Linköping University | Department of Computer and Information Science
Master's thesis, 30 ECTS | Statistik och Maskininlärning
2023 | LIU-IDA/STAT-A--23/005--SE

Regime Based Analysis Using Gated Bayesian Network

– An implementation of Gated Bayesian Network for a Basketball Team

---

Basketanalys med regimskiftmetod

Dimitra Muni

Supervisor : Jose M. Peña
Examiner : Krzysztof Bartoszek

External supervisor : Patrick Lambrix

LINKÖPING UNIVERSITY

Linköpings universitet
SE–581 83 Linköping
+46 13 28 10 00 , www.liu.se

Upphovsrätt

Detta dokument hålls tillgängligt på Internet - eller dess framtida ersättare - under 25 år från publiceringsdatum under förutsättning att inga extraordinära omständigheter uppstår.

Tillgång till dokumentet innebär tillstånd för var och en att läsa, ladda ner, skriva ut enstaka kopior för enskilt bruk och att använda det oförändrat för ickekommersiell forskning och för undervisning. Överföring av upphovsrätten vid en senare tidpunkt kan inte upphäva detta tillstånd. All annan användning av dokumentet kräver upphovsmannens medgivande. För att garantera äktheten, säkerheten och tillgängligheten finns lösningar av teknisk och administrativ art.

Upphovsmannens ideella rätt innefattar rätt att bli nämnd som upphovsman i den omfattning som god sed kräver vid användning av dokumentet på ovan beskrivna sätt samt skydd mot att dokumentet ändras eller presenteras i sådan form eller i sådant sammanhang som är kränkande för upphovsmannens litterära eller konstnärliga anseende eller egenart.

För ytterligare information om Linköping University Electronic Press se förlagets hemsida `http://www.ep.liu.se/`.

Copyright

The publishers will keep this document online on the Internet - or its possible replacement - for a period of 25 years starting from the date of publication barring exceptional circumstances.

The online availability of the document implies permanent permission for anyone to read, to download, or to print out single copies for his/hers own use and to use it unchanged for non-commercial research and educational purpose. Subsequent transfers of copyright cannot revoke this permission. All other uses of the document are conditional upon the consent of the copyright owner. The publisher has taken technical and administrative measures to assure authenticity, security and accessibility.

According to intellectual property law the author has the right to be mentioned when his/her work is accessed as described above and to be protected against infringement.

For additional information about the Linköping University Electronic Press and its procedures for publication and for assurance of document integrity, please refer to its www home page: `http://www.ep.liu.se/`.

© Dimitra Muni

Abstract

In the dynamic sport of professional basketball, a team may encounter several tangible and intangible alterations over a period of time. Regime detection, a widely used method in the domain of financial market decision-making and medical treatment planning, may assist a team management personnel to gain insight into how a team is changing with time based on historical data[1]. In this project, we have explored how multiple Bayesian networks could model the dynamics for a certain time frame. We have utilised basketball records starting from season 1983-84 to season 2020-21 for the NBA team Chicago Bulls. First, we identified the most important parameters to describe a team's performance using SHAP[2] analysis. Next, we identify the regimes based on the Metropolis-Hastings MCMC sampling of the posterior distribution of the dataset. After identifying the regimes in the dataset, we hypothesise different regime transition structures and identify the most optimal regime transition structure. Lastly, the optimal regime transition structure is used to create a Gated Bayesian network (GBN) and parameterise the GBN using Gaussian processes distribution. We discuss the limitations of this approach and our work. Our finding has been visualised along with historical evidence, which helps to provide insight into which parameter may be important for a certain period of history and its possible application for the team officials.

Keywords: probabilistic modelling, SHAP analysis, Shapley Values, Bayesian network, directed acyclic graphs, basketball analytics, gated Bayesian network, regime-based analysis, gaussian processes optimisation, hill climbing, NBA, Chicago Bulls

[1]The similar study has been conducted by (Bendtsen M., 2017) for the career trajectories of the baseball players. https://link.springer.com/article/10.1007/s10618-017-0510-5

[2]SHAP: **SH**apley **A**dditive ex**P**lanation

Acknowledgments

I thank the project supervisor Jose M. Peña for his continuous guidance and counsel; the long conceptual discussions about several topics have been the most intriguing and enriching part of this project. My external supervisor Patrick Lambrix has provided invaluable support and feedback; I cannot thank him enough for trusting the vision of this thesis. Furthermore, I would like to thank examiner Krzysztof Bartoszek for the invaluable suggestions for improvement. I owe much gratitude to my opponent Nour Elhouda Qweder, whose remarks and critique have helped improve the draft's quality. I thank Julia Lindohf for inspiring me to pursue this project.

I could not have done this project without constant support from my close confidant Anjali Pabari; her encouraging words and wisdom have reinvigorated my enthusiasm for my project work. My parents and sister have been the strongest supporters and cheerleaders; their love and support have been the primary motivator in continuing this endeavour.

Contents

List of Figures

List of Tables

1 A General Introduction to Basketball Analysis

National Basketball Association (NBA) is a commercial sports organisation in North America; this league is formed by 30 teams based in the United States and Canada. However, the popularity of basketball transcends across the continents.

There has been a paradigm shift in basketball data analytics in recent years with the advent of sophisticated sensors providing tracking data and increased commercial interest in logging the event data on the court. In this project, we aim to utilise event data for a professional team in NBA during an epoch to analyse their on-court performance using probabilistic graphical modelling qualitatively.

1.1 Introduction to Basketball

Basketball has changed fundamentally after its introduction to the world almost a century ago; it has evolved into its current format of a high-speed game. Although a team sport, basketball has peculiar characteristics in that a professional team is usually formed around two or three-star players. These players are the usual ones who draw the maximum attention from the audience, composing a solvent and thriving team.

A basketball game is played between two teams, each comprising five players at a time on the court. The game is divided into four quarters of fifteen minutes; a team wins points based on the successful attempt to throw the ball into the opponent's basket.

Many players change their position on the court after being traded to a different team or after an injury. In their sunset years, some well-known NBA players decided to play in minor leagues in the United States or the basketball leagues in Europe; these changes in the team compositions have a lasting impact on team performance.

This research project will cover the selected NBA team Chicago Bulls and the parameters describing their on-court play, which could assist the general manager or coach of the team in investigating the cause of the change in the regime. The result might contribute to a better understanding of how to manage a team made up of a herd of solid and eccentric

personalities. We utilise the probabilistic graphical model to analyse the relationship between the parameters of interest.

1.2 Purpose of the Project

The overall purpose of this Master Project is to explore the usage of the Gated Bayesian Network to evaluate the performance of a professional basketball team through history, which could provide important insight into a team's playing dynamics for the coach and other team strategists. The on-court combating quality of a basketball team could be measured from many aspects. The apparent intention is to find out if we could use the gated-Bayesian network, which was used previously by Bendtsen (2016) [1] to assess the change of regimes of baseball players through their careers.

1.3 Research Objectives

The following three research objectives are thoroughly addressed in this degree project work. We list the research objectives as follows,

1. Explore the essential features from the historical basketball data that significantly impact the team's performance.
2. Scruitinse the applicability of the Gated Bayesian Network framework to the basketball data to learn the performance dynamics of a team.
3. Validate whether the aforementioned framework enhanced the understanding of a team's dynamics using statistical measures and historical evidence.

1.4 Scopes of the Master Project

The scope of this Master Project is bound to scrutinise the performance statistics of the NBA team Chicago Bulls between the seasons 1983-84 to 2020-21. Furthermore, in this analysis, each regular season, we consider that the individual team plays during 82 games, excluding the playoffs games. The reason for such a consideration is primarily because teams play quite differently in the playoffs than in the regular season, so the same model would not be an apt choice. Additionally, the non-commercial data is utilised in this project available on the `basketball-reference` website at gratis. Lastly, the model described in this discourse is insensitive to the alteration in the NBA regulations over the decades.

1.5 Structure of the Discourse

Chapter 1 introduces readers to the sport of basketball, along with weaving the narrative of the regime around a team over the passage of time. Furthermore, the purpose of this project is stated and further elaborated with research objectives described in this discourse.

Chapter 2 delves into the theoretical background required to address the research objectives. Firstly we introduce readers to SHAP analysis [2], a post-hoc method to identify the most prominent features in the dataset. Next, the framework of Bayesian statistics is explained to address more advanced sampling techniques, such as the Metropolis-Hastings MCMC algorithm. Next, we provide the statistical foundation of the Bayesian network to for conceptual understanding of a Gated Bayesian Network. Furthermore, the concept of the regime and the gating mechanism between the regimes is examined. Lastly, we discuss the hyper-parameters optimisation technique using Gaussian processes.

Chapter 3 features a detailed description of basketball data. Firstly, an overview of variables utilised to describe the team's performance statistics is presented, including box score and other intangible records. Additionally, the missing data in the context of complementary descriptive data, socio-psychological information and team management-related information is scrutinised. Finally, we examine the historical records and image of the team in a particular period, which could provide important insight into the commercial viability of a team during a particular epoch.

Chapter 4 describes statistical methods utilised in the project; firstly, the reader is introduced to the computational architecture of the method utilised, which consists of structure learning of Bayesian network(BN), regime identification and transition between individual BNs, hyper-parameters optimisation for BN transition using Gaussian Processes. Finally, we discuss the implementation of this framework using flow charts.

Chapter 5 presents results obtained after implementing the methods described in the earlier chapter; we scrutinise the results with historical data to realise the validity of the methods utilised.

Chapter 6 provides a nuanced discussion about the theoretical concepts and method implementation and results in a broader context of regime-based modelling; we describe the potential limitations of this approach and the consequence of assumptions in the model. Additionally, we acknowledge the ethical considerations in the project.

Chapter 7 concludes the findings and limitations of this project work and provides suggestions for future work on this project.

1.6 Prior Research

In the century of big data analytics, the statistical analysis of basketball records is utilised by commercial entities, such as sports betting companies, television broadcast enterprises and professional sports leagues, to generate products and services catering to stakeholders across the globe. These stakeholders are basketball fans, sports journalists, news broadcasters, coaches, physiotherapists and game developers. The advancement in data processing technology in the last few decades has simplified data storage-related challenges and addressed processing latency to a greater extent.

According to Page (2015), with the advent of sophisticated wearable sensors, players have access to their bio-mechanical data, which significantly impacts the elite team sport such as basketball [3]. Any basketball team usually has a subset of two or three players who dominate the game and has a decisive effect on the game result; these players are often subjected to massive media scrutiny, partly fueled by the commercial aspect of the business and partly due to curiosity about players amongst fans many aspiring players grew up in African American families of humble economic backgrounds (Lazenby and Bougard, 2015)[4]. For some young basketball players, the newly gained fandom and publicity might have adverse effects on the longevity and consistency of their performance (Eggers, 2021)[5]; for others, the constant media exposure and the stellar status could motivate them to train harder, aiming to reach a new height in their professional career (McCallum, 2012)[6].

Some of the on-court decisions of players may reflect their experiences during their young adulthood; the celebratory parties where alcoholic beverages and narcotics substances are

consumed may have a lasting impact on the team performance, and excessive drinking post-game may affect the recovery time and testosterone levels adversely (Barnes, 2014)[7]. Gauging insight into players' choices before a game may assist in understanding their frame of mind; some players exhibit repetitive, ritualistic routines or superstitions (Dömötör et al., 2016)[8].

In a regular season, every team participates in 82 games; the number of winning games decides whether a team could win a spot in the playoff season. According to Teramoto and Cross (2010), the defensive performance may be of greater importance compared to their offensive performance [9]. Furthermore, Morgulev and Galily (2018) argued that performance pressure could be an important factor during a decisive game of the playoffs, and the threat of elimination may affect the play in a negative way[10].

2 Theoretical Framework

2.1 Problem Definition

In this discourse, we formally define the problem explored in the thesis work. The problem definition is identical to that of the works of Bendtsen (2016, p.4-5) titled *Regime in Baseball Players' Career Data*. The author defined a regime informally as " a steady state of the relationship between the variables in the data"[1].

The author introduced a system $\mathcal{S}$ of random variables $\mathbf{X}$, where $\mathbf{X}$ is a collection of random variables $X_1, X_2, ..., X_m$ ($m \in \mathbb{N}$). The system $\mathcal{S}$ has regime labels $R_1, R_2, ..., R_q$ ($q \in \mathbb{N}$), where the regime corresponds to independence models $M_1, M_2, ..., M_q$ over the variables $\mathbf{X}$ such that joint probability distributions $p_1(\mathbf{X}), p_2(\mathbf{X}), ..., p_q(\mathbf{X})$ are positive. Furthermore, the author has assumed that each independence model is unique, i.e. $M_i \neq M_j$ for $i \neq j$.

The author denoted the complete dataset of $\mathbf{X}$ with $\mathcal{D}$; furthermore, an i^{th} individual observation was denoted by d_i. We assume there are n data points in $\mathcal{D}$. The author has used notation $d_i \sim R_l$ to denote that d_i (where $i \in [1, n]$) is observation of variables $\mathbf{X}$ when system $\mathcal{S}$ is in regime R_l (where $l \in [1, q]$). So if $\mathcal{D} = \{d_1 \sim R_1, d_2 \sim R_2, d_3 \sim R_2, d_4 \sim R_1\}$, it is interpreted as the observations d_1 and d_4 come from the regime R_1 while d_2 and d_3 come from regime R_2, which could be rewritten as $\mathcal{D} = \{R_1, R_2, R_2, R_1\}$, by only specifying the regimes the observations come from. Moreover, based on this $\mathcal{D}$, one can identify regime changes of the system $\mathcal{S}$, which transitions from R_1 to R_2 and then from R_2 to R_1. The author defined this transition mapping as a regime transition structure. For the aforementioned example, the regime transition structure can be specified using arrows as $R_1 \leftrightarrows R_2$. Now we consider another example, where $\mathcal{D} = \{R_1, R_1, R_2, R_2, R_3, R_2\}$, in this case the transition structure can be denoted by $R_1 \rightarrow R_2 \leftrightarrows R_3$.

The author specified that " If dataset $\mathcal{D}$ is a *valid* sample of $\mathcal{S}$, then it identifies true regime structure of $\mathcal{S}$"[1]. Whether a sample is *valid* or not is determined by comparing the true regime transition structure with the sample $\mathcal{D}$. If the true regime transition structure for $\mathcal{S}$ is $R_1 \rightarrow R_2 \leftrightarrows R_3$ then the sample $\mathcal{D} = \{R_1, R_2, R_3, R_2, R_2, R_2\}$ is a valid sample while $\mathcal{D} = \{R_1, R_2, R_3\}$, $\mathcal{D} = \{R_2, R_3\}$, $\mathcal{D} = \{R_1, R_1, R_2, R_2, R_3, R_1\}$ are not valid samples.

According to the author, although the dataset is assumed to be available completely, they also consider the observation available one by one, termed as *stream* $\mathcal{O}$. At time $t = 1$, $\mathcal{O} = \{d_1 \sim R_1\}$, at time $t = 2$, $\mathcal{O} = \{d_1 \sim R_1, d_2 \sim R_1\}$ and at time $t = j$, $\mathcal{O} = \{d_1 \sim R_1, d_2 \sim R_1, ..., d_j \sim R_l\}$.

Aim

We have access to a high-dimensional dataset $\mathfrak{Ds}$ in basketball. Firstly, we select the features which are more important to describe $\mathcal{S}$ using the dataset $\mathcal{D}$ (where $\mathcal{D} \subseteq \mathfrak{Ds}$). For the next steps, we refer to Bendtsen (2016, p.5), where the author described the principle aim as learning model $\mathcal{S}$ from dataset $\mathcal{D}$; furthermore, it is unknown which regime an observation belongs to. Additionally, the number of regimes included in $\mathcal{S}$ is also unknown. Here we list out the aims of this discourse, where aim (1) is included specifically for the basketball dataset, while aims (2) to (5) were described in the works of the author, also applicable for this thesis work [1].

1. Identifying the most important features of $\mathfrak{Ds}$, to construct the dataset $\mathcal{D}$.
2. Identifying the location of regime changes in dataset $\mathcal{D}$.
3. Identifying the regime $R_1, R_2, ..., R_q$ and learning the corresponding independence models $M_1, M_2, ..., M_q$ and joint distribution $p_1(\mathbf{X}), p_2(\mathbf{X}), ..., p_q(\mathbf{X})$.
4. Identifying the regime transition structure of $\mathcal{S}$.
5. Validation of the learned model and regime transition structure by taking a stream of data $\mathcal{O}$ and correctly identifying the regime $\mathcal{S}$ is in.

In section 2.2, we elaborate on feature selection employed to address aim (1).

2.2 SHAP Analysis for Feature Selection

The mathematical framework of Shapley Values was initially laid out in the seminal works by Lloyd Shapley in the domain of Game Theory [11], where it was initially utilised to attribute credits to an individual player within the coalition of players based on their contribution in the game.

Shapley Values

Molnar (2022) described Shapley value to be a measure of contribution by the feature towards the prediction of an instance (data point) for a linear model f [12], as described in equation 2.1.

$$\begin{aligned} \hat{f}(x) &= \beta_0 + \beta_1 x_1 + ... + \beta_p x_p \\ \phi_j(\hat{f}) &= \beta_j x_j - E(\beta_j X_j) = \beta_j x_j - \beta_j E(X_j) \end{aligned} \tag{2.1}$$

In the equation 2.1, $\hat{f}(x)$ is the prediction for data point x, and x_j is the feature value, β_j is the weight corresponding to feature x_j, where $j = 1, 2, ..., p$ ($p \in \mathbb{N}$). β_0 is the bias in the model. $\phi_j(\hat{f})$ is the contribution of feature j towards prediction $\hat{f}(x)$. Furthermore, the author represented the sum of all feature contribution for one instance as follow,

$$\sum_{j=1}^{p} \phi_j(\hat{f}) = \hat{f}(x) - E(\hat{f}(X)) \tag{2.2}$$

In equation 2.2, X is the data matrix. This equation describes the difference between the predicted value for the instance x and the average predicted value.

Molnar (2022) defined Shapley value using a function *val*, here *val* is function of features as described in equation 2.3[12],

$$\phi_j(val) = \sum_{S \subseteq \{1,2,...,p\} \setminus \{j\}} \frac{|S|!(p-|S|-1)!}{p!} \cdot \Big(val(S \cup \{j\}) - val(S) \Big) \tag{2.3}$$

- S is subset of features in the model, $|S|$ is the size of this subset.
- p represents the number of total features.

Estimation of Shapley Value

Molnar(2022) stated that, as the total number of features increases, the number of possible subsets (based on the total number of features) increases exponentially, which makes the exact solution expensive to compute[12]. Štrumbelj and Kononenko (2014) proposed a Monte-Carlo sampling approximation approach to address the above-mentioned problem[13], as described in the equation 2.4.

$$\hat{\phi}_j = \frac{1}{M} \sum_{m=1}^{M} \big(\hat{f}(x^m_{+j}) - \hat{f}(x^m_{-j}) \big) \tag{2.4}$$

- Here x^m_{+j} represents the vector corresponding to iteration m, where x_{+j} combines the instance x as $x_1, ..., x_j$ with another instance z as $z_{j+1}, ..., z_p$, as described in Algorithm 1.
- x^m_{-j} represents the vector corresponding to iteration m, where x_{-j} combines instance x as $x_1, ..., x_{j-1}$ with another instance z as $z_j, ..., z_p$.
- $\hat{f}(x^m_{+j})$ and $\hat{f}(x^m_{-j})$ represents the predicted value for the vector x^m_{+j} and x^m_{-j} accordingly.

Molnar (2022) proposed an algorithm to calculate the Shapley value for a feature j as follows,

Algorithm 1 Approximation of Shapley Estimation for single feature value

Input: Number of iteration M, observation x, feature index j, data matrix X and machine learning model f.
Output: Shapley value for j^{th} feature.

1. For $iteration = 1, 2, ..., M$
   a) Draw random instance z from data matrix X.
   b) Choose a random permutation $\mathfrak{o}$ of feature values.
   c) Reorder instance x as $x_\mathfrak{o}$: $x_\mathfrak{o} = (x_1, ..., x_j, ..., x_p)$
   d) Reorder instance z as $z_\mathfrak{o}$: $z_\mathfrak{o} = (z_1, ..., z_j, ..., z_p)$
   e) Construct two new instances:
      i. with j: $x_{+j} = (x_1, x_2, ..., x_{j-1}, x_j, z_{j+1}, ..., z_p)$
      ii. without j : $x_{-j} = (x_1, x_2, ..., x_{j-1}, z_j, z_{j+1}, ..., z_p)$
   f) Compute marginal contribution : $\phi^m_j = \hat{f}(x_{+j}) - \hat{f}(x_{-j})$
2. Compute Shapley value as the average: $\phi_j(x) = \frac{1}{M} \sum_{m=1}^{M} \phi^m_j$

SHAP framework

The **SH**apley **A**dditive ex**P**lanation (SHAP) framework, which was introduced by Lundberg et al.[2] in 2017, utilised the Shapley values for feature attribution in the tree-based framework[14].

The authors presented the Shapley value for a feature attribution problem in equation 2.5, where ϕ_i is the feature attribution value for feature i.

- In the equation 2.5, S is feature subset of the data $\mathfrak{Ds}$, while the F refers to the set of all the features in $\mathfrak{Ds}$, here $S \subseteq F$. For example, if $F = \{A, B, C\}$ then S can be $\{A\}$, $\{B\}$, $\{C\}$, $\{A, B\}$, $\{B, C\}$, $\{A, C\}$ or $\{A, B, C\}$.

$$\phi_i = \sum_{S \subseteq F \setminus \{i\}} \frac{|S|!(|F| - |S| - 1)!}{|F|!} \cdot \left(f_{S \cup \{i\}}(x_{S \cup \{i\}}) - f_S(x_S) \right) \tag{2.5}$$

- f is linear model, where $f(x) = \sum_{j=1}^{|F|} w_j x_j + b$ here $b = \phi_0(f, x)$ is the bias; w_j is the weight corresponding to j^{th} variable (feature).
- f_S is the model with features of S while excluding the feature i. If $S = \{A, B\}$ and $i = \{C\}$ then f_S has features $\{A, B\}$.
- $f_{S \cup \{i\}}$ is the model with features of S and including the feature i. Using the previous example, $f_{S \cup \{i\}}$ has features $\{A, B, C\}$.
- $x_{S \cup \{i\}}$ is a data point in $\mathfrak{Ds}$ considering features in S and the feature i, and x_S refers to the same data point with features in S but excluding the feature i.

According to Lundberg and Lee (2019, p.3), the SHAP framework satisfies three important properties[14],

- *Local Accuracy* states that the output function can be described as the sum of the feature attribution [14].
- *Missingness* property states that missing features attribute to zero value (no importance)[14].
- *Consistency* property states, "changing a model, so a feature has a larger impact on the model will never decrease the attribution assigned to that feature"[14].

Tree SHAP

Lundberg and Lee (2019) proposed the Tree SHAP algorithm, which utilised the decision-based ensemble model to calculate the feature attribution values[14]. The authors implemented this framework for the nutritional health-related data HANES[1].

- The authors trained an XGBoost model using the aforementioned dataset to model the log odds of mortality. The features are visualised in the order of their global importance using the SHAP (ϕ_i) value in the figure 2.1 [14].
- According to the authors, each dot in the horizontal feature line represents a person's health indicators; furthermore, every individual has a corresponding dot in each horizontal feature line[14].

[1]HANES: Health and Nutrition Examination Survey; `https://www.cdc.gov/nchs/data/series/sr_01/sr01_010a.pdf`

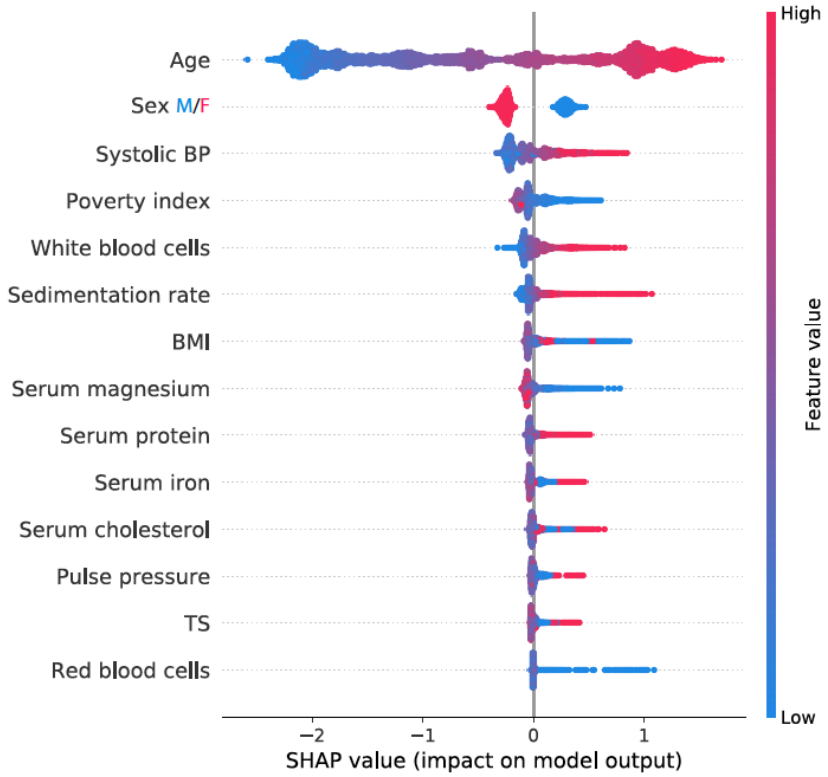


Figure 2.1: SHAP summary plot of mortality data from taken from Lundberg and Lee (2019, p.8)

- The colours from blue to red in the vertical scale correspond to the feature magnitude from lower to higher. The multiple observations describing the same SHAP value are plotted vertically on top, which explains the higher density of dots at certain places.
- In figure 2.1, *Age* is the most important factor affecting mortality; here, older individuals are at more risk of death than younger individuals. Furthermore, the lower value of *Red blood cells* has a higher impact on an individual's death.

We use this framework to identify the most important features of the basketball team performance dataset.

2.3 Bayesian Inference

This section provides the necessary background to address aim (2) (from section 2.1): Identifying the location of regime changes in dataset $\mathcal{D}$.

Gelman et al. (1995, p.32) described Bayesian Inference as "the process of fitting a probability model to a set of data and summarising the result by a probability distribution on the parameter of the model and unobserved quantities such as prediction for new observations."[15]. The authors denoted y to be observed data, θ to be *parameters* of interest.

Bayes' Theorem

Efron (2013) presented Bayes' Theorem[2] for two events A and B, where $p(A)$ and $p(B)$ represent the probabilities of these individual events, $p(A|B)$ represent the probability of event A conditioned on event B; furthermore, the probability of event B conditioned on event A, $p(B|A)$ can be represented as follows assuming that event A has a nonzero probability ($p(A) \neq 0$) [18],

$$p(B|A) = \frac{p(A|B)p(B)}{p(A)} \tag{2.6}$$

Here the equation 2.6 is a rather generalised form of Bayes' theorem for two events. In the context of Bayesian inference, Gelman et al. (1995, p.52) defined a *joint probability distribution* for θ and y as a product of two densities referred to as *prior distribution* $p(\theta)$ and the *data distribution* $p(y|\theta)$ as :

$$p(\theta, y) = p(\theta)p(y|\theta) \tag{2.7}$$

Furthermore, the authors presented the posterior distribution $p(\theta|y)$ as ratio of joint distribution $p(\theta, y)$ from 2.7 and $p(y)$ as follow:

$$p(\theta|y) = \frac{p(\theta, y)}{p(y)} = \frac{p(y|\theta)p(\theta)}{p(y)} \tag{2.8}$$

Here $p(y) = \sum_\theta p(\theta)p(y|\theta)$ is described as the sum over all possible θs. Additionally, the authors provided a form of 2.8 where for a fixed y, the $p(y)$ does not depend on θ and can be considered to be constant, so, in that case, *unnormalised posterior density* $p(\theta|y)$ can be represented as :

$$p(\theta|y) \propto p(y|\theta)p(\theta) \tag{2.9}$$

We focus on the problem, where the intended aim is to model the regime shift within the data set $\mathcal{D}$, where the regime is an interval of time where the data is assumed to follow a unique probability distribution. This probability distribution is assumed to be modelled in a Bayesian manner, based on the equation 2.9 where the posterior distribution $p(\theta|y)$ can be approximated by finding model likelihood $p(y|\theta)$ and prior distribution $p(\theta)$.

Bendtsen (2016, p.10) used the Metropolis-Hastings MCMC algorithm to identify the location of regime change. According to Chib and Greenberg (1995, p.1), this algorithm was developed over the years, initially published in the works of Metropolis et al. (1953) and later generalised in the research by Hastings (1970).

Metropolis-Hastings Algorithm

Metropolis-Hastings algorithm is used to sample the posterior distribution of function when a closed-form expression of the posterior distribution is not easily available. According to Gelman et al. (1995, p.278), " *Metropolis-Hastings algorithm* is a general term for a family of Markov chain simulation methods that are useful for sampling from Bayesian posterior distribution."[15]. The authors presented the Metropolis-Hastings MCMC algorithm in Algorithm 2.

According to Gelman et al. (1995, 292), to reduce the effect of staring values, the first half of the simulated samples are regarded as the *burn-in* or *warm-up* samples, which are discarded from the posterior distribution and only the second half of the simulated samples are considered[15].

[2] According to Bolstad et al. (2016), the eponymous Bayes' theorem was defined by Reverend Thomas Bayes and was posthumously published in 1776 [16][17].

Algorithm 2 Metropolis-Hastings Algorithm

1. Draw a starting point θ^0, for which $p(\theta^0|y) > 0$, from a *starting distribution* $p_0(\theta)$
2. For $t = 1, 2, ...$
   a) Sample a proposal θ^* from a *proposal distribution* at time t, $J_t(\theta^*|\theta^{t-1})$.
   b) Calculate the ratio of the densities,

$$r = \frac{p(\theta^*|y) \cdot J_t(\theta^{t-1}|\theta^*)}{p(\theta^{t-1}|y) \cdot J_t(\theta^*|\theta^{t-1})} \tag{2.10}$$

   c) Set

$$\theta^t = \begin{cases} \theta^* \text{ with probability min}(r,1) \\ \theta^{t-1} \text{ otherwise.} \end{cases} \tag{2.11}$$

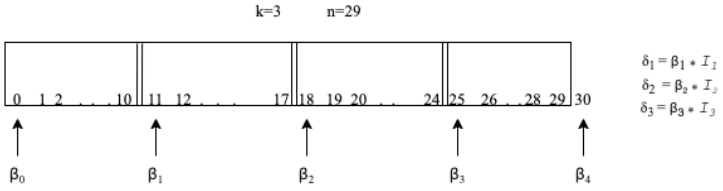


Figure 2.2: Example of split in dataset of n=29 and k=3

Bendtsen (2016, p.10) proposed that if the dataset $\mathcal{D}$ were to split at k unique positions from β_1, β_2, ...,β_k then posterior distribution is represented in the equation2.12[1]. The model likelihood is conditioned on the βs; here, βs are assumed to follow a *discrete uniform distribution*; for example $\mathcal{U}(a; b, c)$ is as discrete uniform distribution of a between b and c.

There are k splits into the data set $\mathcal{D}$ at starting indices δ_1, δ_2, ..., δ_k; where each split is defined based on the relationship between the indicator variable $\mathcal{I}$ and subset index β as follow: $\delta_i = \beta_i \cdot \mathcal{I}_i$, Indicator variables $\mathcal{I}$ are assumed to be Bernoulli distributed random variable with the value of either 0 or 1 (binary).

$$p(\beta_1, ..., \beta_k, \mathcal{I}_1, ..., \mathcal{I}_k|\mathcal{D}) \propto p(\mathcal{D}|\beta_1, ..., \beta_k, \mathcal{I}_1, ..., \mathcal{I}_k) \cdot \prod_{i=1}^{k} \mathcal{U}(\beta_i; \beta_{i-1}, \beta_{i+1}) \cdot \prod_{i=1}^{k} p(\mathcal{I}_i) \tag{2.12}$$

- β_1,β_2,...,β_k are indices that splits $\mathcal{D}$ into subsets, where $\beta_i \sim \mathcal{U}(\beta_i; \beta_{i-1}, \beta_{i+1})$, and $i = 1, 2, ..., k$. Here β_i is discrete uniform distribution from β_{i-1} to β_{i+1}.
- $\mathcal{I}_1$,$\mathcal{I}_2$,...,$\mathcal{I}_k$ are binary indicator variables with the probability distribution function as $\mathcal{I}_i \sim Bern(0.5)$ where $i = 1, 2, ..., k$.
- For the example in figure 2.2, the data-set $\mathcal{D}$ has $n = 29$ data points which has split at three indices $\beta_1 = 11, \beta_2 = 18$ and $\beta_3 = 25$ (here $k = 3$). furthermore, $\beta_0 = 0$ and $\beta_4 = 30$ ($\beta_{k+1} = n + 1$), the corresponding probabilities can be calculated using discrete uniform distribution[3].

[3]Probabilities of βs are, $p(\beta_1) = \frac{1}{\beta_2 - \beta_0} = \frac{1}{18}$, $p(\beta_2) = \frac{1}{\beta_3 - \beta_1} = \frac{1}{14}$, $p(\beta_3) = \frac{1}{\beta_4 - \beta_2} = \frac{1}{12}$

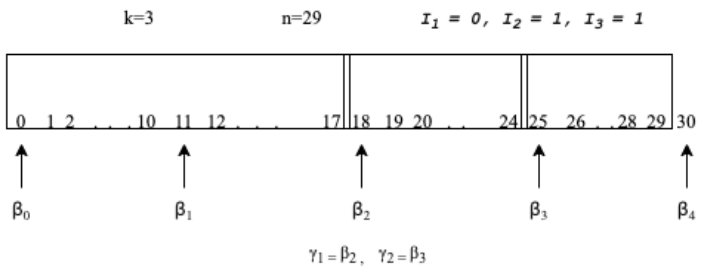


Figure 2.3: Example of the split in the dataset of n=29 with γs

Estimation Model Likelihood

The posterior distribution from the equation 2.12 can be estimated using the Metropolis Hasting MCMC method. Bendtsen (2016, p.10-11) re-parameterised it for a smaller subset of non-overlapping position in the dataset $\mathcal{D}$; these positions are defined as $\gamma_1, ..., \gamma_{k'}$ which are essentially *nonzero* δs. The author formally described $\{\gamma_1, ..., \gamma_{k'}\}$ as subset of $\{\gamma_1, ..., \gamma_k\}$ for which corresponding $\mathcal{I}_1, ..., \mathcal{I}_k$ are equal to one.

The example in figure 2.2, we consider β_1, β_2 and β_3, if the indicator variable $\mathcal{I}$ has nonzero values for $\mathcal{I}_2$ and $\mathcal{I}_3$ and zero value for $\mathcal{I}_1$, we end up with two γs as described in equation 2.13.

$$\begin{aligned}
\beta &= \{\beta_1,\ \beta_2,\ \beta_3\} \\
\mathcal{I} &= \{\mathcal{I}_1 = 0, \mathcal{I}_2 = 1, \mathcal{I}_3 = 1\} \\
\delta &= \{\delta_1 = \beta_1 \cdot \mathcal{I}_1 = 0, \delta_2 = \beta_2 \cdot \mathcal{I}_2 = \beta_2, \delta_3 = \beta_3 \cdot \mathcal{I}_3 = \beta_3\} \\
\gamma &= \{\gamma_1 = \beta_2, \gamma_2 = \beta_3\}
\end{aligned} \tag{2.13}$$

Furthermore, to estimate the likelihood of $p(\mathcal{D}|\beta_1, ..., \beta_k, \mathcal{I}_1, ..., \mathcal{I}_k)$ the model probabilistic distribution from equation 2.12, author re-wrote the equation 2.14 in terms of γs. We utilise the notation, $\mathcal{D}_{\gamma_i}^{\gamma_j - 1}$ to represented the observations of $\mathcal{D}$, starting from γ_i to $\gamma_j - 1$. For example, as the value of γ in equation 2.13 are $\gamma_1 = \beta_2 = 18$ and $\gamma_2 = \beta_3 = 25$. These two γs would create three subsets as described in the figure 2.3 can represent these three subsets, by $\mathcal{D}_1^{\gamma_1 - 1} = \mathcal{D}_1^{17}$, $\mathcal{D}_{\gamma_1}^{\gamma_2 - 1} = \mathcal{D}_{18}^{24}$ and $\mathcal{D}_{\gamma_2}^{n} = \mathcal{D}_{25}^{29}$.

$$\begin{aligned}
p(\mathcal{D}|\beta_1, ..., \beta_k, \mathcal{I}_1, ..., \mathcal{I}_k) &= p(\mathcal{D}_1^{\gamma_1 - 1}, \mathcal{D}_{\gamma_1}^{\gamma_2 - 1}, \mathcal{D}_{\gamma_2}^{\gamma_3 - 1}, ..., \mathcal{D}_{\gamma_{k'}}^{n} | \gamma_1, \gamma_2, ..., \gamma_{k'}) \\
&= p(\mathcal{D}_1^{\gamma_1 - 1} | \gamma_1) p(\mathcal{D}_{\gamma_1}^{\gamma_2 - 1} | \gamma_1, \gamma_2) p(\mathcal{D}_{\gamma_2}^{\gamma_3 - 1} | \gamma_2, \gamma_3) \cdots p(\mathcal{D}_{\gamma_{k'}}^{n} | \gamma_{k'})
\end{aligned} \tag{2.14}$$

In equation 2.14, the conditional probability distribution is re-written as the product of independent distributions ranging between two γs, where $k' \neq 0$.

Furthermore, the author rewrote the individual probability likelihood in equation 2.14 by conditioning each BN and summing over all the BN in the model, as described by marginal

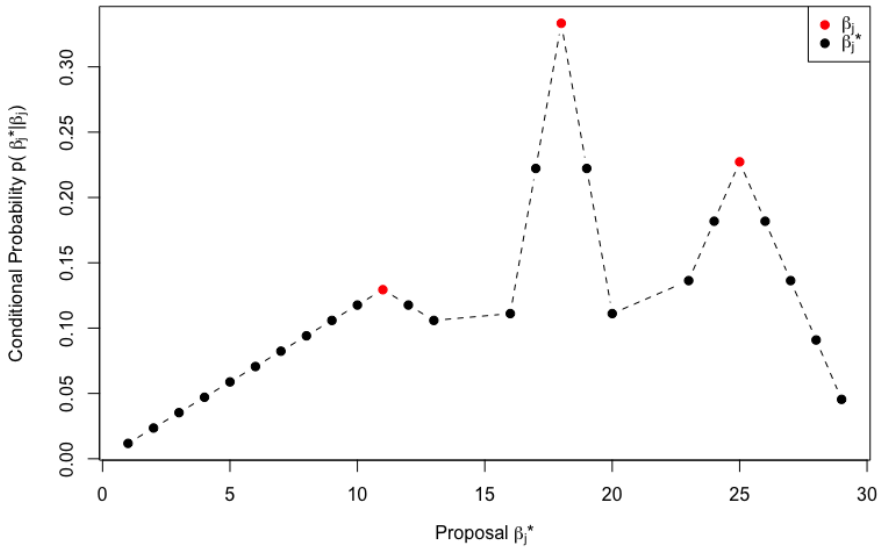


Figure 2.4: Distribution of Proposal β_j^* for the current position $\beta_1 = 11, \beta_2 = 18, \beta_3 = 25$

likelihood of the first two terms $p(\mathcal{D}_1^{\gamma_1-1}|\gamma_1)$ and $p(\mathcal{D}_{\gamma_1}^{\gamma_2-1}|\gamma_1, \gamma_2)$ as follows,

$$\begin{aligned} p(\mathcal{D}_1^{\gamma_1-1}|\gamma_1) &= \sum_{BN_i \in \mathbf{BN}} p(\mathcal{D}_1^{\gamma_1-1}|BN_i)p(BN_i|\gamma_1) \\ p(\mathcal{D}_{\gamma_1}^{\gamma_2-1}|\gamma_1, \gamma_2) &= \sum_{BN_i \in \mathbf{BN}} p(\mathcal{D}_{\gamma_1}^{\gamma_2-1}|BN_i)p(BN_i|\gamma_1, \gamma_2) \end{aligned} \tag{2.15}$$

Although it is possible to compute the terms in equation 2.15, there was an alternate solution provided by [1], where each term was approximated by a learning algorithm $\mathcal{L}$. For $k' > 1$ author presented the model likelihood in the equation 2.16.

$$\begin{aligned} p(\mathcal{D}|\beta_1, ..., \beta_k, \mathcal{I}_1, ..., \mathcal{I}_k) = p(\mathcal{D}_1^{\gamma_1-1}|\mathcal{L}(1, \gamma_1 - 1))p(\mathcal{D}_{\gamma_1}^{\gamma_2-1}|\mathcal{L}(\gamma_1, \gamma_2 - 1)) \cdots \\ p(\mathcal{D}_{\gamma_{k'}}^{n}|\mathcal{L}(\gamma_{k'}, n)); \qquad k' \in \mathbb{N}\backslash\{1\} \end{aligned} \tag{2.16}$$

- In equation 2.16, $\mathcal{L}(1, \gamma_1 - 1)$ refers to the BN that was trained on the data set $\mathcal{D}_1^{\gamma_1-1}$.
- Author had chosen the score-based hill climbing algorithm as the learning algorithm $\mathcal{L}$. We have included a detailed description of this algorithm in the following section.

The author described two special cases when $k' = 0$ and $k' = 1$.

$$\begin{aligned} p(\mathcal{D}|\beta_1, ..., \beta_k, \mathcal{I}_1, ..., \mathcal{I}_k) &= p(\mathcal{D}|\mathcal{L}(1, n)); & k' = 0 \\ p(\mathcal{D}|\beta_1, ..., \beta_k, \mathcal{I}_1, ..., \mathcal{I}_k) &= p(\mathcal{D}_1^{\gamma_1-1}|\mathcal{L}(1, \gamma_1 - 1))p(\mathcal{D}_{\gamma_1}^{n}|\mathcal{L}(\gamma_1, n)); & k' = 1 \end{aligned} \tag{2.17}$$

Formulation of Proposal Distribution

The proposal distribution utilised is defined in equation 2.21, the upper bound and the lower bound in this equation are defined in equation 2.19 and 2.18 accordingly. The marginalising constant $\mathcal{Z}$ is defined in equation 2.22.

$$\mathcal{LB}(\beta_j^*) = \begin{cases} 1, \text{if } j = 1 \\ \beta_j - \lfloor \frac{1}{2}(\beta_j - \beta_{j-1}) \rfloor + 1, \text{otherwise} \end{cases} \tag{2.18}$$

$$\mathcal{UB}(\beta_j^*) = \begin{cases} n, \text{if } j = k \\ \beta_j + \lfloor \frac{1}{2}(\beta_{j+1} - \beta_j) \rfloor - 1, \text{otherwise} \end{cases} \tag{2.19}$$

$$\mathcal{K} = \max(\beta_j - \mathcal{LB}(\beta_j^*), \mathcal{UB}(\beta_j^*) - \beta_j) \tag{2.20}$$

$$p(\beta_j^* = i|\beta_j) = \begin{cases} \frac{1+i-\beta_j+\mathcal{K}}{\mathcal{Z}} \text{ for } \mathcal{LB}(\beta_j^*) \leqslant i \leqslant \beta_j \\ \\ \frac{1-i+\beta_j+\mathcal{K}}{\mathcal{Z}} \text{ for } \beta_j < i \leqslant \mathcal{UB}(\beta_j^*) \end{cases} \tag{2.21}$$

$$\mathcal{Z} = \sum_{i=\mathcal{LB}(\beta_j^*)}^{\beta_j} (1 + i - \beta_j + \mathcal{K}) + \sum_{i=\beta_j+1}^{\mathcal{UB}(\beta_j^*)} (1 - i + \beta_j + \mathcal{K}) \tag{2.22}$$

- Here β_j is the current position value, and β_j^* is the proposed position value.
- The value of β_j^* is constrained by the lower bound $\mathcal{LB}$ (equation 2.19) and the upper bound $\mathcal{UB}$ (equation 2.18).
- Furthermore, $\mathcal{K}$ is placeholder constant (equation 2.20) used in equation 2.21.
- $\mathcal{Z}$ is a normalising constant (equation 2.22) also used in equation 2.21 to find the probability of proposal β_j^*.

The author described three primary constraints concerning the value of βs that are as follows[1]:

1. The value of β can only be within 1 to n and $\beta \in \mathbb{N}$.
2. If $i < j$ then $\beta_i < \beta_j$, as to avoid overlapping of subsets.
3. If the value of proposed β_j^* is bounded as $\mathcal{LB}(\beta_j^*) \leqslant \beta_j^* \leqslant \mathcal{UB}(\beta_j^*)$ then $p(\beta_j^*) > 0$.

Now we consider the running example described in figure 2.2, where $n = 29$ and $\beta_1 = 11, \beta_2 = 18, \beta_3 = 25$. The proposal distribution of β_j^* is described in 2.4, where the probability of proposing β_j^* away from the current β_j is decreasing, while the probability of proposing β_j^* close to current β_j is higher. In appendix (II), we describe the proposal distributions $p(\beta_j^*|\beta_j)$ for three different configurations of β_j.

2.4 Bayesian Network Formalism

In this section, we provide the background and terminology for the probabilistic graphical modelling, followed by the statistical definition of a Bayesian network. The primary goal of this section is to address aim(3) (from section 2.1): identifying the regimes and learning the corresponding independence models (Bayesian networks).

Nodes and Edges

According to Pearl (2000), a graph $\mathcal{G}$ primarily contains two building blocks, the set of vertices (V) (or nodes) and the set of edges (E)[19]. Vertices represent the random variables, while the edge connecting two vertices corresponds to a relation between them. In the diagram 2.5, the vertex **A** represents a random variable **A**, whereas the edge between the node **A** and **B** encodes the probabilistic relationship between these two random variables.

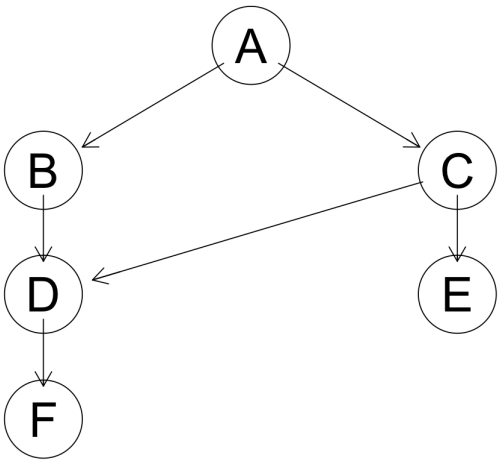


Figure 2.5: Example of A Bayesian Network (or DAG)

Classification of Nodes

Pearl (2000) classified nodes as a *parent*, *child* or *spouse* node based on the position of the node and the direction of adjacent edge(s) in the graph $\mathcal{G}$ [19].

- In the figure 2.5, node **A** is *parent* to nodes **B** and **C** as there are directed edges originating from node **A** and pointing towards the nodes **B** and **C**. Here the nodes **B** and **C** are the *child* nodes of **A**.
- Furthermore, if two or more nodes share a common child node can be termed as *spouse* nodes to one another; for example, nodes **B** and **C** have a common child node **D**, so **B** and **C** can be termed as the *spouse* nodes.
- In terms of hierarchy, nodes **A, B** and **D** are *ancestors* to node **F**, whereas all the nodes from **B** to **F** are *descendants* to node **A**.
- The author defined *family* as a set of nodes that consist of a node and all of its parents. In figure 2.5, sets {A}, {B, A}, {C, A}, {D, B, C}, {E, C}, and {F, D} are families.

Classification of Edges

According to Pearl (2000), an edge can be classified as *directed*, *bi-directional* or *undirected* by either using arrowhead(s) on edge or not including the arrowhead(s)[19] as follows,

- A *directed* edge is a link marked with one directional arrowhead; in this discourse, we have only utilised directed edges in a graph.
- A *bi-directional* edge is marked with two arrowheads on the link in opposite directions.
- A *undirected* edge is devoid of any arrowhead(s) on the link, such a graph is termed as *skeleton*[19].

Directed Acyclic Graph

According to Pearl (2000), a graph which is *directed* and *acyclic* is termed as a *directed acyclic graph* (or DAG). The author elaborated on the directed graph and acyclic graph as follows,

- A *directed* graph may contain a directed cycle (a loop formed by the nodes), but any of the nodes within the graph are not allowed to loop unto itself[19].

- Whereas, an *acyclic* graph cannot have the directed cycle and any node that may loop unto itself.

In this discourse, we only consider DAG to represent the Bayesian network (BN), the terms DAG and BN are used interchangeably.

Mathematical Formulation of BN

The research work by Scutari et al. (2019) described a Bayesian network as a probabilistic graphical model $\mathcal{G}$ that encodes the joint probability distribution over $\mathbf{m}$ ($m \in \mathbb{N}$) random variables such as $X_1, X_2, ..., X_m$, and the set of parameters Θ[20].

- The random variables $\{X_1, X_2, ..., X_m\}$ are collectively denoted by $\mathbf{X}$. For example, variable X_i (where $i = 1, 2, ..., m$) may correspond to an event or an attribute.

$$p(\mathbf{X}|\mathcal{G}, \Theta) = \prod_{i=1}^{m} p(X_i|Pa(X_i), \Theta_{X_i}) \tag{2.23}$$

- In the equation 2.23, the probabilistic likelihood of graph $\mathcal{G}$ is calculated as the multiplication of probability distribution of each node X_i conditioned on its parent node $Pa(X_i)$ and parameter Θ_{X_i}.

Discrete BN

Discrete BN was first introduced in the works of Heckerman et al.(1995)[21]; however, we utilise the definition of Discrete BN elaborated in the research work by Scutari et al.(2019)[20]. The aforementioned research work described a Discrete BN to have the multinomial random variable X_i as described in equation 2.24,

$$X_i|Pa(X_i) \sim Mul(\pi_{ik|j}),\ \pi_{ik|j} = p(X_i = e|Pa(X_i) = j) \tag{2.24}$$

- $\pi_{ik|j}$ refer to the conditional probabilities of random variable X_i to be in state e conditioned on the jth configuration of parent node of X_i (i.e. $Pa(X_i)$)[20].
- Furthermore, Scutari et al. (2019) described three important assumptions about data in the context of Discrete BN, which are listed as follows:
  1. The conditional probabilities $\pi_{ik|j}$ are always positive ($\pi_{ik|j} > 0$).
  2. The conditional probabilities $\pi_{ik|j}$ are independent for different parent configurations, this property is termed as *parameter independence* [21][20].
  3. The conditional probabilities $\pi_{ik|j}$ for different nodes are independent, this property is termed as *parameter modularity*[21][20].

Local Markov Property

A Bayesian network is modelled based on a local Markov property, which states that a child node is independent of all its non-descendants if conditioned on the parent node[22][23]. The probability distribution of the DAG can be represented by denoting random variables $X_1, X_2, ..., X_m$ as nodes in the following manner,

$$\begin{aligned} p(X_1, X_2, ..., X_m) &= p(X_1|Pa(X_1))p(X_2|Pa(X_2)) \cdots p(X_m|Pa(X_m)) \\ &= \prod_{i=1}^{m} p(X_i|Pa(X_i)) \end{aligned} \tag{2.25}$$

In the diagram 2.5, we scrutinise the model depicted using the nodes A, B, C, D, E and F. We can model this statistical dependence as follows:

$$p(A, B, C, D, E, F) = p(A)p(B|A)p(C|A)p(D|B, C)p(F|D)p(E|C) \tag{2.26}$$

2.5 Learning Bayesian Network

One crucial motivation behind Bayesian network modelling is to learn about the conditional distribution within the data $\mathcal{D}$ using a graph $\mathcal{G}$ with parameter Θ; this relation has been explained as joint distribution of $\mathcal{G}$ and Θ, while conditioning on $\mathcal{D}$, in equation 2.27 that is taken from [20].

$$p(\mathcal{G}, \Theta|\mathcal{D}) = p(\mathcal{G}|\mathcal{D}) \cdot p(\Theta|\mathcal{G}, \mathcal{D}) \tag{2.27}$$

The joint probability distribution $p(\mathcal{G}, \Theta|\mathcal{D})$ is further rewritten as the multiplication of conditional distribution of $\mathcal{G}$ and Θ. [4]

- Term $p(\Theta|\mathcal{G}, \mathcal{D})$ refers to parameter learning of Θ conditioned on the graph $\mathcal{G}$ that is learned using the data $\mathcal{D}$.
- The distribution $p(\mathcal{G}|\mathcal{D})$ is computed by structure learning of graph $\mathcal{G}$ for the data $\mathcal{D}$.

Structure Learning

Structure learning refers to the term $p(\mathcal{G}|\mathcal{D})$ from the equation 2.27, to find the closed form expression of $p(\mathcal{G}|\mathcal{D})$ is rewritten by applying Bayes' theorem in equation 2.28.

$$p(\mathcal{G}|\mathcal{D}) = \frac{p(\mathcal{D}|\mathcal{G})p(\mathcal{G})}{p(\mathcal{D})} \tag{2.28}$$

The term $p(\mathcal{D}|\mathcal{G})$ can be further conditioned on Θ according to [20] and re-written by integrating with respect to Θ.

$$\begin{aligned} p(\mathcal{D}|\mathcal{G}) &= \int p(\mathcal{D}|\mathcal{G}, \Theta)p(\Theta|\mathcal{G})d\Theta \\ &= \prod_{i=1}^{m} \int p(X_i|Pa(X_i), \Theta_{X_i})p(\Theta_{X_i}|Pa(X_i))d\Theta_{X_i} \end{aligned} \tag{2.29}$$

In the equation, 2.29, integration is performed on the local model corresponding to each node X_i conditioning on corresponding parameter Θ_{X_i} and the parent node $Pa(X_i)$. Now combining the equation 2.28 and 2.29, term $p(\mathcal{G}|\mathcal{D})$ can be re-written as follows,

$$p(\mathcal{G}|\mathcal{D}) \propto p(\mathcal{G}) \prod_{i=1}^{m} \int p(X_i|Pa(X_i), \Theta_{X_i})p(\Theta_{X_i}|Pa(X_i))d\Theta_{X_i} \tag{2.30}$$

Here the distribution $p(\mathcal{D})$ can be considered to be of a constant value, while $p(\mathcal{G})$ is usually considered to be a uniform distribution according to [20]. The structure learning algorithm could be divided into three classes: Constraint-Based Algorithms, Score-Based Algorithms, and Hybrid Algorithms [22].

Constraint-Based Algorithms

Constraint-based algorithms employ *conditional independence tests* on the undirected graph $\mathcal{G}$ nodes to find a Complete partially Directed Acyclic Graph (CPDAG). Some examples of constraint-based algorithms from the literature are Parent-Child (PC) and Grow Shrink Algorithm[22].

[4] Joint probability distribution of two events, $p(A, B) = p(A)p(B|A)$

Score-Based Algorithms

One highly used score-based structure learning method is the Hill Climbing Algorithm. As the method follows greedy search heuristics, there is a risk that it could converge at the local maxima instead of finding the global maxima. To mitigate the risk of converging at a sub-optimal structure, the algorithm is repeated for several thousand iterations to maximise the search space in our experiments. Scutari et al. (2019, p.6) described the hill climbing algorithm as follows 3[20], [5] In this discourse, we have utilised discrete BN and according to [20],

Algorithm 3 Hill Climbing Algorithm

Input: a data set $\mathcal{D}$ from $\mathbf{X}$, usually an empty DAG $\mathcal{G}$(but not necessarily) and a score function $Score(\mathcal{G}, \mathcal{D})$.
Output: the DAG $\mathcal{G}_{max}$ that maximises $Score(\mathcal{G}, \mathcal{D})$.

1. Compute the score of $\mathcal{G}$, $S_{\mathcal{G}} = Score(\mathcal{G}, \mathcal{D})$, and set $S_{max} = S_{\mathcal{G}}$ and $\mathcal{G}_{max} = \mathcal{G}$.
2. Repeat as long as S_{max} increases:
   a) for every possible arc addition, deletion or reversal in $\mathcal{G}_{max}$ resulting in a DAG:
      i. compute the score of the modified DAG $\mathcal{G}^*$, $S_{\mathcal{G}^*} = Score(\mathcal{G}^*, \mathcal{D})$:
      ii. if $S_{\mathcal{G}^*} > S_{max}$ and $S_{\mathcal{G}^*} > S_{\mathcal{G}}$, set $\mathcal{G} = \mathcal{G}^*$ and $S_{\mathcal{G}} = S_{\mathcal{G}^*}$
   b) if $S_{\mathcal{G}} > S_{max}$, set $S_{max} = S_{\mathcal{G}}$ and $\mathcal{G}_{max} = \mathcal{G}$.

it is possible to approximate the log-likelihood value (score) of $P(\mathcal{D}|\mathcal{G})$ for discrete BN, this is achieved using the conjugate Dirichlet prior as described in the equation 2.31.

$$\begin{aligned} BD(\mathcal{G}, \mathcal{D}; \alpha) &= \prod_{i=1}^{m} BD(X_i|Pa(X_i); \alpha_i) \\ &= \prod_{i=1}^{m} \prod_{j=1}^{q_i} \left[\frac{\Gamma(\alpha_{ij})}{\Gamma(\alpha_{ij} + n_{ij})} \prod_{k=1}^{r_i} \frac{\Gamma(\alpha_{ijk} + n_{ijk})}{\Gamma(\alpha_{ijk})} \right] \end{aligned} \tag{2.31}$$

- Here r_i refers to the number of classes of X_i, whereas q_i refers to the number of different configurations for $Pa(X_i)$.
- Furthermore, n_{ij} is marginal count over kth parent states ($n_{ij} = \sum_k n_{ijk}$). The Dirichlet distribution has α_{ijk} as a hyperparameter.
- If $\alpha_{ijk} = \frac{\alpha_i}{r_i q_i}$, then it is termed as **B**ayesian **D**irichlet **e**quivalent **u**niform (**BDe** or **BDeu**) score.

Hybrid Algorithms

A hybrid algorithm is a combination of constraint-based and score-based algorithms; one of the examples of these algorithms is MMHC (Max-Min Hill Climbing)[24]. The comparative experiments conducted by [20] to determine whether the hybrid algorithm outperforms constraint-based and score-based algorithms in terms of time complexity yielded negative results. Since it is essential to conduct scalable experiments in terms of the running time, the Hybrid algorithm may not be an ideal candidate for the simulations carried out here because of the result mentioned earlier.

[5] Scutari et al. (2019) noted that an initial DAG could be specified based on the expert belief by whitelisting and blacklisting the edges.

2.6 Statistical Framework of Gated Bayesian Network

Definition of Regime :

Bendtsen (2016, p.3) defined a regime as "a steady state of relationships between the variables in data, where these relationships may change between regimes"[1]. It is assumed that the data points within a regime follow independent and identical (i.i.d.) probability distribution. In the context of the basketball team, a regime is defined as a time interval during which a team exhibits similar characteristics in their performance. The Metropolis Hasting MCMC algorithm explained in section 2.3 aids in identifying these time intervals.

Regime Transition Structure

The author argued that the nonzero δs identified in the MCMC algorithm could indicate the structure consisting of a chain of regimes or a more complex transition with forward and backward regime transitions. We explain this using an example in figure 2.6, where three nonzero δs could yield a chain of regimes described by the transition structure $R_1 \rightarrow R_2 \rightarrow R_3 \rightarrow R_4$, where regimes go through a unidirectional transition. However, it is also possible that after $R_1 \rightarrow R_2$ transition, system $\mathcal{S}$ reverts back to regime R_1 instead of R_3 ($R_2 \rightarrow R_1$) and then transits to R_3 ($R_1 \rightarrow R_3$), so the regime transition $R_1 \rightarrow R_2 \rightarrow R_1 \rightarrow R_3$ is also possible. We refer to each possible regime transition structure as *hypothesis*.

Merging Subsets

We explain the merging of regime subsets using the example in figure 2.6, where three nonzero δs yield four subsets d_1, d_2, d_3 and d_4.

When Not to Merge ?

The idea behind implementing regime merging is to populate the pool of hypotheses choices while *avoiding merging regimes with consecutive subsets* to find a hypothesis with the highest marginal likelihood. Here, merging two consecutive subsets would be counter-intuitive as that would contradict the inference obtained by the Metropolis-Hastings MCMC algorithm and indicate that the same data-generating distribution generates the two consecutive subsets.

Hypotheses Population

- Hypothesis H_1 describes the unidirectional transition of regimes, where the regimes R_1, R_2, R_3 and R_4 correspond to subsets d_1, d_2, d_3 and d_4 respectively. The other hypotheses are built upon the changes in hypothesis H_1. R_1 does not have a parent regime, and R_4 does not have any child regime.
- Hypothesis H_2 is obtained by merging the R_1 with R_3 from H_1; the reason it is possible to merge these two is that they have data sets d_1 and d_3, which do not have consecutive indices (i.e. first index of d_3 is not immediately next to the last index of d_1). For H_2 we can visualise that R_1 has subsets d_1 and d_3 while R_2 had subset d_2, and R_3 (re-indexed from R_4 to R_3, as there is one less regime) has subset d_4. Now regime R_1 has regime R_2 and R_3 as child regimes, while regime R_2 has R_1 as a child regime and finally, R_3 does not have any child regime.
- Hypothesis H_3 is created by merging R_1 with R_4 (as d_1 and d_4 do not have consecutive indices). From H_1 the transition remains unchanged as $R_1 \rightarrow R_2 \rightarrow R_3$, however after R_3 the system $\mathcal{S}$ transitions back to R_1.

- Hypothesis H_4 is obtained by merging R_2 with R_4 from H_1. From H_1 the transition remains unchanged as $R_1 \rightarrow R_2 \rightarrow R_3$, however after R_3 the system $\mathcal{S}$ transitions back to R_2.
- Hypothesis H_5 is created by first merging R_1 with R_3 and then merging R_2 with R_4 from hypothesis H_1. Finally, the regime R_1 has subsets d_1 and d_3 while R_2 consist of subsets d_2 and d_4.

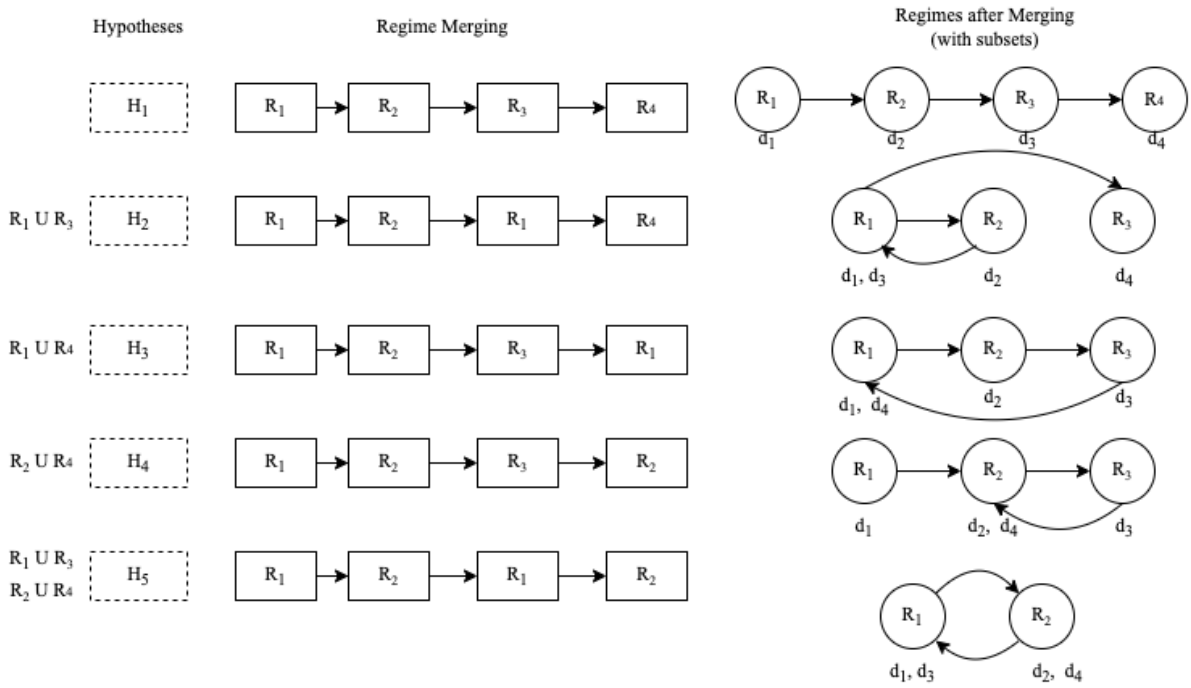


Figure 2.6: Regime Merging Hypotheses for k=3

Number of Possible Hypotheses

As the number of nonzero δs increases, the search space for possible hypotheses increases exponentially. The number of possible hypotheses follows a well-known integer sequence called the Bell numbers [6]. According to OEIS (The **O**nline **E**ncyclopedia of **I**nteger **S**equences), this sequence represents "the numbers of ways to partition set of n labelled elements". This sequence was mathematically described by a recurrence relationship in the works of Wilf(1990, p.39)[25], as follows,

$$b(n+1) = \sum_{j=0}^{n} \binom{n}{j} b(j); \qquad (n \geqslant 0; b(0) = 1). \tag{2.32}$$

[6]OEIS denotes Bell or Exponential Numbers by the sequence **A000110**: `https://oeis.org/A000110`

Using the equation 2.32, we calculate the first four bell numbers as follows,

$$\begin{aligned} b(1) &= \binom{0}{0} b(0) = 1; \\ b(2) &= \binom{1}{0} b(0) + \binom{1}{1} b(1) = 1 + 1 = 2 \\ b(3) &= \binom{2}{0} b(0) + \binom{2}{1} b(1) + \binom{2}{2} b(2) = 1 + 2 + 2 = 5 \\ b(4) &= \binom{3}{0} b(0) + \binom{3}{1} b(1) + \binom{3}{2} b(2) + \binom{3}{3} b(3) = 1 + 3 + 6 + 5 = 15 \end{aligned} \tag{2.33}$$

The results in equation2.33 suggests that if regime identification algorithm,

- fails to obtain any nonzero δ, it would indicate a single regime, which is made up of the entire dataset $\mathcal{D}$ and there would not be any other regime to transition to.
- finds one nonzero δ, then the dataset $\mathcal{D}$ is split into two subsets d_1 and d_2 which are part of regimes R_1 and R_2 accordingly, and possible regime transition structure is $R_1 \rightarrow R_2$.
- finds two nonzero δs, then $\mathcal{D}$ is divided into three subsets d_1, d_2 and d_3, and corresponding regimes are R_1, R_2 and R_3. The first trivial transition structure is $R_1 \rightarrow R_2 \rightarrow R_3$ which is made up of a chain of regimes. and The second transition structure can be obtained by merging regimes with non-adjacent subsets; in this case, these regimes are R_1 and R_3. This transition structure is $R_1 \rightarrow R_2 \rightarrow R_1$, which could also be represented as $R_1 \leftrightarrows R_2$.
- finds three nonzero δs then there are five possible regime transition structures as described in figure 2.6.
- finds four nonzero δs then there are fifteen possible regime transition structures that we represent in figure 5.3.

Gated Bayesian Network

The framework of the Gated Bayesian Network (or GBN) was initially defined in the works of Bendtsen and Peña [26]; the authors utilised the GBN framework for the algorithmic trading use case in [27].

Bendtsen (2016, p.6) defined GBN as a collection of BNs, where the BNs are connected via a *gate* that encodes a unidirectional relationship between parent and child BN citeMarcus2. In this framework, each gate has exactly one parent BN and one child BN[7]. According to the author, a gate is defined to have two states, either *active* or *inactive*.

Bendtsen (2016, p.6) also specified that "a BN could be both parent and child of different gates, and a BN can be a parent or a child of several gates"[1]. For example, we consider the GBN in figure 2.7, corresponding to H_2, where R_1 (and R_2) is the parent and child BN for different gates G_1 and G_2.

Configuration of GBN

Bendtsen (2016, p.15) described the overall idea of GBN configuration "as accepting a stream of observations, for each new observation decide the regime system currently is in"[1]. The author achieved this by defining a *TriggerLogic* for a gate that uses τ recent observation and compares whether the child BN to the gate explains the data better than the parent BN.

[7]This requirement was specific to the definition of GBN in [1] and not to GBN defined in [26], and [27].

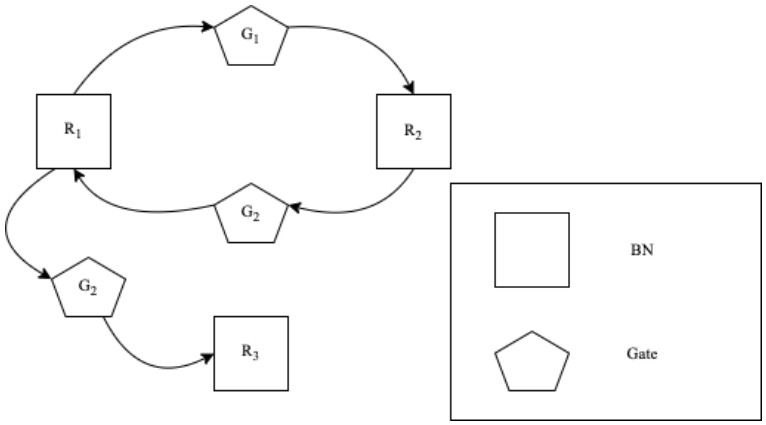


Figure 2.7: Gated Bayesian Network corresponding to Hypothesis H_2, for k=3

This is achieved by calculating the likelihood ratio for τ most recent observations (denoted by $\mathcal{O}_\tau$) for the child BN and the parent BN, and comparing with threshold θ. The author mathematically represented the Trigger Logic for gate G_i in equation 2.34.

$$TL(G_i) : \frac{p(\mathcal{O}_\tau | R_{child(G_i)})}{p(\mathcal{O}_\tau | R_{parent(G_i)})} > \theta \tag{2.34}$$

- Here, the term $R_{child(G_i)}$ refers to the child regime (or child BN) to the gate G_i, while the term $R_{parent(G_i)}$ refers to the parent regime(or parent BN) to the gate G_i.
- Term $p(\mathcal{O}_\tau | R_{child(G_i)})$ denotes the probabilistic likelihood for data points $\mathcal{O}_\tau$ conditioned on the child BN of gate G_i.
- Term $p(\mathcal{O}_\tau | R_{parent(G_i)})$ denotes the probabilistic likelihood for data points $\mathcal{O}_\tau$ conditioned on the parent BN of gate G_i.
- The probabilistic likelihood is calculated by computing the BDe score (described in section 2.6) of the corresponding Bayesian network. The likelihood ratio aids in estimating whether the data points in $\mathcal{O}_\tau$ are more likely to be a part of parent BN or child BN.

Example of Trigger Mechanism

In figure 2.7, the GBN is modelled based on the Hypothesis H_2 (for k=3) from figure 2.6, where the BNs (or Regimes) transitions take place in the following manner, $R_1 \rightarrow R_2 \rightarrow R_1 \rightarrow R_3$. Now, if we were to model trigger logic for gate G_1 for the transition $R_1 \rightarrow R_2$, the equation 2.34 can be re-written as equation 2.35.

$$TL(G_1) : \frac{p(\mathcal{O}_\tau | R_2)}{p(\mathcal{O}_\tau | R_1)} > \theta \tag{2.35}$$

- The gate G_1 is in a *inactive* state when the trigger logic in equation 2.35 is not satisfied, and in this case, the parent BN is considered to be *active* and child BN is in *inactive* state.
- The gate G_1 is *active* when the trigger logic in equation 2.35 is satisfied, where the parent BN considered *inactive* state, while the child BN is activated.

2.7 Hyper-parameter Tuning using Gaussian Processes

In this section, we discuss the necessary background to address the aim (5) (from section 2.1), where after learning the optimal regime transition structure and the corresponding GBN, we tune the hyperparameters associated with it.

The regime transition structure learned from the section 2.6 provides a basic institution about the transition in regime corresponding to each BN, where uni-directional gates connect these BNs. The system $\mathcal{S}$ transits from parent BN to child BN when a gate satisfies the trigger logic for some value of τ, and θ, where τ refers to $\mathcal{O}_\tau$ datastream (which consist of τ observations) and θ refers to a threshold value deciding *active* or *inactive* state of a gate.

Blackbox Optmisation

Bendtsen (2016, p.15) described that the task of choosing appropriate values for τ and θ (or $\Lambda = \{\tau, \theta\}$) required to optimise a computationally expensive *blackbox* function [8]; furthermore, the author proposed to use the Gaussian process (GP) to act as a proxy for the *blackbox* function [1]. Before providing more details about the optimisation problem, we formally describe the Gaussian process.

Gaussian Processes

Rasmussen and Williams (2006, p.2) described a Gaussian process as "a generalisation of the Gaussian probability distribution"[29]. This method utilises vital properties of a Gaussian distribution that any combination of normalisation, marginalisation, summation or conditioning operation over the Gaussian distributed random variables would yield the Gaussian distributed variables[30].

Definition

The authors formally defined a Gaussian process to be "a collection of random variables, any finite number of which have a joint Gaussian distribution"[29] in Rasmussen et al. (2006, p.13).

Mathematical Representation

A Gaussian process $\mathcal{GP}$ for a real process $f(x)$ is defined by a mean function $m(x)$ and covariance function $k(x, x')$, as described in equation 2.37.

$$\begin{aligned} f(x) &\sim \mathcal{GP}\big(m(x), k(x, x')\big) \\ m(x) &= E\big[f(x)\big] \\ k(x, x') &= E\big[(f(x) - m(x))(f(x') - m(x'))\big] \end{aligned} \tag{2.37}$$

- In the previous equation, x refers to the training point, while x' refers to the test point.
- $m(x)$ is defined as the expectation of function f with respect to x, and $k(x, x')$ is the covariance between $f(x)$ and $f(x')$.

[8] Alarie et al.(2021, p.1) described a general form of the optimisation problem as,

$$\max_{x \in \Omega} f(x) \tag{2.36}$$

- where Ω was defined as a feasible region ($x \in \Omega$), and $f : \Omega \to \mathbb{R} \cup \{\infty\}$[28].
- The authors described black box optimisation as the problem where "the structure of objective function f and/or the constraint defining the set Ω is unknown, unexploitable, or non-existent".

Gaussian Process Prediction with Noise-free Observations

Rasmussen and Williams (2006, p.15) defined the joint distribution of training point $\mathbf{f}$ and test point $\mathbf{f}_*$ for the noise-free scenario as follows;

$$\begin{bmatrix} \mathbf{f} \\ \mathbf{f}_* \end{bmatrix} \sim \mathcal{N}\left(0, \begin{bmatrix} K & K_*^T \\ K_* & K_{**} \end{bmatrix}\right) \tag{2.38}$$

In the equation 2.38, there are $\mathbf{n}$ training points, and $\mathbf{n}_*$ test points.

- K is a $\mathbf{n} \times \mathbf{n}$ matrix with variance evaluated between training point pairs. K_{**} is a matrix with variance evaluated between test point pairs; it has dimensions $\mathbf{n}_* \times \mathbf{n}_*$.
- K_* denotes $\mathbf{n} \times \mathbf{n}_*$ matrix with covariance evaluated between all training and test points. K_*^T is the transpose of matrix K_* with dimension $\mathbf{n}_* \times \mathbf{n}$.

The authors derived the noise-free predictive distribution as described in equation 2.39.

$$\mathbf{f}_* | X_*, X, \mathbf{f} \sim \mathcal{N}(K_* K^{-1} \mathbf{f}, K_{**} - K_* K^{-1} K_*^T) \tag{2.39}$$

In the abovementioned equation, X and X_* are matrices representing training and test inputs, while $\mathbf{f}$ and $\mathbf{f}_*$ are function values corresponding to X and X_*.

Specification for the Use Case

Bendtsen (2016, p.16) defined the *objective function* f as the accuracy of the prediction for the current regime. The author argued that although it is possible to evaluate f at a certain Λ (where $\Lambda = \{\tau, \theta\}$) value, there is no close-form expression for f. The author introduced the random variable g as a proxy for f and placed a Gaussian prior on g; furthermore, specified that "for each Λ pair, $g(\Lambda)$ is treated as Gaussian random variable, where $f(\Lambda)$ is a realisation of the random variable $g(\Lambda)$" from Bendtsen (2016, p.16)[1]. The equation 2.37 can be rewritten for the problem at hand in the following way,

$$g(\Lambda) \sim \mathcal{GP}(m(\Lambda), k(\Lambda, \Lambda')) \tag{2.40}$$

In the equation, 2.40, the $m(\Lambda)$ is the mean function, which is assumed to be zero[9], and covariance kernel $k(\Lambda, \Lambda')$.

$$\begin{aligned} m(\Lambda) &= 0 \\ k(\Lambda, \Lambda') &= \sigma_g^2 \cdot \exp\left(-\frac{||\Lambda - \Lambda'||^2}{2 \cdot l^2}\right) \end{aligned} \tag{2.41}$$

In the equation 2.41, $k(\Lambda, \Lambda')$ describes a *squared exponential*(SE) kernel, where σ_g^2 is the variance in g, while l represents the length scale.

Bendtsen (2016, p.17) used the following notation to describe Gaussian parameterisation in equation 2.42, where after collecting $\{\Lambda_{1:i}, f_{1:i}\}$, the posterior distribution for new input Λ_{i+1} is represented by g_{i+1}[1].

- $\Lambda_{1:i}$ represents Λ value at different i, $f_{1:i}$ represents f evaluated for $\Lambda_{1:i}$, such that $f_j = f(\Lambda_j)$. Furthermore, g_j represents the random variable $g(\Lambda_j)$.
- K is the kernel matrix describing variance of $\Lambda_{1:i}$. K_* is the kernel matrix that computes the co-variance between Λ_{i+1} and $\Lambda_{1:i}$.

[9] Bendtsen (2016, p.16) argued that this value could be nonzero based on the prior knowledge[1].

- K_*^T is transpose of kernel matrix K_*. K_{**} is the kernel matrix describing variance of Λ_{i+1}.

$$
\begin{aligned}
&\begin{bmatrix} g_{1:i} \\ g_{i+1} \end{bmatrix} \sim \mathcal{N}\left(0, \begin{bmatrix} K & K_*^T \\ K_* & K_{**} \end{bmatrix}\right) \\
&K = \begin{bmatrix} k(\Lambda_1, \Lambda_1) & \cdots & k(\Lambda_1, \Lambda_i) \\ \vdots & \ddots & \vdots \\ k(\Lambda_i, \Lambda_1) & \cdots & k(\Lambda_i, \Lambda_i) \end{bmatrix} \\
&K_* = \begin{bmatrix} k(\Lambda_{i+1}, \Lambda_1) & \cdots & k(\Lambda_{i+1}, \Lambda_i) \end{bmatrix} \\
&K_{**} = \begin{bmatrix} k(\Lambda_{i+1}, \Lambda_{i+1}) \end{bmatrix} \\
&p(g_{i+1} | \{\Lambda_{i:i}, f_{1:i}\}) = \mathcal{N}(\mu_i(\Lambda_{i+1}), \sigma_i^2(\Lambda_{i+1})) \\
&\mu_i(\Lambda_{i+1}) = K_* K^{-1} f_{1:i} \\
&\sigma_i^2(\Lambda_{i+1}) = K_{**} - K_* K^{-1} K_*^T
\end{aligned}
\tag{2.42}
$$

Acquisition Function

According to Bendtsen (2016, p.17), "the Gaussian process allows encoding prior belief about the objective function and sampling the objective function allows updating the posterior over objective function". The author proposed the Upper Confidence Bound ($\mathcal{UCB}$) criterion as an *acquisition function*, which was initially described in the works of Brochu et al. (2009, p.14)[31].

$$\mathcal{UCB}(\Lambda) = \mu(\Lambda) + \eta \cdot \sigma(\Lambda) \tag{2.43}$$

The $\mathcal{UCB}$ criterion is expressed in equation 2.43, where parameter η represents the trade-off between exploration and exploitation; a higher value of η would lead to more exploration by proposing Λ_{i+1} with higher $\sigma(\Lambda)$, whereas a lower value would lead to exploitation by proposing Λ_{i+1} with a higher $\mu(\Lambda)$. Here $\mu(\Lambda)$ represents the mean of $g(\Lambda)$ while the standard is the standard deviation of $g(\Lambda)$[1].

Evaluation of Accuracy

Bendtsen (2016, p.17) described the objective of the optimisation to find a Λ (or (τ, θ) pair) that maximises the accuracy f in [1]. The author defined a GBN to be *correct* if, after processing the new observation in the stream of data $\mathcal{O}_\tau$, the active BN is the same as the true BN that generated that observation and it is *incorrect* otherwise[1].

3 Basketball Data Descriptions

3.1 An Overview of Basketball

A commercial basketball game in North America is divided into four quarters of twelve minutes each. Each team has five players at a time on the court; a player can earn points for the team through a 3-point shot, 2-point shot and free throw.

- The figure 3.1 provides a basic schema of a basketball court with the above-mentioned five positions in the court[1]. The basketball court is rectangular in shape and is split into two half-courts by the Middle Line.

[1]The image is taken by Photo by Kindel Media and was published on Pexels: https://www.pexels.com/photo/aerial-view-of-basketball-court-9739470/, the original image was edited to include annotation. We have adhered to the licensing protocol listed on https://www.pexels.com/license/.

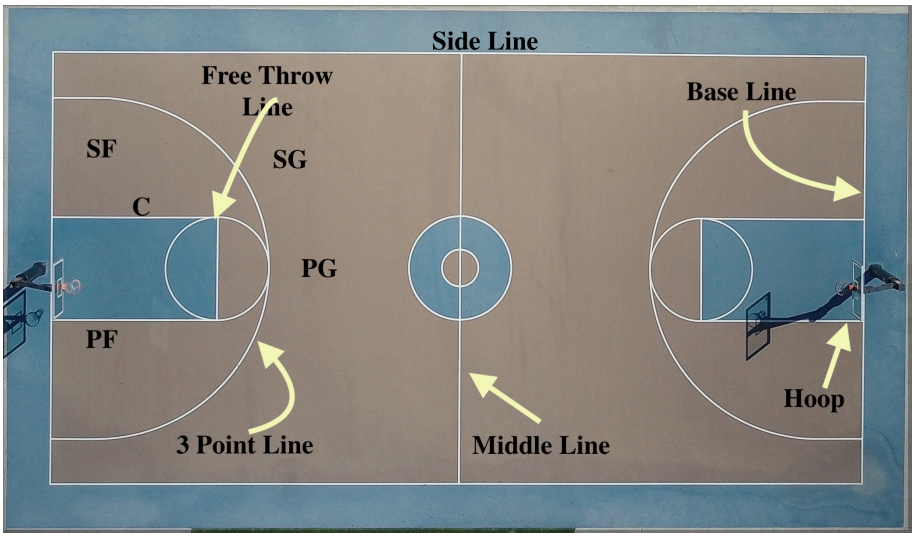


Figure 3.1: Diagram of a Basketball Court taken from pexel.com

- Each team attempts to score a goal in the opponent's basket without losing control of the ball. The high-scoring team of the two teams wins the game.
- A successful goal attempted outside the 3-point line is awarded three points, while the goal scored inside the 3-point line is worth two points.
- A free throw attempt is awarded to a player if the situation ensues when the opponent player commits a foul if that shot is converted into a goal, the player obtains one point for the team[32]. The free throw is attempted from the free throw line, described in figure 3.1.
- Traditionally in basketball, there are five positions: Point Guard (**PG**), Shooting Guard (**SG**), Center (**C**), Small Forward (**SF**) and Power Forward (**PF**). Each position has a set of responsibilities; for example, a player playing as a centre is responsible for gaining control of the ball near the hoop.

Box Score in Professional Basketball League

Box Score is an indicator of team performance in a game [33]. The basic and advanced Box Scores consists of performance statistic of each player in the team as described in figures 3.2 and 3.3 accordingly[2].

	Basic Box Score Stats																	
Starters	**MP**	**FG**	**FGA**	**FG%**	**3P**	**3PA**	**3P%**	**FT**	**FTA**	**FT%**	**ORB**	**DRB**	**TRB**	**AST**	**STL**	**BLK**	**TOV**	**PF**	**PTS**
Dave Greenwood	38:00	2	8	.250	0	0		2	2	1.000	0	7	7	0	0	0	3	5	6
Quintin Dailey	36:00	9	15	.600	0	0		9	11	.818	0	1	1	3	1	0	5	4	27
Orlando Woolridge	33:00	4	8	.500	0	0		4	8	.500	3	5	8	1	0	1	2	5	12
Dave Corzine	28:00	3	10	.300	0	0		8	8	1.000	2	4	6	4	1	0	2	5	14
Ennis Whatley	24:00	3	6	.500	0	0		0	0		0	3	3	8	2	0	4	0	6
Reserves	**MP**	**FG**	**FGA**	**FG%**	**3P**	**3PA**	**3P%**	**FT**	**FTA**	**FT%**	**ORB**	**DRB**	**TRB**	**AST**	**STL**	**BLK**	**TOV**	**PF**	**PTS**
Mitchell Wiggins	36:00	9	20	.450	0	0		8	9	.889	2	7	9	2	2	0	2	3	26
Jawann Oldham	16:00	3	4	.750	0	0		2	2	1.000	0	6	6	0	0	0	2	6	8
Reggie Theus	16:00	2	4	.500	0	0		1	4	.250	1	0	1	4	1	0	1	1	5
Sidney Green	13:00	0	3	.000	0	0		0	0		0	0	0	1	0	0	0	2	0
Team Totals	**240**	**35**	**78**	**.449**	**0**	**0**		**34**	**44**	**.773**	**8**	**33**	**41**	**23**	**7**	**1**	**21**	**31**	**104**

Figure 3.2: Basic Box Score of Chicago Bulls against New Jersey Nets on October 29, 1983, from basketball-reference website

	Advanced Box Score Stats														
Starters	**MP**	**TS%**	**eFG%**	**3PAr**	**FTr**	**ORB%**	**DRB%**	**TRB%**	**AST%**	**STL%**	**BLK%**	**TOV%**	**USG%**	**ORtg**	**DRtg**
Dave Greenwood	38:00	.338	.250	.000	.250	0.0	21.6	11.6	0.0	0.0	0.0	25.3	12.7	53	93
Quintin Dailey	36:00	.680	.600	.000	.733	0.0	3.3	1.8	17.4	1.3	0.0	20.1	28.0	110	94
Orlando Woolridge	33:00	.521	.500	.000	1.000	12.5	17.7	15.3	5.0	0.0	2.0	14.8	16.6	100	92
Dave Corzine	28:00	.518	.300	.000	.800	9.8	16.7	13.5	23.0	1.6	0.0	12.9	22.5	110	90
Ennis Whatley	24:00	.500	.500	.000	.000	0.0	14.6	7.9	55.2	3.8	0.0	40.0	16.9	81	88
Reserves	**MP**	**TS%**	**eFG%**	**3PAr**	**FTr**	**ORB%**	**DRB%**	**TRB%**	**AST%**	**STL%**	**BLK%**	**TOV%**	**USG%**	**ORtg**	**DRtg**
Mitchell Wiggins	36:00	.543	.450	.000	.450	7.6	22.8	15.8	11.6	2.5	0.0	7.7	29.2	110	88
Jawann Oldham	16:00	.820	.750	.000	.500	0.0	43.9	23.7	0.0	0.0	0.0	29.1	17.4	106	88
Reggie Theus	16:00	.434	.500	.000	1.000	8.6	0.0	3.9	41.4	2.8	0.0	14.8	17.1	96	91
Sidney Green	13:00	.000	.000	.000	.000	0.0	0.0	0.0	10.5	0.0	0.0	0.0	9.4	18	96
Team Totals	**240**	**.534**	**.449**	**.000**	**.564**	**22.2**	**80.5**	**53.2**	**65.7**	**6.6**	**1.4**	**17.7**	**100.0**	**97.6**	**91.0**

Figure 3.3: Advanced Box Score of Chicago Bulls against New Jersey Nets on October 29, 1983, from basketball-reference website

The term *Minutes Played*, often denoted by the acronym `MP`, is a critical measure that accounts for the time a player is on-court. The team's on-court playing style could be analysed by

[2]The example of box-score is from the webpage `https://www.basketball-reference.com/boxscores/198310290CHI.html`

concentrating on the players who have spent a relatively higher amount of time on the court in a regular season.

3.2 Game Logs in Professional Basketball League

The basic and advanced Game Logs are designed based on the basic and advanced box scores by aggregating player performance statistics, which is represented as **Team Totals** in figure 3.2 and 3.3. Here the examples of the Basic and Advanced Game Logs are described in the figures 3.4 and 3.7 accordingly.

Basic Game Log Statistics

This section provides the basic understanding of the terms utilised in Basic Gamelog, from figure 3.4.

					Team														Opponent												
Date	Opp	W/L	Tm	Opp	FG	FGA	FG%	3P	3PA	3P%	FT	FTA	FT%	ORB	TRB	AST	STL	BLK	TOV	PF	FG	FGA	FG%	3P	3PA	3P%	FT	FTA	FT%	ORB	TRB	AST	STL	BLK	TOV	PF
1983-10-29	NJN	W	104	97	35	78	.449	0	0		34	44	.773	8	41	23	7	1	21	31	36	77	.468	0	6	.000	25	41	.610	8	36	20	7	5	22	33

Figure 3.4: Regular Game Log of Chicago Bulls against New Jersey Nets on October 29, 1983, from basketball-reference website

- `Tm` denotes the points earned by the team, while `Opp` refers to the points scored by the opponent.
- `FG` denotes the successful field goal worth two or three points in the game. `FGA` refers to the number of field goals attempted in the game. Usually, the higher number of field goal attempts would correspond to higher field goals made; this phenomenon can be observed in the heat map describing the correlation between `FG` and `FGA`, which is 0.91. The heat map is created using the Chicago Bull's basic game log-based performance statistics in 38 seasons.[3] in figure 3.5. Furthermore, the percentage of successful field goals from the attempted field goals is represented as FG% or `FGPer`.
- The goal scored outside the 3-point line earns the team 3 points, which is denoted by `3P`, and the number of 3-point attempts by the team is denoted by `3PA`. Furthermore, the percentage of successful 3 points goals made out of all 3-point attempts is represented by 3P% or `3PPer`. In figure 3.6, we have visualised how the trends in 3 points attempts have changed with the season, as not all the seasons had the same number of matches[4], we have chosen to plot 3 point attempts per game for the each of the 38 seasons. We also include points per game in each season for comparison. It is evident that although the number of 3-point attempts increased each season, the points per game in the seasons have fluctuated but have not seen an increasing trend. These points to changes in playing style over the years, where players choose to shoot 3-point goals, possibly because it is more attractive to the spectators.
- Free throw is awarded to a player in case the opponent player commits a violation; if the free throw is made successfully, it is worth one point. A free throw is attempted from the free throw line by the player; this term is described as `FT`. Furthermore, the number of free throw attempts is denoted by `FTA`. The percentage of successful free throws out of the attempted free throws is described by the feature FT% or `FTPer`.

[3]The heat map is created using the data from the web link:`https://www.basketball-reference.com/teams/CHI/stats_basic_totals.html`.

[4]There have been few seasons in NBA history, where the organisation shortened the number of games in the regular season less than 82, which are often termed as Lockout Season

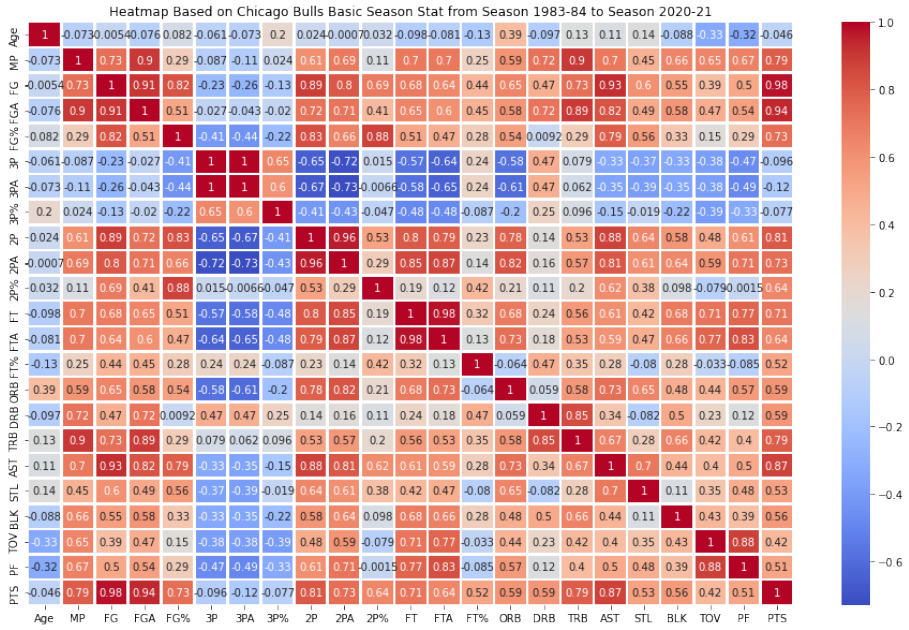


Figure 3.5: Heatmap based on the Correlation between the Features from Basic Stat (season wise)

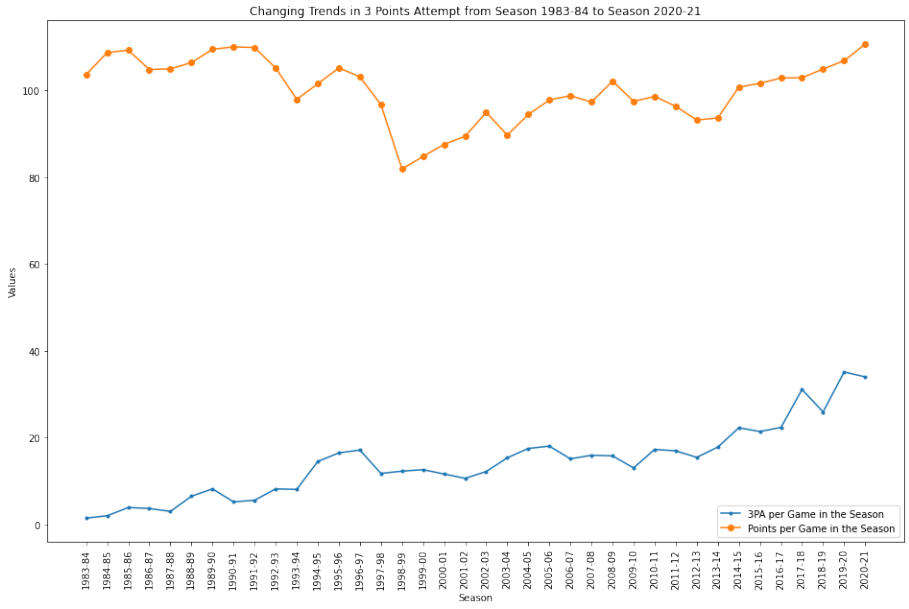


Figure 3.6: Comparison of 3PA per Game and Points per Game, from the Season 1983-84 to Season 2020-21

- The skill to regain control of the ball is called rebounding; based on whether a player or the team is on the offence or defence while scoring the rebound, it is termed either *Offensive* or *Defensive Rebounding*, which are denoted by `ORB` and `DRB` accordingly in the figure 3.4, and *Total Rebounding* refers to the sum of `ORB` and `DRB`, which is denoted by `TRB`.

- According to NBA[5], *Assist* refers to the act of passing the ball to a player that is converted to a successful goal, the number of assists is denoted by `AST`.
- *Turnover* is an event when the team or a player loses control of the ball to the opponent. The team or a player who gains control of the ball is awarded a *Steal*. The frequency of *Turnover* and *Steal* is denoted by `TOV` and `STL` accordingly[32].
- When the attempt to score the goal is prevented by a defending player, the team or a player is awarded a *Block*, the frequency of this event is denoted by `BLK`.
- *Personal Foul* (or `PF`) denotes the number of fouls committed by the player or a team[32].

Advanced Game Log Statistics

					Advanced									Offensive Four Factors				Defensive Four Factors			
Date	Opp	W/L	Tm	Opp	ORtg	DRtg	Pace	FTr	3PAr	TS%	TRB%	AST%	STL%	BLK%	eFG%	TOV%	ORB%	FT/FGA	eFG%	TOV%	DRB%	FT/FGA
1983-10-29	NJN	W	104	97	97.6	91.0	106.6	.564	.000	.534	53.2	65.7	6.6	1.4	.449	17.7	22.2	.436	.468	18.8	80.5	.325

Figure 3.7: Advanced Game Log of Chicago Bulls against New Jersey Nets on October 29, 1983, from basketball-reference website

- Advanced Game Log also has `Tm` denoting team score and `Opp` denoting opponent's score.
- According to Oliver(2004, p.154), *Offensive Rating* (`ORtg`) for a player (or a team) refers to the points produced per 100 possessions of a ball, while *Defensive Rating* (`DRtg`) for a player (or a team) refers to the points conceded per 100 possessions of a ball by an opponent.
- *Pace* factor refers to the number of possessions by a team per 48 minutes.
- *Free Throw Rate* (`Ftr`) refers to the number of free throw attempts per field goal attempt.
- *3-Point Attempt Rate*(`3PAr`) refers to the number of 3-point attempts per field goal attempts.
- *True Shooting Percentage* (`TS%`) is the shooting efficiency that considers free throws, and 2 and 3-point field goals and is calculated as follows,

$$\texttt{TS\%} = 100 \cdot \frac{\texttt{PTS}}{2\,(\texttt{FGA} + 0.44 \cdot \texttt{FTA})} \tag{3.1}$$

- *Total Rebounding Percentage* (`TRB%`) is the estimated total rebounds seized by a player during a game, in percentage.
- *Assist Percentage* (`AST%`) Estimated number of 2-point field goals assisted by a player, in percentage.
- *Steal Percentage* (`STL%`) is the estimated number of opponent's ball possessions regained by the team, in percentage.
- *Block Percentage* (`BLK%`) is the number of opponent's 2-point field goal attempts successfully thwarted, in percentage.

Dean Oliver initially defined four factors in his book *Basketball on Paper* in 2004; according to the author, a team has to control four aspects of a game,

[5] `https://www.nba.com/stats/help/glossary`

1. Shooting percentage
2. Committing turnovers
3. Rebounding
4. Moving the ball near the free throw line

These four aspects can be observed by four statistical measures divided into offensive and defensive counterparts, as follows :

Offensive Four Factors:

- `eFG%` stands for adjusted field goal percentage by the team, which acknowledges that 3 point goal is 50% more valuable than a 2-point goal.
- `TOV%` denotes the percentage of turnover committed by the team, resulting in losing the ball.
- `ORB%` portends offensive rebounding percentage per 100 plays by the team.
- `FT/FGA` expresses the team's free throws per field goal attempt.

Defensive Four Factors:

- `eFG%` indicates the opponent's adjusted field goal percentage.
- `TOV%` signifies how likely the opponents were to lose the ball in percentage.
- `DRB%` symbolises the opponent team's rebounding percentage per 100 plays.
- `FT/FGA` is the ratio of the opponent's free throw to field goal attempts.

Dean Oliver also noted that not all the four factors are equally important for the game outcome, [6] on a scale of 1 to 10, the Shooting percentage has a weight of 10; Turnover has a weight of 5 or 6; rebounding has a weight of 4 or 5 and moving the ball near the free throw line has a weight of 2 or 3.

3.3 Processed Data Features

External Factors

Table 3.1: Processed Data Features

Type of Features	Features

External Influences	`Home_Game`
	`Opponent_PlayOff`
	`Days_Between_Games`

Player's Individual Contribution	`Continuing_Players_WS`
	`Incoming_Players_WS`
	`Leaving_Players_WS`

Team's Overall Performance	`Team_Prospect`
	`WinsInLast15`
	`WinsInLast10`
	`WinsInLast5`

[6] the author determined that using the program called Roboscout:`http://www.rawbw.com/~deano/`

Home Game Advantage:

In National Basketball Association, each team out of 30 teams usually plays for 82 games in a regular season[7], where each team plays 41 games at home-court and the rest on the road. The phenomenon of home-court advantage was studied by the National Collegiate Basketball Association (NCAA) and National Basketball Association (NBA); the observations were that the teams played better at home (González Dos Santo et al., 2022)[35], and the long air travel had adverse effects on player performance (Huyghe et al., 2018)[36].

In this project, `Home_Game` is the Boolean parameter with value `TRUE` indicating the game is played at home court, while the value `FALSE` indicates the away game.

Opponent's Playoff Appearance:

National Basketball Association as a sports league has a fascinating history; based on the scrutiny of seasons 1980-81 to season 2020-21, three teams (Boston Celtics, Chicago Bulls, Los Angeles Lakers and San Antonio Spurs) out of 30 teams have collectively won more than 60% of the National Basketball Association championship titles. Overall in the last 41 seasons, 17 teams out of these 30 have never won the championship, see table A.4; furthermore, out of the 30 current teams and excluding teams which were not operational from the season 1980-81, we narrow down our scrutiny on such 23 teams which played in all 41 seasons.

The opponent's playoff appearance could indicate the opposing team's relative strength and form during the regular season. In this study, the parameter `Opponent_PlayOff` is a Boolean variable with a value, `TRUE` if the opponent appeared in the playoff and `FALSE` if the opponent did not make it to the playoff in the particular season.

Days Between Games

In a regular NBA season, a player usually has to cope with a schedule that requires a lot of travel across North America. Players often have to appear in back-to-back games, which could be hectic and impact their performance. According to work by Esteves et al.(2021, p.5), back-to-back games hurt players' performance; furthermore, the players who had at least one rest day between the games positively affected the game outcome.

In this study, we consider `Days_Between_Games` to be a parameter with value 0 for back-to-back games, and values $1, 2, 3, ...$ based on the days between consecutive games the according to NBA schedule fixture in each season.

Player's Individual Contribution

According to the basketball-reference website[8], Win Share was initially introduced for the game of baseball by Bill James. Dean Oliver utilised the Win Share to attribute credit to individual players based on their offensive and defensive performance (Oliver, 2004)[37].

Players Win-Share

According to the research work by González Dos Santo et al. (2022, p.7), the win share of players who are continuing from the previous season was one of the important predictors of the game outcome. The authors had classified three subsets of players in their analysis,

[7]There are several exceptions to this in the last 40 years when the regular seasons were cut short due to either contractual lockouts [34] or SARS-CoV-2 pandemic `https://www.nba.com/news/coronavirus-pandemic-causes-nba-suspend-season`

[8]Win Share:`https://www.basketball-reference.com/about/ws.html`

Table 3.2: Intervals of `Team_Prospect` as Performance Indicator

`Team_Prospect`	Performance
[-2.43, -0.913]	Poor
(-0.913, -0.262]	Below Average
(-0.262, 0.294]	Average
(0.294, 0.859]	Above Average
(0.859, 2.66]	Extra Ordinary

1. players who are continuing from the previous season.
2. players who are newly signed in the team.
3. players who are leaving in the next season.

In this project, we consider the subsets above of players, assuming that the subset **1** and **2** are mutually exclusive. In contrast, subsets **2** and **3** (and subsets **1** & **3**) are not mutually exclusive.

Team's Overall Performance

We quantify a team's overall performance using two measures; the first is described by the `Team_Prospect` while the second corresponds to the number of wins in the last (5/10/15) games.

Team_Prospect

`Team_Prospect` is computed using the running mean of score difference in the last 20 games. This difference is further discretised into five intervals as described in table 3.2. We provide more detail about this parameter in section 4.1.

Momentum Effect

According to Arkes and Martinez (2011, p.1), *Momentum Effect* refers to the situation in sports when winning in the previous games positively impacts the winning in the next game[38]. Authors studied this effect for the NBA and concluded that the last games' results affect the outcome of the next match (Arkes and Martinez, 2011, p.15).

In this project, the *Momentum Effect* is modelled by the parameter that accounts for the number of wins in the last 5, 10 and 15 games.

4 Methods Description

4.1 Data Collection Procedures

In our experiments, we have relied upon the data available on website `basketball-reference`. We have utilised the information about basic and advanced game logs that documented the team's performance markers in each game in the regular season. We first elaborate on the procedure followed to collect the raw data and then the procedure to obtain the processed data in the following sections.

Procedure for Obtaining Raw Data

Table 4.1: Raw Data Features, from `basketball-reference.com`

Type of Features	Features
Regular BoxScore	`Tm, Opp, FG, FGA, FG%, 3P, 3PA, 3P%, FT, FTA, FT%, ORB, DRB, TRB, AST, STL, BLK, TOV, PF`
Advanced BoxScore	`Tm, Opp, ORtg, DRtg, Pace, FTr, 3PAr, TS%, TRB%, AST%, STL%, BLK%, eFG%, TOV%, ORB%, FT/FGA` (Offensive Four Factors) `eFG%, TOV%, DRB%, FT/FGA` (Defensive Four Factors)

We manually scrapped the raw data with the features described in the table 4.1 for Chicago Bulls from season 1983-84 to season 2020-21. During that time period, we are considering 3041 regular season games.

Procedure for Obtaining Processed Data

As part of our processed data, we have utilised three types of features described in the table 3.1.

External Influences

The first category is described as a set of features called *External Influences*; that consists of three features,

- The first feature `Home_Game` describes whether the game was played at home or away; this parameter was processed based on the value in **Sep** (Separator) column, which has the value @ or whitespace. The marker @ signified the away game (not played at home court), while the whitespace denoted the home game. We further encode the home game with Boolean value `TRUE` while the away game with `FALSE` value.
- The second feature `Opponent_PlayOff` describes whether the opponent team appeared in the playoffs[1] during that particular season, this parameter was further encoded with Boolean value `TRUE` if the opponent team appeared in the playoffs or with `FALSE` if the opponent did not proceed further than the regular season.
- Finally, the third feature `Days_Between_Games` describes the number of days between two consecutive games. This information was accessed and aggregated by finding the difference in days between the current game and the previous game based on the **date** marker in the game log.

Player's Individual Contribution

The second category describes the features based on players' contributions to a game regarding Win Share. We accessed box scores for each match and retrieved the `MP` (Minutes Played) feature value for a particular player in the game. Later on, We access the team roster for each season, where we can find the player's Win Share weighted over 48 minutes. Using these two parameters, we can find out the player-specific Win Share contribution in each game[2]. Furthermore, we combine the cohort of players' Win Share (WS) in the following manner,

- Firstly, we combine the Win Share of the players who were part of the team in the previous season, that is, the summation of the Win Share of all such players in a game and store it as `Continuing_Players_WS`.
- The second feature combines the Win Share of all the new players in the current season; this information is stored in the feature `Incoming_Players_WS`.
- The third feature combines the Win Share of all the players who are leaving the team in the next seasons and stores the value in `Leaving_Players_WS`.

Team's Overall Performance

The third category of features describes the team's overall performance. Here we have four features, out of which the first three store information about the number of wins in the last 5, 10 and 15 games.

- These features were populated by counting the number of wins in the past n games ($n = 5, 10$ and 15).
- The last feature `Team_Prospect` is calculated using the running mean[3] of score difference in a game as described in equation4.1,

$$\delta_t^{20} = \mu_t^{20}(\texttt{Tm}) - \mu_t^{20}(\texttt{Opp}) \tag{4.1}$$

  - Here $\mu_t^{20}(\texttt{Tm})$ denotes the running mean of the team's Score considering twenty values up to time point t.

[1] This information was accessed by visiting season-wise records of NBA playoff Standings, e.g. for season 1983-84 this information can be found on `https://www.basketball-reference.com/playoffs/NBA_1984_standings.html`

[2] For this particular task, we have utilised python package `basketball-reference-scraper` version 1.0.31 `https://pypi.org/project/basketball-reference-scraper/`. This scraper version has stability issues, but we eliminated the bugs during data cleaning for this project.

[3] This idea came to fruition from brainstorming meeting with the main supervisor Jose M. Peña on 11-10-2022

- Whereas $\mu_t^{20}(\texttt{Opp})$ denotes the running mean of the opponent's score considering twenty values up to time point t.
- Furthermore, δ_t^{20} is the difference between the above-mentioned running means at time point t.

All the non-binary variables are normalised using `scale` function before further pre-processing. We intend to use the package `bnlearn` to learn the structure of Bayesian Network, for this reason[4] we discretise all non-binary variables to five `quantile` intervals using `discretize` function.

SHAP Analysis for Feature Selection

In this section, we outline the framework of SHAP Analysis for feature selection[5], the basic idea is to construct tree-based XGBoost (Extreme Gradient Boosting) model using the python package `xgb` and then perform SHAP analysis utilising `shap` python package. We list out the step taken to perform this analysis as follows,

- Firstly, the data set is split into train and validation data set using `train_test_split` function from `sklearn` package with `test_size=0.25`, that is train/test split to be 75%/25%.
- Secondly the train and validation data set are converted into `Dmatrix` format for the XGBoost algorithm.
- Thirdly the model parameters for `xgb.train` function are displayed in the table 4.2 where we have set these parameters randomly; however, we experimented with different values of the parameters `max_depth` (maximum depth of tree) as described in table 4.3.

Table 4.2: `xgb.train` Model with Randomly Set Parameter Values

Parameters	Description	Value
`objective`	Regression with Squared Error Loss	`reg:squarederror`
`eval_metric`	Root Mean Square Error (RMSE)	`rmse`
`num_boost_round`	Number of Boosting Rounds	2000
`early_stopping_rounds`	Early Stopping Rounds	15
`eta`	Learning Rate	0.1
`SEED`	Random Number for Reproducibility	1728

Table 4.3: `xgb.train` Model Experiments to Optimise for `max_depth`

Experiment No.	`max_depth`	`training-rmse`	`validation-rmse`
1	5	1.54754	2.2466
2	10	0.009944	2.3087
3	15	0.026034	2.3674
4	20	0.000876	2.3678
5	25	0.000451	2.3994

[4] `bnlearn` package does not support continuous variable, for this reason, we have converted the cleaned dataset into intervals.

[5] The inspiration for this analysis is attributed to the technical blog about GPU Accelerated SHAP by Parul Pandey published on NVIDIA Developer website, on 05-10-2022 `https://developer.nvidia.com/blog/explain-your-machine-learning-model-predictions-with-gpu-accelerated-shap/`

- As described in table 4.3, we observed that increasing the `max_depth` lowers `training-rmse`, however `validation-rmse` increases, indicating that the model with higher `max_depth` has less generalisation, and performs poorly on validation data. The model with `max_depth=10` performs relatively well with relatively lower `training-rmse` and `validation-rmse`, we utilise the model `max_depth=10` for SHAP analysis.
- Lastly, we utilise `TreeExplainer` function from `shap` package for the tuned model to find the SHAP values using `shap_values` function. Furthermore, the `summary_plot` of SHAP values would aid in visualising the most important features in a beeswarm plot.

4.2 Regime Identification using Metropolis-Hastings MCMC Sampling

This section addresses the method utilised for regime identification, which can be further divided into three parts[6] visualised in the figure 4.1, here we have used the notation of the flow-chart (described in figure A.1) to denote the operations.

- First, we present the parameter initialisation for the algorithm in section **A: Parameter Intialisation**.
- Secondly, the current and proposal values for prior distribution are calculated along with the model likelihood using a structure learning algorithm, which would aid in estimating posterior distribution for each case as described in section **B: From Prior Distribution to Posterior Distribution**.
- Thirdly, we utilise the Metropolis-Hastings algorithm to determine whether the proposal values are accepted based on the ratio, which we elaborate on in section **C: Metropolis-Hastings Simulation**.

A: Parameter Initialisation

The parameter Initialisation subsection takes into account steps [A,1] to [A,7] as indicated in the figure4.1. We list out these steps as follows,

- In step [A,2], the pre-processed data set is obtained based on the method described in the section 4.1 along with the feature identified by the SHAP Analysis. The number of maximum allowable splits `k` is specified[7], along with the number of MCMC iterations.
- In step [A,3], the current value of parameter β is initialised evenly across the k points[8], Indicator variable $\mathcal{I}$ is set 1 (active) for across k values[9] Finally the δ values are set to be multiplication of βs and $\mathcal{I}$s[10]. This step is assigned iteration index 0.
- In step [A,4], the containers for the current value of δ and $\mathcal{I}$ are created; these containers would be assigned values in step [C,7].
- The next step [A,5] is to increase the iteration index by 1. Furthermore, the proposal distribution matrix is created based on the current value of β.

[6] It is based on the pseudo-code outlined in appendix A.1 in the literature[1]

[7] This value enforces the number of possible subsets in the dataset

[8] For example, if the value of $k = 4$ and $n = 1000$ then β is initialised as $\lfloor \frac{(1:k)\cdot n}{k+1} \rceil$ that equates to $[200, 400, 600, 800]$, these values represent the indices from which the subsets are generated.

[9] Using the same example, for k=4 and n=1000, $\mathcal{I}$ value would equate to be $[1, 1, 1, 1]$.

[10] Continuing the above example, δ values would be $[200, 400, 600, 800]$

- In step [A,6], the sampling for the proposal value of β is performed using the probability distribution from step [A,5]. Furthermore, the indicator variable $\mathcal{I}$ follow the Bernoulli distribution with a probability of 0.5.
- Finally, the step [A,7] proposal δs are calculated by multiplying proposal βs and $\mathcal{I}$s in the previous step.

B: From Prior Distribution to Posterior Distribution

Now we shift the attention towards part **B** from the figure 4.1, here the steps [B,1] to [B,8] aid in estimating the posterior distribution of proposal βs and $\mathcal{I}$s, whereas these steps have counterparts on the other-side from [B,9] to [B,16] that aids to estimate the posterior distribution of current βs and Is.

- In steps [B,1] and [B,2], the proposal values of βs and Is are broadcast to step [B,6], which performs a summation of the input distributions.
- In step [B,3], the original data set is split into subsets according to proposal δ values. Furthermore, in step [B,4], the Bayesian structure learning framework is utilised by means of finding a local maximum using hill climbing algorithm[11]. Finally, in step [B,5], the sum of the marginal likelihood value is calculated based on each Bayesian network score.
- The step [B,7] refers to aggregating the prior distribution of proposal β and $\mathcal{I}$, along with the marginal log-likelihood values obtained from the structure learning. Finally, this aggregated value is utilised to calculate the acceptance ratio in step [C,2]
- The step [B,8] represents the transition probability of proposal distribution from the current distribution of β and $\mathcal{I}$ parameters.
- Steps [B,9] to [B,16] are identical in their implementation to that described above, with the change of current parameters instead of proposal parameters.
- The final step [B,17] aggregates the following probability distributions,
  - Posterior likelihood of proposal distribution from step [B,7].
  - Posterior likelihood of current distribution from step [B,16].
  - Transition probabilities of going from current to proposal distribution from step [B,8].
  - Transition probabilities of going from proposal to current distribution from step [B,9].

C: Metropolis-Hastings Simulation

The third part of regime identification is indicated in figure 4.1 by part **C**.

- Here, the initial step [C,1] is to sample a number from a uniform distribution between 0 and 1, which is compared with the acceptance ratio (r) calculated in step [C,2].
- In step [C,3] compares whether the ratio is greater than the uniform number generated from step [C,1] if the ratio is greater than proposed β and $\mathcal{I}$ are set as current β, and $\mathcal{I}$ (from step [C,4]) and the next step is [C,5].

[11] Here we utilise the `hc` function from `bnlearn` library, furthermore, as a Network Score we opt for Bayesian Dirichlet equivalent score (`bde`) in accordance with choice in the literature[1]

- If the ratio is not greater, then the current values of the parameter remain unchanged, and step [C,5] is executed, which compares whether the current iteration is greater than 50% of total iteration then, in step [C,7], we store the current $\mathcal{I}$ and δs values in the container defined in step [A,4]. If the current iteration is not greater than 50% of the total iteration, then we do not store the current $\mathcal{I}$s and δs; the reason for this practice is to remove the burn-in samples obtained by the algorithm.
- The step [C,6] evaluates whether the current iteration has reached the final number; if not, then we jump back to step [A,5] and repeat the above-mentioned steps. When the algorithm has completed the final iteration, the container for δ is evaluated for computing the rounded marginal mean (step [C,8]), which is further sorted in ascending order in step [C,9]. The final values of sorted unique positions are returned.

4.3 Optimisation of Regime Transition Structure

The previous section addressed the regime identification method that yielded the nonzero δs, informing about how to split the dataset $\mathcal{D}$ into subsets for the individual regimes. This section provides insight into how to use the knowledge about nonzero δs to further obtain possible regime transition structures. Furthermore, the search is conducted for an optimum structure from the pool of candidate structures with the highest marginal likelihood while aggregating over all the regimes within a structure as described in figure 4.2.

D1: Partitioning of Original Dataset

The first step of the regime optimisation is to utilise the pre-processed data set and the location of nonzero δs and split the data set into subsets (partitions) as described in the diagram 4.3(step [D1,2] and [D1,3]). The next step is to construct a storage mechanism that keeps track of the data in each subset and the mapping between the parent and child regimes, as this would be utilised in combining the non-adjacent subsets (step [D1,4]).

E: Collapsing the Possible Structures

To find out all the possible structures, the nested loops are utilised to decide whether two subsets are adjacent or not, as described in part **E** of diagram 4.3. Two subsets are adjacent if they contain consecutive indices, e.g. subsets of indices $\{1, 2, ..., 13\}$ and $\{14, ..., 20\}$ are consecutive subsets, while $\{1, 2, ..., 13\}$ and $\{16, ..., 20\}$ are not adjacent.

The outer loop described in step [E,2] would loop over several structures defined in step [D1,4]. Every time a new parent regime and its child regime are found, it is updated in the original container, followed by the recursion call in step [E,6].

F: Combining Non-Adjacent Structures

When two regimes i and j have subsets that are non-adjacent, then it is possible to merge data of j^{th} regime to i^{th} regime and remove the j^{th} regime. As the regimes are set numerically from 1 to k', one has to re-index other regimes.

- Furthermore, if the regime j has child regime(s), then this child regime(s) would have regime i as a parent after j merges with i
- If the regime j is the child of another regime (s), then those regimes would now have i as a child (from step [F,1] to step [F,5]).
- Finally, the updated regimes with new parent-child mappings are updated in the container in step [E,6], followed by the recursion call.

D2: Highest Marginal Likelihood Structure

The final step in optimising regime transition structure is to identify a structure with the highest marginal likelihood from all the possible structures as described in part **D2** of the diagram 4.3.

4.4 Gated Bayesian Network Parameter Optimisation Using Gaussian Processes

In the previous section, we have explained the optimum regime transition structure of the GBN; in this section, we elaborate on tuning the hyperparameters utilised for the gates.

We assume that each *gate* in the GBN has the same threshold (θ) value and look-back window length(τ), as it is computationally infeasible to tune each gate individually. We search for an optimum value for the (τ,θ) pair. We have created 100 synthetic data sets (or Test Sets) based on the optimum transition structure identified in section 4.3.

The hyper-parameter optimisation process commences with learning the optimum structure (in step [2]), where each regime is defined by a Bayesian Network and connected by gates according to the optimum structure.

In step [3], the initial value of the look-back window (τ) and gating threshold (θ) are selected based on an educated guess[12], these initial values would act as the first test points for the Gaussian processes regression implemented in subsequent steps.

The value of coefficient η describes the trade-off between exploration and exploitation in the *Upper Bound Confidence* ($\mathcal{UCB}$) criterion[13], that aids to select a new set of (τ,θ). If the value of η is higher, we could explore larger search space quickly but overshoot and miss the global maxima. If the value of η is lower, then we could exploit it by proposing a new (τ,θ) with less bias, but a large amount of search space would remain unexplored.

How to Create Synthetic Dataset ?

The synthetic data sets (test sets) are created in (step [4]) in such a way that they do not consist of adjacent subsets; for example, if we assume that for $k = 3$, the hypothesis H_4 described in figure **??** represents the optimum transition structure, then the synthetic test dataset is created by first sampling from regime R_1 which has subset d_1, secondly sampling from regime R_2 which has subsets d_2 and d_4 and finally sampling from regime R_3 which has subset d_3 and concatenating all these subsets to create one test dataset, this process is repeated 100 times.

In step [5], function **g** is used as a *proxy function* for accuracy **f**, then we assume that **g** has multivariate Gaussian distribution with infinite random variables. Here the idea is to propose pair of (τ, θ) from the test set pairs that would have higher accuracy (steps [6],[7]), furthermore the (τ, θ) pair is proposed according to an *acquisition function* described as the $\mathcal{UCB}$ criterion that was elaborated earlier.

Steps [7] to [18] refer to evaluating for accuracy using the test sets. We have utilised 100 test data sets, from which we randomly sample five data sets[14]. Now for each subset within

[12] In the experiments initial pair of (τ,θ) is chosen to be (5, 1.5). Here we assume that at a time, we only have access to the data stream of length 5 from the test set.

[13] $\mathcal{UCB}$:- $\mu(\tau, \theta) + \eta \cdot \sigma(\tau, \theta)$ from equation 2.43

[14] We only utilise five datasets at a time to minimise the time to evaluate the accuracy.

the five datasets, we consider data stream $\mathcal{O}$, where we first skip over the initial $\tau - 1$ observation (step [12]) and take into account τ number of observations at a time. For example, if the subset has length 100 and τ has value 10, we first consider the observations at indices in the range 11 to 20, then 12 to 21 and so forth up to 91 to 100.

Now in this data stream of length τ, for each observation, we evaluate whether the observation corresponds to the True BN it was generated from or not.

If the true BN is triggered, we append the value 1 to the accuracy container, and if not, the value 0 is appended (steps [14], [15] and [16]). To determine which BN triggers, the logarithmic ratio is calculated according to equation 2.34 and compared with the threshold value θ.

Lastly, when the specified number of iterations have passed, we would parameterise the GBN with (τ, θ) pair, which corresponds to the highest accuracy $\mathbf{f}$. Finally, the GBN with optimal hyperparameter is returned.

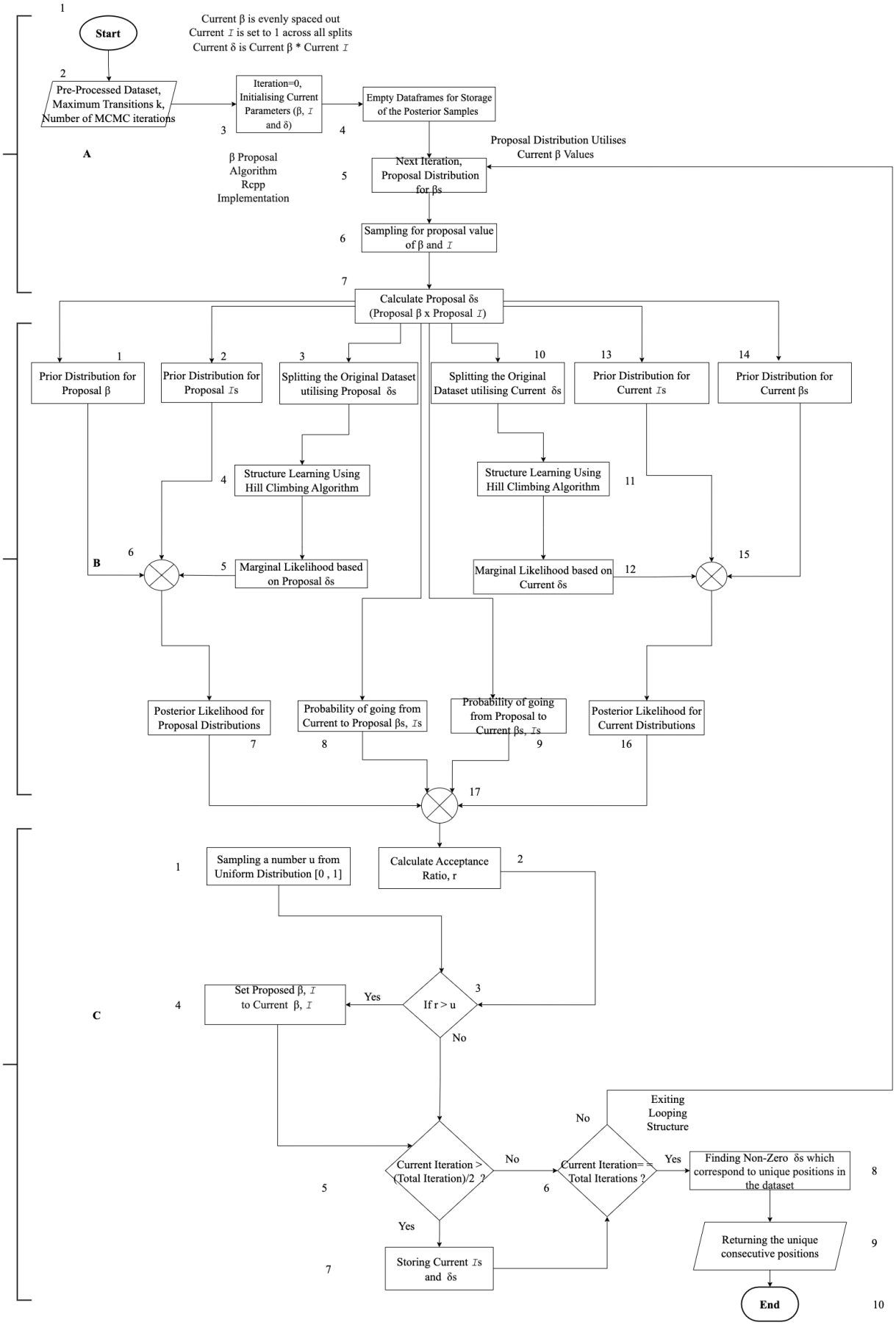


Figure 4.1: Regime Identification using Metropolis-Hastings MCMC Algorithm

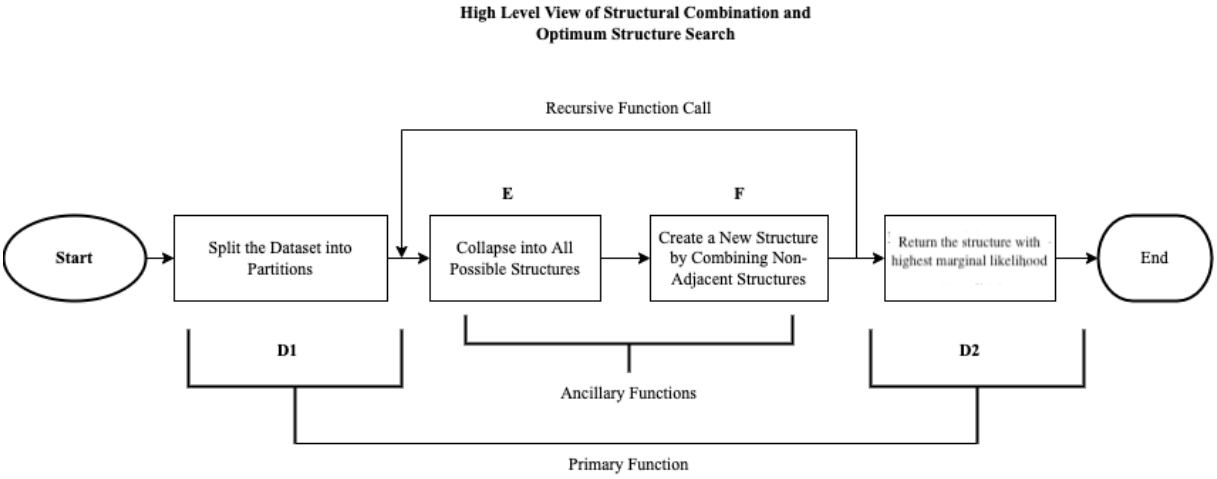


Figure 4.2: Bird's Eye View of Structural Combination of Regimes to Identify the Optimal Structure

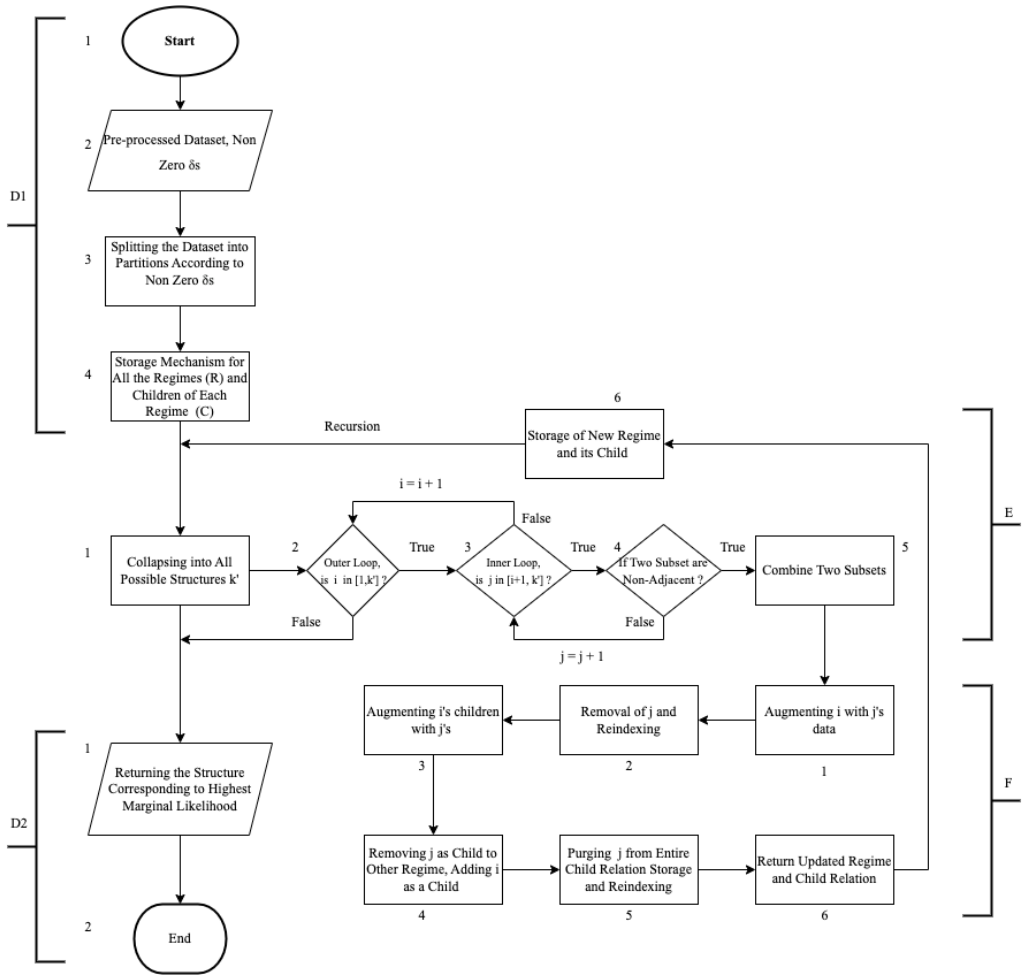


Figure 4.3: Granular View of Structural Combination of Regimes to Identify the Optimal Structure

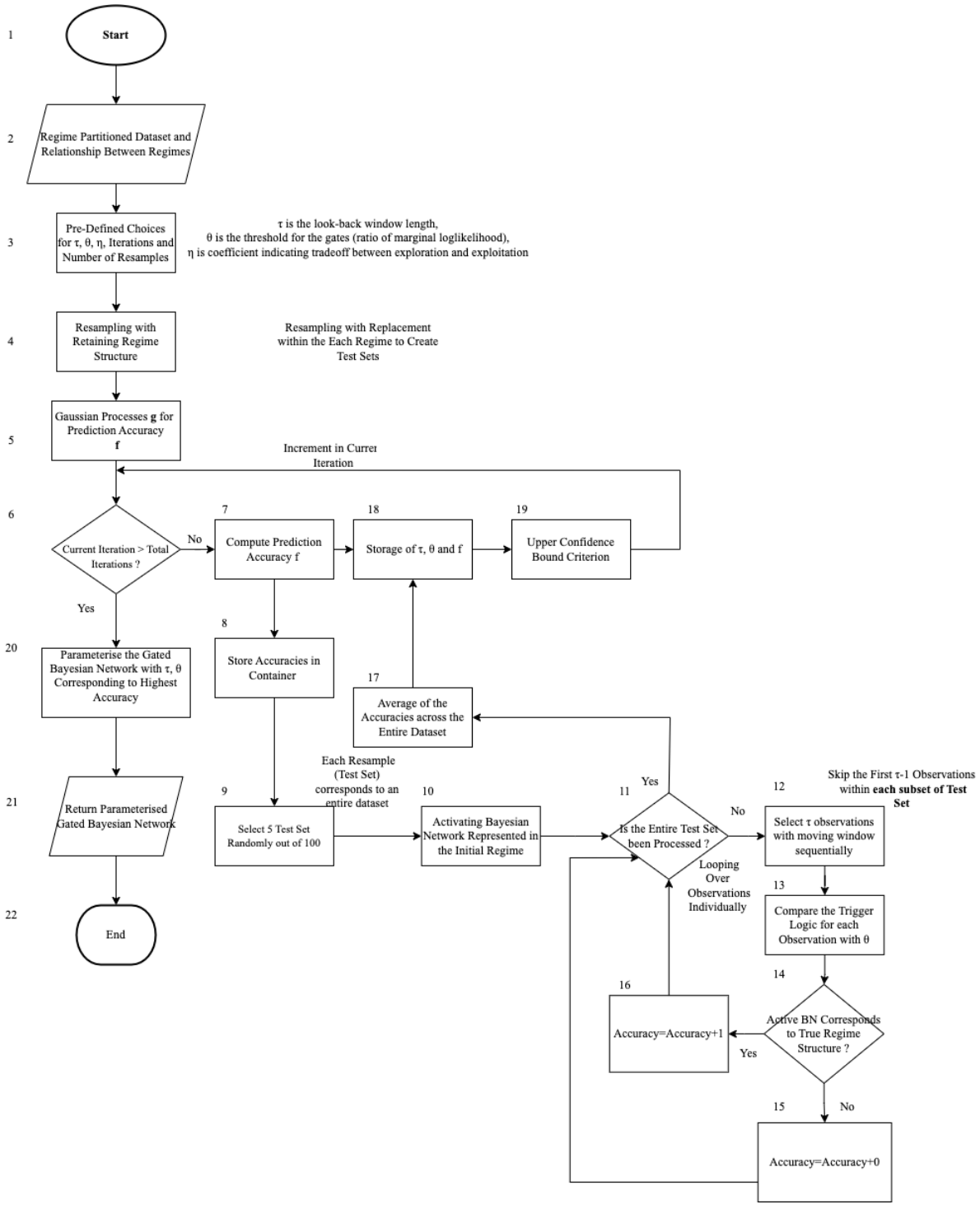


Figure 4.4: Primary Gating Mechanism for Optimisation of the Parameters τ and θ

5 Result

5.1 Feature Selection Using SHAP Analysis

One crucial step before regime identification is feature selection; the regime identification algorithm utilises the Bayesian network to model the distribution of the dataset $\mathfrak{Ds}$, where we consider 3041 observations with 43 features, from season 1983-84 to season 2020-21.

We perform SHAP analysis to search for the twenty most prominent features that have the highest impact on the target feature `Team_Prospect`. The result has been documented as the bee swarm plot in figure 5.1.

5.2 Regime Identification using Metropolis-Hastings MCMC Sampling

It is important to note that as we have utilised the running mean of length 20 for `Team_Prospect` variable, instead of 3041 observations, we consider 3021 observations by discarding the first 20 observations, for this reason, we have the dataset $\mathcal{D}$ with 3021 observations and 15 features as described in table 5.1.

The experimental results for identification of nonzero δs with $k = 4$ for Chicago Bulls have been documented in table 5.2. The experiment was repeated four times, and at each instance, for 10k iterations[1] of the Metropolis-Hastings MCMC algorithm.

The first 5000 iterations (burn-in samples) values are discarded as these samples might be correlated, so the results presented in the table 5.2 are based on the last 5000 iterations of MH MCMC samples, represented by four unique positional values δ_1, δ_2, δ_3 and δ_4. We also present the Bayesian Dirichlet score (log marginal likelihood) for each of the four BNs within an experiment; the results are documented in table 5.3, where the last column represents the total marginal BDe score of all the BNs. We can observe that experiment 3 yielded the highest BDe score, so we utilise the nonzero δs from experiment 3 in further analysis.

[1]The experiment had been initially planned to be carried out for 100k iterations. However, this was time expensive as one experiment would equate to roughly 45 hours of computational time on a machine with 8GB RAM. The results did not change drastically to that obtained for 10k iterations.

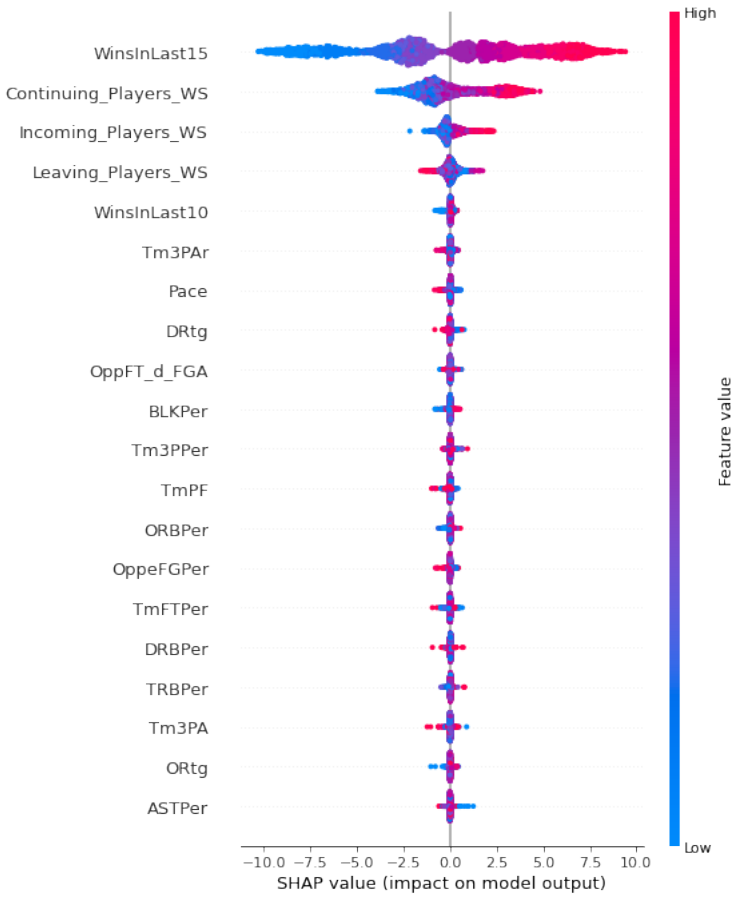


Figure 5.1: Feature Importance Plot for Chicago Bulls data-set, from Season 1983-84 to Season 2020-21

Table 5.1: Features of $\mathcal{D}$, for Regime Identification using MH MCMC

No.	Feature Name
1	WinsInLast15
2	Continuing_Players_WS
3	Incoming_Players_WS
4	Leaving_Players_WS
5	WinsInLast10
6	3PAr
7	DRtg
8	Opponent FT/FGA
9	BLK%
10	PF
11	Opponent eFG%
12	TRB%
13	AST%
14	Team_Prospect
15	Home_Game

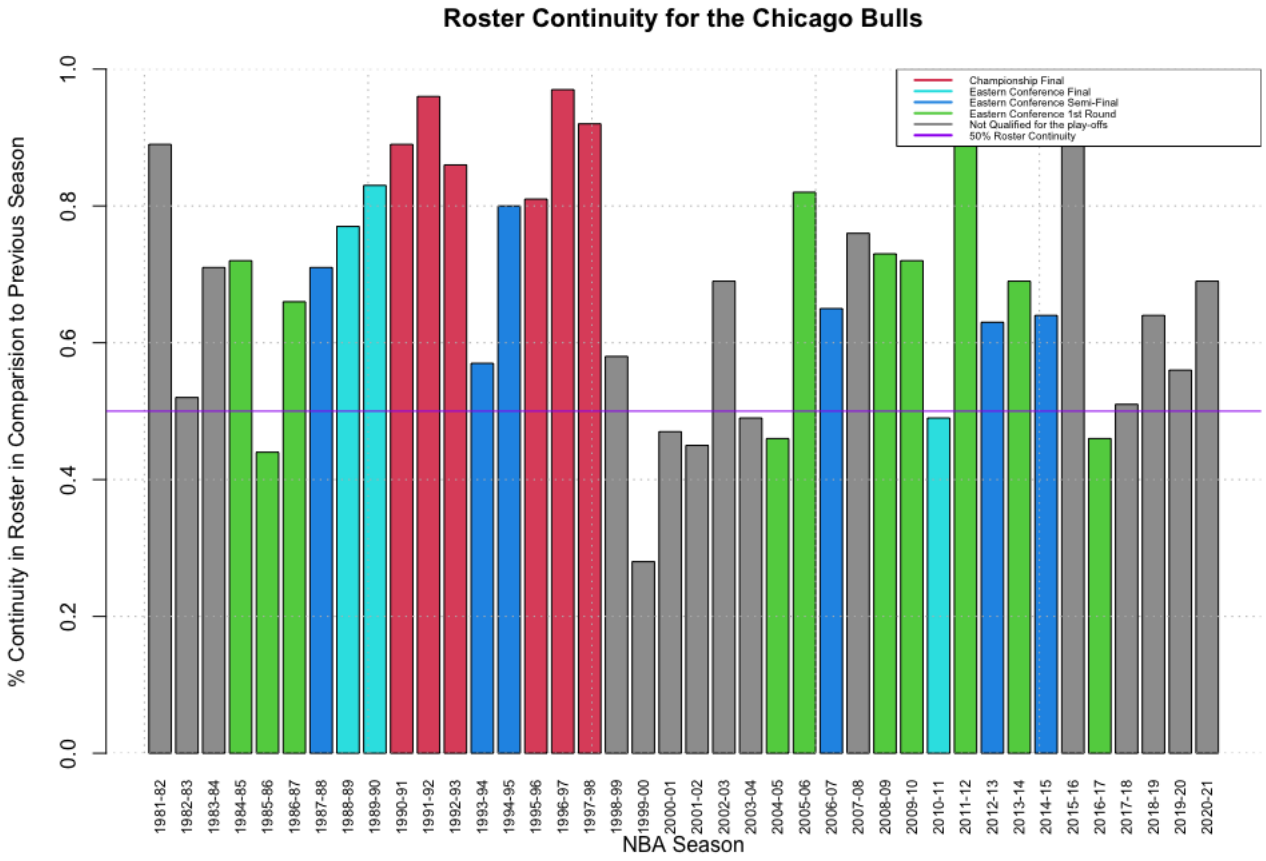


Figure 5.2: Roster Continuity for Chicago Bulls, from Season 1980-81 to Season 2020-21

Table 5.2: Identification of Nonzero δs, for Chicago Bull between Season 1983-84 to Season 2020-21, for $k = 4$ and 10k iterations

Experiment No.	δ_1	δ_2	δ_3	δ_4
1	392	839	1679	2562
2	461	1176	1750	2458
3	390	930	1695	2689
4	542	1225	1760	2455

5.3 Optimisation of Regime Transition Structure

In this section, we present the results obtained from the regime transition structure optimisation process taking into account the nonzero δs identified in the section 5.2. Since the proposed number of nonzero δs are four, this would lead to five unique subsets d_1, d_2, d_3, d_4 and d_5. We present these subsets in table 5.4, where each subset has a corresponding start, end date (yyyy-mm-dd format), and the number of games within that subset. Here we consider 3021 observations with 15 features as described in 5.1.

Table 5.3: Comparison of BDe Score (log marginal likelihood) for the Learned BN for 10k iteration and $k = 4$, for Chicago Bulls between Seasons 1983-84 to 2020-21

Experiment No.	BN_1	BN_2	BN_3	BN_4	BN_5	Total Marginal Likelihood
1	-7438.207	-8476.789	-15741.704	-17945.220	-8695.826	-58297.75
2	8827.758	-13407.245	-10822.621	-14382.637	-10889.465	-58329.73
3	-7400.548	-10536.459	-14027.384	-20152.319	-6142.237	**-58258.95**
4	-10514.67	-12651.33	-10058.06	-14107.84	-10950.62	-58282.52

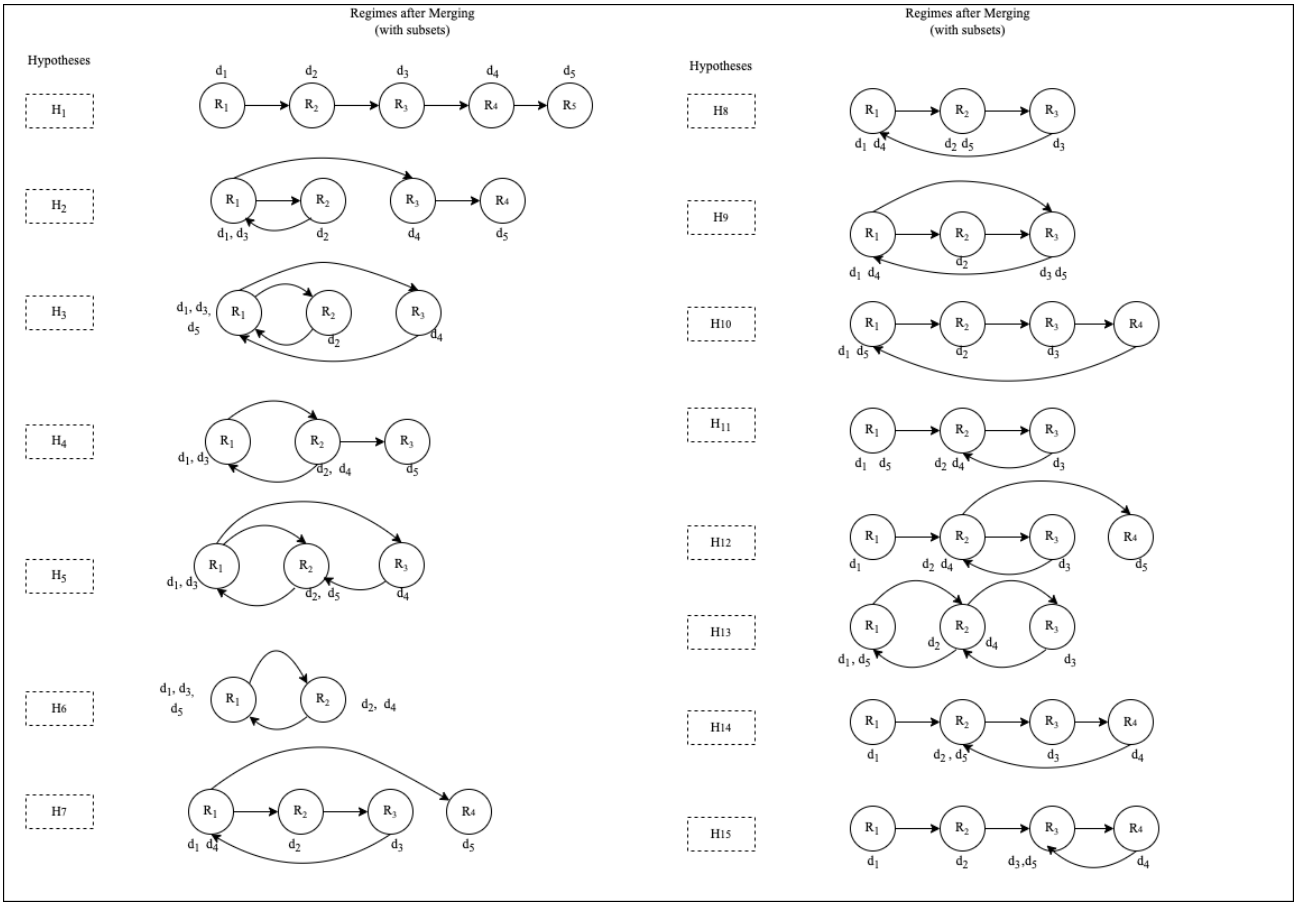


Figure 5.3: Regime Merging Hypotheses for $k = 4$

Table 5.4: Subsets for Chicago Bulls between Seasons 1983-84 to 2020-21

Subset	Start Date	End Date	Number of Games
d_1	1983-12-15	1988-04-22	389
d_2	1988-04-24	1995-02-08	540
d_3	1995-02-09	2004-12-22	765
d_4	2004-12-26	2017-02-01	994
d_5	2017-02-03	2021-05-16	333

Table 5.5: Total Marginal Likelihood for Hypotheses Corresponding four Nonzero δs

Hypothesis	Regimes with Corresponding Subsets	Total Marginal Likelihood
H_1	$R_1 = \{d_1\}, R_2 = \{d_2\}, R_3 = \{d_3\}, R_4 = \{d_4\}, R_5 = \{d_5\}$	**-58258.95**
H_2	$R_1 = \{d_1, d_3\}, R_2 = \{d_2\}, R_3 = \{d_4\}, R_4 = \{d_5\}$	-59123.82
H_3	$R_1 = \{d_1, d_3, d_5\}, R_2 = \{d_2\}, R_3 = \{d_4\}$	-59842.10
H_4	$R_1 = \{d_1, d_3\}, R_2 = \{d_2, d_4\}, R_3 = \{d_5\}$	-59721.36
H_5	$R_1 = \{d_1, d_3\}, R_2 = \{d_2, d_5\}, R_3 = \{d_4\}$	-59840.19
H_6	$R_1 = \{d_1, d_3, d_5\}, R_2 = \{d_2, d_4\}$	-60439.63
H_7	$R_1 = \{d_1, d_4\}, R_2 = \{d_2\}, R_3 = \{d_3\}, R_4 = \{d_5\}$	-59173.67
H_8	$R_1 = \{d_1, d_4\}, R_2 = \{d_2, d_5\}, R_3 = \{d_3\}$	-59890.04
H_9	$R_1 = \{d_1, d_4\}, R_2 = \{d_2\}, R_3 = \{d_3, d_5\}$	-59812.40
H_{10}	$R_1 = \{d_1, d_5\}, R_2 = \{d_2\}, R_3 = \{d_3\}, R_4 = \{d_4\}$	-58954.50
H_{11}	$R_1 = \{d_1, d_5\}, R_2 = \{d_2, d_4\}, R_3 = \{d_3\}$	-59552.04
H_{12}	$R_1 = \{d_1\}, R_2 = \{d_2, d_4\}, R_3 = \{d_3\}, R_4 = \{d_5\}$	-58856.49
H_{13}	$R_1 = \{d_1, d_5\}, R_2 = \{d_2, d_4\}, R_3 = \{d_3\}$	-59552.04
H_{14}	$R_1 = \{d_1\}, R_2 = \{d_2, d_5\}, R_3 = \{d_3\}, R_4 = \{d_4\}$	-58975.32
H_{15}	$R_1 = \{d_1\}, R_2 = \{d_2\}, R_3 = \{d_3, d_5\}, R_4 = \{d_4\}$	-58897.68

5.4 Comparison between Singular BN and GBN

To compare how the GBN based on the hypothesis H_1 represents the underlying structure compared to the singular BN that utilises the entire data set. Firstly, we train the cohort of BNs using the corresponding data sets described by the hypothesis H_1 in table 5.5.

Furthermore, in the table 5.6, the rows correspond to the trained BN, e.g. BN_{d_1} refers to the BN trained using the subset d_1 and $BN_{\mathcal{D}}$ refers to the BN trained using the entire data set $\mathcal{D}$. The columns refer to the BDe score for the corresponding regime, e.g. The highlighted score in the column $Score(R_1 = \{d_1\})$ refers to the BDe score of BN_{d_4} for the regime R_1. While scrutinising the results in 5.6, it can be observed that the cohort of BNs consistently

Table 5.6: Comparison of Marginal Log Likelihood values (BDe Score) for the Cohort of BNs (GBN) against the Singular BN, for each Subset of the Game Logs, for Chicago Bulls between Seasons 1983-84 to 2020-21

BN	Score($R_1 = \{d_1\}$)	Score($R_2 = \{d_2\}$)	Score($R_3 = \{d_3\}$)	Score($R_4 = \{d_4\}$)	Score($R_5 = \{d_5\}$)
BN_{d_1}	-7570.133	-10726.14	-14253.25	-20214.98	-6376.181
BN_{d_2}	7590.641	-10718.81	-14243.71	-20232.19	-6366.401
BN_{d_3}	-7672.84	-10766.64	-14261.41	-20274.16	-6422.612
BN_{d_4}	**-7463.295**	**-10591.67**	**-14123.68**	**-20131.08**	-6246.563
BN_{d_5}	-7580.62	-10725.85	-14433.55	-20276.06	**-6205.887**
$BN_{\mathcal{D}}$	-7779.198	-10895.31	-14355.37	-20289.65	-6523.448

outperformed the singular BN represented by $BN_{\mathcal{D}}$. We have included the Singular BN in appendix II, figureA.6.

5.5 GBN Parameter Optimisation Using Gaussian Processes

The next step after having identified hypothesis H_1 as the optimum structure in the section 5.3 is to tune the hyperparameters pair (τ, θ) that has the highest accuracy **f**. This accuracy (f) is a measure of correctly triggering the four gates described in the equation 5.1. We utilised the Gaussian Processes framework to propose a new set of (τ, θ) values while optimising for the function **f**.

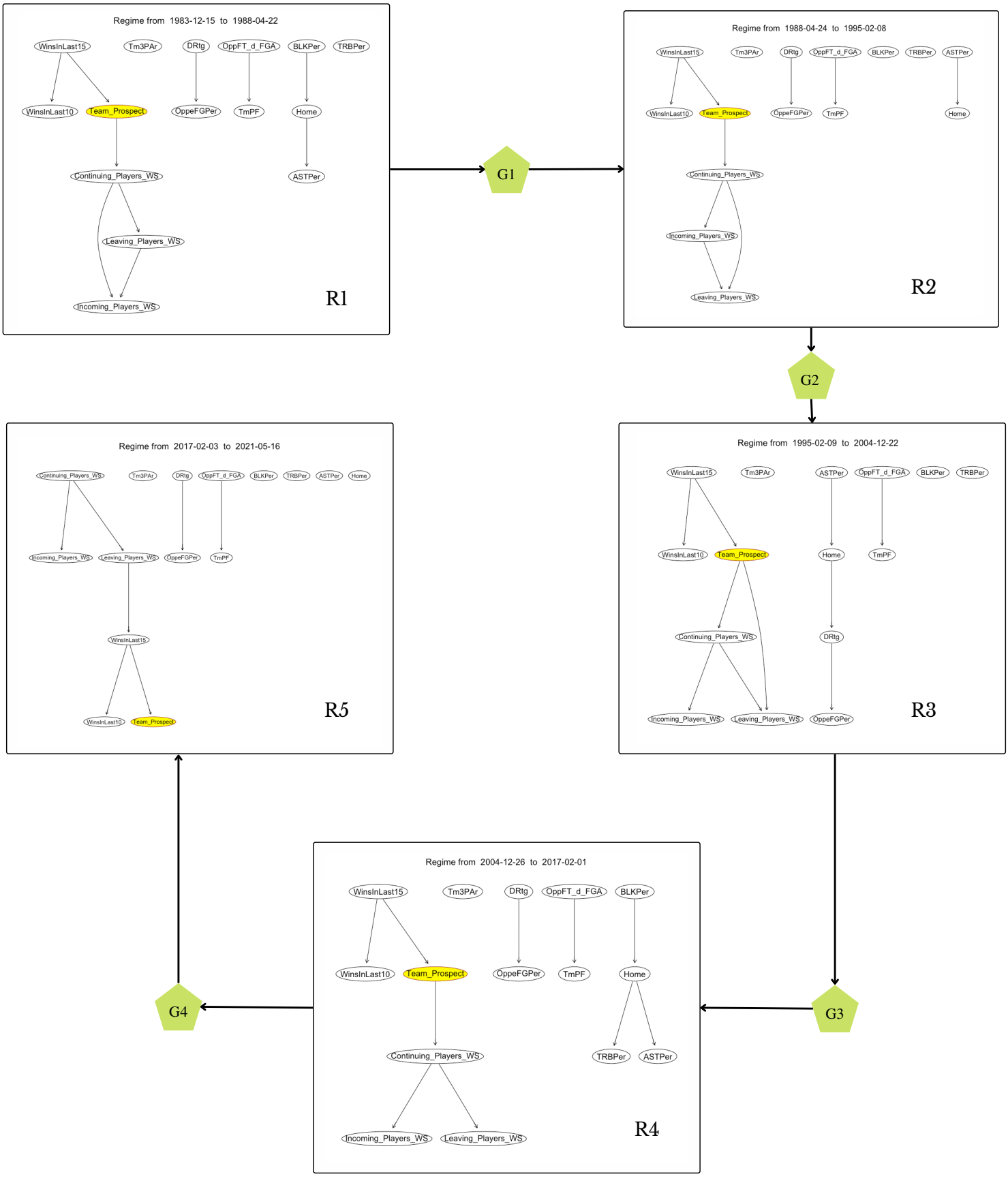
$$\begin{aligned} TL(G_1) :&= \frac{p(d_1^\tau|R_2)}{p(d_1^\tau|R_1)} > \theta \\ TL(G_2) :&= \frac{p(d_2^\tau|R_3)}{p(d_2^\tau|R_2)} > \theta \\ TL(G_3) :&= \frac{p(d_3^\tau|R_4)}{p(d_3^\tau|R_3)} > \theta \\ TL(G_4) :&= \frac{p(d_4^\tau|R_5)}{p(d_4^\tau|R_4)} > \theta \end{aligned} \tag{5.1}$$

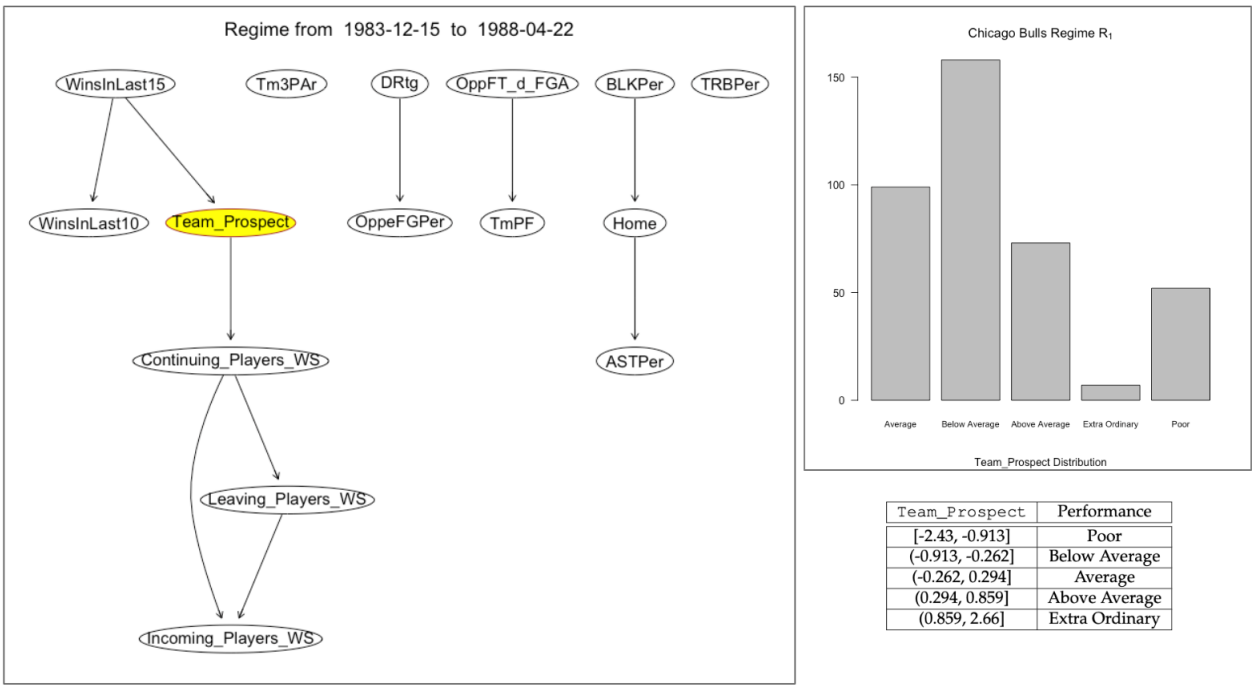
In these equations, we have used the notation d_i^τ (where i=1,2,3,4), which refers to τ data points in the d_i subset. Here we have assumed the data that we have access to is noise-free. Furthermore, we have chosen the squared exponential kernel (SE) with variance parameters $\sigma_g^2 = 50$ and $l = 1$ defined in the equation 2.41.

Table 5.7: Optimisation for Hyperparameter Pair (τ, θ) for Different values of η

No.	η	$max(f)$	τ	θ	$f > 0.85$	$f > 0.90$
1	5	0.9556502	26	56.21053	86	43
2	10	0.9487421	36	48.39474	87	31
3	25	0.9438185	33	71.84211	89	34
4	50	0.9533541	31	71.84211	91	43

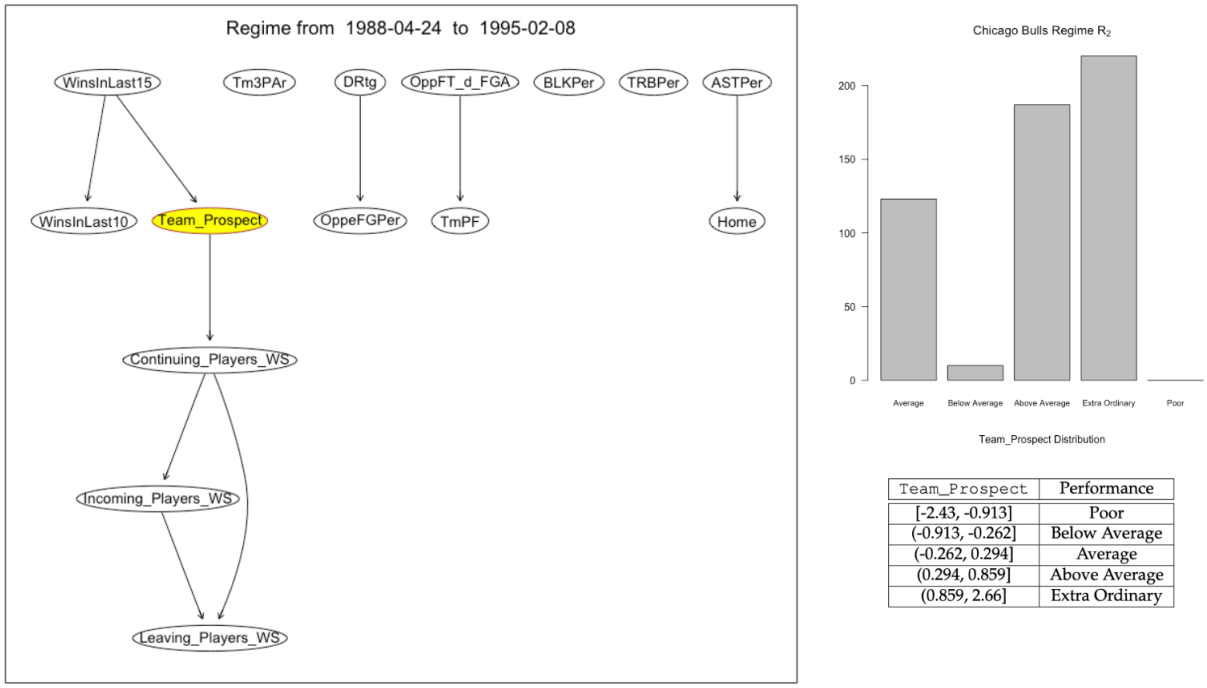
5.6 Chicago Bull GBN Results


Regime from 1983-12-15 to 1988-04-22
WinsInLast15
Tm3PAr
DRtg
OppFT_d_FGA
BLKPer
TRBPer
WinsInLast10
Team_Prospect
OppeFGPer
TmPF
Home
Continuing_Players_WS
ASTPer
Leaving_Players_WS
Incoming_Players_WS
R1
G1
Regime from 1988-04-24 to 1995-02-08
WinsInLast15
Tm3PAr
DRtg
OppFT_d_FGA
BLKPer
TRBPer
ASTPer
WinsInLast10
Team_Prospect
OppeFGPer
TmPF
Home
Continuing_Players_WS
Incoming_Players_WS
Leaving_Players_WS
R2
G2
Regime from 1995-02-09 to 2004-12-22
WinsInLast15
Tm3PAr
ASTPer
OppFT_d_FGA
BLKPer
TRBPer
WinsInLast10
Team_Prospect
Home
TmPF
Continuing_Players_WS
DRtg
Incoming_Players_WS
Leaving_Players_WS
OppeFGPer
R3
G3
Regime from 2004-12-26 to 2017-02-01
WinsInLast15
Tm3PAr
DRtg
OppFT_d_FGA
BLKPer
WinsInLast10
Team_Prospect
OppeFGPer
TmPF
Home
Continuing_Players_WS
TRBPer
ASTPer
Incoming_Players_WS
Leaving_Players_WS
R4
G4
Regime from 2017-02-03 to 2021-05-16
Continuing_Players_WS
Tm3PAr
DRtg
OppFT_d_FGA
BLKPer
TRBPer
ASTPer
Home
Incoming_Players_WS
Leaving_Players_WS
OppeFGPer
TmPF
WinsInLast15
WinsInLast10
Team_Prospect
R5



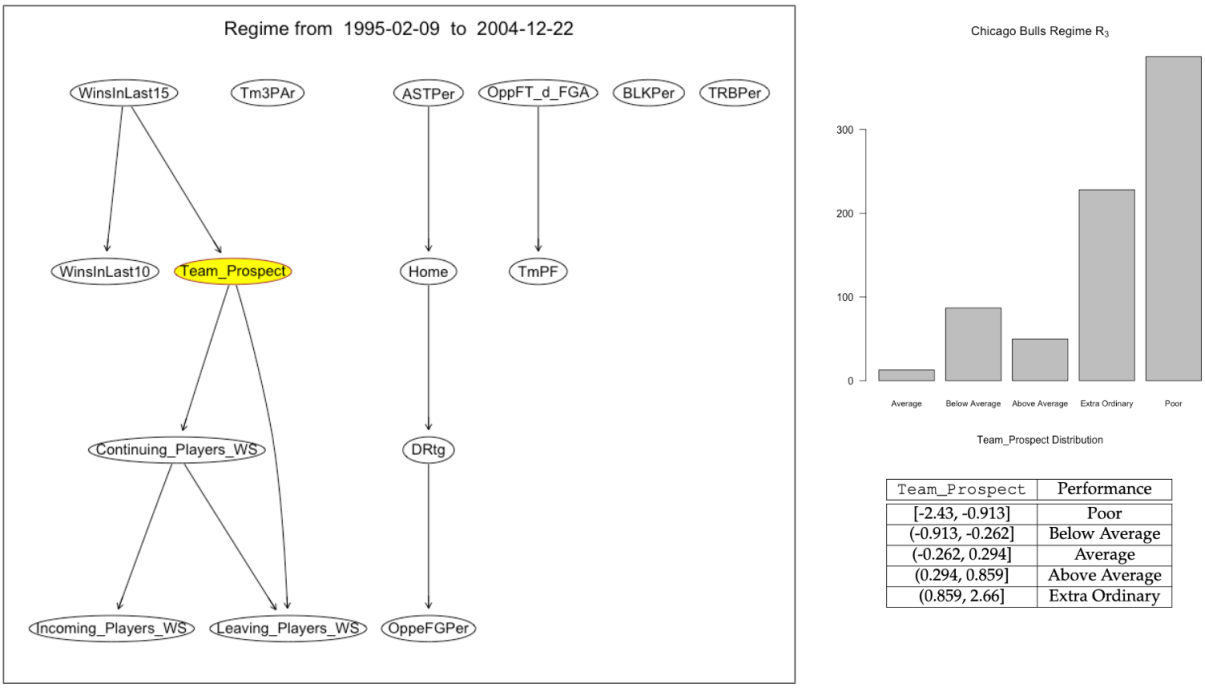
Team_Prospect	Performance
[-2.43, -0.913]	Poor
(-0.913, -0.262]	Below Average
(-0.262, 0.294]	Average
(0.294, 0.859]	Above Average
(0.859, 2.66]	Extra Ordinary

Figure 5.4: Regime R_1, Rookie Years of Chicago Bulls



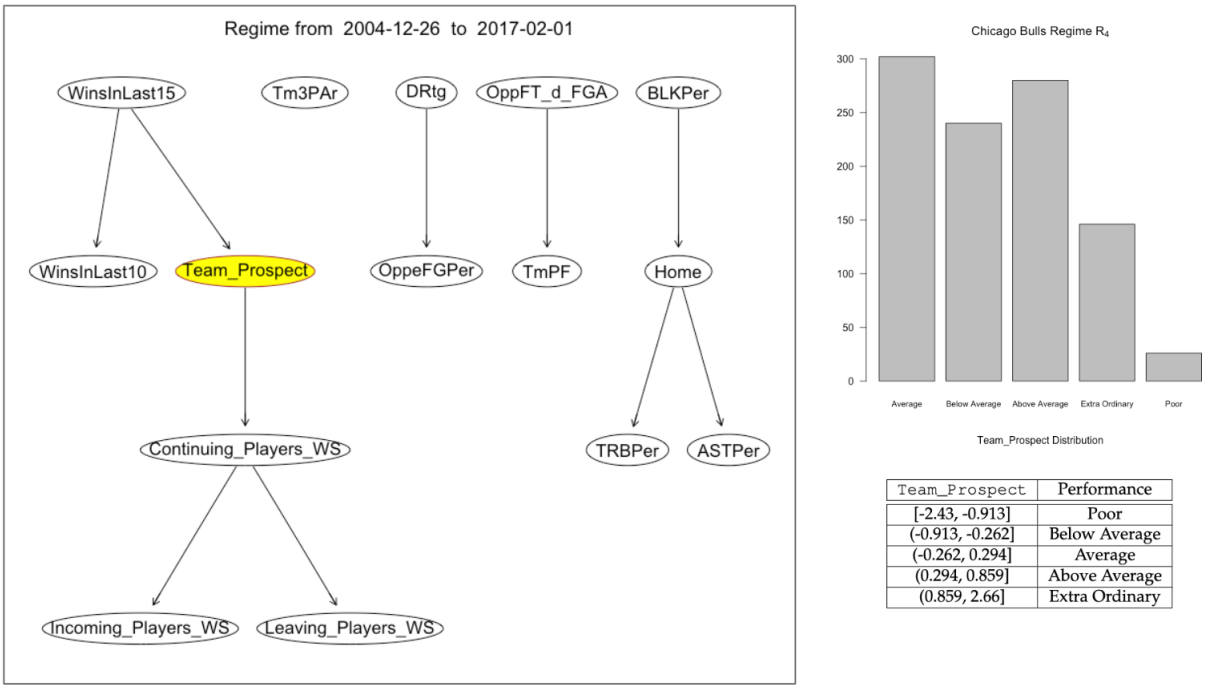
Team_Prospect	Performance
[-2.43, -0.913]	Poor
(-0.913, -0.262]	Below Average
(-0.262, 0.294]	Average
(0.294, 0.859]	Above Average
(0.859, 2.66]	Extra Ordinary

Figure 5.5: Golden Years (I) R_2,Chicago Bulls



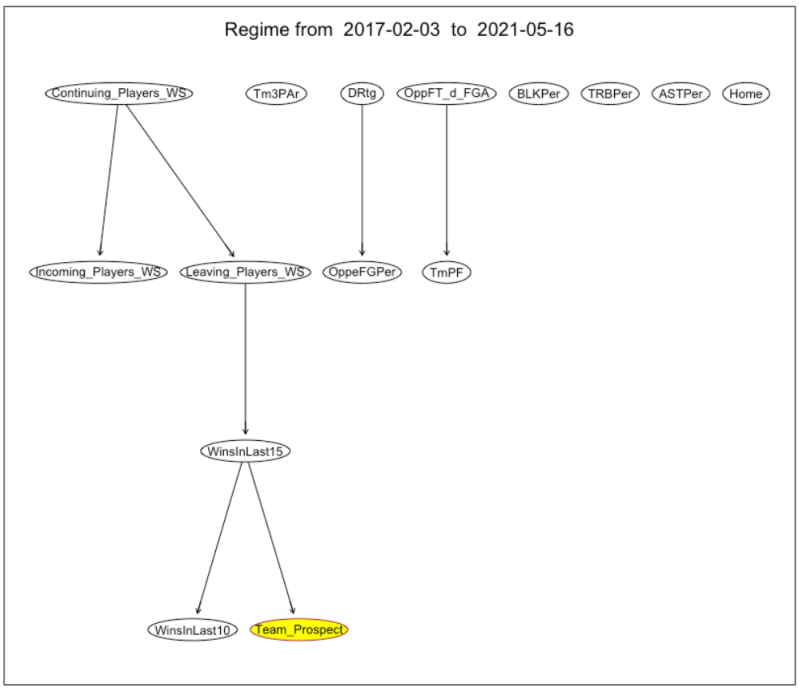
Team_Prospect	Performance
[-2.43, -0.913]	Poor
(-0.913, -0.262]	Below Average
(-0.262, 0.294]	Average
(0.294, 0.859]	Above Average
(0.859, 2.66]	Extra Ordinary

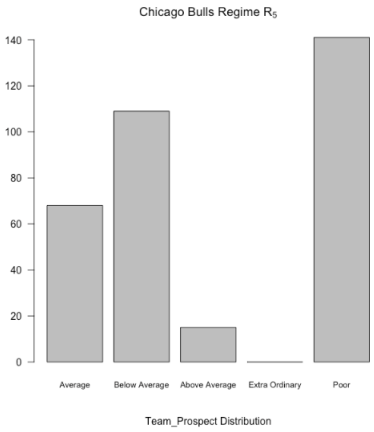
Figure 5.6: Golden Years(II) and Aftermath R_3, Chicago Bulls



Team_Prospect	Performance
[-2.43, -0.913]	Poor
(-0.913, -0.262]	Below Average
(-0.262, 0.294]	Average
(0.294, 0.859]	Above Average
(0.859, 2.66]	Extra Ordinary

Figure 5.7: R_4: Resurgence, Chicago Bulls





Team_Prospect	Performance
[-2.43, -0.913]	Poor
(-0.913, -0.262]	Below Average
(-0.262, 0.294]	Average
(0.294, 0.859]	Above Average
(0.859, 2.66]	Extra Ordinary

Figure 5.8: Regime R_5, Chicago Bulls

6 Discussion

The work conducted by Bendtsen (2016) in regime transition research for baseball provided insight into a player's performance at a granular level, with the underlying assumption that during the specific temporal partition, the player exhibits a unique set of attributes [1]; this aspect has been reproduced in the case of the basketball team, where the game logs (basic and advanced) along with the external parameters described in chapter 3 between seasons 1983-84 to 2020-21 are utilised to model the regime transitions.

There has been massive development in the last forty years in the basketball field that has reshaped the structure of the National Basketball Association in North America. In season 1983-84, 23 teams were incorporated into the league, and four divisions were formed. Forty years later, the total number of competing teams is 30, divided into six divisions. Chicago Bulls is one of the teams which has been included in the league since season 1966-67. They have played in 56 seasons, making it a comparatively mature franchise. Throughout history, they have advanced 35 times into the playoffs seasons [39] and won the NBA title six times.

Requirement for Feature Selection

According to OEIS[1], the number of possible DAG for n labelled nodes follow the sequence A003024, where the first 5 numbers corresponding to $1, 2, 3, 4, 5$ nodes are $1, 3, 25, 543, 29281$. These values are calculated using the recurrence relation formula in the works of Rodinov(1992, p.320)[40], where $a(n)$ represents the number of DAGs for n labelled nodes, as described in equation 6.1, where the i ranges from $1, 2, ..., n$.

$$\begin{aligned} a(n) &= \sum_{i=1}^{n} (-1)^{i+1} \cdot \binom{n}{i} \cdot 2^{i(n-i)} \cdot a(n-i) \\ a(0) &= 1 \end{aligned} \tag{6.1}$$

In figure A.5, we plot the number $a(n)$, where n ranges from $1, 2, ..., 40$ on the x-axis, and the y-axis the number of DAGs are represented in log scale, for example, number of DAGs for $n = 40$ are approximately $1.1241e + 276$.

[1]The On-Line Encyclopedia of Integer Sequences, Sequence A003024: `https://oeis.org/A003024`

As the original dataset, $\mathfrak{Ds}$ has 43 features (or nodes in DAG), it would be computationally infeasible to perform a heuristic search to find a DAG with the highest marginal likelihood. To mitigate this problem, we used the feature attribution model to find the 20 most important features that could aid in describing the target variable `Future_Prospect`.

Primary Findings from SHAP Analysis

In the results, the feature that has the highest effect on `Team_Prospect` is `WinsInLast15`, i.e. the number of wins in the last fifteen games; this is unsurprising as the `Team_Prospect` takes into account the running mean of score difference for the last twenty games, so there is likely a higher correlation between these two events. We observed a similar result for another feature `WinsInLast10`, which accounts for several wins in the last ten games, albeit the feature importance is less than that of `WinsInLast15`.

The next three most important features are related to the player's individual contribution. As described previously, the first category of the players are those who are going to continue playing in the next season; these players have a higher impact on the `Team_Prospect` than the contribution from the players who either have newly joined the team in the current season or the players from the current season who are going to be leaving the team in the next season.

Roster Continuity and Player Contribution

We momentarily turn our attention to the figure 5.2, which describes the relationship between Chicago Bull's roster continuity (from season 1980-81 to season 2020-21) value and the team's performance in the season. Roster continuity measures the percentage of the current season roster made up of players from the previous season. In this figure, the roster continuity is plotted on the y-axis ranging between 0 to 1 (or 0% to 100%) along with the corresponding season on the x-axis. The season where a team only played in the regular season is marked with grey coloured bars, and when the team is advancing to other levels, such as playing in Conference 1st Round, Conference Semi-Final, Conference Final or Championship Final represented by green, blue, turquoise and red colours accordingly. We can visually observe that seasons, where Chicago Bulls had advanced to the Conference Semi-Final or above are described by blue, turquoise and red bars, which correspond to more than 50% roster continuity value (except for season 2010-11 for which roster continuity is 49%), which means that there is a significant correlation between roster stability of a team and team's performance. Now, if we consider the stability of the roster in terms of players who continue to play in the next season, this would translate into the player's contribution towards win share.

There are few features from the feature importance plot which did not have a causal relationship with any other features in any of the regimes; these features are `Pace`, `TmFTPEr`, `ORBPer` and `ORtg`. These features are not included in the final data set used for Regime Identification experiments.

Regime Identification

One crucial parameter in the regime identification algorithm is the number of maximum allowable splits (k) in the data set. This number could be any positive integer, ranging from 1 to n (here, n is the number of data points in the data set). However, it is unlikely that a large number of k would generalise the learned structure well, as a higher value of k would likely cause a low bias but a higher variance in the model. We have experimented with values of $k = 4, 5$, and 6; however, based on the empirical evidence, the results for $k = 4$ yielded the optimal model.

Optimisation of Regime Transition Structure

The number of unique hypotheses (structural arrangement of subsets) for five unique subsets is 15; we have documented these results in table 5.5. The highest total marginal likelihood for the training data set corresponds to the hypothesis H_1.

Comparision of Singular BN with GBN

We have compared the marginal likelihood values for GBN, and singular BN in table 5.6, where it can be observed that the GBN trained on the subsets d_1 to d_5 outperforms the BN trained on dataset $\mathcal{D}$.

GBN Optimisation

To find the optimum value of (τ, θ) pair, we have experimented with different values of η ranging from 5, 10, 25 and 50. η is a coefficient for which its higher value would lead to more exploration while the lower value would yield higher exploitation. We searched for the optimum (τ, θ) in the grid where the window length (τ) varied from 5 to 50 and the gating threshold value (θ) ranged from 1.5 to 150. We repeated the process of finding the optimum Λ pair for each η presented in figure 5.7; it can be observed that the maximum accuracy obtained for the $\eta = 5$ is the highest amongst the four experiments. The close 2nd is the maximum accuracy for the $\eta = 50$. Finally, the four gates are parameterised with $\tau = 26$ and $\theta = 56.21$.

Discussion about Chicago Bulls GBN

Chicago Bulls was a rather mediocre team in the early 1980s; from the season 1980-81 to the season 1983-84, they did not even qualify for the playoffs and the team was ranked 19th out of 23[2]. The normalised `Team_Prospect` values are split into five intervals corresponding to the performance indicator as displayed in the table 3.2.

The first regime R_1 began in the season 1983-84, which brought several changes; the team had a new coach Kevin Loughery along with four rookies and only two players[3] with experience of five years maximum. In the following season, the team had a new executive, Jerry Krause and a rookie Shooting Guard Michael Jordan; this made several long-lasting impacts on the franchise. Chicago Bulls started qualifying for the playoffs from 1984-85. The first regime started from the season 1983-84[4]. Regime R_1 lasted until season 1987-88, which can be visualised in figure 5.4. We can observe that winning the previous games and the players' win share have an impact over the `Team_Prospect`; this has been consistent through all the regimes. In figure 5.4, the distribution of `Team_Prospect` is positively skewed with performance *Below Average* and *Poor* performance.

Chicago Bulls had hit their stride in the regimeR_2, which started in season 1988-89 and lasted until season 1994-95. The distribution of `Team_Prospect` here has a left tail indicating their consistent domination on the court; the performance during this period was exemplary, which earned the Chicago Bulls their first threepeat (three consecutive wins)of NBA title. After the sudden departure of Michael Jordan from the team before season 1993-94, Scottie Pippen helmed the leadership, but the team had a performance dip in two consecutive seasons in the absence of Michael Jordan.

[2]Using the Net Rtg (Net Rating),`https://www.basketball-reference.com/teams/CHI/1983.html`

[3]Dave Corzine, who played as the Centre and Reggie Theus, the point guard.

[4]In our analysis, we had chosen to ignore the first 20 matches at the beginning of the first season as we had modelled the target feature `Team_Prospect` considering the last 20 games at any given point of time.

The next regime R_3 Michael Jordan's 2nd retirement coincided with this era, and multiple prominent players like Dennis Rodman and Scottie Pippen were traded-off to another team; this was a volatile time; the team hired several rookies from NCAA, Ed Curry and Tyson Chandler, and the team had one of the worst seasons in 2001 with 16 consecutive losses in the regular games, in the following years the franchise underwent several different coaches, it was only in the season 2004-05, the Chicago Bulls appeared in the play-off after six years.

Regime R_4 started in the season 2004-05 and lasted until the season 2017-18; this was the era of resurgence; three coaches who coached the team over the years, Scott Skiles(for four years), Tom Thibodeau (for five years) and Fred Hoiberg (for four years). This regime covered 13 seasons; Chicago Bulls had advanced to Playoffs in 11 seasons.

The last era is made of only three seasons; based on the search heuristics in figure 5.8, the home-game advantage is not evident in this regime. After the season 2016-17, Chicago Bulls did not appear in the playoffs in the following seasons.

Limitation of the Work

Data

- In the research work, we have utilised several features for our analysis that we think are interesting for a manager or coach of a basketball team. However, as we have not consulted any personalities that are stakeholders in the NBA team, this work provides a limited understanding of the real team dynamics.
- Bio-mechanical profiling of a player may help the coach devise an on-court strategy and understand the player's weaknesses. However, this data has been collected by proprietary technologies and may not be readily available, as is the case for performance statistics.

Method

- The hill climbing implementation using `bnlearn` has a parameter `blacklist`, which can help to encode prior beliefs about the relationship between certain nodes in the model. In our investigation, we experimented with blacklisting arcs between the variables, which quantify independent events. For example, the team's personal fouls `PF` do not affect whether a team plays at home or away, so one can encode this belief in the model to find an optimum model. However, the model with this design choice was computationally expensive and not feasible in **R**.
- We have used Gaussian Processes Regression to tune the hyperparameters τ and θ for a fixed value of variance σ_g^2 and length scale l for the squared exponential kernel. Ideally, the parameter space should be tuned for all four hyperparameters instead of just two. There is the risk that the results are locally optimal but not globally optimal.

Results

- Individual regimes identified in the previous chapter reveal important shortcomings of this analysis, as some of the edges between certain variables do not correspond to the observable event in basketball history.
- The regime structures identified in the figures 5.4 to 5.8 have relatively little changes to the structure composed of the top five features identified from the SHAP value diagram.

Ethical Consideration

In this section, we list the ethical concerns that are important in the context of data access and the intended use of the knowledge gained from the research work.

- The data set utilised here is hosted on the website `basketball-reference.com` which seems to operate on a combination of ad revenue based freemium[41] model[5], we have only accessed the available data without any cost.
- We have adhered to the *Terms of Use* [6] and *Information Sharing*[7] guidelines laid out by the parent company of the data hosting website, Sports Reference LLC, to the best of our knowledge.
- Primarily, the data is utilised for academic research purposes. Furthermore, the research we conducted here is intended for constructive development in the sports analytics community (if any) without any commercial ambition.

[5]The term 'freemium' is made up of the words 'free' and 'premium', which refers to the tiers in the economic hierarchy, where the 'premium' subscription provides access to more advanced targeted searches in addition to access to 'free' data.

[6]Terms of Use Guidelines: `https://www.sportsreference.com/termsofuse.html`

[7]Information Sharing Guidelines:`https://www.sports-reference.com/sharing.html`

7 Conclusion

7.1 Summary

The research primarily focuses on the statistical performance analysis of the famous NBA team Chicago Bulls between the season 1983-84 and to season 2020-21. The team has undergone several changes over the decades, from a relatively unknown team in the 1980s to the peak of its fame in the 1990s following a sudden demolition under the new management in the early 2000s to multiple attempts to reinstate the lost glory in the 2010s.

Our initial research objective was to discover the most critical variables demonstrating how the team had performed over time. To address this objective, we conducted a post hoc analysis using SHAP (**SH**apley **A**dditive ex**P**lanation) method that identified the twenty features that had the highest impact on the team performance.

The second research objective was to implement the appropriate causal framework to model the team's on-court performance within a specific time frame (regime). In order to simulate such a framework, firstly, we identified the temporal subset within the data by simulating the posterior distribution for such a model using the Metropolis-Hastings MCMC algorithm. Secondly, we identified the regimes and searched for an optimum regime structure; using the optimum model, we constructed the Gated Bayesian Network(GBN). Finally, we tuned the hyperparameters utilised in the GBN by calculating the accuracy of the gating mechanism using the hyperparameters proposed using Gaussian Processes. We demonstrated that GBN explains the data better overall than a singular BN.

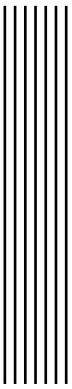
Our tertiary research objective was to verify the results obtained from the GBN and compare them with historical data to determine whether these effects are observed retrospectively. This objective has been only partially fruitful because several eccentricities in the results could not be explained.

7.2 Future Work

In this section, we list the possible extension to future project work to mitigate the limitation discussed above and expand the utility of the research work.

- In this degree project, we have only focused on a single team (i.e. Chicago Bulls) for the historical analysis, which is not necessarily adequate to determine the wider applicability of the methodology. If the experiment is repeated for several teams, we could more adequately understand the capabilities and limitations of the method.
- One of the crucial bottlenecks in this project work has been the running time of the regime identification algorithm in **R** framework. We have utilised the package `bnlearn` to conduct the structure learning, which isn't necessarily time optimal when the algorithm is repeated for 100k iterations. This problem can be addressed by translating the code in a faster language such as **C**, where the C program such as GOBNILP [1][42][43] could fasten the structure learning.
- As a part of our data set, we have included the regular season game logs but did not include the playoff game logs. A possible extension to this project could include the playoff data in the original data set and compare whether the model learned from the combined data set provides greater insight into the team dynamics.
- There are specific socio-psychological parameters we referred to in chapter 1, such as a player's off-court lifestyle choices and contractual salary negotiations that may affect the team's performance are not readily available for various reasons; it may require additional computational and economic resources to perform such analysis using text mining.

[1]GOBNILP: Globally Optimal Bayesian Network learning using Integer Linear Programming, link: `https://www.cs.york.ac.uk/aig/sw/gobnilp/`



Bibliography

[1] Marcus Bendtsen. "Regimes in baseball players' career data." In: *Data Mining Knowledge Discovery* 31.6 (2016), pp. 1580–1621. ISSN: 13845810. URL: https://link.springer.com/content/pdf/10.1007/s10618-017-0510-5.pdf.

[2] Scott M Lundberg and Su-In Lee. "A unified approach to interpreting model predictions". In: *Advances in neural information processing systems* 30 (2017).

[3] Thomas Page. "Applications of wearable technology in elite sports". In: *i-manager's Journal on Mobile Applications and Technologies* 2.1 (2015), p. 1.

[4] Roland Lazenby and Olivier Bougard. *Michael Jordan: The life*. Talent sport, 2015.

[5] Kerry Eggers. *Jail blazers: How the Portland Trail Blazers became the bad boys of basketball*. Sports Publishing, 2021.

[6] Jack McCallum. *DREAM TEAM: How Michael, Magic, Larry, Charles and the Greatest Team of All Time Conquered the World and Changed the Game of Basketball Forever*. Vol. 59. 7. Risk and Insurance Management Society, Inc., 2012, p. 43.

[7] Matthew J. Barnes. "Alcohol: Impact on Sports Performance and Recovery in Male Athletes". In: *Sports Medicine* 44.7 (2014), pp. 909–919. DOI: 10.1007/s40279-014-0192-8. URL: https://doi.org/10.1007/s40279-014-0192-8.

[8] Zsuzsanna Dömötör, Roberto Ruíz-Barquín, and Attila Szabo. "Superstitious behavior in sport: A literature review". In: *Scandinavian Journal of Psychology* 57.4 (2016), pp. 368–382. DOI: https://doi.org/10.1111/sjop.12301.

[9] Masaru Teramoto and Chad L. Cross. "Relative Importance of Performance Factors in Winning NBA Games in Regular Season versus Playoffs". In: *Journal of Quantitative Analysis in Sports* 6.3 (2010). DOI: doi:10.2202/1559-0410.1260. URL: https://doi.org/10.2202/1559-0410.1260.

[10] Elia Morgulev and Yair Galily. "Choking or Delivering Under Pressure? The Case of Elimination Games in NBA Playoffs". In: *Frontiers in Psychology* 9 (2018), p. 979. ISSN: 1664-1078. DOI: 10.3389/fpsyg.2018.00979. URL: https://www.frontiersin.org/article/10.3389/fpsyg.2018.00979.

[11] LLOYD S Shapley. *Quota solutions of n-person games*. Tech. rep. RAND CORP SANTA MONICA CA, 1952.

[12] Christoph Molnar. *Interpretable Machine Learning. A Guide for Making Black Box Models Explainable*. 2nd ed. 2022. URL: https://christophm.github.io/interpretable-ml-book.

[13] Erik Štrumbelj and Igor Kononenko. "Explaining prediction models and individual predictions with feature contributions". In: *Knowledge and information systems* 41.3 (2014), pp. 647–665.

[14] Scott M. Lundberg, Gabriel G. Erion, and Su-In Lee. "Consistent Individualized Feature Attribution for Tree Ensembles". In: *CoRR* abs/1802.03888 (2018). arXiv: 1802.03888. URL: http://arxiv.org/abs/1802.03888.

[15] Andrew Gelman, John B. Carlin, Hal S. Stern, and Donald B. Rubin. *Bayesian data analysis*. 3rd ed. Statistics texts. Chapman Hall, 1995. ISBN: 9781439840955. URL: http://www.stat.columbia.edu/~gelman/book/BDA3.pdf.

[16] Thomas Bayes. "LII. An essay towards solving a problem in the doctrine of chances. By the late Rev. Mr. Bayes, FRS communicated by Mr. Price, in a letter to John Canton, AMFR S". In: *Philosophical transactions of the Royal Society of London* 53 (1763), pp. 370–418.

[17] William M Bolstad and James M Curran. *Introduction to Bayesian statistics*. John Wiley & Sons, 2016.

[18] Bradley Efron. "Bayes' Theorem in the 21st Century". In: *Science* 340.6137 (2013), pp. 1177–1178. DOI: 10.1126/science.1236536. eprint: https://www.science.org/doi/pdf/10.1126/science.1236536. URL: https://www.science.org/doi/abs/10.1126/science.1236536.

[19] Leland Gerson Neuberg. "Causality: models, reasoning, and inference, by judea pearl, cambridge university press, 2000". In: *Econometric Theory* 19.4 (2003), pp. 675–685.

[20] Marco Scutari, Catharina Elisabeth Graafland, and José Manuel Gutiérrez. "Who learns better Bayesian network structures: Accuracy and speed of structure learning algorithms". In: *International Journal of Approximate Reasoning* 115 (2019), pp. 235–253. ISSN: 0888-613X. DOI: https://doi.org/10.1016/j.ijar.2019.10.003. URL: https://www.sciencedirect.com/science/article/pii/S0888613X19301434.

[21] David Heckerman, Dan Geiger, and David M Chickering. "Learning Bayesian networks: The combination of knowledge and statistical data". In: *Machine learning* 20.3 (1995), pp. 197–243.

[22] Marco Scutari. "Learning Bayesian Networks with the bnlearn R Package". In: *Journal of Statistical Software* 35.3 (2010), pp. 1–22. DOI: 10.18637/jss.v035.i03. URL: https://www.jstatsoft.org/index.php/jss/article/view/v035i03.

[23] David Heckerman. "A Tutorial on Learning with Bayesian Networks". In: *Innovations in Bayesian Networks: Theory and Applications*. Ed. by Dawn E. Holmes and Lakhmi C. Jain. Berlin, Heidelberg: Springer Berlin Heidelberg, 2008, pp. 33–82. ISBN: 978-3-540-85066-3. DOI: 10.1007/978-3-540-85066-3_3. URL: https://doi.org/10.1007/978-3-540-85066-3_3.

[24] Ioannis Tsamardinos, Laura E Brown, and Constantin F Aliferis. "The max-min hill-climbing Bayesian network structure learning algorithm". In: *Machine learning* 65.1 (2006), pp. 31–78.

[25] Herbert S. Wilf. "Chapter 1 - Introductory Ideas and Examples". In: *Generatingfunctionology*. Ed. by Herbert S. Wilf. Academic Press, 1990, pp. 1–26. ISBN: 978-0-12-751955-5. DOI: https://doi.org/10.1016/B978-0-12-751955-5.50004-6. URL: https://www.sciencedirect.com/science/article/pii/B9780127519555500046.

[26] Marcus Bendtsen and Jose M Peña. "Gated Bayesian Networks". In: *Twelfth Scandinavian Conference on Artificial Intelligence: SCAI 2013*. Vol. 257. IOS Press. 2013, p. 35.

[27] Marcus Bendtsen and Jose M. Peña. "Gated Bayesian networks for algorithmic trading." In: *International Journal of Approximate Reasoning* 69 (2016), pp. 58–80. ISSN: 0888-613X.

[28] Stéphane Alarie, Charles Audet, Aımen E Gheribi, Michael Kokkolaras, and Sébastien Le Digabel. "Two decades of blackbox optimization applications". In: *EURO Journal on Computational Optimization* 9 (2021), p. 100011.

[29] Christopher KI Williams and Carl Edward Rasmussen. *Gaussian processes for machine learning*. Vol. 2. 3. MIT press Cambridge, MA, 2006.

[30] Kilian Weinberger. *"Lecture 15: Gaussian Processes"*. https://www.cs.cornell.edu/courses/cs4780/2018fa/lectures/lecturenote15.html. Accessed: 2022-11-11. 2018.

[31] E Brochu, V Cora, and N de Freitas. *A Tutorial on Bayesian Optimization of Expensive Cost Functions, with Application to Active User Modeling and Hierarchical Reinforcement Learning*. Tech. rep. 2009. URL: https://ora.ox.ac.uk/objects/uuid:9e6c9666-5641-4924-b9e7-4b768a96f50b.

[32] NBA Media Ventures LLC. *NBA Rulebook 2019-20*. 2020. URL: https://official.nba.com/rulebook/.

[33] Paola Zuccolotto, Marica Manisera, and Marco Sandri. "Alley-oop! Basketball analytics in R". In: *Significance* 18.2 (2021), pp. 26–31. DOI: https://doi.org/10.1111/1740-9713.01507. eprint: https://rss.onlinelibrary.wiley.com/doi/pdf/10.1111/1740-9713.01507. URL: https://rss.onlinelibrary.wiley.com/doi/abs/10.1111/1740-9713.01507.

[34] Melanie J Formentin. "Crisis Communication and the NBA Lockout: Exploring Fan Reactions to Crisis Response Strategies in Sport". In: *Reputational Challenges in Sport*. Routledge, 2018, pp. 117–134.

[35] González Dos Santos Teno, Chunyan Wang, Niklas Carlsson, and Patrick Lambrix. "Predicting Season Outcomes for the NBA". In: *Machine Learning and Data Mining for Sports Analytics*. Ed. by Ulf Brefeld, Jesse Davis, Jan Van Haaren, and Albrecht Zimmermann. Cham: Springer International Publishing, 2022, pp. 129–142. ISBN: 978-3-031-02044-5.

[36] Thomas Huyghe, Aaron T Scanlan, Vincent J Dalbo, and Julio Calleja-González. "The negative influence of air travel on health and performance in the national basketball association: a narrative review". In: *Sports* 6.3 (2018), p. 89.

[37] Dean Oliver. *Basketball on paper: rules and tools for performance analysis*. Potomac Books, Inc., 2004.

[38] Jeremy Arkes and Jose Martinez. "Finally, Evidence for a Momentum Effect in the NBA". In: *Journal of Quantitative Analysis in Sports* 7.3 (2011). DOI: doi:10.2202/1559-0410.1304. URL: https://doi.org/10.2202/1559-0410.1304.

[39] Sports Reference LLC. *Basketball-Reference.com - Basketball Statistics and History.* 2021. URL: https://www.basketball-reference.com/.

[40] V.I. Rodionov. "On the number of labeled acyclic digraphs". In: *Discrete Mathematics* 105.1 (1992), pp. 319–321. ISSN: 0012-365X. DOI: https://doi.org/10.1016/0012-365X(92)90155-9. URL: https://www.sciencedirect.com/science/article/pii/0012365X92901559.

[41] Thomas M Wagner, Alexander Benlian, and Thomas Hess. "Converting freemium customers from free to premium—the role of the perceived premium fit in the case of music as a service". In: *Electronic Markets* 24.4 (2014), pp. 259–268.

[42] James Cussens and Mark Bartlett. *GOBNILP: Globally Optimal Bayesian Network learning using Integer Linear Programming*. English. 2013.

[43] Mauro Scanagatta, Antonio Salmerón, and Fabio Stella. "A survey on Bayesian network structure learning from data". In: *Progress in Artificial Intelligence* 8.4 (2019), pp. 425–439.

A Appendix

A.1 Abbreviations, Notations and Basketball Miscellaneous

Table A.1: Description of Abbreviated Terms in the NBA Basic Game Log

Term	Description
FG	Field Goal
FGA	Field Goal Attempts
FG%	Field Goal Percentage
3P	3 Point Goals
3PA	3 Points Goal Attempted
3P%	3 Points Goal Percentage
FT	Free Throw
FTA	Free Throw Attempts
FT%	Free Throw Percentage
ORB	Offensive Rebound
DRB	Defensive Rebound
TRB	Total Rebound
AST	Assist
STL	Steal
BLK	Block
TOV	Turnover
PF	Personal Fouls

Table A.2: Description of Abbreviated Terms in the NBA Advanced Game Log

Term	Description
ORtg	Offensive Ratings
DRtg	Defensive Ratings
Pace	Pace Factor
FTr	Free Throw Attempt Rate
3PAr	3-Points Attempt Rate
TS%	True Shooting Percentage
TRB%	Total Rebound Percentage
AST%	Assist Percentage
STL%	Steal Percentage
BLK%	Block Percentage
eFG%	Effective Field Goal Percentage
TOV%	Turnover Percentage
ORB%	Offensive Rebound Percentage
DRB%	Defensive Rebound Percentage
FT/FGA	Free Throw per Free Goal Attempt

In table **??**, * refers to the teams formed after the Season 1980-81.

Table A.3: Features Utilised for SHAP Analysis

Type of Features	Features
Team's Overall Performance	Team_Prospect WinsInLast15 WinsInLast10 WinsInLast5
External Influence	Home_Game(If the game was played at Home, 1/0) Opponent_PlayOff (Opponent Team appeared in PlayOff in the Season, 1/0) Days_Between_Games
Team's Regular BoxScore	FG, FGA, FG%, 3P, 3PA, 3P%, FT, FTA, FT%, ORB, TRB, AST, STL, BLK, TOV, PF
Advanced BoxScore	ORtg, DRtg, Pace, FTr, 3PAr, TS%, TRB%, AST%, STL%, BLK%, eFG%, TOV%, ORB%, FT/FGA (Offensive Four Factors) eFG%, TOV%,DRB%,FT/FGA (Defensive Four Factors)

Table A.4: Championship Title Holder, from Season 1980-81 to Season 2020-21

Team	Number of Championship Titles
Boston Celtics	4
Chicago Bulls	6
Cleveland Cavaliers	1
Dallas Mavericks	1
Detroit Pistons	3
Golden State Warriors	3
Houston Rockets	2
Los Angeles Lakers	10
Miami Heat	3
Milwaukee Bucks	1
Philadelphia 76ers	1
San Antonio Spurs	5
Toronto Raptors	1

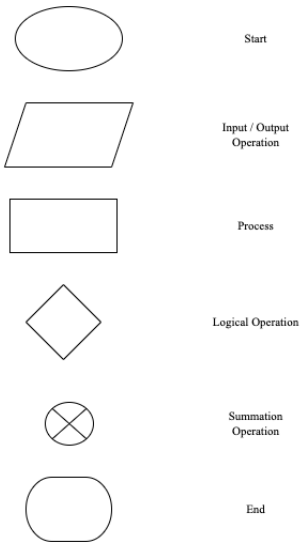


Figure A.1: Flow-Chart Notations

Table A.5: Team Play-Off Appearance, from Season 1980-81 to Season 2020-21

Team	Number of Play-Off Appearance
Atlanta Hawks	26
Boston Celtics	31
Brooklyn (New Jersey) Nets	21
Charlotte Hornets (Bobcats)*	10
Chicago Bulls	26
Cleveland Cavaliers	19
Dallas Mavericks	23
Denver Nuggets	24
Detroit Pistons	23
Golden State Warriors	13
Houston Rockets	29
Indiana Pacers	27
Los Angeles Clippers	13
Los Angeles Lakers	33
Memphis (Vancouver) Grizzlies*	11
Miami Heat*	22
Milwaukee Bucks	25
Minnesota Timberwolves*	9
New Orleans Pelicans (Hornets)*	7
New York Knicks	22
Oklahoma City Thunder (Seattle SuperSonics)	27
Orlando Magics*	16
Philadelphia 76ers	24
Phoenix Suns	25
Portland Trail Blazers	33
Sacramento (Kansas City) Kings	12
San Antonio Spurs	35
Toronto Raptors*	12
Utah Jazz	30
Washington Wizards (Bullets)	16

Table A.6: Mathematical Notation with Corresponding Explanations

Notation	Description
$\mathbb{N}$	Set of Natural Numbers (i.e. $1, 2, 3, ...$)
$E[.]$	Expectation
$\mathcal{D}$	Dataset
$\mathfrak{D}_{\mathfrak{S}}$	High Dimensional Dataset
$\mathcal{U}$	Uniform Probability Distribution
$\propto$	Proportional to
$\sim$	Distributed as
$\prod$	Product Operation
$\sum$	Summation Operation
$\in$	Belongs to, e.g. $1 \in \mathbb{N}$
$\subset$	Subset of, e.g. $\{1\} \subset \mathbb{N}$
$\subseteq$	Subset of or equal to
$\lfloor x \rfloor$	floor function of x, e.g. if $x = 3.9$ then $\lfloor x \rfloor = 3$
$\lceil x \rceil$	rounding function of x, e.g. if $x = 3.9$ then $\lceil x \rceil = 4$
$\setminus$	describes set subtraction, e.g. $\mathcal{N} \setminus \{1\}$ refers to natural numbers 2, 3, 4, ...
$\lvert\ \rvert$	absolute value, e.g. $\lvert -5 \rvert$=5, $\lvert 5.5 \rvert = 5.5$
!	factorial, e.g. $2! = 2 * 1 = 2$, $1! = 1$, $0! = 1$
$\mathcal{K}$	Placeholder Constant in equation 2.20
$\mathcal{Z}$	Normalising Constant in equation 2.22
$\mathcal{I}$	Indicator Variable
β_i	Index describing positions in $\mathcal{D}$
δ_i	Subsets of the dataset $\mathcal{D}$
$\mathcal{L}$	Learning Algorithm
$log(x)$	Natural Logarithm of x
r	Acceptance Ratio in Metropolis-Hastings Algorithm
$\mathcal{G}$	Graph
Bern(p)	Bernoulli Distribution with probability p
X_i	Random variable X_i, where $i \in \mathbb{N}$
$Pa(X_i)$	Parent of variable X_i
$\pi_{ik\|j}$	Condition probability in Discrete BN equation 2.24
Θ	Set of parameters in equation 2.23
$\Gamma(x)$	Gamma Function of x, $\Gamma(x) = \int_0^\infty t^{x-1} e^{-t} dt$, where $x > 0$
$\mathcal{GP}$	Gaussian Processes: $f \sim \mathcal{GP}(m(x), k(x, x'))$
σ_g^2	Variance of noise-free data
τ	Window length
θ	Threshold value for gate triggering mechanism
Λ	$\Lambda = (\tau, \theta)$
η	Exploitation - Exploration Tradeoff
ϕ	SHAP value

Table A.7: Acronyms of NBA Teams

Acronym	Team Name
ATL	Atlanta Hawks
BOS	Boston Celtics
NJN	Brooklyn (New Jersey) Nets
CHA	Charlotte Hornets (Bobcats)
CHI	Chicago Bulls
CLE	Cleveland Cavaliers
DAL	Dallas Mavericks
DEN	Denver Nuggets
DET	Detroit Pistons
GSW	Golden State Warriors
HOU	Houston Rockets
IND	Indiana Pacers
LAC	Los Angeles Clippers
LAL	Los Angeles Lakers
MEM	Memphis (Vancouver) Grizzlies
MIA	Miami Heat
MIL	Milwaukee Bucks
MIN	Minnesota Timberwolves
NOH	New Orleans Pelicans (Hornets)
NYK	New York Knicks
OKC	Oklahoma City Thunder (Seattle SuperSonics)
ORL	Orlando Magics
PHI	Philadelphia 76ers
PHO	Phoenix Suns
POR	Portland Trail Blazers
SAC	Sacramento (Kansas City) Kings
SAS	San Antonio Spurs
TOR	Toronto Raptors
UTA	Utah Jazz
WAS	Washington Wizards (Bullets)

Table A.8: Acronyms with Corresponding Description

Acronym	Description
NBA	National Basketball Association
NCAA	National Collegiate Athletic Association
SHAP	**SH**apley **A**dditive ex**P**lanation
XGBOOST	Extreme Gradient Boosting
OEIS	The **O**nline **E**ncyclopedia of **I**nteger **S**equences
DAG	Directed Acyclic Graph
BN	Bayesian Network
CPDAG	Complete partially Directed Acyclic Graph
$\mathcal{UB}$	Upper Bound
$\mathcal{LB}$	Lower Bound
GBN	Gated Bayesian Network
$\mathcal{UCB}$	Upper Confidence Bound Criterion
TL	Trigger Logic
ABN	Additive Bayesian Network
GOBNILP	**G**lobally **O**ptimal **B**ayesian **N**etwork learning using **I**nteger **L**inear **P**rogramming

A.2 Diagrams

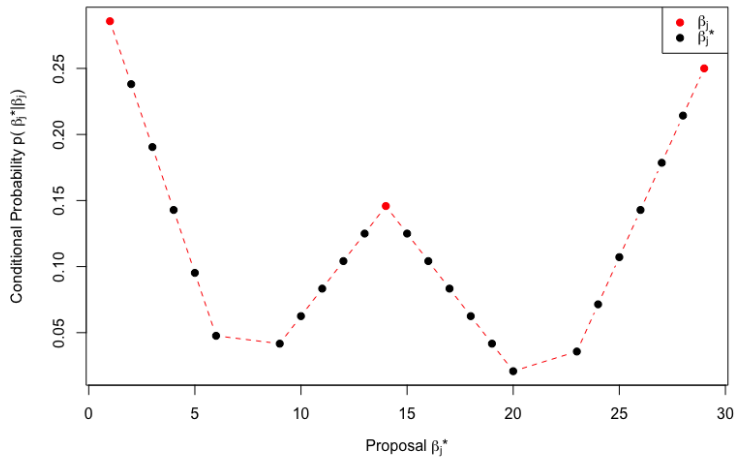


Figure A.2: Distribution of Proposal β_j^* for the current position,$\beta_1 = 1, \beta_2 = 14, \beta_3 = 19$

- In the figure A.2, distribution of proposal β_j^* is displayed when the current β_js are spread evenly across the index 1 to 29, where the conditional probability $p(\beta_j^*|\beta_j)$ is decreasing as moving away from the current β_js.
- Next in the figure A.3, β_2 and β_3 are relatively closer to each other, $p(\beta_j^*|\beta_j)$ is higher between these two β_js, however, the number for proposed β_j^*s are only two. $p(\beta_j^*|\beta_j)$ around β_1 is although relatively lower but the number of β_j^*s around β_1 is higher.
- Lastly, in the figure A.4, all three β_js are concentrated around the the center, $p(\beta_j^*|\beta_j)$ is zero, and $p(\beta_j^*|\beta_j)$ are decreasing slowly as moving away from β_1 and β_3.

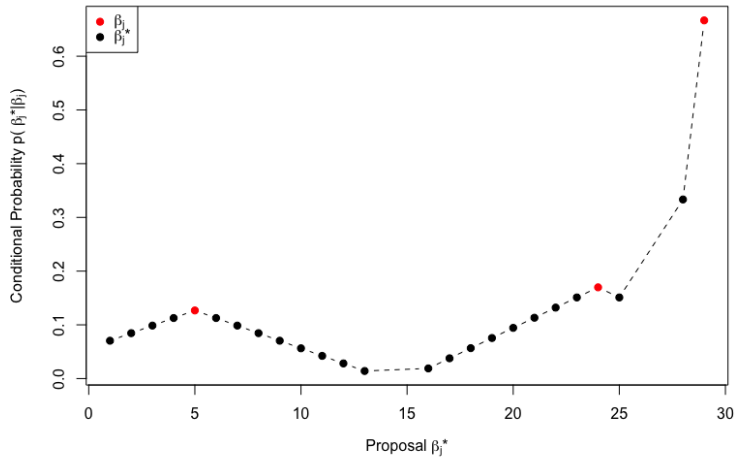


Figure A.3: Distribution of Proposal β_j^* for the current position,$\beta_1 = 5, \beta_2 = 24, \beta_3 = 29$

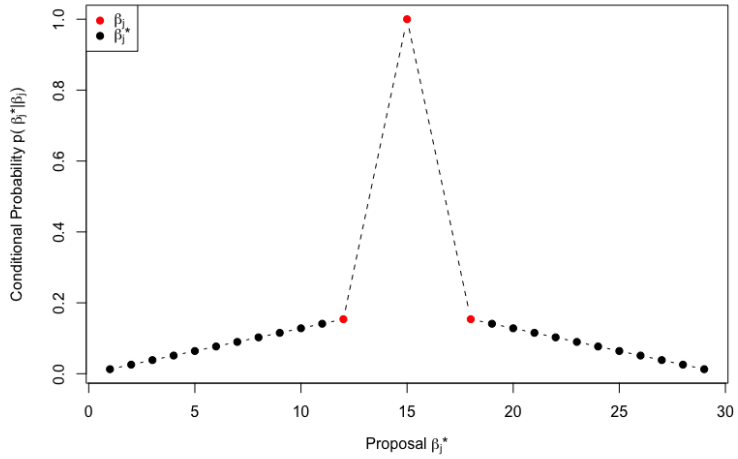


Figure A.4: Distribution of Proposal β_j^* for the current position,$\beta_1 = 12, \beta_2 = 15, \beta_3 = 18$

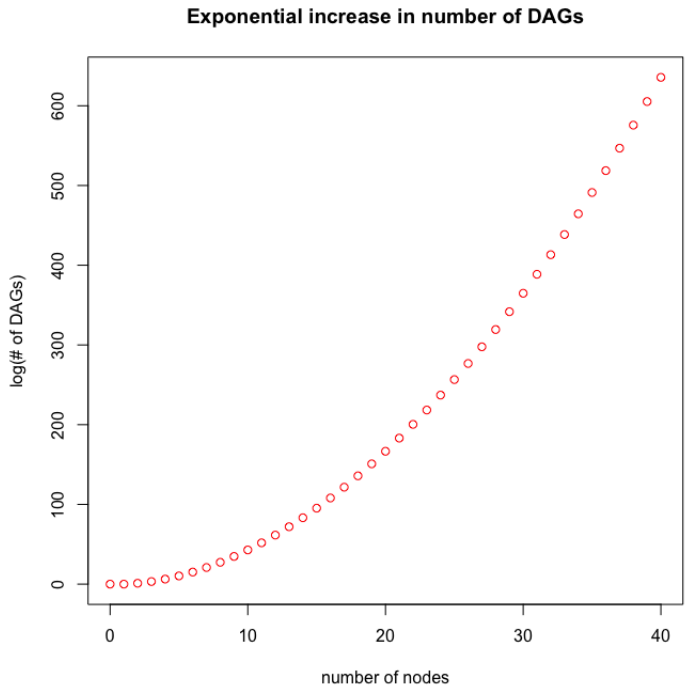


Figure A.5: Increase in Number of DAGs with increasing Number of Nodes

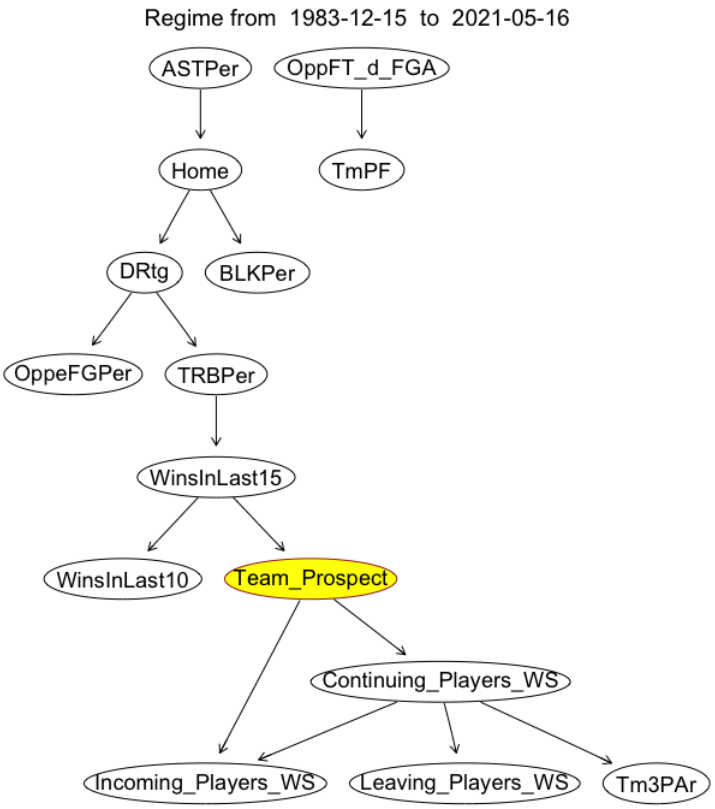


Figure A.6: Singular Bayesian Network based on entire Chicago Bulls using dataset $\mathcal{D}$