分类号_____ 学号 M200976059

学校代码 10487 密级_____

華中科技大學

硕士学位论文

基于贝叶斯网络的 NBA 比分预测和球员能力评估模型

学位申请人　牛兆捷

学 科 专 业：软件工程

指 导 教 师：沈　刚　教授

答 辩 日 期：2012.5.18

A Thesis Submitted in Partial Fulfillment of the Requirements for the Degree for the Master of Engineering

A Model of NBA Score Predication and Player Skills Evaluation Based on Bayesian Network

Candidate : Niu ZhaoJie

Major : Software Engineering

Supervisor : Prof. Shen Gang

Huazhong University of Science and Technology

Wuhan 430074, P. R. China

May, 2012

摘 要

体育比赛的胜负结果不仅取决于竞技双方的实力，还存在很大的不确定性，这种随机性使得胜负和比分的预测十分困难。近年来，很多研究者希望借助于比分的预测研究来加强人们对于复杂对象的理解。其中，概率统计方法被广泛应用到在胜负预测中：首先建立一个用于预测的模型，该模型根据当前球队的实力，计算比赛预期的胜负概率；然后通过球队之间的历史战绩数据进行训练，根据实际的比赛结果按照模型的实力更新规则计算出球队新的实力值。但是目前的方法主要是在队伍的级别上进行预测，没有在球员的级别上进行细化。

实际上，因为球队之间的比赛就是队员之间的对抗，通常情况下队员的能力值在攻防过程中往往会起到关键性的作用，决定比赛的胜负，但队员的发挥，上场时间，伤病情况，转会等因素也会对球队的实力造成影响，所以这些因素也不应该忽略。对比赛中的攻防情况和队员能力值建立合理的因果关系，从队员的能力值级别上进行建模，利用得分，命中率，篮板，抢断，失误等多个数据指标去对球员的能力进行评估，建立较为复杂的贝叶斯网络模型，考虑了球员上场时间，发挥等因素，并使用 EM 算法对球员能力参数值进行学习，然后用测试数据集进行比赛预测验证，以达到预期的效果。针对上述问题，提出了一个混合的模型：利用潜在变量建立一个比赛模型，得出在提倡比赛中，对手得分机会的期望值；利用一只团队的成员能力与得分之间的概率图模型，建立一个得分模型，得出不同队伍将得分机会转化为得分的期望；将二者结合起来，就可以预测比赛的得分。因此，并不是单纯地直接对胜负进行预测，而是通过概率统计的方法通过对历史详细战报进行分析，通过建立合理的概率图模型，对球队球员的各项能力值进行评估，并对比赛中每一次攻防过程中的得分情况进行预测。

使用 NBA 的比赛真实的比赛详细战报数据集，能有效的评估出球员的能力值，并且发现对球员能力值粒度细化的模型在比分预测应用中比粗粒度的模型更加准确。

关键词：贝叶斯网络　　潜在变量模型　　EM 算法

Abstract

The outcome of sports games quiz widespread daily life, but athletics is not only a strength factor, there is great uncertainty, this randomness makes forecasting very difficult, in this paper is not to predict the right outcome.analyzed by the history of the game battlefield but through the methods of probability and statistics through the establishment of a reasonable probabilistic graphical model, to assess the value of the team players, the intent to improve the prediction accuracy of forecasting for the rest of the score to make certaincontribution.

Probability statistics method is widely used in the outcome prediction by the historical record of data between team training, the establishment of the predicted probability model, the model is usually based on the current value of the team's strength, to calculate the probability of the outcome of the game expected, and then the team's new strength value calculated in accordance with the strength of the model update rules according to the actual results of the competition. The present method is mainly on the level of the team predicted that did not refine the player's level. In fact, however, consider more reasonable, because the competition between the teams is the confrontation between the players and capacity value of the players in the offensive and defensive process usually tend to play a key role, decided the game wins from the team level negative, but the play of the players, playing time, injuries, transfer and other factors will be the strength of the team's cause a great impact. Established the relationship between the team and the players, from the level of the players the ability to value modeling, the use of the score, hit rate, rebounds, steals, turnovers and other data indicators to assess the ability of players to establish a reasonable shellfish Bayesian network model, and use the EM algorithm to train the players ability to value and validate the test data set, in order to achieve the desired results.

Key words：Bayesian network　　Latent variable model　　Exception maximum

目 录

摘 要..... I

Abstract..... II

1 绪论

1.1 论文研究背景与意义 ..... (1)

1.2 国内外研究概况 ..... (1)

1.3 论文研究内容与组织结构 ..... (2)

2 相关研究和技术

2.1 贝叶斯估计与推论 ..... (5)

2.2 图模型 ..... (6)

2.3 潜在变量模型 ..... (8)

2.4 EM 算法 ..... (12)

2.5 非线性最小 2 乘法 ..... (14)

2.6 2 元响应的 LOGIT 和 PROBIT 模型 ..... (15)

2.7 本章小结 ..... (16)

3 比分预测与能力评估模型

3.1 任务目标和问题描述 ..... (17)

3.2 传统模型的缺陷分析 ..... (19)

3.3 改进的比分预测与能力评估模型 ..... (21)

3.4 模型参数的学习算法 ..... (27)

3.5 本章小结 ..... (30)

4 实验设计及结果分析

4.1 数据集 .....(32)

4.2 实验设计 .....(33)

4.3 结果分析 .....(38)

4.4 本章小结 .....(40)

5 总结与展望

5.1 全文总结 .....(41)

5.2 展望 .....(42)

致 谢.....(43)

参考文献.....(44)

1 绪论

1.1 论文研究背景与意义

竞技体育是受关注度和普及度最高的项目之一，其最大的特点就是不可预测性，因为高强度，快节奏的比赛，往往在一分钟甚至一秒钟的时间内，比赛的结果都有可能不同，影响胜负的因素很多，虽然说实力因素是主导因素，但是临场的发挥，心理因素，运气成分，一些突发的情况等也是决定比赛胜负的重要因素，而这些因素也是随机，不确定的，导致了竞技比赛的赛果难以预测。

近年来，体育竞技的分析已经演变成为一个重要的领域。球队的老板每年在各自球队身上花费成千上万的钱财，希望避免在那些对球队实力没有帮助的球员身上耗费资金。分析师花费大量的时间试图预测每年哪一支球队将赢得冠军，从而在商业上获取更高的经济效益，球迷也想在整个赛季赢得比赛竞猜的胜利。使用统计学，竞技体育已经从比赛转变成为一门学科，研究者使用大量的概率统计模型去对比赛进行分析，能够比较准确的预测出系列比赛之间的胜负，赛季的冠军，由于体育竞技多为团队之间的对抗，队员之间的配合也十分的重要，将队员的个人数据统计因素考虑进去，可以更加准确的分析出队员对胜负的影响，对球队的贡献。

1.2 国内外研究概况

对于球队 A 和球队 B 之间的比赛，通常有 2 种结果（A 胜利，A 失败），目前存在大量很好理解的技术将球员的表现压缩称为球队的实力，这种模型比较简单，将多个球员的参数近似为一个球队的整体实力参数，即将球员之间的配合和对抗抽象的看作是 2 支球队整体之间的对决。ELO[1,2,3]是最有名的这类预测方法。ELO 是一种概率模型，它将球队 A 战胜球队 B 的概率近似为这 2 支球队的实力差距的函数表示。随之很快，这种模型被优化为效果更好的 Bradley-Terry 模型[4,5]。其他的胜负预测模型，例如 Glicko[6,7]，使用逻辑与统计函数建立胜利的概率模型，2 者在分布上十分相似但是拥有不同的收敛属性，并且在尾部分布上也不同。

近期的预测系统发展涉及球队的行为和表现，实力影响表现的贝叶斯网络，并且将多种表现变量结合起来用以推断胜利的概率。TrueSkill[8,9,10]在单人或者团队的竞技项目中均可以使用，使用了 Gaussian 分布作为实力和表现的先验假设。Whole-History Rating[11,12,13]则侧重于随着时间推移动态变化的实力参数上，统计了球员随着经验的增长的能力值的改进和随着年龄衰老的退步。

最初，Elo 系统主要用于在 2 个球员的比赛项目中计算球员的相对实力等级，但是已经被适用到团队的运动项目中去，例如足球，篮球和棒球等。球员或者球队的表现通过与其他球员或者球队的胜，负，平交战记录中推测出来，但是依赖于对手的排名和与其交锋的比分情况。Elo 仅仅考虑最终的比分情况，但是最终的比分往往并不会反映出比赛的过程，并且当用于球队的情况时，也只是将球队看作是一个整体而没有单独考虑每个球员的表现。因此，从比赛的结果中并不能体现出球员的价值，无法对球员的能力值进行评估。近年来，随着 TrueSkill 模型的提出，引入了球员能力值评估的概念，通过对球员能力值的学习，对交锋双方的得分情况进行预测，球员能力值的学习过程是采取的贝叶斯推论的方法，在 TrueSkill 模型中使用的是 Expectation Propagation[14,15,16]算法，经实验验证，其预测准确率为 64.42%。但是其并没有对球员的能力值进行细化，仅仅只有一个球员能力值变量，而赛场上的所有攻防情况都共享依赖于该变量。本文试图构建一种更加复杂的模型，使用贝叶斯网络对每位球员建立 6 种不同的实力参数（3 个进攻参数和 3 个防守参数）。通过这样做，我们能够在每一次攻防过程中对每个球员对球队的贡献上做出更好的推论。论文[20]对 TrueSkill 模型进行改进，提出了一种新的贝叶斯推论[17,18,19]的近似技术，其预测准确率为 64.56%。论文[21,25,26]则是使用一种矩阵分解的方法，利用高斯过程加入了影响比赛比分结果的因素，最高准确率达到了 70%左右。

篮球这种团队竞技也给基于贝叶斯预测领域带来了更多关键的挑战：比赛的结果不是 2 元的胜负输出；仅仅依靠比赛的结果值几乎无法区分球员之间的能力值；结果被多于 2 个球员的实力差异所影响；一个球员的实力值至少拥有 2 个参数-进攻与防守。

1.3 论文研究内容与组织结构

本文给出了胜负预测问题的形式化描述，分析了该解决该问题的难点：随机性

因素过多，本文在已有模型和技术的基础上，提出了用于预测的比赛模型和球队模型。比赛模型对比赛进行建模，其实质是一个潜在变量模型，试图通过球队风格，战术体系，主客场等可能影响比赛的潜在因素对能对比赛中各球队的进攻的回合数进行预测。球队模型则是对球员进行能力值评估的模型，其实质是一个贝叶斯网络，试图通过完整的赛况数据集，利用 EM 算法对各队球员的能力值进行学习，对球员能力值进行评估，然后结合球员上场的组合情况对回合的得分概率和球队得分期望进行计算，最后根据比赛模型中得到的回合数预测实现比分的估计。我们的工作不仅提出了一个比赛模型，还对传统的基于球员能力值的预测模型进行了改进，我们对传统模型在某个攻防过程中的得分状态进行了拆解，细化了球员的能力值参数，将得分状态和相应的球员能力值联系起来，建立了合理的球员能力值和比赛中得分状态的因果关系。在球队模型参数学习的过程中，分别假设了服从了 Logit 和 Probit 分布进行实现，并对其效果进行了比较。使用 NBA 三个赛季的真实数据集，将我们的模型和已有的贝叶斯网络模型进行比较，经实验证明，我们的模型在对比赛过程预测的准确率高于已有模型。

全文一共分为五个章节。

第一章为绪论。叙述了胜负预测和对球员能力值进行评估的现实意义和价值，分析了其面临的困难问题，并且简单的相关工作的研究现状。

第二章为相关技术的详细介绍。首先详细介绍了贝叶斯估计和推论技术，列举了出了主流的两类计算方法，并进行简要的介绍。接着介绍了图模型，并举例说明。然后介绍了比赛模型中所采用的潜在变量模型，并介绍了对潜在变量参数进行学习的非线性最小二乘法。接下来，介绍球队模型中用到的一种贝叶斯推论算法，Expectation Maximum 算法，并对其迭代求解的过程进行了详细的介绍。最后简单介绍了球队模型中计算所假设的 Logit[30,31]和 Probit[32,33]模型。

第三章介绍了贝叶斯实力评估模型，首先对我们的任务目标进行描述并对已存在的难点进行分析，然后对目前的研究工作中存在的问题进行了列举，并提出了改进的方向。接着提出了我们提出了基于潜在变量的比赛模型，并对基于贝叶斯网络模型的球队模型进行改进，然后对 2 个模型中参数学习算法进行了详细的描述。

第四章是实验设计和结果分析，首先对我们的数据集进行介绍，然后对实验设计和评估方法进行了介绍，对已有模型和我们提出的模型进行的对比，最后列举出了实验结果并用图表形式进行展示。

第五章是总结与展望，总结了论文的核心内容，并在展望中提出了值得进一步深入研究的方向。

2 相关研究和技术

为了后续讨论顺利进行，本章介绍六种关键技术的相关研究工作，分别是贝叶斯推论，图模型，潜在变量模型，EM 算法，非线性最小二乘法和 2 元响应的 Logit 和 Probit 模型。

2.1 贝叶斯估计与推论

假设 x 是观测值并且 θ 是生成 x 的模型的未知参数。贝叶斯估计即指参数估计，是指从不完整，不确定并且带有噪声的数据中计算出参数的近似值。贝叶斯推论用于随机变量，其过程是使用先验信念和给出的观测值 x 推导出随机变量θ的后验概率 $\rho(\theta \mid x)$。

最常见的参数估计方法是 ML（极大似然[22]）。根据这种方法，ML 估计按照（2.1）式获取

$$\theta_{ML} = \arg\max_{\theta} \rho(x;\theta), \tag{2.1}$$

基于生成观测值 x 的假设模型，$\rho(x;\theta)$ 描述了观测值与参数之间的概率关系。$\rho(x;\theta)$ 是一个由参数 θ 所表示的函数，称为似然函数。

在很多情况下，直接计算似然函数 $\rho(x;\theta)$ 是复杂，困难甚至是不可能去直接计算它。在这种情况下，通过引入隐变量 z 来辅助计算似然度。这些引入的随机变量使用贝叶斯定律作为连接观测值和未知变量的链接。因变量的选择是视问题而定的。然而，正如隐变量的名字，这些变量是无法被观测到的并且它们提供了足够的关于观测值的信息所以条件概率 $\rho(x \mid z)$ 容易计算得到。

一旦隐变量和关于它们的先验概率 $\rho(z;\theta)$ 被引入，就能得到似然度或者称为边缘似然度，因为如（2.2）式所示，它通过对全部隐变量求积分得到

$$\rho(x;\theta) = \int \rho(x,z;\theta)dz = \int \rho(x \mid z;\theta)\, p(z;\theta)dz \tag{2.2}$$

这个看似简单的积分是贝叶斯方法的关键，因为就是依靠这种方法，我们不仅

能够获取似然函数，而且通过使用贝叶斯定律，根据公式（2.3），隐变量的后验概率也可以计算出来。

$$\rho(z|x,\theta)=\frac{\rho(x|z,\theta)p(z;\theta)}{p(x;\theta)} \tag{2.3}$$

一旦后验概率是可利用的，对于隐变量的推论就可以按照（2.3）式计算得到。尽管上面的方法看着十分简单，但是在多数情况下（2.2）式中的积分难以或者无法计算。因此，贝叶斯推论的主要研究就集中在寻找避开或者近似逼近积分的方法上。

这样的方法主要分为 2 类。第一种是数值采样[23,24,25]的方法被称为蒙特卡洛[25,26]技术，第二种是确定性近似的方法。EM 算法就是属于蒙特卡洛家族的贝叶斯推论方法，它假设了后验知识 $\rho(z|x;\theta)$ 然后迭代的去最大化似然函数而不是准确的去计算。这个方法的一个严重不足之处就是很多情况下这种后验并不能获取。然而，贝叶斯的近期研究通过近似获取这个后验概率绕开了这个困难。这种方法称为变分贝叶斯。

2.2 图模型

图模型为在统计模型难题提供了一种表示随机变量之间的依赖的框架并且提供了直观的方式表示概率系统中实体之间的交互关系。图模型使用节点表示随机变量，边则表示随机变量之间的依赖。从节点 A 指向节点 B 的有向边表示变量 B 依赖于随机变量 A 的值。图模型分为有向图和无向图，无向图的也通常被称为马尔科夫随机域[27,28,29]，而有向图模型，也通常被称为贝叶斯网络，所有的边从父节点到儿子节点，表示了对应随机变量的条件依赖。除此之外，还做了有向五环图的假设。

定义 $G=(V,E)$ 是有向无环图，V 是节点的集合，E 是有向边的集合。令 χ_s 表示与节点 s 相关联的随机变量并且 $\pi(s)$ 是节点 s 的父节点的集合。与每个节点 s 相关联的也是一个条件概率密度 $\rho(\chi_s|\chi_{\pi(s)})$，其定义了给出其父变量的值后关于 χ_s 的分布。因此，对于一个完整定义的图模型，除了其图的结构，每一个节点的条件概率分布也应该被指定。一旦这些分布已知了，所有变量集合的联合分布如（2.4）的乘积形式：

$$\rho(x)=\prod_s \rho(\chi_s|\chi_{\pi(s)}) \tag{2.4}$$

有向图模型是一个概率分布的集合，依赖于具体图的结构，按照上式的方式进行分解。图 2.1 是一个有向图模型的例子。节点 a，b，c，d 随机变量，每个节点表示了依赖于其父节点的条件概率。使用概率的链式规则，图 2.1 的联合概率如（2.5）式所示：

$$\rho(a,b,c,d;\theta)=\rho(a;\theta_1)\rho(b\mid a;\theta_2)\rho(c\mid a,b;\theta_3)\rho(d\mid a,b,c;\theta_4) \tag{2.5}$$

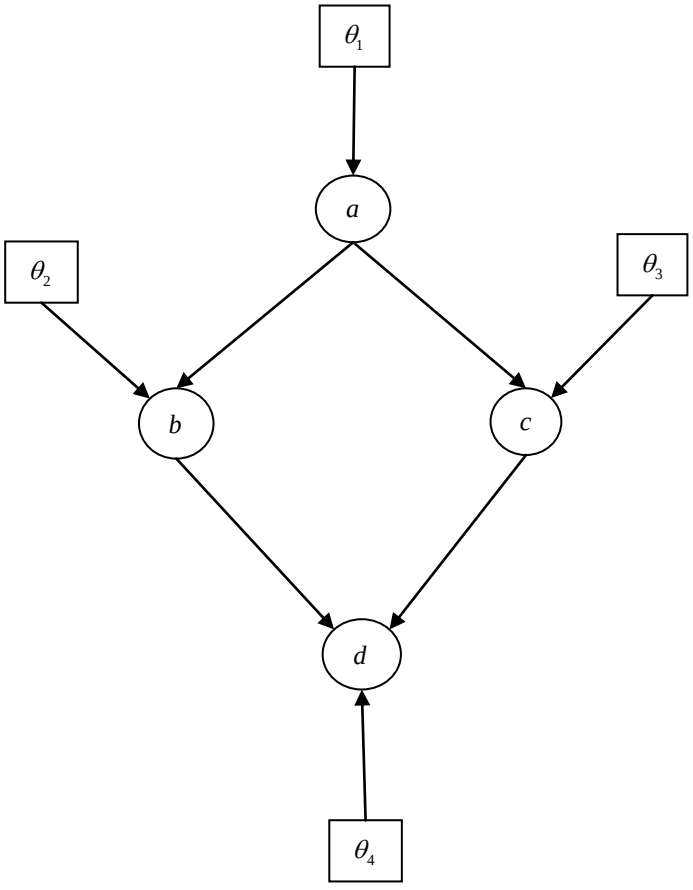

图 2.1　图模型

在图模型中，随机变量分为可观测的和不能直接观测的隐变量，通常隐变量作为计算中间的采样和计算步骤最终生成可观测的变量。图模型分为参数化或者非参数化的两类。假如模型是非参数化的，即参数已经完全已知，可以应以一些推论问题，例如计算随机变量子集的边缘分布，计算在给定剩余变量的情况下计算变量子

集的条件分布等。假如模型是参数化的，即参数出现在一些图节点的条件概率分布中，则就需要在给定观测数据集的情况下学习参数的值，通常在这个参数学习的过程中，有一些已有的推论技术，2.3 节中，我们将介绍一种的代表 EM 算法。

2.3 潜在变量模型

在模式识别和机器学习领域的其中一个难题是密度估计，即是给定一个来自于某个分布的优先数据样例，构造一个关于一个概率分布的模型。在这节中，我们将考虑建立连续变量分布模型的问题，连续变量$t_1,...,t_d$用向量$\vec{t}$表示。

密度估计的标准方法涉及参数估计模型，其密度的表达式包括大量参数，这些参数值由包含 N 个数据向量可观测数据集$D=\{t_1,...,t_N\}$所决定。最为广泛使用的参数模型是正态分布，表示为：

$$\rho(t\mid\mu,\Sigma)=(2\pi)^{-d/2}\left|\Sigma\right|^{-1/2}\exp\left\{-\frac{1}{2}(t-\mu)\Sigma^{-1}(t-\mu)^{\mathrm{T}}\right\} \tag{2.6}$$

μ是均值，Σ是协方差矩阵，$\left|\Sigma\right|$表示矩阵Σ的行列式。求解这些参数值的其中一种技术是极大似然估计，例如给定参数的关于观测数据的概率对数：

$$L(\mu,\Sigma)=\ln\rho(D\mid\mu,\Sigma)=\sum_{n=1}^{N}\ln\rho(t_n\mid\mu,\Sigma) \tag{2.7}$$

假设数据向量t_n独立于分布进行采样。表达式$\rho(D\mid\mu,\Sigma)$被视为关于μ和Σ的函数，被称为似然函数。最大化关于μ和Σ似然函数（或对数似然函数），即给定可观测数据集，求解参数。对于符合正态分布的对数似然函数，使其最大化的相应μ和Σ表示为：

$$\mu=\frac{1}{N}\sum_{n=1}^{N}t_n \tag{2.8}$$

$$\Sigma=\frac{1}{N}\sum_{n=1}^{N}(t_n-\mu)(t_n-\mu)^{\mathrm{T}} \tag{2.9}$$

各自对于样本均值和样本方差。除了极大似然估计，我们也能使用贝叶斯理论在$\vec{\mu}$和$\vec{\Sigma}$上定义先验，与可观测的数据一起，决定后验概率，对于这个正态分布的

贝叶斯推论的介绍在中给出。

尽管简单的正态分布被广泛使用，它存在于一些明显的限制。特别的，它通常由于模型中独立参数过多变得过于复杂。这个问题可能用过引入连续隐变量解决。另一方面，正态分布由于只能表示一元模态的分布显得不够灵活。一个更为一般的分布家族能够通过采用高斯混合模型获得，对应于离散隐变量。

考虑正态分布中的参数数量。因为$\sum$是对称的，它包含$d(d+1)/2$个独立的参数，再加上μ中的d个独立参数，使得总共有$d(d+3)/2$个参数。对于比较大的数d，由于参数过多，需要大量的数据点用于极大似然估计。一种减少模型参数的方式就是考虑一个对角协方差矩阵，其仅仅拥有d个参数。但是这里做了一个很严格的假设，就是观测数据t的组成在统计上是独立的，因此就无法获得这些不同组成之间的关系。

既要使模型的自由程度可控制，又要保证其关联性，所以引入了潜在（隐）变量。潜在变量模型的目标即是将表示分布$\rho(t)$的变量$t_1,...,t_d$，替换为数目更少的潜在变量$x=(x_1,...,x_q)$，这里$q<d$。这通过首先将联合概率分布$\rho(t,x)$分解成为潜在变量的边缘分布$\rho(x)$和给出隐变量关于数据变量的条件分布$\rho(t|x)$的乘积来实现。通常比较方便假设条件分布在数据变量之上进行分解，所以联合分布变成：

$$\rho(t,x)=\rho(x)\rho(t|x)=\rho(x)\prod_{i=1}^{d}\rho(t_i|x) \qquad (2.10)$$

这个分解的属性能使用贝叶斯网络进行图形化的表示，如图 2.2 所示。

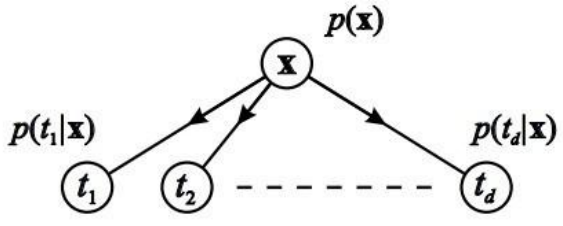

图 2.2　潜在变量分布的贝叶斯网络

公式（2.10）给出的关于潜在变量分布的贝叶斯网络表示，给出隐变量x，数据变量$t_1,...,t_d$是独立的。

紧接着，我们将条件分布 $\rho(t|x)$ 表示为从潜在变量到数据变量的映射，所以有：

$$t = y(x;w) + u \tag{2.11}$$

这里 $y(x;w)$ 是一个关于参数 w 的潜在变量 x 的函数，u 是一个独立于 x 的噪声过程。假如 u 的组成是不相关的，对于 t 的条件分布将按照（2.10）式进行分解。从几何学上，函数 $y(x;w)$ 定义了一个在数据空间中的流形，如图 2.3 所示。

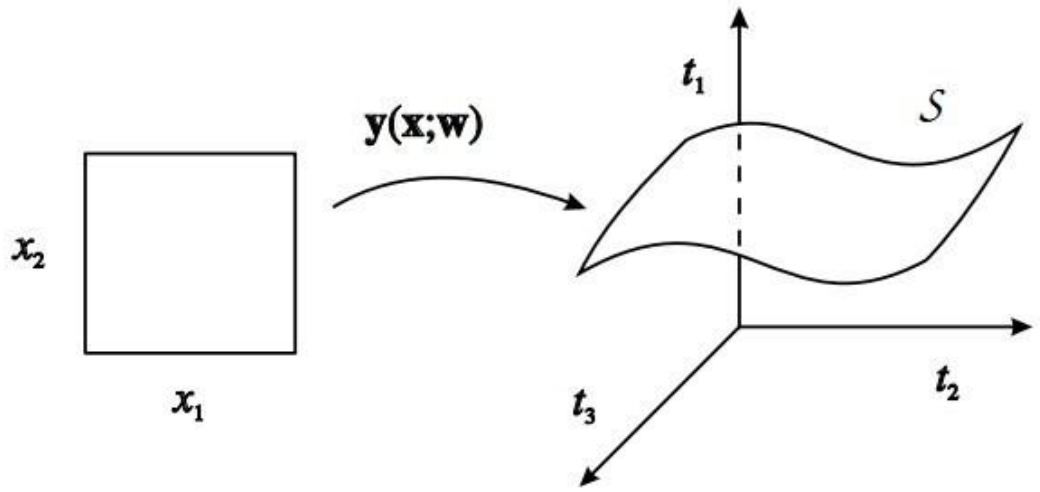

图 2.3　非线性函数 $y(x;w)$ 定义了一个流形 S，嵌入在映射 $x \to y$ 的数据空间

潜在变量模型的定义通过指定分布 $\rho(u)$，映射 $y(x;w)$ 和边缘分布 $\rho(x)$ 来完成。正如我们后面将介绍的，十分方便将 $\rho(x)$ 看作是基于潜在变量上的先验分布。

对于数据分布 $\rho(t)$ 的模型通过边缘化潜在变量获得：

$$\rho(t) = \int \rho(t|x)\rho(x)dx \tag{2.12}$$

这个积分，除非使用特殊形式的分布 $\rho(t|x)$ 和 $\rho(x)$，否则分析会十分困难。

最简单的一种潜在变量模型被称为因子 分析，其基于一种线性映射 $y(x;w)$，使得：

$$t = Wx + \mu + u \tag{2.13}$$

W 和 μ 是自适应的参数。分布 $\rho(x)$ 被当做一个 0 均值的单位协方差高斯分布 $\mathbb{N}(0,\mathrm{I})$，然而对于 u 的噪声模型也是一个带有协方差矩阵 Ψ 的 0 均值高斯，Ψ 是对

角矩阵。由公式（2.12）易知分布 $\rho(t)$ 也是高斯分布，均值为 μ，协方差矩阵为 $\psi+WW^{\mathrm{T}}$。模型的参数，由 W,Ψ 和 μ 组成，能够被极大似然估计所决定。然而没有一个直接求解的形式，所以它们的值只能够通过迭代的过程来进行求解。对于 q 个隐变量，在 W 中有 $q\times d$ 个参数,在 ψ 中有 d 个，μ 中有 d 。.在这些参数之间也存在一些冗余，并且一个更加仔细的研究分析表明在这个模型中独立参数的数量为：

$$(d+1)(q+1)-q(q+1)/2 \qquad (2.14)$$

可见总的参数数量按照 d 成线性增长，并且模型仍然保持了数据变量之间的主要关系。

到目前所考虑的密度模型十分局限，尽管他们可以建立多样的概率分布，但是他们仅仅能够表示一元模态的分布。然而，通过考虑 M 个更加简单的参数分布的概率混合，它们能够形成密度模型的通用的表达形式。其密度模型的形式为：

$$\rho(t)=\sum_{i=1}^{M}\pi_i\rho(t\,|\,i) \qquad (2.15)$$

其中 $\rho(t\,|\,i)$ 表示混合模型的独立组成成分，并且可能由公式（2.6）的正态分布的形式组成，每个拥有各自独立的均值 μ_i 和协方差矩阵 $\sum_i$。在（2.15）中的参数 π_i 被称为混合协同系数并且满足条件 $0\le\pi_i\le 1$，$\sum_i\pi_i=1$，所以 $\rho(t)$ 为非负数并且积分为1（假设各自独立密度函数也拥有这些属性）。将（2.15）中的表示为一个简单的贝叶斯网络，如图 2.4 所示。

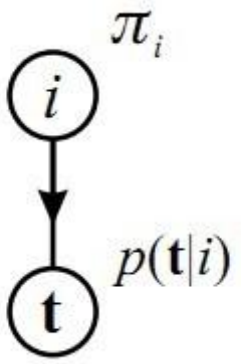

图 2.4　简单混合分布的贝叶斯网络表示

对于标签i的值，混合协方差系数能够表示为先验概率。对于一个给定的数据点t_n，我们能使用贝叶斯理论去计算其对应的后验概率，如式（2.16）所示。

$$R_{ni} \equiv \rho(i \mid t_n) = \frac{\pi_i \rho(t_n \mid i)}{\sum_j \pi_j \rho(t_n \mid j)} \tag{2.16}$$

混合分布的对数似然函数为：

$$L(\{\pi_i, \mu_i, \Sigma_i\}) = \sum_{n=1}^{N} \ln \left\{ \sum_{i=1}^{M} \pi_i \rho(t \mid i) \right\} \tag{2.17}$$

最大化混合情况下的对数似然函数比一个单独的组成更加复杂，因为在对数中出现了求和运算。一种执行该优化过程的算法被称为期望最大化算法（EM），将在2.4 节中详细介绍。EM 算法基于观察值，假如我们给出了指示器变量的集合z_{ni}表示哪个成分i负责生成每一个数据点t_n，对数似然函数变为：

$$L_{comp}\left(\{\pi_i, \mu_i, \Sigma_i\}\right) = \sum_{n=1}^{N} \sum_{i=1}^{M} z_{ni} \ln \{\pi_i \rho(t \mid i)\} \tag{2.18}$$

它的优化过程是比较简单的，其结果是每一个成分独立适应于数据点的相应组，并且混合协方差系数通过每一个组中的小部分点所给定。

$\{z_{ni}\}$被视为缺失的数据，并且数据集$\{t_n\}$也是不完整的。将$\{t_n\}$和$\{z_{ni}\}$组合起来，我们得到了相应完整的数据集，对应的对数似然估计由（2.18）所表示。当然，$\{z_{ni}\}$的值是未知的，但是他们的后验分布能够使用贝叶斯理论计算得到，并且在这个分布下z_{ni}的期望恰恰是式（2.16）R_{ni}的集合。EM 算法基于最大化由式（2.18）给出的完整数据的对数似然度。

$$\left\langle L_{comp}\left(\{\pi_i, \mu_i, \Sigma_i\}\right)\right\rangle = \sum_{n=1}^{N} \sum_{i=1}^{M} R_{ni} \ln \{\pi_i \rho(t \mid i)\} \tag{2.19}$$

其在 E-step 和 M-step 之间切换，E-step 使用公式（2.16）计算R_{ni}的值，M-step 中，给出改进的参数值集合，使（2.19）最大化。在 EM 算法的每一次循环过程中，对数似然度逐步增加直到其达到局部最大。

2.4 EM 算法

对（2.2）式的似然函数取对数得到对数似然函数， 如 2.20 式所示。

$$\ln \rho(x;\theta) = F(q,\theta) + KL(q \| \rho) \tag{2.20}$$

其中

$$F(q,\theta) = \int q(z) \ln(\frac{\rho(x,z;\theta)}{q(z)}) dz \tag{2.21}$$

$$KL(q \| p) = -\int q(z) \ln(\frac{p(z|x;\theta)}{q(z)}) dz \tag{2.22}$$

其中 $q(z)$ 是任意的概率密度函数。$KL(q \| \rho)$ 是 $\rho(z|x;\theta)$ 和 $q(z)$ 之间的 Kullback-Leibler[39,40]差异并且因为 $KL(q \| p) \geq 0$，保证了 $\ln \rho(x;\theta) \geq F(q,\theta)$。可见，$F(q,\theta)$ 是对数似然函数的下界，当且仅当 $KL(q \| \rho) == 0$ 时，$\rho(z|x;\theta) = q(z)$。EM 算法和相关贝叶斯推论的确定性近似的近期进展可以被看作是有关密度 q 和参数 θ 最大化下界 $F(q,\theta)$ 的过程。

EM 算法是一种最大化下界 $F(q,\theta)$ 的 2 步（E 步骤和 M 步骤）迭代算法。假设当前的参数值是 θ^{OLD}。在 E 步骤中，下界 $F(q,\theta^{OLD})$ 被最大化。当 $KL(q \| \rho) == 0$ 时，其取得最大值，此时的 $q(z) = \rho(z|x;\theta^{OLD})$。在这种情况下，下界等于对数似然函数。在接下来的 M 步骤中，$q(z)$ 是固定值，考虑 θ 的变化，给定新的 θ^{NEW} 值使得下界 $F(q,\theta)$ 被最大化。使得下界增长的结果也会使得对应的对数似然度增长。因为 $q(z)$ 是由 θ^{OLD} 来决定的并且在 M 步骤中保持不变，它也不等于新的后验概率 $\rho(z|x;\theta^{NEW})$，因此 KL 距离将不再为 0。这样，对数似然度的增长大于下界的增长。假如我们将 $q(z) = \rho(z|x;\theta^{OLD})$ 带入下界公式中对（2.21）进行展开得到

$$\begin{aligned} F(q,\theta) &= \int \rho(z|x;\theta^{OLD}) \ln \rho(x,z;\theta) dz - \int \rho(z|x;\theta^{OLD}) \ln \rho(z|x;\theta^{OLD}) dz \\ &= Q(\theta,\theta^{OLD}) + const \end{aligned} \tag{2.22}$$

常量是 $\rho(z|x;\theta^{OLD})$ 的熵，函数

$$Q(\theta,\theta^{OLD})=\int \rho(z\mid x;\theta^{OLD})\ln\rho(x,z;\theta)dz \tag{2.23}$$

是对数似然度的期望，在 M 步骤最大化。总结起来说，EM 算法是一种迭代算法，涉及 2 个步骤：

E-步骤：计算

$$\rho(z\mid x;\theta^{OLD}) \tag{2.24}$$

M-步骤：求值

$$\theta^{NEW}=\arg\max_{\theta}Q(\theta,\theta^{OLD}) \tag{2.25}$$

2.5 非线性最小 2 乘法

考虑 m 个数据点的集合，$(x_1,y_1),(x_2,y_2),...,(x_m,y_m)$，曲线（模型函数）为 $y=f(x,\beta)$，依赖于变量 x 也依赖于 n 个参数，$\beta=(\beta_1,\beta_2,...,\beta_n)$，$m\geq n$。最小二乘法意图找到合适的参数矩阵拟合曲线以最佳适应给定的数据集，即使得错误 r_i 的平方和最小：

$$S=\sum_{i=1}^{m}r_i^2 \tag{2.26}$$

$$r_i=y_i-f(x_i,\beta) \tag{2.27}$$

其中 $i=1,2,...,m$。

S 的最小值在梯度为 0 的时候取得，因为模型包括 n 个参数，所以拥有 n 个梯度等式：

$$\frac{\partial S}{\partial\beta_j}=2\sum_i r_i\frac{\partial r_i}{\partial\beta_j}=0 \qquad (j=1,...,n) \tag{2.28}$$

在一个非线性系统中，$\frac{\partial r_i}{\partial\beta_j}$ 是关于独立变量和参数的函数，这些等式不能直接求解。取而代之，为这些参数选择初值，然后这些参数在迭代过程中不断更新，参数值通过迭代逐次逼近进行求解。

$$\beta_j\approx\beta_j^{k+1}=\beta_j^k+\overrightarrow{\Delta\beta}_j \tag{2.29}$$

这里，k 是迭代次数，向量增量 $\overrightarrow{\Delta\beta}$ 是一个移位向量。在每一次迭代过程中，模型按一阶泰勒展开式近似线性化，展开式的参数为 β^k

$$f(x_i,\beta)\approx f(x_i,\beta^k)+\sum_j\frac{\partial f(x_i,\beta^k)}{\partial\beta_j}(\beta_j-\beta_j^k)\approx f(x_i,\beta^k)+\sum_j J_{ij}\Delta\beta_j \qquad (2.30)$$

雅克比行列式 J，是一个关于常量，独立变量和参数的函数，所以它在迭代过程中不断变化。因此，按照线性模型，$\frac{\partial r_i}{\partial\beta_j}=-J_{ij}$，误差表示为：

$$r_i=\Delta y_i-\sum_{s=1}^{n}J_{is}\Delta\beta_s;\ \Delta y_i=y_i-f(x_i,\beta^k) \qquad (2.31)$$

替换这些值为梯度方程，有

$$-2\sum_{i=1}^{m}J_{ij}(\Delta y_i-\sum_{s=1}^{n}J_{is}\Delta\beta_s)=0 \qquad (2.32)$$

重新整理，得到 n 维线性方程组，其标准方式为：

$$\sum_{i=1}^{m}\sum_{s=1}^{n}J_{ij}J_{is}\Delta\beta_s=\sum_{i=1}^{m}J_{ij}\Delta y_i\quad(j=1,...,n) \qquad (2.33)$$

使用矩阵符号的形式表示为：

$$(J^{\mathrm{T}}J)\Delta\beta=J^{\mathrm{T}}\Delta y \qquad (2.34)$$

2.6 2 元响应的 Logit 和 Probit 模型

线性概率模型[37,38]存在局限性，主要表现为如下 2 点：

（1）被预测的隐变量可能不被支持

（2）对于所有的解释变量来说部分影响是常量

在 2 元响应模型中，最为关键的就是响应概率

$$\Pr(y=1|x)=\Pr(y=1|\chi_1,\chi_2,...,\chi_k) \qquad (2.35)$$

为了消除线性概率模型的限制，做如下假设：

$$\Pr(y=1|x)=F(\beta_0+\beta_1x_1+\beta_2x_2+...+\beta_kx_k) \qquad (2.36)$$

这里函数 $F(.)$ 是一个函数：$F:x\to[0,1],\forall x\in R$，对于 $F(.)$ 有多种函数，最常用

的有两类，即下面进行介绍的 Logit 模型和 Probit 模型。

首先是 Logit 模型，即假设函数 $F(.)$ 符合一个逻辑分布，

$$F(x)=\frac{\exp(x)}{1+\exp(x)} \tag{2.37}$$

另一个就是 Probit 模型，即假设函数 $F(.)$ 符合一个正态分布，

$$F(x)=\phi(x)=\int_{-\infty}^{x}\varphi(z)dz \tag{2.38}$$

$\varphi(z)$ 为正态密度函数。

2.7 本章小结

这一章我们主要介绍四种相关技术，分别是贝叶斯估计与推论，图模型，EM算法和 2 元响应的Logit和Probit模型。在贝叶斯网络中，介绍了其估计与推论的具体过程，并将其与图模型进行联系，同时对有向无环图进行了介绍。接着介绍了比赛模型中使用的潜在变量模型方法，并对其潜在变量拟合的非线性最小 2 乘法进行了介绍。然后对球队模型关于球员能力值的推论算法EM算法的计算过程进行了详细说明，其是球员能力值参数学习过程中的主要算法。最后简要介绍了 2 元响应的Logit和Probit模型，在球队模型中，在训练和推论的过程中都需要对比赛回合中得分的情况进行计算，计算过程中所假设的模型就是这里介绍了高斯和Logit模型。

3 比分预测与能力评估模型

在本章，我们将详细介绍我们提出的基于贝叶斯网络的球员实力评估模型。3.1 节介绍我们的任务目标并对问题进行描述。3.2 节分析了传统模型的不足之处。3.3 节将详细介绍我们提出的基于贝叶斯网络的比分预测和球员能力评估模型，其分为比赛模型和球队模型，我们将分别进行介绍。3.4 节将介绍模型参数的学习算法，包括比赛模型中影响比赛进攻回合数的潜在因素和球队模型中影响得分情况的球员能力参数。3.5 节将对本章进行总结。

3.1 任务目标和问题描述

在一般预测任务中，目标是对进行比赛的两支球队的各自总得分进行预测，然后得到胜负情况。但是在本文中，我们更希望得到交锋双方各自的进攻回合数，以及在每次进攻回合中的进攻方的得分情况，例如因为失误，被封盖，投篮不中导致的不得分，因对手防守犯规罚球得到 1 分，命中 2 分和命中 3 分，这里不考虑比较少见的得到 3 分以上的情况。这样通过预测得到的进攻回合数和每个进攻回合中的得分情况就可以得到球队的得分情况，从而实现对比分的预测。

在此基础上，我们还希望将每一次进攻回合中的得分情况和球队或者球员的能力建立合理的因果关系，例如在某一回合中 3 分命中，并不能评价进攻球队或者球员的 2 分命中或者罚球命中的能力，也不能评价防守球队的内线防守能力，其应该对应进攻球队的 3 分球命中率和防守球队的外线防守能力。所以我们试图通过每一次进攻回合的得分情况统计数据，评估出球员的各项能力值，从而得到关于球员的评价，以供球队管理层决策。

与此同时，NBA篮球比赛的预测也与其他竞技比赛的预测问题不同，其拥有其他预测问题所没有的特殊性，下面将描述NBA篮球比赛预测问题的特殊性：

（1）数据噪声

在NBA篮球比赛中，当球队之间的分差比较大的时候，通常在第四节末尾双方就回达成默契，换下各自的主力队员，由替补球员上场比赛，消耗掉余下的时间，

这称之为垃圾时间。这种垃圾时间由于都是纯替补球员组成，所以并不是球队真实实力的体现，所以垃圾时间的数据对预测的准确率会造成很大的干扰。通常为了降低数据的噪声，最简单有效的方式就是对于比分差距比较大的比赛仅仅使用前 3 节的数据进行训练，而对于比分接近或者有加时赛的比赛，则可以使用全场的数据，这样，就能很大程度上降低噪声数据对预测准确率的干扰，能够比较准确的捕获球队实力以及技战术对比赛胜利的影响。

（2）攻防转换

篮球比赛不同于其他的比赛，例如足球比赛中，球队前锋的防守通常对球队的整体防守能力造成太多影响，前锋在某一回合进行或者不进行防守通过不会对进攻球队的得分情况造成直接影响，但是由于篮球比赛基本上是五个位置上一一对抗的比赛，所以球员之间能力的对抗，往往决定了进攻回合中的得分情况，例如外线球员的防守松懈有可能就会让进攻球队在外线 3 分命中。所以得分的情况由比赛双方所有球员的攻防能力共同决定，需要联合起来考虑。

（3）球队战术与进攻机会

NBA篮球比赛因为每只球队必须在24秒之内完成进攻,不同于其他的比赛项目，可以控制进攻的情况，所以导致两只球队总体来回合数都偏高，所以准需要确预测出每个回合中的得分情况。但是由于不同球队的风格和教练指定的战术体系不同，导致两只球队的进攻回合数存在差异，例如以防守为主的球队的进攻回合数就会少于以炮轰为主的球队。所以在对比赛的回合数进行预测，需要考虑哪些潜在因素对比赛回合数的影响因子比较高。

（4）主客场因素

NBA篮球比赛中的主客场对胜负关系的影响十分明显，例如裁判因素，观众因素，球员体力因素等都会对球队胜利造成影响，所以需要在预测过程中不能对忽略主客场因素。

（5）球员轮换

NBA的比赛名单包含十二名球员，但是在球队每一次的持球比赛过程中仅仅只有五位球员能够在场上。不同于其他的竞技体育，NBA比赛在死球的时候就可以对

球场上的球员做出调整，而且没有换人次数的限制，所以球队的实力往往由于不同球员的组合情况存在着较大的差异，所以在预测过程中需要着重考虑球员的上场情况，以对每个回合中的得分效率进行准确的估计。

（6）多元的得分情况

为NBA比赛中每次攻防阶段建立模型，将其结果作为一个独立同分布的随机变量。其得分的情况是多元的，在每次进攻过程中可能不得分，得到一分，两分，三方甚至更多分，所以为了得到合理的得分因果关系，也需要对考虑球员各种情况下的攻防能力值，以使得和得分情况对应起来，得到合理的解释。

3.2 传统模型的缺陷分析

大多数传统模型缺乏对比赛回合数进行预测的模型，而是直接依据球队球员的能力值结合其他的影响因素对比赛的比分进行预测，但是这显得过于笼统，没有进行细化，因为往往在比赛的不同的时间段，因为不同上场球员的组合情况，导致了球队在不同时间段之间的实力差异明显，对得分的情况带来明显的影响。所以我们希望能够建立一种比赛模型，能够细化到球队的每一次攻防回合，通过对比赛的攻防回合数进行预测，再结合对攻防效率的估计从而得预测出总得分情况。除此之外，对比赛进行拆解，细化到每一个攻防回合的好处，就是可以根据当前在场球员的组合以及得分情况，对球队的球员的各项能力值进行更加准确的评估。

有少数的基于贝叶斯网络的传统的模型对比赛过程进行了如上所说的分解，细化到了每一次的攻防回合。其通常是假设球员拥有一个进攻能力值和一个防守能力值，并符合一定的概率分布。球队的实力值则是球员能力值之和，使用贝叶斯推论的方法，利用比赛的历史数据进行训练，对球员的能力值参数进行估计。然后根据球队球员能力值的差异，结合影响比赛的其他因素，对比赛的结果进行预测。传统模型的贝叶斯网络如图 3.1 所示。

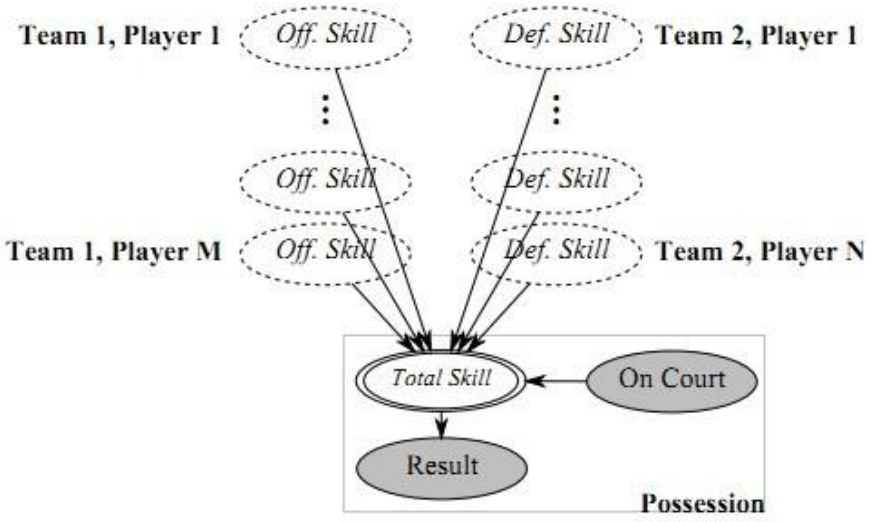

图 3.1　传统模型的贝叶斯网络

由图 3.1 可以看出，球队由多名球员组成，每名球员拥有进攻和防守能力值，其决定了交锋双方球队的能力值，然后根据在场球员的组合情况，决定攻防过程中的得分情况，结果变量（R）可能为下面 4 种输出中的一种：

$R=0$ 进攻球队没有得分，只有其他数据的改变（例如：失误，防守篮板等）

$R=1$ 进攻球队得到 1 分

$R=2$ 进攻球队得到 2 分

$R=3$ 进攻球队得到 3 分

On Court随机变量表示这些球员在场上或者不在场上。

但是这类传统模型存在着如下几点的不足：

（1）上述的传统模型[4]当结果有多项的值的输出时（例如上述的贝叶斯网络中，结果变量 R 存在 4 种的可能输出情况），但是却依赖于球队球员共同的进攻和防守能力值，难以获取到合理的因果关系，无法将比赛过程中的详细赛况信息和球员的能力值进行关联。例如在攻防过程中得到 3 分，由于其依赖于球队球员公共的进攻和对方防守能力值，我们无法得知该球员到底 3 分球能力值如何，很有可能这位球员的能力值很高但是他只是一名优秀的内线中锋，其更多的得分是在篮下得到 2 分，实际上其 3 分球命中率很低。因为模型没有细粒度的对攻防过程之中的得分情况与球员的能力值进行细化并建立相应的映射关系，而只是笼统的当做一个整体攻防实

力值考虑，比赛中的攻防情况都共同依赖于该攻防实力值，所以无法得到攻防阶段的得分情况与球员相应的能力值之间的关系，然而实际上，得分情况应该依赖于球员不同的能力值（1 分球依赖于球员的罚球命中率，2 分球依赖于球员的 2 分球命中率和对手的内线防守能力，3 分球依赖于球员的 3 分球命中率和对手的外线防守能力）。例如 $\Pr\{R=2\}$ 和 $\Pr\{R=3\}$ 都依赖于球员总的攻防能力值，结果的可能输出分布 $\Pr\{R=2\}$ 和 $\Pr\{R=3\}$ 都将在场上的所有单位的公共攻防能力值（例如：上场球员的所有组合分配）上共享。然而实际上，结果条件分布应该依赖于球员的不同能力值（例如：2 分球命中率，3 分球命中率），而且对于不同的场上球员组合情况也应该有不同的分布。

（2）存在欠拟合的问题。实际中，上半场得分$\mathrm{R}=3$，下半场得分$\mathrm{R}=1$的球队与上下半场各自得到$\mathrm{R}=2$的球队在实力上是对等的。然而，传统的模型将不公平的在其他球队的基础上惩罚其中一支球队的似然度，这就带了了欠拟合的问题，目前是想通过加入一个随机变量，表示所得分数的期望来解决这个问题，不过还没有实现。

（3）在目前基于贝叶斯的能力差异的胜负预测模型中的研究中，存在着一个公共的争议，关于能力或者表现的逻辑分布和高斯分布。我们也将试图在改进的模型网络中通过假设不同的概率密度函数对其进行比较。

3.3 改进的比分预测与能力评估模型

本节针对问题的特点探讨解决方法，首先提出改进的贝叶斯网络模型，使之适应篮球比赛的特殊性，要考虑比赛中每次攻防过程中的多值输出结果，攻防过程中球员能力值和得分情况的合理因果关系以及球员之间配合对结果的影响。改进的比分预测和球员能力评估模型包括球队模型和比赛模型。

球队模型对传统模型进行改进，对攻防过程中的状态进行拆解，对应于球员相应的能力值，通过对球员能力值参数进行学习，得到每位球员各项能力值，然后计算出两支球队在某一次攻防过程中得分的情况，也可以求出每支球队在攻防过程中的得分期望。比赛模型是我们提出的潜在变量模型，通过假设每支球队的风格，战术，进攻体系等潜在因素，主客场情况，对比赛中两支球队的进攻回合

数进行预测，然后通过球队模型中推论出来的球队的得分期望，可以得到最终的比分结果。

3.3.1 比赛模型

比赛模型是一种潜在变量模型，潜在变量包括攻防球队的风格，教练的战术体系，不同球队的战术风格和体系通常会决定这只比赛比赛中的进攻回合数，通过对参与比赛的球队加入人为先验，考虑可能会影响比赛进攻回合数的球队因素，当然由于缺乏十分专业篮球知识，我们将动态调整球队因素潜在变量的数目，通过实验来选择对球队进攻回合数起决定性作用的潜在变量组合。我们在这里假设这些因素均服从正态分布。除此之外，主客场因素也会对进攻回合数带来影响。综合考虑这些因素，对参与比赛的两支球队建立如图 3.2 所示的比赛模型，试图对交锋双方的进攻回合数进行预测。

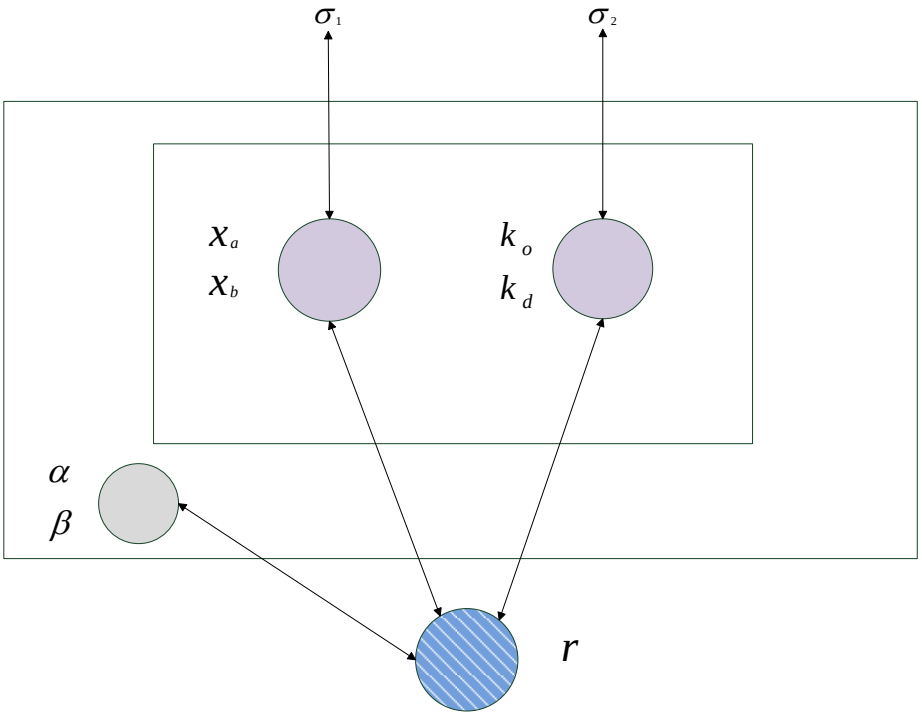

图 3.2 比赛模型

$$r_{a\to b}=[\alpha \quad \beta]\begin{bmatrix}\overrightarrow{x_a} & 0\\ 0 & \overrightarrow{x_b}\end{bmatrix}\begin{bmatrix}\overrightarrow{k_o}^{\mathrm{T}}\\ \overrightarrow{k_d}^{\mathrm{T}}\end{bmatrix} \tag{3.1}$$

$$r_{b\to a}=\begin{bmatrix}\alpha & \beta\end{bmatrix}\begin{bmatrix}\overrightarrow{x_b} & 0\\ 0 & \overrightarrow{x_a}\end{bmatrix}\begin{bmatrix}\overrightarrow{k_o}^{\mathrm{T}}\\ \overrightarrow{k_d}^{\mathrm{T}}\end{bmatrix} \tag{3.2}$$

这里我们采用矩阵分解的方法进行计算，$r_{a\to b}$是球队a和球队b比赛，球队a的进攻回合数，$r_{b\to a}$是球队b的进攻回合数，参数α表示主场系数，参数β表示客场系数。向量$\overrightarrow{x_a}=[x_{a1},...,x_{ai}]$和$\overrightarrow{x_b}=[x_{b1},...,x_{bi}]$分别是球队a和球队b影响比赛回合数的球队因素。向量$\overrightarrow{k_o}=[k_{o1},...,k_{oi}]$是球队影响因素的进攻权重系数，向量$\overrightarrow{k_d}=[k_{d1},...,k_{di}]$是球队影响因素的防守权重系数。通过动态调整球队的影响因素的个数，并利用数据集进行潜在变量参数的学习，并验证其预测准确率，从而得到最佳的影响因素的数目，从而能够更准确的对比赛的进攻回合数进行预测。

3.3.2 球队模型

为了解决在 3.2 节中描述的问题,我们提出了如图 3.3 所示的新贝叶斯网络模型。每一次攻防过程分为进攻方和防守方，进攻球队拥有十二名球员，防守球队也拥有十二名球员。图中，在场随机变量，标识球员在某一时刻是否上场比赛。每位球员在图中使用Pi表示，其对应的能力值使用Si表示，i为 1,2,3，i表示球员命中i分球的能力，在图中，球员的能力值是随机变量。在一次攻防过程中进攻球队可能得到 1 分，2 分，3 分或者不得分，在图中是随机变量，该随机变量依赖于球员相应的得分能力随机变量以及球员是否在场随机变量，而且不同得分情况的随机变量之间不是独立的，而是存在相互联系。进攻过程中的得分结果表示在这次攻防过程中实际得到的分数，在图中是随机变量，其依赖于得分情况随机变量。参数ε表示一些异常因素，也是一个随机变量。

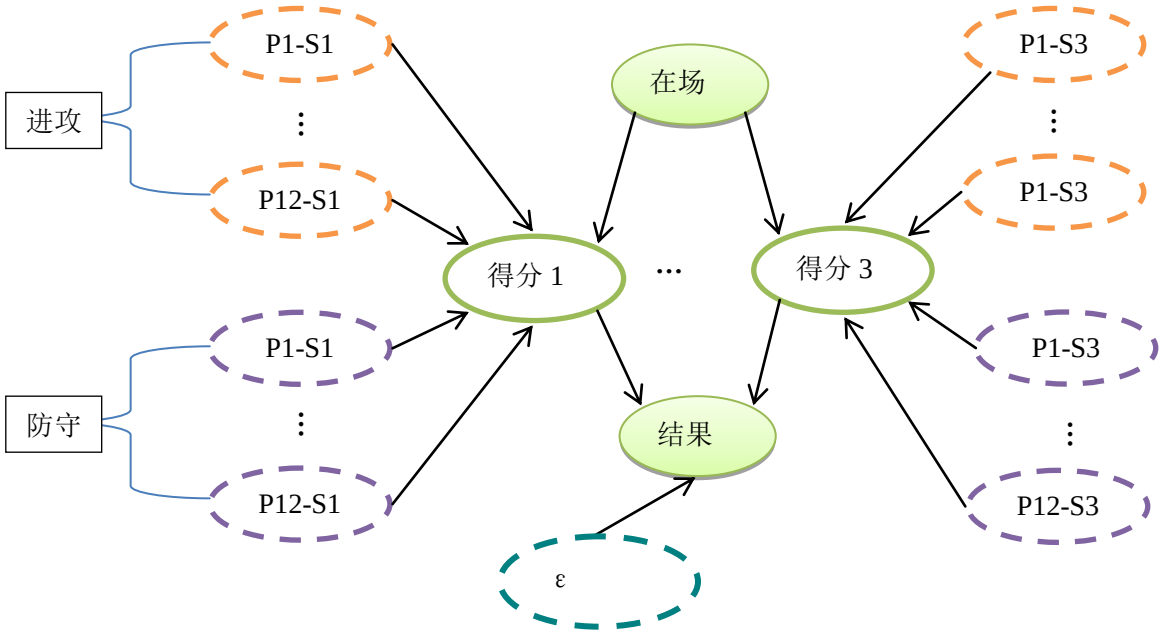

图 3.3　球队模型

Pi-Sj表示球队中第i名球员的第j项能力值，其是模型的参数，在场变量表示改球员是否在场上，得分i表示这次进攻回合中的得分情况的关系，结果表示这次进攻过程中的实际得分， ε 表示对结果的影响的模型参数

在改进的贝叶斯网络中，每位球员的能力细分为 3 个方面的能力参数：

Skill 1：命中（防守）一分球机会的能力

Skill 2：命中（防守）二分球机会的能力

Skill 3：命中（防守）三分球机会的能力

球队A的整体进攻和防守能力的符号表示：

$$OffA(Skill\,k)=\sum_i(Player\,i\,Off.Skill\,k)*\vec{C}_i \quad (3.1)$$

$$DefA(Skill\,k)=\sum_i(Player\,i\,Def.Skill\,k)*\vec{C}_i \quad (3.2)$$

$\vec{C}_i$ 是表示向量 $\vec{C}$ 的第i个元素，$\vec{C}$ 是一个关于指示函数的向量。对于比赛中的每一次攻防，向量 $\vec{C}$ 中仅仅只有 10 个元素的值为 1（赛车上同时只能有 10 位球员在尝试），该向量中的其他元素值为 0.

在比赛的每一次攻防过程中，存在 3 个由带参数的概率函数所定义的 2 值隐随机变量，该概率函数的参数是需要学习的。这 3 个 2 值（Boolean类型）隐随机变量表示在一次攻防过程中得到了几分，其依赖于在场上的球员的对应得分能力值（需要通过学习得到）和上场情况：

$Score_1$：True 表示有一个可保证的概率在攻防过程中的某一时刻得到 1 分；

$Score_2$：True 表示有一个可保证的概率在攻防过程中的某一时刻得到 2 分；

$Score_3$：True 表示有一个可保证的概率在攻防过程中的某一时刻得到 3 分；

每一个 $Score_k$ 事件拥有一个使用向量参数 $\overrightarrow{\theta_k}$（在学习过程中进行学习）表示的条件概率分布，其依赖于球员的能力值。在基本的球员能力模型中，5 人组成的球队的能力值分别是每个球员的能力值的和。这 3 个事件之间并不是独立的，彼此之间存在着相互影响，例如在这次回合中得到 3 分的事件会影响得到 2 分和 1 分的事件，得到 2 分的时间会影响得到 1 分的事件。

分别使用logit和probit模型对球队A与球队B比赛中球队A的 $Score_k$ 事件的概率进行表示，Bradley-Terry对应的logit模型为：

$$\Pr\{Score_k = True \mid Skill\,k\} = \frac{OffA(Skill\,k)}{OffA(Skill\,k) + DefB(Skill\,k)} \tag{3.3}$$

Thurstone Case V对应的probit模型为：

$$\Pr\{Score_k = True \mid Skill\,k\} = \varphi(OffA(Skill\,k) - DefB(Skill\,k)) \tag{3.4}$$

Logit（Bradley-Terry）和probit（Thurstone Case V）在预测排行的研究中是最为常用的 2 值回应模型。

通过对比赛回合中得分的概率进行计算，可以得到在某个进攻回合中不得分，得到 1 分，2 分和 3 分的概率，其 4 个概率值的满足 $P\{s_i\}<1$ 和并且 $\sum_i P\{s_i\}=1$，当需要对该回合得分进行预测的时候，通过产生一个 $[0,1]$ 的随机数，然后与 $p\{s_i\}$ 进行比较得到。

为了便于用符号表示，使用 $s_k = s_k^1$ 表示 $Score_k = True$，$s_k = s_k^0$ 表示

$Score_k = False$。下面举例说明$\{Score_k\}$和结果变量之间的关系，假设有支由 5 人组成的球队，参数θ是已知，结果概率为：

$$\Pr\{S_1 = s_1^1\} = 5\%, \Pr\{S_1 = s_1^0\} = 95\% \quad (3.5)$$

$$\Pr\{S_2 = s_2^1\} = 35\%, \Pr\{S_2 = s_2^0\} = 65\% \quad (3.6)$$

$$\Pr\{S_3 = s_3^1\} = 10\%, \Pr\{S_3 = s_3^0\} = 90\% \quad (3.7)$$

这表示，在每一次的攻防过程中：

进攻球队拥有10% 的概率得到 3 分；

进攻球队拥有$(100\% - 10\%) \times 35\%$的概率得到 2 分；

进攻球队拥有$(90\% \times 65\% \times 5\%)$的概率得到 1 分；

进攻球队拥有55.575%的概率不得分

基本上，为了简化，结果变量R为攻防过程中有效的最高得分选项。例如$Score_3$为True，结果变量R即为 3；假如$Score_3$为False，$Score_2$为True，结果变量即为 2。结果变量有如表 3.1 所示的条件概率分布，参数ε表示为模型中的错误（例如进攻或者防守失误），该参数可以理解为球员临场发挥好坏的因素，受noisy-max的启发，在结果变量的条件概率分布中，参数ε越小，我们的模型更加适用于实际的比赛。

表 3.1　进攻回合得分概率

$R \mid S_1, S_2, S_3$	$R = r^0$	$R = r^1$	$R = r^2$	$R = r^3$
$s_3^1 s_2^1 s_1^1$	ε	ε	ε	$(1-3\varepsilon)$
$s_3^1 s_2^1 s_1^0$	ε	ε	ε	$(1-3\varepsilon)$
$s_3^1 s_2^0 s_1^1$	ε	ε	ε	$(1-3\varepsilon)$
$s_3^1 s_2^0 s_1^0$	ε	ε	ε	$(1-3\varepsilon)$
$s_3^0 s_2^1 s_1^1$	ε	ε	$(1-3\varepsilon)$	ε
$s_3^0 s_2^1 s_1^0$	ε	ε	$(1-3\varepsilon)$	ε
$s_3^0 s_2^0 s_1^1$	ε	$(1-3\varepsilon)$	ε	ε
$s_3^0 s_2^0 s_1^0$	$(1-3\varepsilon)$	ε	ε	ε

3.4 模型参数的学习算法

将每一位球员的能力值视为得分事件的条件概率分布的固定参数表示，ε也是参数，我们使用有缺失的数据集对模型的参数进行估计。来自我们数据集合中每一次攻防过程被视为是独立同分布的观测值，结果变量R和标识是否上场的变量*OnCount*也总是可观察的，得分事件$Score_1$， $Score_2$， $Score_3$是不可观测的隐变量。

该算法过程实际为一个使用 EM 算法迭代求解参数使对数似然函数的期望最大值的过程，如式所示其中D是观测值，D_r是结果变量的观测值，D_c是*OnCount*变量的观测值，它总是可观测的并且假设其具有一个先验概率分布。似然函数 L 由（3.8）所示

$$
\begin{aligned}
&L=\prod\nolimits_D \Pr\{D\mid\vec{\theta_1},\vec{\theta_2},\vec{\theta_3},\varepsilon)= \\
&\prod\nolimits_D \sum\nolimits_{\substack{s_1\in Val(s_1)\\ s_2\in Val(s_2)\\ s_3\in Val(s_3)}} \Pr\{R=D_r,S_1=s_1,S_2=s_2,S_3=s_3,D_c\mid\vec{\theta_1},\vec{\theta_2},\vec{\theta_3},\varepsilon\}= \\
&\prod\nolimits_D \sum\nolimits_{\substack{s_1\in Val(s_1)\\ s_2\in Val(s_2)\\ s_3\in Val(s_3)}} \Pr\{R=D_r\mid S_1=s_1,S_2=s_2,S_3=s_3,D_c,\vec{\theta_1},\vec{\theta_2},\vec{\theta_3},\varepsilon\} \\
&\Pr\{S_1=s_1\mid C=D_c,\vec{\theta_1},\vec{\theta_2},\vec{\theta_3},\varepsilon\} \\
&\Pr\{S_2=s_2\mid C=D_c,\vec{\theta_1},\vec{\theta_2},\vec{\theta_3},\varepsilon\} \\
&\Pr\{S_3=s_3\mid C=D_c,\vec{\theta_1},\vec{\theta_2},\vec{\theta_3},\varepsilon\}
\end{aligned}
\tag{3.8}
$$

在 E-step 中，对得分事件$Score_1$， $Score_2$， $Score_3$的 8 种可能组合做推论并计算在当前参数下，得分结果的 8 种可能的组合情况的概率。

在 M-step 中，利用 E-step 中的概率组合情况可得到似然函数，它是一个关于参数的函数，对参数$\vec{\theta_1}$， $\vec{\theta_2}$， $\vec{\theta_3}$和ε进行极大似然估计，即是在已知观测值 D 的情况下，使得对数似然函数的期望取得最大值，此时的参数值作为下一次迭代的新值。

在 M-step 中，对数似然函数表示为（3.9）：

$$
\begin{aligned}
&l= \\
&(\sum\nolimits_D l\{R=D_r\mid S_1=D_1,S_2=D_2,S_3=D_3,\varepsilon\}) \\
&\sum\nolimits_D l\{S_1=D_1\mid C=D_c,\vec{\theta_1}\}) \\
&\sum\nolimits_D l\{S_2=D_2\mid C=D_c,\vec{\theta_2}\}) \\
&\sum\nolimits_D l\{S_3=D_3\mid C=D_c,\vec{\theta_3}\})
\end{aligned}
\tag{3.9}
$$

3.4.1 M-Step：极大似然估计 ε

对表的条件概率分布用 $\Pr\{R=D_r \mid S_1=D_1, S_2=D_2, S_3=D_3, \varepsilon\}$ 表示，可以得到关于参数 ε 的极大似然估计：

$$\begin{aligned}&\arg\max_{\varepsilon}\{\prod_{D}\Pr\{R=D_r \mid S_1=D_1, S_2=D_2, S_3=D_3, \varepsilon\}\}= \\ &\arg\max_{\varepsilon}\{(1-3\varepsilon)^{M_{\mathrm{mod}elled}}(\varepsilon)^{M_{noise}}\}\end{aligned} \tag{3.10}$$

可得到参数 ε

$$\varepsilon=\frac{1}{3}\frac{M_{noise}}{M_{noise}+M_{\mathrm{mod}elled}} \tag{3.11}$$

其中

$$M_{\mathrm{mod}elled}=M[r^3,s_3^1]+M[r^2,s_3^0,s_2^1]+M[r^1,s_3^0,s_2^0,s_1^1]+M[r^0,s_3^0,s_2^0,s_1^0] \tag{3.12}$$

并且 M_{noise} 是余下观测值的数量，使得 $M_{\mathrm{mod}elled}+M_{noise}=M$，$M$ 为观测值的总数。

3.4.2 M-Step：极大似然估计 θ_i

在实现过程中，我们实现了 2 类成对的比较模型：Bradley-Terry 和 Thurstone Case V，分别对应了 logistic 和 probit 模型。

在 Bradley-Terry 模型中，每一个得分事件随机变量 $Score_i$ 服从一个 logistic 分布：

$$\Pr\{S_i=s_i^0 \mid \vec{c},\vec{\theta_i}\}=\frac{1}{1+\exp(\Delta_i)} \tag{3.13}$$

所以

$$\Pr\{S_i=s_i^1 \mid \vec{C},\vec{\theta_i}\}=\frac{1}{1+\exp(-\Delta_i)} \tag{3.14}$$

其中

$$\Delta_i=[\theta_{i,Off.P1},...,\theta_{i,Off.P12},...,\theta_{i,Def.P1},...,\theta_{i,Def.P12}]\vec{C} \tag{3.15}$$

并且 OnCourt 向量 $\vec{C}$ 是一个指示函数的向量集合。对于任意的攻防过程，向量 $\vec{C}$ 中只有 10 个元素的值为 1，其他的元素值为 0。

为了不失一般性，这里假设防守参数 θ_i 是负数，当进行求和的时候就会降低总的 $\Pr\{Score\}$ 值，而不必减去正的 θ_{Def} 值。

目前，我们需要计算：

$$\arg\max_{\vec{\theta}_i}\{\prod_D \Pr\{S_i = D_i \mid \vec{\theta}_i\}\} \quad (3.16)$$

S_i是关于统计参数$\vec{\theta}_i$的观测值，这退化为普通的逻辑回归模型，并且是由于在E-step的中计算了分配到不同得分情况下的概率不同，即回归是一个带权重的逻辑回归模型。实现中使用了带有权重变量的基本Newton-Raphson逻辑回归技术[6]。

在Thurstone Case V模型中，每一个得分随机变量$Score_i$随机变量服从一个高斯分布：

$$\Pr\{S_i = s_i^0 \mid \vec{\theta}_i\} = \varphi(\Delta_i / \sigma) \quad (3.17)$$

使用probit函数取代logit函数，在实现中使用了争对probit回归的Newton-Raphson带权变种技术，这里取$\sigma^2 = 10$作为球员的表现的方差，输出10位球员的表现数值的和。

3.4.3 期望最大：E-step

在E-step过程中，所有的参数都是固定的，所以我们直接对概率进行计算。对于每一个数据观测点$\langle D_r, D_c \rangle$我们可以获取其8种可能的情况：

$\langle D_r, D_c, s_1^1, s_2^1, s_3^1 \rangle$带权正比于$\Pr\{D_r, D_c, s_1^1, s_2^1, s_3^1 \mid \vec{\theta}, \varepsilon\}$

$\langle D_r, D_c, s_1^1, s_2^1, s_3^0 \rangle$带权正比于$\Pr\{D_r, D_c, s_1^1, s_2^1, s_3^0 \mid \vec{\theta}, \varepsilon\}$

……

$\langle D_r, D_c, s_1^0, s_2^0, s_3^0 \rangle$带权正比于$\Pr\{D_r, D_c, s_1^0, s_2^0, s_3^0 \mid \vec{\theta}, \varepsilon\}$

使用贝叶斯网络图模型的链式规则，可以方便的计算$\Pr\{D_r, D_c, s_1^i, s_2^i, s_3^i \mid \vec{\theta}, \varepsilon\}$的准确概率值。

3.4.4 期望最大：初始化

在实现时使用一些初始化策略，一种策略就是将所有的能力参数值初始化为相同的值，这里为：

$$\theta = \vec{0} \tag{3.18}$$

对所有的分数 i 假设 $\Pr\{s_i\} = 0.5$，因此有

$$\varepsilon = \frac{1}{3}\frac{M\left[r^3\right]/2 + M\left[r^2\right]/4 + M\left[r^1\right]/8 + M\left[r^0\right]/8}{M} \tag{3.19}$$

另外一种策略是近似的表示在一次攻防过程中得分的概率，对于得到 3 分的概率，我们初始化为：

$$s_3^{init} = \frac{M\left[r^3\right]}{M} \tag{3.20}$$

类似的，由 S_3 和 S_3 可以得到在一次攻防过程中得到 2 分的概率：

$$\frac{M\left[r^2\right]}{M} \to \Pr\left\{r^2\right\} \approx \left(1 - \Pr\left\{S_3 = s_3^1\right\}\right) \times \Pr\left\{S_2 = s_2^1\right\} \tag{3.21}$$

所以初始化 $\Pr\left\{S_3 = s_3^1\right\}$ 为：

$$s_2^{init} = \left(M\left[r^2\right]/M\right)/\left(1 - s_3^{init}\right) \tag{3.22}$$

同理可得：

$$s_1^{init} = \left(M\left[r^1\right]/M\right)/\left(1 - s_2^{init}\right) \tag{3.23}$$

初始化完成之后，EM 算法从 M-step 开始进行计算。

3.5 本章小结

在这一章里，我们首先分析了传统的贝叶斯预测模型，然后对其存在的问题进行了介绍，提出了对模型进行改进的方法和目标，争对篮球竞技的特点，将粗粒度的球员能力值进行细化，将其分解成为更加细粒度的多个能力值，进攻和防守能力各三个，将攻防过程中的得分情况和球员的不同能力值进行合理的因果关联，得到了新的贝叶斯网络结构。然后利用有效的数据集，将其划分为训练数据和测试数据，。对该模型进行训练和验证。

在对模型的参数进行训练过程中，使用了 EM 算法，在训练过程中，比赛每一个攻防过程中的得分情况是可观测的变量，球员的能力参数是隐变量，EM 算法的迭

代过程就是不断的求解参数化的期望值，然后使其在得分观测值的情况下最大化，迭代结束时，近似的得到了模型参数，即得到了每个球员的能力值。然后使用测试数据，对该模型的准确率进行评估。

4 实验设计及结果分析

在这一章里，我们将对我们的实验情况进行介绍，首先对我们的数据集和其相应的格式进行计算，接着我们选择了 06-09NBA 常规赛季的数据集，使用部分数据集做训练，部分数据集做测试，得到每次进攻回合中的得分情况的准确率统计并和已有的 Trueskill 算法进行比较，发现我们的模型对进攻回合中得分情况的预测准确率高于 Trueskill 算法。接着我们选取了 08 赛季的数据集，部分做训练部分做测试，对比赛模型中的进攻回合数做拟合实验，并统计拟合的准确率。然后我们选取了 08 赛季西南赛区中 5 支球队的战报进行分析，对每个回合的得分概率和当前在场的球员组合情况进行分析，并争对火箭和灰熊的一场战报进行分析，得到球员的能力值与得分效率之间的合理解释。在此基础上，根据各队球员的上场时间作为权重，对球队总的得分期望（得分效率）进行计算，并根据比赛模型中估计出来的进攻回合数，对总的得分情况进行估计。最后，通过对模型参数的学习，得到了赛季各个球员的各项能力值，在实验中，我们对球员的各项能力值进行排名，并将结果列举出来，并进行相关的分析。

4.1 数据集

我们收集了 NBA2006-2007，2007-2008，2008-2009 常规赛季各场比赛的详细赛况，其包括的数据信息如表 4.1 所示：

表 4.1 数据集格式

字段	描述
a1	客队场上球员 1
a2	客队场上球员 2
a3	客队场上球员 3
a4	客队场上球员 4
a5	客队场上球员 5
h1	主队场上球员 1
h2	主队场上球员 2
h3	主队场上球员 3
h4	主队场上球员 4

续表 4.1　数据集格式

h5	主队场上球员 5
period	比赛节数
time	小节剩余时间
team	持球球队名
etype	发生的事件，包括 jump ball（跳球），shot（投篮），foul（犯规），free throw（罚球），rebound（篮板），turnover（失误），timeout（暂停），foul（犯规），violation（违例），sub（换人）
assist	助攻球员
steal	抢断球员
block	盖帽球员
enter	换人时换上场的球员
left	换人时下场的球员
num	当前罚球次数
opponent	犯规时被侵犯的球员
out of	罚球次数
player	发生投篮，返回，罚球，篮板，失误，犯规，违例事件的当前球员
points	得到的分数
reason	发生法规，失误的原因例如传球失误，过人失误，技术性犯规，战术性犯规
result	投篮或者罚球的结果

4.2　实验设计

我们使用数据集中 25%的攻防过程作为测试数据和 75%的攻防过程作为训练数据。在训练数据中，对超过 100000 次攻防过程的得分情况进行计算：

$$E[R] \approx \frac{1}{m} \sum_{R} rM[r] = 1.0486 \qquad (4.1)$$

得到每次攻防过程的期望得分为 1.0486，在测试过程中，对超过 788 次攻防过程的得分情况进行计算：

$$E_{traing}\left[R|C\right]=1.0183 \qquad (4.2)$$

得到每次攻防过程的期望得分为 1.0183，测试数据的实际得分期望为 1.0178，可见我们模型的结果与实际的结果已经十分接近。使用相同的数据集和实验方法，将 Trueskill 模型和我们的模型进行比较，取了 06-07,07-08 两个赛季的数据进行训练，08-09 赛季的数据进行测试，对数据集每一次攻防过程中的得分情况进行预测，结果如图 4.1 所示。

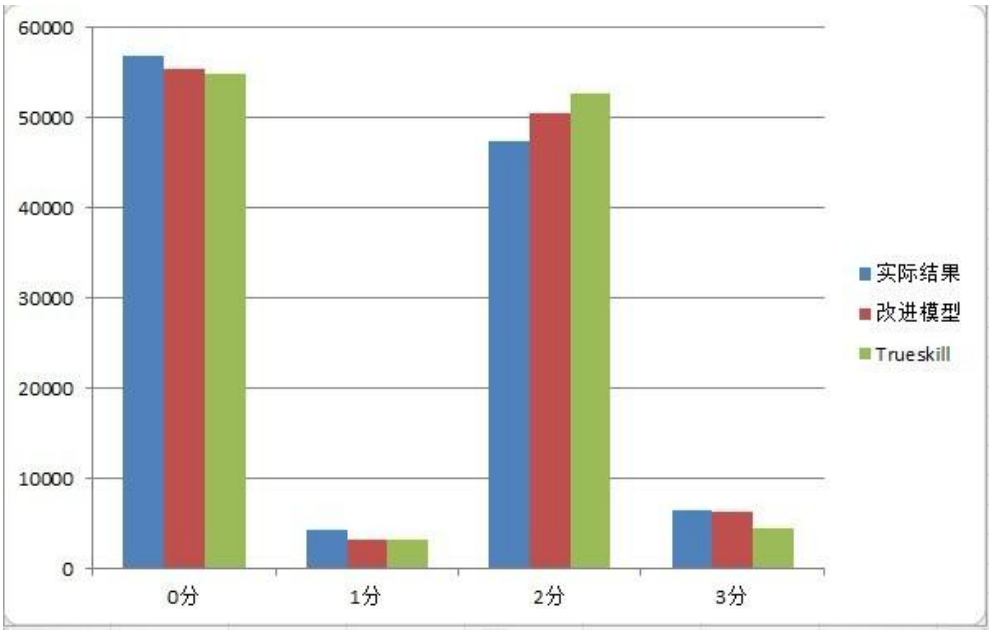

图 4.1　结果对比

我们分别使用了 Bradley-Terry (logit)和 Thurstone Case V (probit)模型对球员的能力值进行评估，得到了排名靠前的球员和数据列表，如表 4.2 和表 4.3 所示。进攻能力值（正数）越高越好，防守能力值（负数）越低越好。

表 4.2　Bradley-Terry 模型球员能力

进攻			防守			球员
1pt	2pt	3pt	1pt	2pt	3pt	(num. possessions)
-547.377	0.478	-1.836	179.184	-0.528	12.050	Jason Terry(728)
1520.724	0.890	-0.624	915.671	-0.106	-12.436	Dikembe Mutombo(47)
-469.428	0.306	1.141	-555.166	-0.659	-13.181	Ryan Bowen(62)
997.244	0.620	-5.364	636.398	-0.270	-3.430	Yao Ming(690)
-499.753	0.352	-22.760	222.773	-0.321	-5.103	Marko Jaric(153)
266.299	0.445	1.948	309.153	-0.180	-2.017	Kurt Thomas(382)

续表 4.2　Bradley-Terry 模型

-396.846	0.376	0.814	-354.445	-0.212	11.228	Jose Juan Barea(427)
1465.266	0.391	-3.987	970.594	-0.179	-5.035	Luis Scola(767)
-120.838	0.362	-1.943	-199.037	-0.185	-8.298	Antoine Wright(484)
537.833	0.290	-4.034	101.455	-0.192	5.926	Sean Marks(309)
-562.385	-0.004	-2.357	-80.260	-0.481	12.638	Gerald Green(77)
220.429	0.188	-1.023	90.823	-0.288	-0.948	Matt Bonner(543)
-77.228	0.205	-2.996	49.357	-0.261	10.115	Jason Kidd(863)
-284.500	0.212	6.018	-42.883	-0.214	0.222	Kyle Lowry(872)
590.692	0.240	1.164	308.431	-0.185	-3.068	Tim Duncan(857)
515.277	0.065	-3.019	-582.764	-0.351	-0.222	David West(887)
619.201	0.130	3.299	20.162	-0.238	1.379	Tyson Chandler(417)
955.361	0.313	-6.397	868.131	-0.020	-3.392	Chuck Hayes(349)
1260.731	0.357	-5.341	660.640	0.038	-4.346	Carl Landry(421)
439.383	-0.036	-2.051	-501.853	-0.349	7.807	Jerry Stackhouse(61)
109.231	0.019	-1.694	3.963	-0.263	-0.471	Devin Brown(355)
1003.931	0.088	0.742	89.625	-0.183	-6.313	Fabricio Oberto(179)
-270.817	0.022	4.640	605.664	-0.236	-0.335	Mike Conley(860)
-857.377	0.125	-4.902	119.645	-0.130	-7.144	Chris Paul(830)
-191.039	0.073	1.461	27.733	-0.154	4.664	Melvin Ely(214)
-10.198	0.110	0.523	-510.975	-0.102	-1.283	Marc Gasol(949)
254.362	0.317	-0.386	515.945	0.114	-1.746	Drew Gooden(92)
-276.790	0.183	-0.419	4.769	0.003	2.356	Anthony Tolliver(120)
-613.060	0.022	-4.736	-144.385	-0.140	-22.593	Devean George(141)
343.744	-0.003	-9.103	-3.963	-0.145	1.392	Hilton Armstrong(258)
-875.890	-0.190	0.400	-1198.651	-0.314	-0.742	Von Wafer(414)
-148.658	-0.119	-2.194	178.507	-0.237	-6.116	Josh Howard(665)
65.185	0.114	-6.401	-115.816	0.080	-2.784	Hakim Warrick(760)
303.439	-0.050	-18.882	-314.960	-0.046	0.880	Julian Wright(243)
-414.110	-0.194	-1.583	-261.687	-0.183	-1.854	Tony Parker(756)
548.341	0.009	1.490	27.244	0.054	0.776	Peja Stojakovic(540)
471.534	-0.125	4.482	-225.106	-0.078	-1.608	O.J. Mayo(1269)
-590.135	-0.033	-0.257	41.858	0.054	0.712	Bruce Bowen(479)
183.037	-0.025	-4.620	313.388	0.067	-2.375	Greg Buckner(301)
88.732	-0.080	-14.946	-507.651	0.022	-12.767	Hamed Haddadi(9)
43.961	0.692	-14.933	660.123	0.800	-12.768	Matt Carroll(14)
-239.838	-0.254	-4.023	-1275.994	-0.132	-0.886	Darko Milicic(503)
243.403	-0.153	-1.855	-281.965	-0.023	-0.334	Ime Udoka(290)
-879.942	0.028	0.719	-726.612	0.168	-1.536	Tracy McGrady(310)
-660.346	-0.144	-4.252	-325.539	0.029	1.412	James Posey(553)
-45.441	0.098	2.568	-365.673	0.270	-6.875	Erick Dampier(673)
160.952	-0.039	-8.166	477.428	0.134	3.655	Rudy Gay(1108)
-57.805	-0.326	1.361	-400.539	-0.079	-8.675	James Singleton(262)
-16.634	-0.209	1.281	-246.591	0.051	3.079	Manu Ginobili(392)

续表 4.2　Bradley-Terry 模型

-81.431	-0.016	-1.267	-283.603	0.250	4.416	Michael Finley(815)
-267.966	-0.147	-4.499	-555.824	0.126	-0.486	Darrell Arthur(559)
-578.976	-0.257	-1.470	-282.521	0.054	3.520	Roger Mason(782)
95.233	-0.684	3.461	-466.944	-0.355	2.268	Darius Miles(40)
-875.872	-0.411	0.452	-397.212	-0.081	4.943	Shane Battier(520)
-337.505	-0.272	-1.579	-320.760	0.147	-0.885	George Hill(336)
705.553	-0.427	-2.936	-150.680	0.001	-49.872	Shawne Williams(94)
-154.807	-0.190	2.492	56.795	0.265	-8.015	Antonio Daniels(337)
89.145	-0.266	-9.617	42.456	0.199	-0.812	Quinton Ross(557)
-1300.804	-0.480	0.367	-1076.560	0.033	3.251	Ron Artest(751)
-937.496	-0.235	6.892	-902.506	0.357	6.433	Rafer Alston(269)
-352.168	-0.195	0.944	24.381	0.452	1.305	Rasual Butler(784)
496.072	-0.292	1.692	-610.860	0.374	-6.388	Brandon Bass(470)
-77.456	-0.412	5.924	-739.228	0.330	0.328	Aaron Brooks(546)
-2090.758	-0.792	5.030	-578.207	-0.001	5.621	Luther Head(123)
-136.440	-0.436	-19.532	-201.301	0.392	5.023	Malik Hairston(28)
496.377	-0.482	0.353	107.357	0.398	-6.913	Dirk Nowitzki(951)
-91.307	-0.241	4.775	-930.931	0.655	-6.321	DeSagana Diop(171)
-995.555	-0.787	-1.242	-1172.180	0.148	1.649	Brent Barry(258)
-333.059	-0.813	-0.440	-594.142	0.125	-0.612	Jacque Vaughn(49)
-841.425	-0.575	7.946	85.983	0.503	6.197	Morris Peterson(126)
495.967	0.062	-13.836	-312.682	1.147	-6.992	Ryan Hollins(19)

表 4.3　Thurstone Case V 模型

进攻			防守			球员
1pt	2pt	3pt	1pt	2pt	3pt	(num. possessions)
-2028.591	1.003	2.255	-2449.287	-2.255	-19.422	Ryan Bowen(62)
-1648.566	1.541	-0.956	622.888	-1.504	4.871	Jason Terry(728)
4141.276	2.688	-2.873	1584.462	-0.267	-13.319	Dikembe Mutombo(47)
1503.596	1.783	-3.676	1774.545	-0.901	-1.516	Yao Ming(690)
-421.198	1.549	-7.519	849.735	-1.021	-2.440	Marko Jaric(153)
543.398	1.467	0.990	1169.503	-0.541	-1.852	Kurt Thomas(382)
-1130.560	1.198	-0.632	-767.618	-0.704	3.297	Jose Juan Barea(427)
-369.141	1.192	-1.428	-897.197	-0.607	-3.675	Antoine Wright(484)
2661.467	1.028	-3.251	1761.895	-0.579	-2.476	Luis Scola(767)
222.533	0.602	-1.552	558.974	-0.985	-2.332	Matt Bonner(543)
-95.499	0.917	3.053	-982.321	-0.650	-0.316	Kyle Lowry(872)

续表 4.3　Thurstone Case V 模型

939.348	0.942	-0.067	917.036	-0.620	1.843	Sean Marks(309)
936.816	0.270	0.496	-2178.286	-1.185	-0.750	David West(887)
360.515	0.610	-3.312	732.698	-0.773	2.075	Jason Kidd(863)
-1670.263	-0.069	-2.785	531.296	-1.415	5.832	Gerald Green(77)
976.168	0.726	-0.509	1167.356	-0.609	-3.493	Tim Duncan(857)
1610.184	0.413	0.845	467.511	-0.814	1.173	Tyson Chandler(417)
-355.721	-0.150	-0.726	-613.890	-1.129	0.843	Jerry Stackhouse(61)
69.391	0.184	2.832	930.370	-0.768	-0.349	Mike Conley(860)
2589.485	0.256	-1.015	835.838	-0.640	-4.399	Fabricio Oberto(179)
745.136	0.041	0.135	93.667	-0.818	-0.727	Devin Brown(355)
372.151	0.787	-4.005	2583.904	-0.067	-1.422	Chuck Hayes(349)
2014.097	0.957	-3.821	1814.035	0.117	-3.862	Carl Landry(421)
-266.813	0.256	0.041	38.491	-0.580	0.974	Melvin Ely(214)
-746.818	0.446	-0.311	-1213.852	-0.310	0.530	Marc Gasol(949)
-806.850	0.593	-1.402	-44.990	-0.076	-0.760	Anthony Tolliver(120)
-2008.368	0.254	-1.999	610.435	-0.383	-3.835	Chris Paul(830)
471.258	0.981	-0.606	664.254	0.429	-4.816	Drew Gooden(92)
-1606.875	0.066	-3.494	-838.560	-0.462	-3.375	Devean George(141)
-1320.307	-0.533	1.289	-2430.496	-1.019	0.041	Von Wafer(414)
939.926	-0.108	-2.990	-46.937	-0.529	0.392	Hilton Armstrong(258)
-605.086	-0.431	-1.715	620.911	-0.627	-1.452	Josh Howard(665)
-314.275	0.400	-1.862	64.026	0.271	-0.849	Hakim Warrick(760)
954.229	-0.166	-26.323	-1941.121	-0.136	0.006	Julian Wright(243)
941.323	0.091	1.323	326.974	0.183	0.203	Peja Stojakovic(540)
-791.879	-0.680	-0.620	-883.424	-0.561	-1.167	Tony Parker(756)
1518.822	-0.481	0.839	83.646	-0.284	-2.661	O.J. Mayo(1269)
-1397.593	-0.154	-0.301	-232.059	0.177	0.286	Bruce Bowen(479)
-1414.737	0.189	0.336	-1370.053	0.574	-3.712	Tracy McGrady(310)
259.151	-0.441	-0.179	-846.664	-0.049	-0.620	Ime Udoka(290)
49.672	2.373	-18.995	1395.407	2.784	-14.503	Matt Carroll(14)
811.989	-0.207	-3.523	1211.259	0.205	-1.479	Greg Buckner(301)
-325.385	0.393	2.972	-1553.296	0.843	-2.280	Erick Dampier(673)
-1239.700	-0.857	-2.228	-2939.979	-0.400	1.208	Darko Milicic(503)
66.400	-0.361	-18.668	-943.839	0.126	-16.138	Hamed Haddadi(9)
-2064.105	-0.517	-2.976	-1956.433	0.095	0.947	James Posey(553)
-343.265	-1.120	0.836	-844.660	-0.391	-4.079	James Singleton(262)
305.161	-0.287	-4.351	1192.910	0.446	-0.058	Rudy Gay(1108)
-83.438	-0.672	1.658	-219.630	0.190	1.198	Manu Ginobili(392)
-175.086	-0.041	0.053	-851.058	0.825	3.630	Michael Finley(815)
-1577.152	-0.527	-2.038	-1234.826	0.402	-0.098	Darrell Arthur(559)
217.185	-2.204	4.718	-490.852	-1.196	3.372	Darius Miles(40)
-1379.432	-0.815	-0.947	-847.511	0.197	1.666	Roger Mason(782)
-572.698	-1.294	1.360	-399.857	-0.275	1.791	Shane Battier(520)

续表 4.3　Thurstone Case V 模型

-372.369	-0.603	1.022	648.783	0.852	-4.029	Antonio Daniels(337)
-681.781	-0.897	-0.329	-1201.146	0.566	-0.632	George Hill(336)
-3492.217	-1.398	0.006	-2111.204	0.127	-0.014	Ron Artest(751)
2450.054	-1.488	-1.408	-70.602	0.065	-32.755	Shawne Williams(94)
-130.744	-1.024	-6.652	474.656	0.662	-1.333	Quinton Ross(557)
-1938.612	-0.623	3.574	-2604.635	1.200	4.700	Rafer Alston(269)
897.671	-0.900	1.353	-1541.262	1.024	-3.162	Brandon Bass(470)
-584.409	-0.632	-3.462	158.658	1.526	-0.161	Rasual Butler(784)
140.149	-1.175	3.422	-1840.660	1.110	-1.547	Aaron Brooks(546)
-5291.794	-2.397	1.913	-2183.345	0.074	3.173	Luther Head(123)
-349.487	-0.695	2.224	-2845.544	1.967	-3.430	DeSagana Diop(171)
898.828	-1.589	0.966	184.994	1.222	-2.916	Dirk Nowitzki(951)
38.491	-1.508	-22.444	-599.285	1.359	3.760	Malik Hairston(28)
-2311.382	-2.425	-0.325	-2585.166	0.448	0.847	Brent Barry(258)
897.348	0.268	-17.682	-1523.477	3.397	-3.404	Ryan Hollins(19)
-487.516	-2.748	-0.469	-799.452	0.442	0.184	Jacque Vaughn(49)
-2205.014	-1.889	2.755	1258.575	1.692	1.761	Morris Peterson(126)

4.3　结果分析

使用比赛模型和球队模型共同计算出来的结果,对 2008 赛季比赛进行比分预测,并和直接结果进行比较,如图 4.2 所示.我们使用了非线性最小二乘法对潜在变量模型的参数进行拟合，使用 08 赛季的部分数据进行训练，部分进行测试的时候，其关于进攻回合数的预测准确率为 65.39%，预测与实际回合数的误差均值为 11.7953，均方差为 9.5379。实际回合数的均值为 215.4898，均方差为 18.1330。预测回合数的均值为 215.4904，均方差为 9.9325。

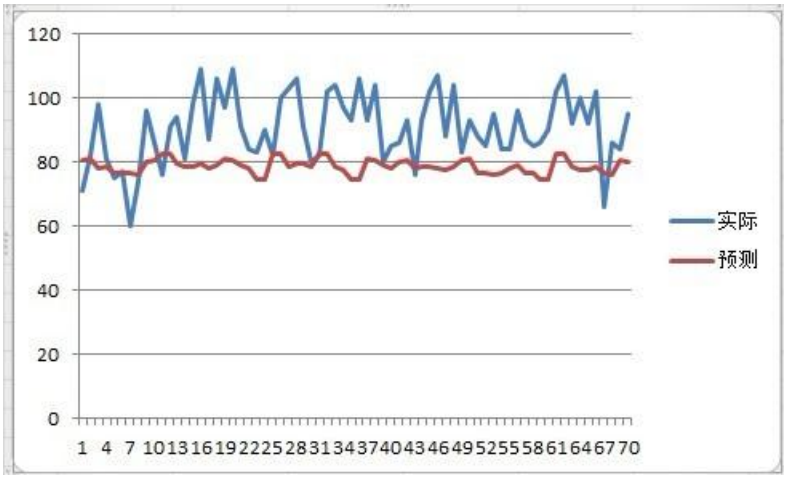

图 4.2　预测与实际结果比较

选取火箭和灰熊在 08 年的一场比赛进行分析，对每一次进攻回合当前在场球员的组合情况，对该次进攻回合中的得分概率进行计算，发现 43 回合的随着核心球员麦迪的下场休息得分概率从 42 回合的

$$P(0)=0.4178, P(1)=0.0154, P(2)=0.5087, P(3)=0.0581$$

变为了

$$P(0)=0.7798, P(1)=0.0154, P(2)=0.1742, P(3)=0.0307$$

在 54 回合随着另外一个核心球员姚明的下场，得分概率变为

$$P(0)=0.8353, P(1)=0.0154, P(2)=0.1178, P(3)=0.0316$$

随着 69 回合麦迪归来，得分概率又有所增强，得到

$$P(0)=0.6754, P(1)=0.0154, P(2)=0.2781, P(3)=0.0312$$

其回合数和相应的在场球员如表 4.4 所示。

表 4.4　回合在场球员

回合数	球员 1	球员 2	球员 3	球员 4	球员 5
42	巴里	兰德里	阿尔斯通	麦迪	姚明
43	巴里	兰德里	海德	阿泰斯特	姚明
…					
54	巴里	兰德里	海耶斯	海德	阿泰斯特
…					
68	巴里	兰德里	海耶斯	海德	阿泰
69	巴里	兰德里	海耶斯	海德	麦迪

可见球队的得分效率受上场球员的组合情况所影响，能力值比较高的球员组合情况下的球队得分效率明显高于能力值差的球员组合，这也合理解释了球队的先发阵容强于替补阵容。

最后对 08 赛季西南赛区的 5 支球队之间的交锋战报进行了统计分析，通过球队球员的上场时间作为权重，对球队的总的得分概率和期望进行计算，并结合实际的战报对结果进行分析，这里选取了火箭和马刺的 3 场战报进行分析。战报如表 4.5 所示。

表 4.5 火箭 vs 马刺战报

比赛	球队 1	球队 2	回合	得分	实际效率
1	火箭	马刺	184	75	0.4076
1	马刺	火箭	182	77	0.4230
2	火箭	马刺	201	85	0.4228
2	马刺	火箭	202	88	0.4356
3	火箭	马刺	204	87	0.4264
3	马刺	火箭	218	85	0.3899

在 Logit 假设模型下，计算得到的球队得分效率：

火箭总的得分概率（进攻效率）为：

$$P(0)=0.3484, P(1)=0.0227, P(2)=0.3215, P(3)=0.3073$$

马刺总的得分概率（进攻效率）为：

$$P(0)=0.3316, P(1)=0.0227, P(2)=0.3176, P(3)=0.3281$$

加入权重得到其得分期望：

火箭为 0.419769，马刺为 0.425775，可见与真实的效率十分接近。

4.4 本章小结

在这一章里，我们介绍了实验的详细设计，并对实验结果进行了分析，将我们的模型和已有算法的对 06-09 常规赛的比赛的进攻回合中的得分情况进行预测，并对其准确率进行了比较，发现我们模型对 NBA 比赛进攻回合中的得分情况预测的准确率高于 TrueSkill 算法。接着我们利用比赛模型对比赛进攻回合数进行估计，统计预测准确率，并计算预测数据的均方差并和真实的得分情况进行比较。然后，我们对实验数据进行分析，得到了进攻回合中的得分效率和上场球员的组合情况的合理解释，然后对球队总的得分概率进行计算，并得到球队的得分情况，利用比赛模型中估计得到的回合数，得到球队之间比赛的预测得分情况，并与真实得分情况进行比较。最后对球员能力值进行分析评估并进行排名。

5 总结与展望

5.1 全文总结

在这篇文章里，我们全面介绍了竞技体育预测模型的研究现状，分析了目前已有的预测模型，包括模型的算法，解决了什么问题，结果怎样。学术界也在不断的使用新的数学方法对预测模型进行改进，以提高其准确率。我们选择了其中几个代表性的数学方法进行了详细的介绍，以便于理解传统模型的不足之处以及改进模型的优点。

在充分了解了当前预测模型的发展现状及其应用到的数学方法，我们对我们需要解决的目标问题以及需要达到的任务目标进行描述，详细列举了 NBA 篮球比赛相对于其他预测问题的特殊之处，分析了该预测问题的难点之所在。接着，争对这些特殊问题，介绍了传统模型的解决办法以及其不足之处，并提出了改进的方向。最后介绍我们提出的改进模型，对改进的方法详细的形式化描述，并通过实验进行验证，分析实验得到的数据结果，与其他模型进行比较。我们是按照提出问题，现有方法综述，提出方法，实验论证这样的路线对论文的内容进行组织，符合合理的科学研究方法。

我们提出方法由比赛模型和球队模型构成，比赛模型是新提出的一种潜在变量模型，目的是对一些复杂的模型进行简化，将影响比赛的可能因素当做潜变量利用历史战报数据集进行训练，然后利用学习得到的潜在变量对比赛的进攻回合数进行预测，这样避免了对这些相关因素建立过于复杂的模型。球队模型则是对传统的基于贝叶斯网络的模型进行改进，将比赛进攻回合数中的得分状态进行拆解，与球员相应的得分能力相对应，建立合理的因果果关系。得分状态之间也不是独立的，而是相互影响，共同决定了进攻回合中的得分结果。

在明确了问题，并提出了改进的方案之后，我们通过对 06-09 年之间常规赛季的 NBA 战报数据进行分析和整理，对我们的方案进行了实现，并在该数据集上面进行验证。经验证，我们的方法对攻防回合中的得分情况预测准确率高于传统模型，这

也说明对球员能力值的评估更加准确，在攻防回合数和总得分预测上的准确率在65%-70%之间，和传统的方法的结果相当。

5.2 展望

虽然我们提出的模型取得了比较好的实验结果，但是后续我们还有很多值得改进的地方。近期的目标有两个，一是对球员的能力值进行合理的介绍，对得到球员的能力值参数简历合理的评估模型，以评估球员对球队的价值。二是目前如果使用不同赛季的比赛进行训练和预测，由于赛季不同，在进攻或者防守强度上的不同，存在着预测值和实际值的偏差，这个也是未来的改进目标。

致 谢

首先感谢我的导师沈刚教授，是他最早将我领入计算机科学的殿堂。我能够完成学业，掌握很多知识，都是沈老师对我教导的功劳。如果日后我能侥幸取得一点成绩的话，也要感谢沈老师在我学生生涯督促我打下的坚实基础，引导我领悟的治学方法，和以身作则教会我的做人道理。

我还要感谢在学期间结交的同道中人，我们因各种机缘巧合从五湖四海走到一起，感谢他们每个人都有那么多优秀的闪光点，让我时刻不敢骄傲，让我总能找到前进的方向。

最后感谢我的家人，有他们的坚定支持是我能够努力做好这一切的最前提条件。

参考文献

[1] Goldberg d., nichols d., oki b.M.Et al.Using collaborative filtering to weave an information tapestry. Communications of acm, 1992, 35(12): 61-70

[2] Su xiaoyuan, khoshgoftaar taghi m.A survey of collaborative filtering techniques.Advances in artificial intelligence, 2009: 19-21

[3] Adomavicius g., tuzhilin a.Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions.Ieee transactions on knowledge and data engineering, 2005, 17: 734-749

[4] Melvillev prem, sindhwani vikas. Recommender systems. Encyclopedia of machine learning, 2010, 705: 829-838

[5] Goldberg k.Eigentaste: a constant time collaborative filtering algorithm.Information retrieval, 2001(4): 133-151

[6] Miller B.N., konstan j.a., riedl j.Pocketlens: toward a personal recommender system.Acm transactions on information systems, 2004, 3: 437-476

[7] Resnick paul, iacovou neophytos, mitesh suchak, et al. Grouplens: an open architecture for collaborative filtering of netnews.In: proceedings of the acm conference on computer supported cooperative work.New york, 1994: 175-186

[8] Linden G., Smith B., york, j.Amazon.com recommendations: item-to-item collaborative filtering.Ieee internet computing, 2003, 7(1): 76-80

[9] Ansari A., essegaier S., kohli r.Internet recommendation systems.Journal of marketing research, 2000, 37(3): 363-375

[10] Sarwar B. M.Item-based collaborative filtering recommendation algorithms.In: proceedings of the 10th international conference on world wide web, 2001: 285-295

[11] Mclaughlin m.R., herlocker j.L.A collaborative filtering algorithm and evaluation metric that accurately model the user experience.In: proceedings of 27th annual international acm sigir conference on research and development in information retrieval, 2004: 329-336

[12] Herlocker J. L., konstan joseph a., terveen loren g.Et al.Evaluating collaborative filtering recommender systems.Acm transactions on information systems, 2004,

22(1): 5-53

[13] Salton g., mcgill m.Introduction to modern information retrieval.Mcgraw-hill, 1983

[14] Karypis g.Evaluation of item-based top-n recommendation algorithms.In: proceedings of the international conference on information and knowledge management, 2001: 247-254

[15] Deshpande m.And karypis g.Item-based top-n recommendation algorithms.Acm transactions on information systems, 2004, 22(1): 143-177

[16] Herlocker j.L.An algorithmic framework for performing collaborative filtering.In: proceedings of the conference on research and development in information retrieval, 1999: 230-237

[17] Breese J., heckerman D., kadie C. Empirical analysis of predictive algorithms for collaborative filtering.In: proceedings of the 14th conference on uncertainty in artificial intelligence, 1998: 43-52

[18] Miyahara K., pazzani M. J. Collaborative filtering with the simple bayesian classifier.In: proceedings of the 6th pacific rim international conference on artificial intelligence, 2000: 679-689

[19] Su x., khoshgoftaar t.M.Collaborative filtering for multi-class data using belief nets algorithms.In: proceedings of the international conference on tools with artificial intelligence, 2006: 487-504

[20] Ungar L. H., foster d.P.Clustering methods for collaborative filtering.In: proceedings of the workshop on recommendation systems.Aaai press, 1998

[21] Chee s.H.S., han J., wang k.Rectree: an efficient collaborative filtering method.In: proceedings of the 3rd international conference on data warehousing and knowledge discovery, 2011: 145-151

[22] Hofmann T. Latent semantic models for collaborative filtering.Acm transactions on information systems, 2004, 22(1): 89-115

[23] Shani G., heckerman D., brafman R. I.An mdp-based recommender system.Journal of machine learning research, 2005, 6: 1265-1295

[24] Basu C., hirsh H., cohen W. Recommendation as classification: using social and content-based information in recommendation.In: proceedings of the 15th national conference on artificial intelligence, 1998: 714-720

[25] Billsus D., pazzani M.Learning collaborative information filters.In: proceedings of the 15th international conference on machine learning, 1998

[26] Hu Y., koren Y. volinsky C. Collaborative filtering for implicit feedback datasets.In: proceedings of 8th ieee international conference on data mining, 2008: 163-272

[27] Hofmann T. Latent semantic models for collaborative filtering.Acm transactions on information systems, 2004, 22: 89-115

[28] Salakhutdinov R., mnih A., hinton G. Restricted boltzmann machines for collaborative filtering.In: proceedings of the 24th unnual international conference on machine learning, 2007: 791-798

[29] Blei D., Ng A., jordan M. Latent dirichlet allocation.Journal of machine learning research, 2003: 993-1022

[30] Hu Y., koren Y., volinsky C. Collaborative filtering for implicit feedback datasets.In: proceedings of ieee international conference on data mining, 2008: 263-272

[31] Sarwar B. M. Application of dimensionality reduction in recommender system—a case study.In proceedings of kdd workshop on web mining for e-commerce: challenges and opportunities, 2000

[32] Takacs G., pilaszy I., nemeth B. Et al.Major components of the gravity recommendation system.Acm sigkdd explorations newsletter, 2007(9): 80-84

[33] Pan rong, zhou yunhong, cao bin et al.One-class collaborative filtering.In: proceedings of ieee 8th international conference on data mining, 2008: 502-511

[34] Schmidt-thieme l.Compound classification models for recommender systems.In: proceedings of 5th ieee international conference on data mining, 2005: 378-385

[35] Rendle s.Bpr: bayesian personalized ranking from implicit feedback.In: proceedings of the 25th conference on uncertainty in artificial intelligence, 2009: 452-461

[36] Rendle s., schmidt-thieme l.Online-updating regularized kernel matrix factorization models for large-scale recommender systems.In: proceedings of the 2008 acm conference on recommender systems.New york, usa, 2008: 251-258

[37] Cao bin, li nathan nan, yang qiang.Transfer learning for collective link prediction in multiple heterogeneous domains.In: proceedings of the 27th international conference on machine learning.Haifa, israel, 2010: 159-166

[38] Li bin, yang qing, xue xiangyang.Can movies and books collaborate? Cross-domain

collaborative filtering for sparsity reduction.In: proceedings of the 21st international joint conference on artificial intelligencei.Pasadena usa: aaai press, 2009: 2052-2057

[39] Li bin, yang qiang, xue xiangyang.Transfer learning for collaborative filtering via a rating-matrix generative model.In: proceedings of 26th annual international conference on machine learning.New york usa: acm, 2009: 617-624

[40] Singh ajit, gordon geoffrey.Relational learning via collective matrix factorization.In: proceedings of the 14th acm sigkdd international conference on knowledge discovery and data mining.New york usa: acm, 2008: 650-658

[41] Xu zhao, kersting kristian, tresp volker.Multi-relational learning with gaussian processes. In: proceedings of the 21st international joint conference on artificial intelligence. Pasadena usa: aaai press, 2009: 1309-1314