 2   5 -6791.4  2 14.393  0.0007494 ***
## ---

## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1plot(lm3)
```

Code Chunk 9: Linear models for PL and likelihood ratio tests comparing the models.

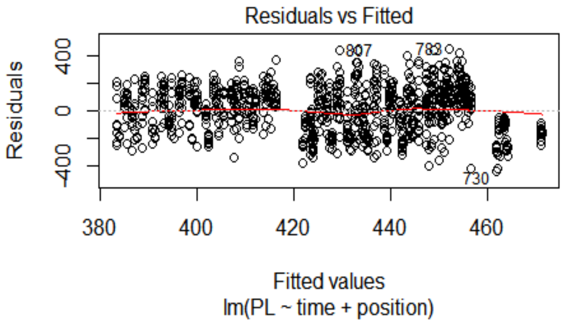


Figure 10.1: Residuals vs Fits plot.

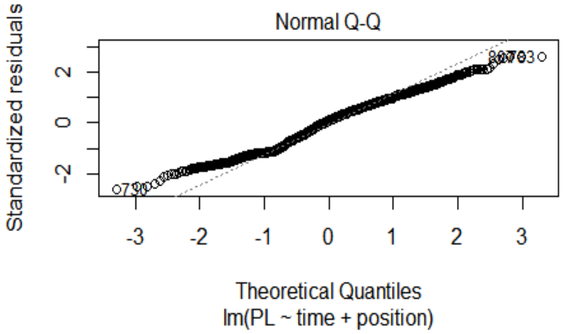


Figure 10.2: Normal Q-Q plot

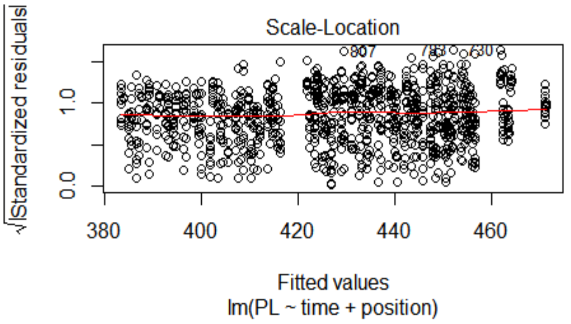


Figure 10.3: Scale-Location plot.

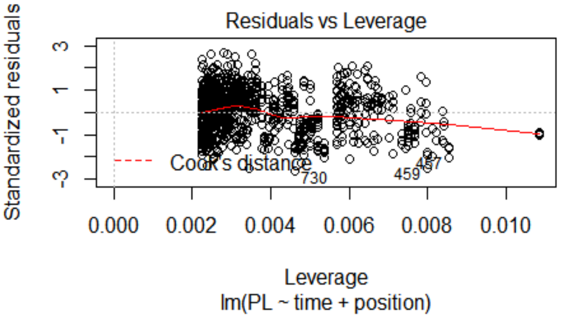


Figure 10.4: Residuals vs Leverage plot.

Player load, like total distance, is a continuous variable. In Code Chunk 9 upon viewing the log-likelihood scores, we can see that position does not have a major impact on PL which is unsurprising. The amount of load a player is under can be a result of variables other than TD, such as number of tackles committed/sustained. If a wide receiver realises they are unlikely to make up more ground, they may choose to run the ball out of bounds instead of bracing themselves for the inevitable tackle(s). This smart play prevents unnecessary trauma and possible injury after an accumulation of heavy tackles over the course of a game. In Figure 10.2 in the Q-Q plot, all the points fall approximately along this reference line, so we can assume normality. The Residuals vs Fitted plot in Figure 10.4 shows a relatively linear result as there is a noticeable increase or drop off in PL. Although the p-value here is not as small as it was for TD, it still provides enough evidence to assume that time has a significant effect on PL. We can therefore reject $H_0: \beta_1 = \beta_2 =\ldots \beta_p = 0$. This tells us that regression here is significant. The R^2 value is 0.01879 which implies that it is probably not fitting the data as well as we would like. We therefore have reason to believe that the model is not giving the true description of the relationship, which indicates that improvements can be made.

4.2 Linear Mixed Model

As the data are longitudinal, I expect the linear mixed model, which is produced with the nlme package in R, to be able to explain the data more accurately than a simple linear model as it allows for individual intercepts and slopes. First, I constructed a linear mixed model to estimate the effect of time on total distance in which individual intercepts were allowed, but not individual slopes. Next, I constructed the same model but this time I allowed players to have both their own intercepts and slope. Below, I carried out a log-likelihood ratio test to determine which model fits the data better.

```
lme1 <- lme(TD ~ time + position, data=dat, random = ~1|id) # linear random i
ntercept model - allows players to have their own intercepts, but all have th
e same slope
lme2 <- lme(TD ~ time + position, data=dat, random = ~time|id) # linear rando
m intercept and slope model - allows players to have their own intercepts and
slope
lrtest(lme1, lme2) # suggests random slopes are not necessary

## Likelihood ratio test
##
## Model 1: TD ~ time + position
## Model 2: TD ~ time + position
##   #Df  LogLik Df  Chisq Pr(>Chisq)
## 1   6 -8961.5
## 2   8 -8961.2  2 0.5816     0.7477
```

Code Chunk 10: Linear mixed effects model for TD and likelihood ratio tests.

In Code Chunk 10 there was only an infinitesimal difference in the log-likelihood scores. The p-value for (lme2) is 0.7477 which is not small, indicating to me that random slopes may not be necessary here and that there is not a linear relationship between the variables. I expected that increasing the complexity of the (lme1) model with polynomial regression would improve the model fit.

I also used a linear mixed model to estimate the effect of time on player load. I again carried out a log-likelihood ratio test to determine which model fits to the data better.

```
lme11 <- lme(PL ~ time + position, data=dat, random = ~1|id) # linear random
intercept model - allows players to have their own intercepts, but all have t
he same slope
lme22 <- lme(PL ~ time + position, data=dat, random = ~time|id) # linear rand
om intercept and slope model - allows players to have their own intercepts an
d slope

lrtest(lme11, lme22) # suggests random slopes are not necessary

## Likelihood ratio test
##
## Model 1: PL ~ time + position
## Model 2: PL ~ time + position
##   #Df  LogLik Df  Chisq Pr(>Chisq)
## 1   6 -6731.3
## 2   8 -6731.2  2 0.2517     0.8817
```

Code Chunk 11: Linear mixed effects models for PL and likelihood ratio tests.

I found again there was only an infinitesimal difference in the log-likelihood scores Code Chunk 11 and the p-value is not small here either. This indicated to me that random slopes may not be necessary here and that there is not a linear relationship between the variables. We will now take the (lme11) model that allows for individual intercepts but equal slopes and determine if increasing the complexity of the model with polynomial regression will improve the model fit.

4.3 Polynomial model

Here, I created a new variable $time^2$ and constructed three more differing models for comparison. The (lme3) polynomial model allowed players to have their own intercepts, but all have the same slope and quadratic change. The (lme4) allowed for both individual intercepts and slopes but all having the same quadratic change. However, the (lme5) polynomial model allowed for individual intercepts, slope and quadratic change. Again, a log-likelihood ratio test was used to compare the models and determine which fitted the data the best.

```
lme3 <- lme(TD ~ poly(time, 2) + position, data=dat, random = ~1|id) # quadratic random intercept model - allows players to have their own intercepts, but all have the same slope and quadratic change
lme4 <- lme(TD ~ poly(time, 2) + position, data=dat, random = ~time|id) # quadratic random intercept + slope model - allows players to have their own intercepts, slope but all have the same  quadratic change
lme5 <- lme(TD ~ poly(time, 2) + position, data=dat, random = ~time + I(time^2)|id) # quadratic full random effect model - allows players to have their own intercepts, slope and quadratic change

lrtest(lme1, lme3) # suggests quadratic model is superior to linear

## Likelihood ratio test
##
## Model 1: TD ~ time + position
## Model 2: TD ~ poly(time, 2) + position
##   #Df  LogLik Df  Chisq Pr(>Chisq)
## 1   6 -8961.5
## 2   7 -8899.7  1 123.65  < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

lrtest(lme3, lme4) # suggests quadratic model with random intercepts only is better than including random slopes

## Likelihood ratio test
##
## Model 1: TD ~ poly(time, 2) + position
## Model 2: TD ~ poly(time, 2) + position
##   #Df  LogLik Df Chisq Pr(>Chisq)
## 1   7 -8899.7
## 2   9 -8898.7  2 2.105     0.3491
```

Code Chunk 12: Polynomial regression models for TD and likelihood ratio tests.

On comparing the (lme1) and (lme3) polynomial models in Code Chunk 12 the log-likelihood scores along with the p-value being very close to zero showed that there was a noticeable improvement in the (lme3) model fit when we allowed players to have their own intercepts, but all have the same slope and quadratic change. While with the (lme3) and (lme4) models however there was no significant change that would justify adding more parameters to the model by allowing each player to have their own slope. The p-value for (lme4) is also large at 0.3012. We have now determined that (lme3) is the optimal model for TD.

```
lme33 <- lme(PL ~ poly(time, 2) + position, data=dat, random = ~1|id) # quadr
atic random intercept model - allows players to have their own intercepts, bu
t all have the same slope and quadratic change
lme44 <- lme(PL ~ poly(time, 2) + position, data=dat, random = ~time|id) # qu
adratic random intercept + slope model - allows players to have their own int
ercepts, slope but all have the same  quadratic change
lme55 <- lme(PL ~ poly(time, 2) + position, data=dat, random = ~time + I(time
^2)|id) # quadratic full random effect model - allows players to have their o
wn intercepts, slope and quadratic change

lrtest(lme11, lme33) # suggests quadratic model is superior to linear

## Likelihood ratio test
##
## Model 1: PL ~ time + position
## Model 2: PL ~ poly(time, 2) + position
##   #Df  LogLik Df  Chisq Pr(>Chisq)
## 1   6 -6731.3
## 2   7 -6664.2  1 134.24  < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

lrtest(lme33, lme44) # suggests quadratic model with random intercepts only i
s better than including random slopes

## Likelihood ratio test
##
## Model 1: PL ~ poly(time, 2) + position
## Model 2: PL ~ poly(time, 2) + position
##   #Df  LogLik Df  Chisq Pr(>Chisq)
## 1   7 -6664.2
## 2   9 -6663.5  2 1.3516     0.5088
```

Code Chunk 13: Polynomial regression models for PL and likelihood ratio tests.

When I swapped PL for TD in Code Chunk 13 and compared (lme11) and the (lme33) polynomial model, the log-likelihood scores showed that there was a noticeable improvement in the (lme33) model fit. This was because I allowed for individual intercepts but kept the same slope and quadratic change while the p-value was also very small. Between the (lme33) and (lme44) models, however, there was no significant change that would justify adding more parameters to the model by allowing individual intercepts and slopes but all having the same quadratic change. The p-value also increased significantly to 0.5285 so we cannot reject the null hypothesis here. We can conclude that (lme33) is optimal for PL.

4.4 Linear Spline Mixed Model

Linear mixed models presume that both the within cluster residuals and random effects are normally distributed.

Total Distance Comparing fixed effects

We compared linear spline models with a single knot at day 10, 20, 30 and 40. I carried this out with a likelihood ratio test using log(TD) versus time as H_0 while I used log(TD) against time and time spline10/20/30/40 as the H_a.

```
# created spline variables for day 10/20/30/40

dat$spline10 <- ifelse(dat$time > 10, dat$time - 10, 0)
dat$spline20 <- ifelse(dat$time > 20, dat$time - 20, 0)
dat$spline30 <- ifelse(dat$time > 30, dat$time - 30, 0)
dat$spline40 <- ifelse(dat$time > 40, dat$time - 40, 0)

# created linear spline mixed models with individual intercepts but all playe
rs having the same slopes and quadratic change, with knots at day 10/20/30/40
slme1 <- lme(TD ~ time + position + spline10, random = ~1|id, data = dat)
slme2 <- lme(TD ~ time + position + spline20, random = ~1|id, data = dat)
slme3 <- lme(TD ~ time + position + spline30, random = ~1|id, data = dat)
slme4 <- lme(TD ~ time + position + spline40, random = ~1|id, data = dat)

lrtest(slme1, slme2) # spline at day-20 is best here
```

```
## Likelihood ratio test
##
## Model 1: TD ~ time + position + spline10
## Model 2: TD ~ time + position + spline20
##   #Df  LogLik Df  Chisq Pr(>Chisq)
## 1   7 -8947.3
## 2   7 -8915.6  0 63.306  < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
lrtest(slme2, slme3) # spline at day-30 is best here
```

```
## Likelihood ratio test
##
## Model 1: TD ~ time + position + spline20
## Model 2: TD ~ time + position + spline30
##   #Df  LogLik Df  Chisq Pr(>Chisq)
## 1   7 -8915.6
## 2   7 -8866.2  0 98.769  < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

lrtest(slme3, slme4) # spline at day-20 is best here

## Likelihood ratio test
##
## Model 1: TD ~ time + position + spline30
## Model 2: TD ~ time + position + spline40
##   #Df  LogLik Df  Chisq Pr(>Chisq)
## 1   7 -8866.2
## 2   7 -8894.8  0 57.146  < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Code Chunk 14: Comparing linear spline mixed models for TD.

Based on the log-likelihood scores in Code Chunk 14, it is clear that (slme2) with a spline at day 20 is a better fit than (slme1) which has a spline at day 10. However, adjusting this with a spline at day 30 for (slme3) again improves the model which is reflected in its log-likelihood score in comparison to (slme2). When I tried to adjust this further with a spline at day 40 it made the model worse. I have now found that the (slme3) model is preferred in which we allow a knot at day 30, giving us the optimal spline model. Next, I will compare random effect options for this model.

Total Distance Comparing random effects

```
slme3 <- lme(TD ~ time + position + spline30, random = ~1|id, data = dat)
slme3_1 <- lme(TD ~ time + position + spline30, random = ~time|id, data = dat
) # linear spline mixed model with individual intercepts and slopes for playe
rs but all having the same quadratic change
# slme33_2 <- lme(PL ~ time + position + spline30, random = ~time + spline30|
id, data = dat) ... doesn't converge

lrtest(slme3, slme3_1) # slme3 model is optimum here

## Likelihood ratio test
##
## Model 1: TD ~ time + position + spline30
## Model 2: TD ~ time + position + spline30
##   #Df  LogLik Df  Chisq Pr(>Chisq)
## 1   7 -8866.2
## 2   9 -8864.8  2 2.7992     0.2467
```

Code Chunk 15: Comparing linear spline mixed models with random effects.

The random intercept only model for a spline with knot at day 30 is optimal. We can now compare this to the optimal polynomial model.

```
lrtest(slme3, lme3) # slme3 model is optimum here for TD

## Likelihood ratio test
##
## Model 1: TD ~ time + position + spline30
## Model 2: TD ~ poly(time, 2) + position
##   #Df  LogLik Df  Chisq Pr(>Chisq)
## 1   7 -8866.2
## 2   7 -8899.7  0 66.984  < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Code Chunk 16: Likelihood ratio test shows model with spline at day-30 is best for TD.

The results from the Likelihood Ratio Test in Code Chunk 16 show the model with a knot at day 30 is preferred, with a log likelihood of -8884. The p-value for the difference in likelihood scores is only 2.2e-16. As a result of this, there is sufficient evidence to reject H_0 and therefore we have now found that the spline model with a single knot at day 30 is best. However, this dataset is tracking the same group of players over an 89-day period and therefore each individual's observations are not independent of one another. In order to account for the variability between the players, a random effect should be added to the model.

Player Load Comparing fixed effects

Again, knots were placed at days 10, 20, 30 and 40. Here, these models are compared using the likelihood ratio test.

```
# created linear spline mixed models allowing for individual intercepts but a
ll players having the same slopes and quadratic change, with knots at day 10/
20/30/40

slme11 <- lme(PL ~ time + position + spline10, random = ~1|id, data = dat)
slme22 <- lme(PL ~ time + position + spline20, random = ~1|id, data = dat)
slme33 <- lme(PL ~ time + position + spline30, random = ~1|id, data = dat)
slme44 <- lme(PL ~ time + position + spline40, random = ~1|id, data = dat)

lrtest(slme11, slme22) # spline at day-20 is best here

## Likelihood ratio test
##
## Model 1: PL ~ time + position + spline10
## Model 2: PL ~ time + position + spline20
##   #Df  LogLik Df Chisq Pr(>Chisq)
## 1   7 -6715.6
## 2   7 -6680.8  0 69.45  < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
lrtest(slme22, slme33) # spline at day-30 is best here

## Likelihood ratio test
##
## Model 1: PL ~ time + position + spline20
## Model 2: PL ~ time + position + spline30
##   #Df  LogLik Df  Chisq Pr(>Chisq)
## 1   7 -6680.8
## 2   7 -6625.3  0 111.02  < 2.2e-16 ***

lrtest(slme33, slme44) # spline at day-30 is best here

## Likelihood ratio test
##
## Model 1: PL ~ time + position + spline30
## Model 2: PL ~ time + position + spline40
##   #Df  LogLik Df  Chisq Pr(>Chisq)
## 1   7 -6625.3
## 2   7 -6653.3  0 55.869  < 2.2e-16 ***
```

Code Chunk 17: Comparing linear spline mixed models for PL.

The linear spline mixed model with a knot at day 30 is optimal. The random effects now need to be decided.

Player Load Comparing random effects

Again, I wanted to use the linear random intercept and slope model to compare random effects and determine if allowing only individual intercepts or also individual slopes would make any difference by improving the model with a knot at day 30.

```
slme33 <- lme(PL ~ time + position + spline30, random = ~1|id, data = dat)
slme33_1 <- lme(PL ~ time + position + spline30, random = ~time|id, data = da
t) # linear spline mixed model with individual intercepts and slopes for play
ers but all having the same quadratic change
# slme33_2 <- lme(PL ~ time + spline30, random = ~time + spline30|id, data =
dat) ... doesn't converge

lrtest(slme33, slme33_1) # random slopes model (slme33_1) not necessary here

## Likelihood ratio test
##
## Model 1: PL ~ time + position + spline30
## Model 2: PL ~ time + position + spline30
##   #Df  LogLik Df  Chisq Pr(>Chisq)
## 1   7 -6625.3
## 2   9 -6624.4  2 1.8353     0.3995
```

Code Chunk 18: Comparing linear spline mixed models with knot at day-30 is best for PL.

When we run a log-likelihood test comparing (slme33) and (slme33_1) in Code Chunk 18 we find there is no evidence to support the inclusion of random slopes. Although both models converge, there is an insignificant difference between log-likelihood scores of the models. As a result there is insufficient evidence to justify adding more parameters to the model by allowing each player to have their own slope as well as their own intercept. It should be noted also that the (slme33_2) model, which allows players have their own individual intercept, slope and spline change, does not converge. We can conclude there is no evidence to allow players to have a different change over time.

We can conclude that the random intercept only model with a knot at day 30 is optimal for PL. It has a log-likelihood score of -6625 and a p-value <2.2e-16.

4.5 Tabulating Results

```
results_splineTD<- cbind(coef(summary(slme33)),as.data.frame(intervals(slme33
)[1]$fixed)) # created variable summarising results for optimum TD model
results_splineTD<- results_splineTD[ ,c(1,6,8,5)] # improved variable to incl
ude only intercepts, lower and upper bound values and p-values in results tab
le
results_splinePL<- cbind(coef(summary(slme33)),as.data.frame(intervals(slme33
)[1]$fixed)) # created variable summarising results for optimum TD model
results_splinePL<- results_splinePL[ ,c(1,6,8,5)] # improved variable to incl
ude only intercepts, lower and upper bound values and p-values in results tab
le
round(rbind(results_splineTD, results_splinePL), 2) %>% kable # summarised re
sults rounded to two decimal places and presented in easy to read kable table
```

Code Chunk 19: Results summarised and presented in table format.

	Value	lower	upper	p-value
(Intercept)	164.17	102.97	225.37	0.00
time	12.28	10.58	13.97	0.00
Linebacker & Tight End	-2.74	-86.52	81.04	0.95
Linemen	-38.02	-102.98	26.95	0.24
spline30	-15.12	-17.07	-13.18	0.00
(Intercept)1	164.17	102.97	225.37	0.00
time1	12.28	10.58	13.97	0.00
Linebacker & Tight End1	-2.74	-86.52	81.04	0.95
Linemen1	-38.02	-102.98	26.95	0.24
spline301	-15.12	-17.07	-13.18	0.00

Figure 11: Results including intercepts for each position.

Clearly in Figure 11, the results above are very similar for both TD and PL with the random intercept only model with a knot at day 30 optimal for both. We carried out a log-likelihood test of TD & PL against time as our null model and TD & PL against time and time-spline30. The resulting p-value is very small here, so we have enough evidence to reject the null hypothesis and suggest the spline. This tells us that the addition of the spline and random effects improves the fit of the model.

5. Modelling Injury

5.1 Results

	Injured	***Uninjured***	***Total***
Backs, Wide Receivers & Secondary	7	3	10
Linebackers & Tight Ends	3	1	4
Linemen	8	1	9
Total	18	5	23

Figure 12: Number of players injured and uninjured by position.

Our overall results in Figure 12 make for interesting reading. Out of a squad of 23 players, 18 were injured at some point during the 89-day observational period while only 5 managed to remain injury-free. This high incidence of injuries equates to 78% of the playing squad which is a staggering statistic. If we break this down further Backs, Wide receivers & Secondary positions had 7 out of 10 players side-lined at some point. Linebackers & Tight Ends had a 75% injury rate as only 1 member remained unscathed. However, without doubt, the most alarming results lay with the Linemen who had 8 injured players out of a group of 9 in this time which is a frightening 89% injury rate. To have such a high return on injuries is very worrying for any team but even more so for a college team with young athletes [21]. In order to see if we can gain a greater understanding of these results, I will attempt to model the injuries. By doing so, I hope to find some correlations between TD, PL and individual positions that may help to explain the high injury rate somewhat.

5.2 Modelling

In order to model injuries, I will use a class of generalized linear models known as logistic regression. Here, I will use the logit function as our link function and the binomial distribution as the probability distribution to give use the logistic regression model. The logistic function is a genuine link function for the binomial distribution because it gives back values between 0 and 1 for arbitrary inputs. Logistic regression can be used as a classification algorithm to predict a categorical variable (Yes/No, Injured/Uninjured) based on a group of independent variables. However, an issue with binary logistic regression is that it treats each row as independent. So, having multiple observations for each variable may produce inaccurate results and make interpretation of the results challenging. As I noted at the beginning of this project, delivering my results clearly and concisely so that they may be understood by anyone is of utmost importance. Below we can see the data displayed when the model regards each observation as an individual player, i.e. 1039 players altogether.

As it is assumed that the logit transformation of the outcome variable displays a linear relationship with the predictor variables, interpretation of the regression coefficients may be difficult. We therefore transform the probabilities of the variables above so as the odds increase so too does the probability. By taking the exponential of the coefficients we can determine the odds for the different variables which are displayed below.

As TD in the data is in meters, I decided to create a new variable by dividing it by 1000 and displaying it in km. This will make interpretation of the results easier by avoiding unnecessary confusion.

```
glm1 <- glm(ever_injured ~ TDkm + position, data = dat, family = binomial(lin
k = "logit")) # generalised linear model considering effect of TDkm and posit
ion on ever getting injured

results_glm1 <- cbind(coef(summary(glm1)),as.data.frame(confint(glm1))) # cre
ated results variable containing summary of (glm1) and confidence interval
results_glm1<- results_glm1[ ,c(1,5,6,4)] # improved variable to include only
intercepts, 95% confidence interval and p-values in results table
results_glm1[,1:3] <- exp(results_glm1[,1:3]) # finding the exponential of th
e results to give the odds of incurring an injury per position for TDkm
round(rbind(results_glm1), 2) %>% kable # results rounded to two decimal plac
es, presented in easy to read kable table
```

Code Chunk 20: Modelling injuries for TD using generalised linear model.

	Estimate	2.5 %	97.5 %	Pr(>|z|)
(Intercept)	1.00	0.64	1.57	0.99
TDkm	1.21	1.10	1.34	0.00
Linebacker & Tight End	1.47	0.99	2.22	0.06
Linemen	4.24	2.94	6.22	0.00

Figure 13: Injury odds by position for TD.

Figure 13 shows that, for every km covered, across the squad the odds of a player picking up an injury increased by 21% on average. The 95% confidence interval (CI) suggests that 1km extra increases the odds by between 9.5 and 34% in the population of all players, with the p-value of 0.0002 suggesting evidence for an effect of distance on injury. For anyone playing any of the Linebacker & Tight End positions, their odds of picking up an injury is 47% on average but can vary between 0% (no effect) and 122%. Most notably, Linemen were over 3x more likely to sustain an injury for every extra km, while in extreme cases they were 5x as likely compared to Backs & Wide Receivers.

```r
glm2 <- glm(ever_injured ~ PL + position, data = dat, family = binomial(link = "logit")) # generalised linear model considering effect of PL and position on ever getting injured

results_glm2 <- cbind(coef(summary(glm2)),as.data.frame(confint(glm2))) # created results variable containing summary of (glm2) and confidence interval
results_glm2<- results_glm2[ ,c(1,5,6,4)] # improved variable to include only intercepts, 95% confidence interval and p-values in results table
results_glm2[,1:3] <- exp(results_glm2[,1:3]) # finding the exponential of the results to give the odds of incurring an injury per position for PL
round(rbind(results_glm2), 2) %>% kable # results rounded to two decimal places in kable table
```

Code Chunk 21: Modelling injuries for PL using generalised linear model.

	Estimate	2.5 %	97.5 %	Pr(>|z|)
(Intercept)	0.87	0.57	1.34	0.52
PL	1.00	1.00	1.00	0.00
Linebacker & Tight End	1.44	0.97	2.17	0.08
Linemen	3.92	2.73	5.70	0.00

Figure 14: Injury odds by position for PL.

We can see that in Figure 14, as player load increases the odds of someone on the team picking up an injury increased by only 0.2% on average. However, for Linebackers & Tight Ends the odds could be anywhere between 0% and 120% but around 43% on average while for Linemen they are again usually 3x more likely to get injured due to increased load. I suspect that the high odds of injury seen for both TD and PL could be due to the 1039 observations that the model considered to be players. This may very well have skewed the results due to the increased sample size in which far more injured Linemen would have been accounted for, for example.

Evidently, these results do not accurately represent the data due to there being eleven observations per player which are not recognised by the model. I will therefore upgrade our model to a generalized linear mixed model. Now using mixed effects, logistic regression considers the clustering between and within players. The results with random effects included are displayed below.

```
glmm1 <- glmer(injured ~ TDkm + position + (1|id), data = dat, family = binom
ial(link = "logit")) # generalised linear mixed model considering effect of T
Dkm and position on ever getting injured

results_glmm1 <- data.frame(cbind(coef(summary(glmm1)))) # created variable s
ummarising the results for (glmm1)
results_glmm1$CIL <- results_glmm1$Estimate - 1.96*results_glmm1$Std..Error #
improved variable to include results for lower confidence interval
results_glmm1$CIU <- results_glmm1$Estimate + 1.96*results_glmm1$Std..Error #
improved variable to include results for upper confidence interval
results_glmm1<- results_glmm1[ ,c(1,5,6,4)] # improved variable to include on
ly intercepts, upper and lower confidence limits and p-values in results tabl
e
results_glmm1[,1:3] <- exp(results_glmm1[,1:3]) # finding the exponential of
the results to give the odds of incurring an injury per position for TDkm
round(rbind(results_glmm1), 2) %>% kable # presenting the results rounded to
two decimal places in a kable table
```

Code Chunk 22: Accounting for variation between players using a generalised linear model.

	Estimate	CIL	CIU	Pr...z..
(Intercept)	0.01	0.00	0.03	0.00
TDkm	1.31	1.00	1.71	0.05
Linebacker & Tight End	0.90	0.28	2.85	0.86
Linemen	1.56	0.66	3.69	0.32

Figure 15: Updated injury odds by position for TD.

We can now see that in Figure 15 for every additional km, the odds of a member of the playing squad sustaining an injury ranges from 0.2-71% but on average is 31% with the 0.048 p-value confirming that TD in fact has an effect on injury likelihood. Compared to Backs & Wide Receivers, Linebackers & Tight Ends have similar odds of injury with an odds ratio of 0.9. The 95% CI suggests odds of injury is anywhere from 72% decreased odds to 185% increased odds while the p-value is high here at 0.86. The wide interval here is due to the low number of injuries (29) out of the total sample of 1039 measurements. Between 34% decreased odds and 267% increased odds reflects the likelihood of Linemen getting injured. The large p-values here for both Linebackers & Tight Ends and Linemen suggests there may not be a significant effect of TD on injury likelihood regarding individual player positions.

Due to the profile of a typical Lineman and their role on the pitch, it is not surprising that they are at greater risk of injury the more ground they cover, although they usually cover minimal ground anyway. They are the biggest and strongest players on the team, standing at least 6’3” and 300-plus lbs in many cases. As a result, their bodies are less amenable to running/sprinting, making them more susceptible to soft tissue injuries. This is reflected in the reduced injury odds of Linebackers & Tight Ends who although typically are around

6’3’’, weigh far less at ~200-plus lbs. This build marries speed and strength, enabling these players to run fast enough to catch receivers whilst also having enough power and size to dominate on the line and tackle running backs. In essence, their bodies are far better equipped for running than Linemen.

```
glmm2 <- glmer(injured ~ PL + position + (1|id), data = dat, family = binomia
l(link = "logit")) # generalised linear mixed model considering effect of PL
and position on ever getting injured

results_glmm2 <- data.frame(cbind(coef(summary(glmm2)))) # created variable s
ummarising the results for (glmm2)
results_glmm2$CIL <- results_glmm2$Estimate - 1.96*results_glmm2$Std..Error #
improved variable to include results for lower confidence interval
results_glmm2$CIU <- results_glmm2$Estimate + 1.96*results_glmm2$Std..Error #
improved variable to include results for upper confidence interval
results_glmm2<- results_glmm2[ ,c(1,5,6,4)] # improved variable to include on
ly intercepts, upper and lower confidence limits and p-values in results tabl
e
results_glmm2[,1:3] <- exp(results_glmm2[,1:3]) # finding the exponential of
the results to give the odds of incurring an injury per position for PL
round(rbind(results_glmm2), 2) %>% kable # presenting the results rounded to
two decimal places in a kable table
```

Code Chunk 23: Accounting for variation between players using a generalised linear model.

	Estimate	CIL	CIU	Pr...z..
(Intercept)	0.01	0.00	0.03	0.00
PL	1.00	1.00	1.00	0.04
Linebacker & Tight End	0.85	0.27	2.68	0.78
Linemen	1.33	0.59	2.99	0.50

Figure 16: Updated injury odds by position for PL.

When considering PL in Figure 16, the odds are very similar to that of TD. Within the squad, the odds of injury as PL increases is 0.25% with a p-value of 0.036. This provides enough evidence to reject the null hypothesis that PL does not have an impact on injury likelihood. For Linebackers & Tight Ends it could be between 73% decreased odds and 168% increased odds while for Linemen it is 33% on average. However, the p-values for both of these are large, suggesting that PL and position together may not be an accurate indicator of injury likelihood. Although the CI for Linemen is very wide here, their injury odds are noticeably smaller in comparison to what it was for TD with anywhere between 41% decreased odds and 199% increased odds. They again have a higher likelihood of injury than Linebackers & Tight Ends as their role is consistently more abrasive in some regards. Virtually every play that Linemen are involved in involves committing heavy tackles, either trying to protect their own Quarterback in offence or to put pressure on the opposing Quarterback in defence. While Linebackers & Tight Ends are also heavily involved in play, their roles are a little more varied between chasing Wide Receivers, putting in tackles and attacking as receivers when the opportunity arises. In essence performing the same moves and tackles in every play place's extraordinary pressure on the same areas of Linemen's bodies. As a result, the brunt of the load they endure is magnified on certain areas of their bodies. As a result, their load threshold may not be as high as that of other players whose roles may be far more varied. This results in their bodies being worked more evenly and thus are able to deal with a greater spread of workload.

6 Discussion

6.1 Summary

This project involved building various models with predictors Total Distance, Player Load and Position to determine the chances of injury across the squad of players. Initially, I compared linear models and linear mixed models for modelling training load which showed that linear mixed models are a far better fit for the data as linear models only have one regression fit for an entire population. As linear mixed models are more complex, I was able to consider the high variation between individual players TD and PL. I then followed on from this with generalized linear models (logistic regression) to model injuries which classified players as injured or uninjured. Generalized linear mixed models (logistic regression with mixed effects) accounted for the clustering between and within players which provided me with more reliable results. I found that mixed models are an effective statistical method for analysing repeated measures. I was able to attain homoscedascity through log transformation which satisfied our assumptions for the models. The estimates we obtained from our mixed models would be easy to understand in a sporting environment while the models we built were not overly complicated. As a result, overfitting was not a concern. From our data visualization, we knew that TD and PL increased significantly over time so we decided to fit a spline to the data to see would this improve the model. We found a spline at day 30 statistically significant which confirms that there was noticeable rise in both TD and PL around this time. We found the models to be accurate for predicting the likelihood of injury for the playing squad based off predictors TD and PL. However, when we added position into these models, it produced wide confidence intervals. For example, for PL regarding Linebackers & Tight Ends, we found their likelihood of incurring injury was anywhere between 73% decreased odds and 168% increased odds. It is likely that partly the reason for this is the limited sample size of 23 players of which only 4 are Linebackers & Tight Ends. It is possible that freak injuries, for example, would have a greater bearing on the results. For example, injuries like broken bones are unpreventable which could skew our results and produce wider confidence intervals. If we were to do the same study for the whole NCAA American Football league, I would expect to see a far narrower interval.

6.2 Strengths & Limitations

The narrow time frame I had to complete this project was an obstacle from the outset. Although I covered many models as well as splines, some of which are not taught here in NUIG, I would have liked to have explored other statistical techniques. Another limiting factor was the rather small data set I used. It would have been fascinating to have had access to injury data from every NCAA college football team. This would have provided me with a greater amount of information. More players per position would have resulted in greater accuracy in injury prediction injuries. I would also have liked access to the types of injuries sustained, length of recovery times, and whether or not players reinjured. Age, height and weight would also have been interesting variables to consider although they are somewhat accounted for by positions. With a greater sample size and more predictor variables, I would have liked to have developed more complex user-friendly interfaces like Shiny App which could be used in management's everyday analysis and plans.

It would have been beneficial to know what GPS trackers were used in this study and what company they were produced by. There have been instances in past studies in which GPS trackers have struggled during short-distance, high-speed movements, for example [22]. If I were completing research in this area, I would ensure that the trackers were calibrated correctly so that I could have full confidence in their accuracy. Also, knowing the manufacturer of the GPS trackers may have indicated how PL was measured. This would have given more meaning to the data and helped in our analysis of it.

6.3 Future Possibilities

The initial research and analysis carried out on this sample of college athletes has shown huge promise for future research. Preferably an athlete-monitoring and analytics developer, such as Kitman Labs, would work with a major sporting organization like the NFL to conduct a large-scale study under its umbrella. Committing to a major study like this over an extended length of time could yield significant findings for a sport in which player injuries is a known problem [23]. Although the study has demonstrated some notable results, we cannot popularize them until they are trialled over a larger athlete population.

6.4 Conclusion

Overall, I am happy with my dissertation and with what I have learned from it. Although my background is in Applied Mathematics I thoroughly enjoyed the statistical nature of this project as it is an area I am eager to delve into in the future. I found learning about linear mixed models and splines and their importance in modern sports science analysis extremely interesting. I had some experience in R before carrying out this project, but I have now greatly enhanced and built upon these base skills. As I mentioned, there is huge potential to extend this project further with extra variables and increased testing of the models. However, the initial testing and prediction models I built provide a solid basis for future study and analysis.

Bibliography

[1] Humphreys, B.R. and Ruseski, J.E., 2009. Estimates of the dimensions of the sports market in the US. *International Journal of Sport Finance*, *4*(2), p.94.

[2] Neville, J., Rowlands, D., Wixted, A. and James, D., 2012. Application of GPS devices to longitudinal analysis on game and training data. *Procedia Engineering*, *34*, pp.443-448.

[3] Connolly, F. and White, P., 2017. *Game changer*. Simon and Schuster.

[4] Yasmin Anwar “Everything big data claims to know about you could be wrong” June 2018 https://news.berkeley.edu/2018/06/18/big-data-flaws/

[4] Stephen Smith “What is the real cost of injuries in professional sport” April 2106 https://www.kitmanlabs.com/what-is-the-real-cost-of-injuries-in-professional-sport/
Öztürk, S. and Kılıç, D., 2013. What is the economic burden of sports injuries? *Joint Diseases and Related Surgery*, *24*(2), pp.108-111. URL
https://www.ncbi.nlm.nih.gov/pubmed/23692199

[6] Lambert, Mike & Borresen, Jill. (2010). Measuring Training Load in Sports. International journal of sports physiology and performance. 5. 406-11. 10.1123/ijspp.5.3.406.

[7] Play Division I Sports
http://www.ncaa.org/student-athletes/play-division-i-sports

[8] Reich, J.B., 2012. When Getting Your Bell Rung May Lead to Ringing the Bell: Potential Compensation for NFL Player Concussion-Related Injuries. *Va. Sports & Ent. LJ*, *12*, p.198.

[9] Wickham, H., 2006. An introduction to ggplot: an implementation of the grammar of graphics in R. *Statistics*.

[10] Anthony Turner, Strength and Conditioning Journal: February 2011 - Volume 33 - Issue 1 - p 34-46 Anthony Turner “The science and practice of periodization: A brief review” February 2011.   URL
https://journals.lww.com/nscascj/fulltext/2011/02000/the_science_and_practice_of_periodization__a_brief.6.aspx

[11] Neter, John, et al. Applied linear statistical models. Vol. 4. Chicago: Irwin, 1996.

[12] Ostertagova, Eva. (2012). Modelling Using Polynomial Regression. Procedia Engineering. 48. 500–506. 10.1016/j.proeng.2012.09.545.

[13] Ruppert, D., Wand, M.P. and Carroll, R.J., 2003. *Semiparametric regression*. Pages 15-20, 46-48. Cambridge university press.

[14] Osborne, J., 2010. Improving your data transformations: Applying the Box-Cox transformation. *Practical Assessment, Research, and Evaluation*, *15*(1), p.12.

[15] Ruppert, D. and Carroll, R.J., 1999. *Penalized regression splines*. Cornell University Operations Research and Industrial Engineering.

[16] Liang, K.Y. and Zeger, S.L., 1986. Longitudinal data analysis using generalized linear models. *Biometrika*, *73*(1), pp.13-22.

[17] McCullagh, Peter; Nelder, John (1989). *Generalized Linear Models, Second Edition*. Chapman & Hall/CRC. ISBN 0-412-31760-5.

[18] UCLA "Introduction to linear mixed models" https://stats.idre.ucla.edu/other/mult-pkg/introduction-to-linear-mixed-models/

[19] Laird, n. M., & Ware, J. H. (1982). Random-effects models for longitudinal data. Biometrics, 38(4), 963-974.

[20] Ruppert, D., Wand, M.P. and Carroll, R.J., 2003. *Semiparametric regression*. Pages 195-197. Cambridge university press.

[21] Dalton, S.E., 1992. Overuse injuries in adolescent athletes. *Sports medicine*, *13*(1), pp.58-70.

[22] Owen Walker "GPS (Wearables): Part 1 – Technology, Validity, and Reliability" January 2017 https://www.scienceforsport.com/gps-wearables-validity-and-reliability/

[23] Ehrlich, S.C., 2018. A More Perfect (NFL Players) Union: Secret Side Deals, The NFLPA, and the Duty of Fair Representation. *Ohio NUL Rev.*, *44*, p.33.