An Oracle method to predict NFL games

E. Cabral Balreira
Department of Mathematics
Trinity University
One Trinity Place
San Antonio, TX 78212-7200
ebalreir@trinity.edu

Brian K. Miceli
Department of Mathematics
Trinity University
One Trinity Place
San Antonio, TX 78212-7200
bmiceli@trinity.edu

Thomas Tegtmeyer
Department of Mathematics & Computer Science
Truman State University
100 E. Normal
Kirksville, MO 63501
ttegt@truman.edu

Submitted: xxx; Accepted: xxx
MR Subject Classifications: 05A05, 05A15, 05A19

Abstract

Multiple models are discussed for ranking teams in a league and introduce a new model called the Oracle method. This is a Markovovian method that can be customized to incorporate multiple team traits into its ranking. Using a foresight prediction of NFL game outcomes for the 2002–2013 seasons, it is shown that the Oracle method correctly picked 64.1% of the games under consideration, which is higher than any of the methods compared, including ESPN Power Rankings, Massey, Colley, and PageRank.
Keywords: NFL, rankings, Massey, PageRank, Oracle method

1 Introduction

Former NFL Head Coach Bill Parcells authored the well-known quote, "You are what your record says you are." While this may be true for leagues deciding which teams will move into the playoffs from a regular season, it may not be the case that the win-loss record is the sole indicator of which team is more likely to win a head-to-head matchup. For example, suppose we compare Team A that has a final regular season record of 8 wins and 8 losses against Team

B that has a final regular season record of 6 wins and 10 losses, where Team A has never played Team B. If we were to see that the bulk of Team A's victories come against teams with seven or fewer wins while many of Team B's victories are against teams with eight or more wins, then that makes it seem more likely that Team B could beat Team A in a head-to-head contest. Moreover, head-to-head competition is not enough to give us information on which teams are better than one another. As an example, we could have a 3-team league, say comprised of Teams C, D, and E, where every team plays every other team once, and the outcomes are Team C beats Team D, Team D beats Team E, and Team E beats Team C.

Consider a tournament with n teams, $T_1, \ldots, T_n$, playing K rounds of games, $R_1, \ldots, R_K$. We let $\mathcal{M}^{n\times n}$ denote the set of $n \times n$ real matrices and let $(a_{ij}) \in \mathcal{M}^{n\times n}$ be the matrix with a_{ij} as the i,j-th entry. For each round played in this tournament, we create a matrix whose i,j-th entry is the number of times that T_i has defeated T_j and in the case of a tie between T_i and T_j, we define that each team is awarded 0.5 wins for that match. In a round-robin tournament with no ties allowed, this entry is either 0 or 1. Define for any round, R_m, the matrix $\mathbf{A}^m$ to be the sum of these matrices, that is, $\mathbf{A}^m$ is the matrix whose i,j-th entry is the number of times T_i has defeated T_j in the first m rounds of the tournament. In addition, this gives the sum of the entries of the i-th row to be the total number of wins accumulated by T_i in rounds $1,2,\ldots,m$ and the sum of the entries of the i-th column to be the total number of losses accumulated by T_i in rounds $1,2,\ldots,m$. With each matrix $\mathbf{A}^m$ there is a canonical associated network, denoted by N^m, see Horn and Johnson (1990), with nodes $1,2,\ldots,n$ such that there is a directed edge from j to i for every instance in which T_i beats T_j. We will say that this network is *strongly connected* if, given any two nodes u and v, there is both a directed path from u to v and a directed path from v to u. In the case where N^m is strongly connected, then it will also be the case that $\mathbf{A}^m$ is irreducible, which is a condition that we need in order to employ the new application proposed in this paper.

Given $T_1, \ldots, T_n$, a rating vector $\mathbf{r} = [r_1 \;\; r_2 \;\; \cdots \;\; r_n]^T \in \mathbb{R}^n$ is constructed via some predetermined method, where r_i represents the strength of T_i. We then use $\mathbf{r}$ to form a ranking of these teams, that is, whenever $r_i > r_j$ we say that T_i is ranked higher than T_j. Given a tournament, we wish to rank the nodes of the network corresponding to a given round, say R_m. We then use this ranking to predict the outcomes of games to be played in the next round, R_{m+1}. More precisely, if T_i is set to play T_j in R_{m+1} and node i is ranked above node j in N^m, then we predict that T_i will beat T_j in R_{m+1}. In the case where T_i and T_j have the same rank, we cannot necessarily predict which team is more likely to win. However, in the application of the methods in this paper, namely, predicting the outcomes of NFL games, we predict that the home team will win when identical rankings occur .

There are many well-known methods for creating a ranking system for a given tournament. Some use simple data or human judgement, such as the Issacson-Tarbell Postulate by Easterbrook (2008) or the Experts Pick method, where we use the Power Rankings given by ESPN (2014). Some methods consider the analysis of data from paired comparisons experiments,

where one estimates the probability that a team may defeat another. For instance, this is the case in the models given by Bradley and Terry (1952) and Thurstone (1927). Methods for paired comparisons are still very influential, and we refer the reader to Agresti (2002), David (1963), and Hunter (2004) for details and other applications. Other methods, such as the Massey (1997) and Colley (2002) methods, use elementary ideas from Linear Algebra to solve, or find a best-fit solution to, a linear system $\mathbf{Ar} = \mathbf{b}$ to produce a rating vector. Here the matrix $\mathbf{A}$ contains information about the number of games played between two teams in the tournament and the vector $\mathbf{b}$ encodes a variation on the win-loss differential or the cumulative point differential for each team in the tournament. There are also other ranking methods that use the Perron-Frobenius Theorem to find the steady-state solutions of a Markov process, such as Keener (1993) and Google's PageRank algorithm by Brin and Page (1998). We refer the reader to the recent book of Langville and Meyer (2012) for a greater perspective on several other ranking methods as well as further details on the methods listed in this paper.

In this article, we focus on the ranking problem in the sports setting and introduce a new ranking method, called the *Oracle method*. This method is a customizable ranking algorithm that is influenced by the PageRank method. Recent efforts to optimize the PageRank algorithm, such as the work of Gleich (2011), have yielded important improvements in performance and customization geared toward the Web page ranking industry. In this work with sports rankings, we will see that the Oracle method has the ability to include various traits when compiling a ranking of teams, such as the strength of a team's schedule, the margin of victory, and number of wins.

We organize this paper as follows. In Section 2 we discuss the general notion of rating and ranking teams in a league, and we discuss a few of the more popular methods employed for compiling such rankings. In Section 3 we discuss the Oracle method for ranking teams, and in Section 4, we measure how this new ranking method compares to other well-known methods, using as a barometer of accuracy the foresight prediction outcomes of NFL games in the Super Bowl era. In Section 5 we give a discussion of the results from the previous sections, along with some strengths and weaknesses pertaining to this application of the Oracle model. Finally, in Section 6, some concrete examples of how to implement the Oracle model are laid out, and additional tables of data are given.

2 Ranking Methods

In this section we provide a more detailed description of some previously-known ranking methods.

2.1 The Win-Home Method

In an ESPN.com Tuesday Morning Quarterback article by Easterbrook (2008), he cites a reader proposed method for choosing the winner of NFL games which he calls the *Issacson-Tarbell Postulate*, and which we call the *Win-Home (WH)* ranking. Wishing to find the simplest way to predict NFL game outcomes while still getting good results, this method fits the bill, where if T_i and T_j are scheduled to play in R_{m+1}, we predict that T_i will beat T_j if

(i) T_i has more wins than T_j in the first m rounds, or

(ii) T_i and T_j have the exact number of wins in the first m rounds, but T_i is the home team.

In Section 4, we will use this ranking system as a baseline for comparing all ranking systems in terms of correctly picking the outcomes of games between Weeks 4 (and 11) and the penultimate week of the 1966–2013 NFL seasons.

2.2 The Experts Pick Method

In this method, an expert (or group of experts) ranks the teams of a league. This is the type of ranking we see on ESPN (2014), and typically, it does not deviate far from the rankings one gets when using the WH method given above. That said, this is the one method we give that allows for humans to make predictions using any information available, including data that cannot be incorporated into any of the other models presented in this article (for example, if the starting quarterback for a team gets injured).

2.3 Massey and Colley Methods

Some methods for establishing a ranking require solving, or finding a best-fit solution to, the linear system $\mathbf{Ar} = \mathbf{b}$. Here, based on some set of predetermined conditions, $\mathbf{A}$ and $\mathbf{b}$ are formed using the results from the tournament, and the $n \times 1$ vector $\mathbf{r}$ corresponds to a rating of the teams $T_1, \ldots, T_n$. The two predominant methods in this area are those of Massey (1997)[1], which accounts for the scores of tournament games, and Colley (2002), which does not.

Massey's method finds the least squares solution, $\mathbf{r}$, to the system $\mathbf{Mr} = \mathbf{y}$, where $\mathbf{M} = (M_{ij}) \in \mathcal{M}^{n\times n}$ and $\mathbf{y} = (y_i) \in \mathcal{M}^{n\times 1}$ are constructed in the following manner:

(i) The entry M_{ii} is the number of games played by T_i for each $1 \leq i \leq n-1$, M_{ij} is the negative of the number of games played between T_i and T_j for $i \neq j$ and $1 \leq i \leq n-1$, and $M_{nj} = 1$ for every j.

[1]The method of Massey used for the BCS Rankings in college football is proprietary, and thus not publicly available. The method we discuss is the original idea of Massey (1997), which he developed for an honors thesis as an undergraduate at Bluefield College.

(ii) The coordinate y_i is the total number of points scored against T_i subtracted from the total number of points scored by T_i for $1 \leq i \leq n-1$ and $y_n = 0$.

We then let the i-th coordinate of the least squares solution $\mathbf{r}$ be the rating of T_i. In Massey's method, the solution $\mathbf{r}$ has the property that $r_i - r_j$ is the expected score differential of the match between T_i and T_j.

Colley's method finds the exact solution to the system $\mathbf{Cr} = \mathbf{t}$, where $\mathbf{C} = (C_{ij}) \in \mathcal{M}^{n\times n}$ and $\mathbf{t} = (t_i) \in \mathcal{M}^{n\times 1}$ are constructed in the following manner:

(i) The entry C_{ii} is two plus the number of games played by T_i for each i, and C_{ij} is the negative of the number of games played between T_i and T_j for $i \neq j$.

(ii) The coordinate t_i is given by $t_i = 1 + \frac{1}{2}(w_i - \ell_i)$, where w_i is the number of wins by T_i and ℓ_i is the number of losses by T_i.

We then let the i-th coordinate of the exact solution $\mathbf{r}$ be the rating of T_i. Here, scores of tournament games are not accounted for, but rather, Colley's idea is that a team should receive more credit for defeating a stronger opponent than for defeating a weaker opponent, regardless of the score of the contest.

2.4 Bradley-Terry Model

An alternate approach to view a ranking system is to use the ratings of the teams to determine the probability that one team will defeat another. This is the basic assumption of the pair preference model by Bradley and Terry (1952). More precisely, Bradley and Terry produce a rating vector, $\mathbf{r}$, and then they define

$$\pi_{ij} = \frac{r_i}{r_i + r_j} \tag{2.1}$$

to be is the probability that T_i defeats T_j. Details and further information on the Bradley-Terry model can be found in Agresti (2002). A simple iterative algorithm for finding the maximum likelihood estimate, π_{ij}, has been known for a long time, see Zermelo (1929). Recent work in Hunter (2004) produces iterative maximum likelihood estimation algorithms for a wide class of generalizations of the Bradley-Terry model.

When there are not enough paired comparisons and the comparison table is sparse, this method can have undesirable features and does not provide meaningful results, see Agresti and Hitchcock (2005). In fact, there are known conditions that ensure whether or not the algorithms to find π_{ij} converge, see Ford (1957) and Hunter (2004). Namely, one must show that in every possible partition of the teams into two nonempty subsets, some team in the second subset beats some team in the first subset at least once. This condition was first observed in Ford (1957) and it is equivalent to showing that the associated directed network corresponding to this tournament is strongly connected.

When not enough games are played to ensure that the associated directed network corresponding to a tournament is strongly connected, one can opt to use an approach in Keener (1993), where the games scores are used to provide a reasonable estimate of π_{ij} in Equation (2.1). Namely,

$$\pi_{ij} \approx \frac{s_{ij}}{s_{ij} + s_{ji}}.$$

where s_{ij} is the total number of points scored by T_i against T_j. In this fashion, if $s_{ij} > 0$, one adds a directed edge to from T_j to T_i. Hence, in sports such as the NFL where teams usually score some points in each match, in order to satisfy the strong connectivity given in Ford (1957) to obtain unique rankings, we must check if the undirected network is strongly connected.

In this paper, we check for strong connectivity of the directed network of NFL tournaments and in fact, some seasons – 1972, 1976, 2007, and 2008 – are never strongly connected as a directed graph, since there was an undefeated or a winless team. In general, over 12 weeks of play are required for an NFL season to satisfy Ford's condition. This is in contrast with the (mostly) undirected network which is always strongly connected after week 3. Hence, in this paper we use the estimative approach of Keener in order to obtain the Bradley-Terry data[2] in Section 4.

2.5 Markov Methods

For a Markov ranking method, consider a tournament which has completed R_m and consider a random walker on N^m. Here, we let the ranking for T_i be an indication of the long-run proportion of time this random walker spends at node i. To be more specific, imagine placing a random walker at one of the n nodes at some starting time $t = 0$. At each time step this random walker will be allowed to leave its current node and move along a single directed edge of N^m to a (possibly) new node in the network. Then, as the number of time steps increases, we compute the proportion of time, r_i, that this random walker spends at each node i, and we will allow that proportion to be the rating of each team T_i. Constructing a probabilistic ratings vector $\mathbf{r} = [r_1 \;\; r_2 \;\; \cdots \;\; r_n]^T$ in this fashion forces the conditions each $r_i \geq 0$ and $r_1 + r_2 + \cdots + r_n = 1$. We see from this description that one must define the transitional probabilities for each node, that is, the probability that if this walker is at node i, then this walker will move to node j at the next time step. It is here that the flexibility of this approach becomes more evident, as one can incorporate the score outcomes of contests between T_i and T_j, the total number of wins for each of T_i and T_j, etc. when defining each transitional probability. Before we get ahead of ourselves, however, we must first be able to show that such long-run proportions are well-defined, which is where the Perron-Frobenius Theorem, stated below, comes into play. For a proof, we refer the reader to Keener (1993).

[2]For this data, we compute the rating vector based on the algorithm and MATLAB routine given in Hunter (2004).

Theorem 2.1 (Perron-Frobenius)**.** *Let $\boldsymbol{A}$ be an $n \times n$ matrix with associated network N. If N is strongly connected, then there exists a positive, real eigenvalue λ of $\boldsymbol{A}$ such that*

(i) $\lambda \geq |\tau|$ *for any eigenvalue τ of $\boldsymbol{A}$,*

(ii) there exists an eigenvector, $\boldsymbol{r} = [r_1 \;\; r_2 \;\; \cdots \;\; r_n]^T$ of $\boldsymbol{A}$ associated with λ such that $r_i \geq 0$ for all i and $r_1 + r_2 + \cdots + r_n = 1$, and

(iii) if $\boldsymbol{x} = [x_1 \;\; x_2 \;\; \cdots \;\; x_n]^T$ is any eigenvector of $\boldsymbol{A}$ such that $x_i > 0$ for every i, then $\boldsymbol{r}$ and $\boldsymbol{x}$ are multiples of one another.

The λ and $\mathbf{r}$ that are defined in Theorem 2.1 are called the *Perron value* and *Perron vector*, respectively. Further, we see from part (ii) of this theorem that $\mathbf{r}$ satisfies the properties of a ratings vector in this probabilistic setting. Accordingly, once the transitional probabilities for a Markov ranking method are defined, we allow the ratings vector of the tournament be the corresponding Perron vector. This is accomplished in the following manner. First, from a strongly connected network, N^m, we consider a corresponding, weighted matrix, $\mathbf{A}_w^m = (a_{ij}^w)$, where each a_{ij}^w is a nonnegative value that incorporates some statistic taken from the game played between T_i and T_j. This could simply be done by letting $\mathbf{A}_w^m = \mathbf{A}^m$, or for example, suppose T_i beats T_j by a score of 27-10. We may let $a_{ij}^w = 17$, the score difference between these teams, and we set $a_{ji}^w = 0$. We then make a column-stochastic matrix $\mathbf{P}^m = (p_{ij})$ by defining

$$p_{ij} = \frac{a_{ij}^w}{\sum_{k=1}^n a_{kj}^w}, \tag{2.2}$$

and then allowing p_{ij} to be the transitional probability, that is, the probability that a random walker will proceed from node j to node i. In this case, one can show that the Perron value of $\mathbf{P}^m$ is $\lambda = 1$.

2.5.1 The PageRank Method

We now discuss a well-known ranking method, Google's PageRank algorithm. Developed by Brin and Page (1998) and Page, Brin, Motwani, and Winograd (1999), the PageRank ranking algorithm is an important part of the original Google search engine. Using an adjacency matrix formed out of Web page links, this algorithm constructs a Hyperlink matrix that will be used to compute the rating scores of each Web page prior to the user's query. In order to guarantee that the Hyperlink matrix satisfies the Perron-Frobenius Theorem, the algorithm tweaks this matrix to guarantee that every Web page can be visited from any other Web page by adjusting the dangling nodes, i.e., nodes with no outlinks, and adding a special rank one, teleportation matrix. We will summarize the details below using the above terminology of tournaments, but we refer the reader to Langville and Meyer (2006) for complete details.

In the application of tournaments, we view the adjacency matrix as

$$a_{ij} = \begin{cases} 1 & \text{if } T_j \text{ beats } T_i, \text{ and} \\ 0 & \text{otherwise.} \end{cases}$$

Next, we form the Hyperlink matrix $\mathbf{H}$ as

$$h_{ij} = \begin{cases} 1/\sum_{k=1}^{n} a_{kj} & \text{if } T_j \text{ beats } T_i, \text{ and} \\ 0 & \text{otherwise.} \end{cases}$$

Here, the dangling nodes would correspond to undefeated teams which would have no directed edges coming out of them. Assuming that T_ℓ is undefeated, the aforementioned tweak would be to replace the ℓ-th column of $\mathbf{H}$ with the vector $\frac{1}{n}\mathbf{e}$, where $\mathbf{e}$ is the $n \times 1$ vector with all 1's as its entries and then rename this new matrix as $\mathbf{H}$. In the setting of a random walk, this is analogous to moving from node ℓ to any other node in the network with equal probability. In order to ensure that the corresponding network is strongly connected, the PageRank algorithm chooses a value $\alpha \in (0,1)$ and a vector $\mathbf{v} = [v_1\ v_2\ \cdots\ v_n]^T$ such that each $v_i \geq 0$ and $v_1 + v_2 + \cdots + v_n = 1$. The last step is to define the Google Matrix:

$$\mathbf{G}_\alpha = \alpha\mathbf{H} + (1-\alpha)\mathbf{v}\mathbf{e}^T. \tag{2.3}$$

The vector $\mathbf{v}$ is called a personalization vector and the usual choice of $\mathbf{v} = \frac{1}{n}\mathbf{e}$ is the usual PageRank method. If we think of a random walk in heuristic terms, the choice of α yields that with probability α the random walker will follow the links from the network, while with probability $1-\alpha$ the walker will be teleported to different node in the network given by the probabilities of the personalization vector[3]. The PageRank rating, and consequent ranking, is obtained from the unique rating vector $\mathbf{r}$ of $\mathbf{G}_\alpha$ as guaranteed by the Perron-Frobenius Theorem.

In this article, we consider different choices of the personalization vector, $\mathbf{v}$, and the corresponding model is denoted as PageRank($\mathbf{v}$). For instance, one can use $\mathbf{v}$ to account for the number of wins or the number of points scored. It is worth noting at this point that in the Web search industry there is a great deal of ongoing research, such as that of Constantine and Gleich (2010), on understanding how the choice of the personalization vector and teleportation probability changes the Web search results.

2.5.2 Keener and Biased Random Walker Ratings

Two other well-established Markov ranking algorithms are the method developed by Keener (1993) and the Biased Random Walker method developed by Callaghan, Mucha, and Porter (2007).

Keener's approach is to use game scores to compute ratings, employing, in part, Laplace's rule of succession to define his transitional probabilities. Indeed, Keener defines what we denote

[3] In Langville and Meyer (2004), it is given that Google originally used $\alpha = 0.85$.

as a weighted matrix (a_{ij}^w) with

$$a_{ij}^w = h\left(\frac{s_{ij}+1}{s_{ij}+s_{ji}+2}\right),$$

where $h(x) = \frac{1}{2} + \frac{1}{2}\text{sgn}\left(x-\frac{1}{2}\right)\sqrt{|2x-1|}$ and s_{ij} is the total number of points scored by T_i against T_j. We note that h is continuous, $h\left(\frac{1}{2}\right) = \frac{1}{2}$, and away from $\frac{1}{2}$, h goes rapidly to zero or one. Finally, using Equation (2.2), we define the transitional probabilities to move from one node to another.

In the method of Callaghan et al. (2007)[4] , the random walker is biased, based on game outcomes, in its movements about the network. Here, the biased random walker finds himself at a node, then he selects a match and chooses, with some fixed probability $p \in (\frac{1}{2}, 1)$, to move to the winner of that match, whether or not the winner is the current node. More precisely, if T_i has defeated T_j, then we add an edge from i to itself with weight p and an edge from i to j with weight $1-p$. In addition, we also add an edge from j to itself with weight $1-p$ and an edge from j to i with weight p. Since each team may play several games, in order for the Biased Random Walker method to be viewed as a Markov process, we must then add all the weights from the possible multiple edges. Then, again using Equation (2.2), we define the transitional probabilities to move about the network.

3 The Oracle Method

The new ranking method defined in this paper, called the *Oracle method*, is a customizable Markov method that has the ability to consider multiple teams' traits at once. The main idea is to introduce an unbiased computer expert to aid our random walker in deciding where to proceed in the network. To accomplish this, we create a new node in the network, called the *Oracle* node, to be the $(n+1)$-st node. This idea is a generalization of the personalization vector in the PageRank method that addresses a fundamental flaw when using certain Markov methods to rank a tournament, namely, when an undefeated team loses to a winless team, and subsequently, that previously winless team rises up near the top of the rankings. The Oracle method not only provides a novel approach to this problem, but introduces several ways to take other factors of potential interest into consideration when ranking teams in a tournament. While we discuss the basic machinery of the Oracle method here, a detailed example of this algorithm is given in Section 6.1.

Formally, we consider the Oracle to be a new, $(n+1)$-st node in the network, N^m, associated to the tournament. We denote the new network, $N_{\mathbf{O}}^m$, to be the network N^m with a new node, $n+1$, that has a directed edge to and from node $n+1$ to each node $1, 2, \ldots, n$. This new node changes the associated $n \times n$ matrix $\mathbf{A}^m$ into an $(n+1)\times(n+1)$ Oracle matrix, $\mathbf{O}^m$, given by

[4]In Callaghan et al. (2007), the authors named their method as Random Walker Ranking. As that description may also fit other Markov methods, we refer to it as Biased Random Walker.

$$\mathbf{O}^m = \left(\begin{array}{c|c} \mathbf{A}^m & \mathbf{e} \\ \hline \mathbf{e}^T & 0 \end{array} \right). \tag{3.1}$$

Using Equation (2.2) to compute the transitional probabilities, we have that

$$p_{n+1,i} = \frac{1}{\ell_i + 1} \text{ and } p_{i,n+1} = \frac{1}{n},$$

where again, ℓ_i is the number of losses by T_i.

This shows that when the random walker visits teams with fewer loses, it will be more likely move up to the Oracle. Also, once this walker visits the Oracle, it will subsequently move to any team with equal likelihood, similar to the personalization vector modification of the PageRank method. Observe that if we assign nonnegative transitional probabilities in moving to and from the Oracle node, we ensure that the associated network is strongly connected. Hence, the Perron-Frobenius Theorem can always be applied to find an Oracle rating vector, $\mathbf{r}^{\mathbf{O}} \in \mathbb{R}^{n+1}$. We then define the rating vector, $\mathbf{r} \in \mathbb{R}^n$, by

$$r_i = \frac{r_i^{\mathbf{O}}}{\sum_{j=i}^{n} r_i^{\mathbf{O}}}. \tag{3.2}$$

3.1 Oracle Variants

As aforementioned, one of the advantages of this newly introduced method is the fact that it is customizable. In particular, for $1 \leq i, j \leq n$, the Oracle method has the ability to consider multiple traits separately in assigning the transitional probabilities while moving from node i to node j, the transitional probabilities in moving from node i to the Oracle node, and the transitional probabilities in moving from the Oracle node to node i. A discussion of how or why the user might choose one customization over another is given in Section 5.

We shall refer to the moves from node i to the Oracle node as *up moves* and moves from the Oracle node to node i as *down moves*. Hence, while traversing $N_{\mathbf{O}}^m$, the random walker may follow the original network, N^m, or go up to the Oracle node and subsequently move down to a team by considering the Oracle as an unbiased computer expert that will guide the random walker. In practice, the user assigns, via the transitional probabilities, the traits that the Oracle will consider to be most important, and then the edges connected to the Oracle will be weighted based on these traits. In order to accomplish this, let us refer to the vector $\mathbf{u}$ as the *up direction vector* and the vector $\mathbf{d}$ as the *down direction vector* as depicted in Figure 1.

This modification, together with the classical modifications of the matrix $\mathbf{A}^m$, changes the Oracle matrix to be a customized, weighted matrix given by

$$\mathbf{O}_w^m(\mathbf{u}, \mathbf{d}) = \left(\begin{array}{c|c} \mathbf{A}_w^m & \mathbf{d} \\ \hline \mathbf{u}^T & 0 \end{array} \right). \tag{3.3}$$

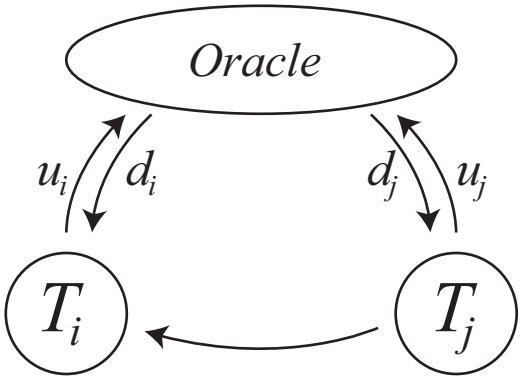

Figure 1: After the end of m rounds, the Oracle is viewed as T_{n+1} in the network N^m that has a directed edge to and from each T_i. The direct edges can be weighted.

The entries of the vectors $\mathbf{u}$ and $\mathbf{d}$ may be chosen in several different ways to reflect statistics which are particular to the interaction between two teams. We shall refer to a particular modification of the Oracle as given in (3.3) by w-Or$(\mathbf{u},\mathbf{d})$, and in the case where $\mathbf{A}_w^m = \mathbf{A}^m$ we will simply denote this model and corresponding matrix as Or$(\mathbf{u},\mathbf{d})$ and $\mathbf{O}^m(\mathbf{u},\mathbf{d})$, respectively.

While several variants for the up and down direction vectors may exist, the two modifications that we consider the most natural for a sports tournament are as follows. First, one can replace $\mathbf{u}$ and/or $\mathbf{d}$ with the win vector $\mathbf{w}$, whose i-th coordinate is w_i, the number of wins by T_i. Second, we could replace either of the direction vectors with the score vector $\mathbf{s}$, whose i-th coordinate is s_i, the sum of the margin of victories of the games T_i has won. We also define for any vector $\mathbf{v} \in \mathbb{R}^n$, the vector $\mathbf{v}^+$ such that $v_i^+ = v_i + 1$. We observe that in some applications it is possible that a team is winless or that there may be multiple scoreless matches (such as in soccer), meaning that the network is not strongly connected even with the addition of the Oracle node. Using $\mathbf{w}^+$ or $\mathbf{s}^+$ in these cases will always ensure that the network is strongly connected.

Finally, one can also view the introduction of the Oracle node as a fictitious competitor who wins one game and loses one game to every real opponent. This interpretation would not change the relative winning records of the other teams, but does ensure that the condition in Ford (1957) is satisfied, and hence the ordinary Bradley-Terry model can be used on our augmented Oracle matrix. We shall refer to this modification as the Or-Bradley-Terry model.

3.2 An Example to Illustrate the Oracle Method

Let us consider a round-robin tournament with six teams, $T_1, T_2, \ldots, T_6$, where we only have information about the win or loss outcome, summarized in Table 1.

This example will illustrate how the Oracle method addresses what we consider a fundamental flaw when using other Markov ranking methods, in particular the PageRank method. Consider the game played in the last round, where the previously undefeated T_1 loses to the

Round 1	Round 2	Round 3	Round 4	Round 5
T_1 beats T_4	T_1 beats T_5	T_1 beats T_3	T_1 beats T_2	T_6 beats T_1
T_3 beats T_5	T_2 beats T_3	T_4 beats T_2	T_5 beats T_4	T_3 beats T_4
T_2 beats T_6	T_4 beats T_6	T_5 beats T_6	T_3 beats T_6	T_2 beats T_5

Table 1: A simulated round-robin tournament with six teams.

previously winless T_6. To find the PageRank rating for this tournament, the initial step is to construct $\mathbf{A}^5$ and the associated network, N^5. In order to find the Oracle ranking, we introduce the Oracle node and consider the matrix $\mathbf{O}^5$ and the associated network, $N^5_{\mathbf{O}}$. These networks are depicted in the left and right side of Figure 2, respectively.

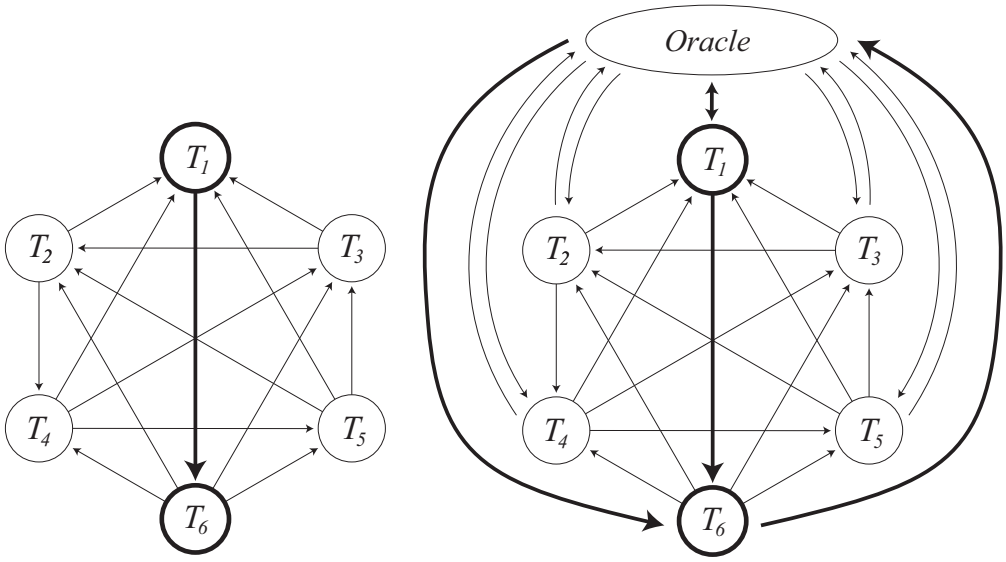

Figure 2: The associated networks N^5 and $N^5_{\mathbf{O}}$ after completion of the simulated, six-team tournament.

Once we have the networks in place, we consider two PageRank ratings, one with the uniform Google method and the other using the personalization vector $\mathbf{w}^+$; we also consider the three Oracle models Or($\mathbf{e},\mathbf{w}^+$), Or($\mathbf{e},\mathbf{w}$), and Or($\mathbf{w},\mathbf{w}$). For each of these five methods, the final rankings of the six teams are given in Table 2. We see from this table that both PageRank methods, and in fact many other Markov methods as well, will promote T_6 to the 2nd highest ranking[5], simply because they have beaten the best team. The main reason for this happening is quite straightforward, as if one looks at the corresponding network on the left of Figure 2, we see that anytime the random walker happens upon the node for T_1, they must always proceed to

[5]All the variations of PageRank we have tested, always promote T_6 to the 2nd highest ranking.

the node for T_6 at the next time step. The alternative provided by the Oracle method, where the random walker now has the option of proceeding to the Oracle node after it visits T_1, greatly changes these rankings depending on the statistics used in dictating the walker's movements. In fact, using the three Oracle customizations here, one may have T_6 as the 3rd, the 4th, or even the 6th best team, which highlights different features for each customization.

	Win-Loss	PageRank	PageRank($\mathbf{v}^+$)	Or($\mathbf{e}$, $\mathbf{w}^+$)	Or($\mathbf{e}$, $\mathbf{w}$)	Or($\mathbf{w}$, $\mathbf{w}$)
T_1	4-1	1st	1st	1st	1st	1st
T_2	3-2	3rd	3rd	2nd	2nd	2nd
T_3	3-2	4th	4th	4th	3rd	3rd
T_4	2-3	5th	5th	5th	5th	4th
T_5	2-3	6th	6th	6th	6th	5th
T_6	1-4	2nd	2nd	3rd	4th	6th

Table 2: Final rankings of the simulated tournament using PageRank and Oracle methods.

4 NFL predictions – 1966–2013

In this section we apply the Oracle method Or($\mathbf{w}^+, \mathbf{s}^+$) towards predicting the outcome of NFL games. We predict only those games which transpire between Weeks 4 and the penultimate week of each season in the Super Bowl era, 1966–2013, and the data[6] given uses only these weeks unless otherwise stated.

Using the WH method as a baseline, we compare the season-by-season accuracy of the Oracle method to the prediction accuracy of the eight previously defined ranking systems from Section 2: Experts Pick, Massey, Colley, Bradley-Terry model (Keener estimate), PageRank with $\alpha = 0.85$, Keener, and Biased Random Walker with $p = 0.75$. In the Appendix, Table 4 shows the prediction results for several other Oracle variants, PageRank for various values of α and personalization vectors, and Biased Random Walker for other values of p. For the Experts Pick method, we use the weekly ESPN Power Rankings (ESPN PR) as the expert ranking. The earlier rankings available online only date back to 2002, so the predictions are split into two categories: 1966–2013 and 2002–2013.

The reason for using the WH method as a baseline for predictive power (in fact, the impetus for choosing this application for the Oracle method in the first place) is an article by Gregg Easterbrook (2008). In this article, Easterbrook states the opinion that the WH method is a relatively reliable way of choosing games, in fact, better than most pundits do with whatever up-to-date information they have; moreover, it requires very little information to apply ("you don't

[6] All computations were performed using MATLAB R2013a.

even need to know who's playing", Easterbrook (2008)) and no computing power whatsoever. The reason for only including games up through the penultimate week is simple, for in this same article by Easterbrook, it is also states that the WH method "does have a weakness," which is that is does not do well in predicting outcomes of games in the final week of the season Indeed, the playoff seeds for a team may already be set in the last week of the NFL and a team may not play all of it starters, hence the outcomes of games in the final week of the season do not reflect the true ability of a team. Thus, to be fair we exclude this week from all prediction comparisons in this section, which are summarized in Table 3. We have also included an identical prediction comparison taking into consideration the final week of each season, and this is shown in the Appendix in Table 8.

We can consider the starting week for our predictions as early as Week 4 and as late as Week 11. The reason for beginning with games in Week 4 is because we need the tournament networks to be strongly connected for *all* models considered before we can use the corresponding irreducible matrices to make foresight predictions. Through MATLAB we verified that the networks for all models do become strongly connected in either Weeks 2 or 3 of each season, i.e., we can employ all of the computerized ranking methods to make predictions beginning with the Week 4 games of each season. From a practical standpoint, beginning after three completed weeks of competition allows each model to gather more "starting data" in order to hopefully make better predictions. The reason to also consider the predictions starting at Week 11 is to allow higher reliability of the rankings as majority of the season has been played and we may be more confident in the rankings at that point. We also did not consider foresight predictions starting after Week 11 since earlier NFL seasons (prior to 1978) had only 14 weeks and because we are excluding the last week, we wanted to have at least three weeks where we could predict games outcomes. We remark that the 1982 NFL season was shortened to nine weeks due to a player strike. Hence, we did not included it in the late foresight predictions starting at Week 11. In addition, one could consider a modification of the foresight prediction method where we start predicting games at Week 11, using the rank data up to Week 10 and predict the subsequent weeks as if each week is the eleventh week. This approach potentially avoid mixing different levels of reliability and variations of the ranking methods after the results from week to week. The results of this prediction model, which we call the 10-*week fixed* foresight model, are similar to the other foresight predictions in this paper, and we include them in the Appendix in Table 7.

Finally, in choosing the WH method as the baseline, we also programmed it to be the case that if any method gives two NFL teams the same rank in a week where they will be playing each other, the routine chooses the home team as the winner.

When starting the foresight predictions in Week 4 and Week 11, the WH method correctly picked 62.89% and 63.94%, respectively, of the NFL games played in the 1966–2013 seasons and 62.25% and 65.76%, respectively, of the games played in the 2002–2013 seasons. These numbers are given in the lefthand side of Table 3, as well as the prediction percentages for the

	1966–2013		2002–2013	
Starting Week	Week 4	Week 11	Week 4	Week 11
WH	62.89%	63.94%	62.25%	65.76%
ESPN PR	n/a	n/a	**63.02%**	65.48%
Massey	62.77%	**64.99%**	**63.80%**	**67.41%**
Colley	61.79%	63.15%	61.99%	65.32%
PageRank	59.33%	61.44%	58.67%	60.61%
Keener	60.08%	62.73%	61.04%	65.07%
Biased Voter	60.80%	62.86%	61.17%	64.19%
Bradley-Terry Model	62.49%	**64.23%**	**63.58%**	**66.98%**
Oracle(w+,s+)	**63.41%**	**65.09%**	**64.10%**	**66.63%**

Table 3: Average of correct foresight prediction percentage of all NFL games starting either in Week 4 or in Week 11. In bold, we indicate methods whose predictions, on average, are better than or equal to the WH method.

other methods. For instance, the Or($\mathbf{w}^+, \mathbf{s}^+$) method correctly picked 63.41% of games starting from Week 4 and 65.09% of all games starting in Week 11 of the NFL games played in the 1966–2013 seasons, outperforming all of the methods given in Section 2. During similar predictions for NFL games played in the 2002–2013, the Or($\mathbf{w}^+, \mathbf{s}^+$) method is only outperformed by the Massey method during the foresight predictions starting from Week 11. We highlight that the other Markov methods - PageRank, Keener, Biased Random Walker - had fewer foresight predictions than the WH method during all seasons.

5 Discussion

The Oracle method used in this paper to rank teams in a tournament was initially developed to address a flaw when using certain Markov methods to rank teams in a tournament. Namely, when an undefeated team loses to a winless team, and subsequently, that previously winless team rises up to be ranked as the second best team, this is viewed by most as an incorrect ranking. The method to generalize the notion of teleportation by introducing an Oracle node, and then using the choices of the up and down vectors given in the previous sections, indeed solves this problem when ranking teams in a sports tournament. Once this problem had been addressed, we chose to further validate the Oracle method rankings by using a particular variant, Or($\mathbf{w}^+, \mathbf{s}^+$), to perform foresight predictions of NFL contests and then comparing this Oracle method's predictive powers against other well-known ranking methods. We decided to use the NFL, instead of another major sports because the NFL has a very structured schedule and matches that rarely end up in tie. We chose a foresight prediction method to evaluate the ranking

methods as this closely approach the experience of a sports fan. Namely, a fan predicts the outcome of the games in the next round based on all the cumulative information of the season up to that round. We provide results starting predictions at Week 4 to best compare with the expert analyses that begin as early as rosters are announced. We also present our prediction numbers when starting our predictions at Week 11 so that all ranking methods have more of the season to compile their ranks, and accordingly, the predictions could be more reliable. In addition, we can also consider the starting week of our predictions to be any week between Week 4 and Week 11, and we provide the accompanying data in Tables 5 and 6 of the Appendix.

We find that the primary strength of this model lies in its ability to consider team traits which are customized according to the whims of the user. For example, a user may believe that some team statistic, such as time of possession, is one of the most important traits in ranking NFL teams. Accordingly, the user can implement the Oracle method by encoding the time of possession statistic into the up and down vectors, essentially making it the most important factor in this user's rankings. Despite the possibilities for personalizations for the Oracle method and other ranking methods, the traits that performed the best were still the traits that any successful team would like to have - a large number of victories and scoring as many points as possible, which are of course, not unrelated. Another strength of the Oracle method is that by having positive up and down vectors, the introduction of the Oracle node in the network always makes the network strongly connected, thereby providing an actual rank with only a small set of data[7].

In terms of weaknesses of the Oracle method relative to other, non-human algorithms, there is no *a priori* method for assigning the traits of the up and down vectors to give the best predictions. In fact, the overall performances of many the different Oracle variants are very close to each other with respect to NFL game predictions, as we see in Table 4, and so in practice, it is quite possible that only a few traits ever need to be considered. Another shortcoming is that the Oracle method, as implemented here, gives no bias to a team playing at home unless they have an identical ranking to their current opponent, which almost never occurs[8]. Since we are comparing the Oracle method (and all others) to the WH method, it seems like this would be good to incorporate somehow, and we note that there are probabilistic models which account for home field advantage, such as the model of Bradley and Terry (1952). Finally, unlike the Massey method, the Oracle method is not constructed so as to rank teams in a way that allows the user to predict score outcomes of games.

We also observe from Table 4 that the Or-Bradley-Terry model, has similar performance to other ranking methods, but it is still lower than the WH method and also lower than the Keener

[7]One could certainly argue that this addition of the Oracle node opens the possibility for a distortion of the rankings in some way, especially by artificially forcing connectedness early in season. However, the data supports that, at least for *standard* choices of the statistics - score differential, wins, etc. - incorporated into the up and down vectors, this does not happen.

[8]Teams with identical ranks for all methods other than WH meet, on average, less than once per year in the weeks considered. In the WH method, teams with the same record meet, on average, 24 times per season in the weeks considered.

variant we use throughout his paper. This leads one to posit that the improvement from the Oracle method is due to using Markov estimation methods and not simply by adding a "new team" in the tournament. The natural interpretation of the direct edges to and from the Oracle indicates that under the Markovian method, the *standard* choices of the statistics, such as wins and scores, are good measures to rank the teams in the NFL.

In summary, we have provided a viable and novel alternative to previously studied Markovian processes to rank tournaments. Furthermore, the groundwork developed here may be adapted by others according to their particular dataset and desired validation methods.

Acknowledgments

All game data used in the compilation of the rankings in this article was retrieved from Sports Reference (2014).

The authors thank the anonymous referees for their thoroughly meticulous review of our initial manuscript and subsequent revisions. Their suggestions helped us make more accurate the data provided in this paper, add more models for comparison, and ultimately, allowed us to tremendously improve the presentation of this paper.

6 Appendix

One of the advantages in using the Oracle method is the possible customization that can be implemented. We will now show in more detail how these customizations are done and how the ranks are computed. We also include an expanded list of the methods we considered in the NFL predictions from Section 4.

6.1 Oracle Implementation

Let us take a second look into six-team the round-robin tournament with outcomes summarized in Table 1. As stated before, one can consider different up and down vectors for the Oracle method. Indeed, having only information about the outcomes of the games, we use only $\mathbf{e}$, $\mathbf{w}$, and $\mathbf{w}^+$ as the possible up and down vectors in the ranking, and after five rounds we have $\mathbf{w} = [4\ \ 3\ \ 3\ \ 2\ \ 2\ \ 1]^T$ and $\mathbf{w}^+ = [5\ \ 4\ \ 4\ \ 3\ \ 3\ \ 2]^T$. The corresponding Oracle matrices and their corresponding column-stochastic matrix $\mathbf{P}^m = (p_{ij})$ would be as follows:

$$\mathbf{O}^5(\mathbf{e},\mathbf{w}^+)=\begin{pmatrix}0&1&1&1&1&0&5\\0&0&1&0&1&1&4\\0&0&0&1&1&1&4\\0&1&0&0&0&1&3\\0&0&0&1&0&1&3\\1&0&0&0&0&0&2\\1&1&1&1&1&1&0\end{pmatrix} \text{ and } \mathbf{P}^5(\mathbf{e},\mathbf{w}^+)=\begin{pmatrix}0&\frac{1}{3}&\frac{1}{3}&\frac{1}{4}&\frac{1}{4}&0&\frac{5}{21}\\0&0&\frac{1}{3}&0&\frac{1}{4}&\frac{1}{5}&\frac{4}{21}\\0&0&0&\frac{1}{4}&\frac{1}{4}&\frac{1}{5}&\frac{4}{21}\\0&\frac{1}{3}&0&0&0&\frac{1}{5}&\frac{3}{21}\\0&0&0&\frac{1}{4}&0&\frac{1}{5}&\frac{3}{21}\\\frac{1}{2}&0&0&0&0&0&\frac{2}{21}\\\frac{1}{2}&\frac{1}{3}&\frac{1}{3}&\frac{1}{4}&\frac{1}{4}&\frac{1}{5}&0\end{pmatrix}.$$

$$\mathbf{O}^5(\mathbf{e},\mathbf{w})=\begin{pmatrix}0&1&1&1&1&0&4\\0&0&1&0&1&1&3\\0&0&0&1&1&1&3\\0&1&0&0&0&1&2\\0&0&0&1&0&1&2\\1&0&0&0&0&0&1\\1&1&1&1&1&1&0\end{pmatrix} \text{ and } \mathbf{P}^5(\mathbf{e},\mathbf{w})=\begin{pmatrix}0&\frac{1}{3}&\frac{1}{3}&\frac{1}{4}&\frac{1}{4}&0&\frac{4}{15}\\0&0&\frac{1}{3}&0&\frac{1}{4}&\frac{1}{5}&\frac{3}{15}\\0&0&0&\frac{1}{4}&\frac{1}{4}&\frac{1}{5}&\frac{3}{15}\\0&\frac{1}{3}&0&0&0&\frac{1}{5}&\frac{2}{15}\\0&0&0&\frac{1}{4}&0&\frac{1}{5}&\frac{2}{15}\\\frac{1}{2}&0&0&0&0&0&\frac{1}{15}\\\frac{1}{2}&\frac{1}{3}&\frac{1}{3}&\frac{1}{4}&\frac{1}{4}&\frac{1}{5}&0\end{pmatrix}.$$

$$\mathbf{O}^5(\mathbf{w},\mathbf{w})=\begin{pmatrix}0&1&1&1&1&0&4\\0&0&1&0&1&1&3\\0&0&0&1&1&1&3\\0&1&0&0&0&1&2\\0&0&0&1&0&1&2\\1&0&0&0&0&0&1\\4&3&3&2&2&1&0\end{pmatrix} \text{ and } \mathbf{P}^5(\mathbf{w},\mathbf{w})=\begin{pmatrix}0&\frac{1}{5}&\frac{1}{5}&\frac{1}{5}&\frac{1}{5}&0&\frac{4}{15}\\0&0&\frac{1}{5}&0&\frac{1}{5}&\frac{1}{5}&\frac{3}{15}\\0&0&0&\frac{1}{5}&\frac{1}{5}&\frac{1}{5}&\frac{3}{15}\\0&\frac{1}{5}&0&0&0&\frac{1}{5}&\frac{2}{15}\\0&0&0&\frac{1}{5}&0&\frac{1}{5}&\frac{2}{15}\\\frac{1}{5}&0&0&0&0&0&\frac{1}{15}\\\frac{4}{5}&\frac{3}{5}&\frac{3}{5}&\frac{2}{5}&\frac{2}{5}&\frac{1}{5}&0\end{pmatrix}.$$

For the ranking methods PageRank and PageRank($\mathbf{w}^+$), we simply used

$$\mathbf{H}=\begin{pmatrix}0&\frac{1}{2}&\frac{1}{2}&\frac{1}{3}&\frac{1}{3}&0\\0&0&\frac{1}{2}&0&\frac{1}{3}&\frac{1}{4}\\0&0&0&\frac{1}{3}&\frac{1}{3}&\frac{1}{4}\\0&\frac{1}{2}&0&0&0&\frac{1}{4}\\0&0&0&\frac{1}{3}&0&\frac{1}{4}\\1&0&0&0&0&0\end{pmatrix}$$

and $\alpha=0.85$. Hence we obtained the two column-stochastic matrices

$$\mathbf{G}_\alpha=\alpha\mathbf{H}+\frac{1}{6}(1-\alpha)\mathbf{e}\mathbf{e}^T \text{ and } \mathbf{G}_\alpha(\mathbf{w}^+)=\alpha\mathbf{H}+\frac{1}{6}(1-\alpha)\mathbf{w}^+\mathbf{e}^T.$$

Finally, the rankings in Table 2 were obtained by finding the Perron vector of each of the column-stochastic matrices above.

6.2 Expanded Results for NFL Predictions

In the predictions, the winner is clearly determined by the score in each game, but the actual score may also be used as another statistic for each match. In addition to the usual incidence

matrix $\mathbf{A}^m$ that captures the number of wins of each team, we also consider the weighted score matrix $\mathbf{A}^m_s = (a^s_{ij})$, where if T_i beats T_j, we have a^s_{ij} is the score difference and $a^s_{ji} = 0$. In the case of NFL games, the margin of victory can also be measured in the number of possessions (mostly touchdowns) that team is ahead. Hence, we consider the weighted margin matrix $\mathbf{A}^m_m = (a^m_{ij})$, where $a^m_{ij} = a^s_{ij}/7$. Ranking methods using the score and margin matrix are denoted using a prefix of *s*- and *m*-, respectively. Finally we consider $\mathbf{w}^+$ and $\mathbf{s}^+$ as possible customizations for the PageRank and Oracle variants.

In Table 4, we give an expanded list of models, again with the percentage of games each method picked correctly starting in Week 4 and Week 11. In this table, numbers given in bold are those that outperform the WH method. We note that Oracle variants which do not incorporate the score into either of the up or down vectors do a bit worse than those that do use this statistic, which would lead one to the conclusion that the score difference between teams in NFL games does give some insight into the quality of the teams in that league.

	1966–2013		2002–2013	
Starting Week	Week 4	Week 11	Week 4	Week 11
WH	62.89%	63.94%	62.25%	65.76%
ESPN PR	n/a	n/a	**63.02%**	65.48%
Massey	62.77%	**64.99%**	**63.80%**	**67.41%**
Colley	61.79%	63.15%	61.99%	65.32%
Bradley-Terry Model	62.49%	**64.23%**	**63.58%**	**66.98%**
Or-Bradley-Terry Model	61.71%	63.21%	61.90%	65.50%
Keener	60.08%	62.73%	61.04%	65.07%
Biased Voter ($p = 0.60$)	60.85%	62.44%	60.99%	64.88%
Biased Voter ($p = 0.75$)	60.80%	62.86%	61.17%	64.19%
Biased Voter ($p = 0.90$)	60.53%	63.23%	60.95%	63.57%
Or($\mathbf{e}, \mathbf{e}$)	60.06%	62.02%	59.48%	61.30%
Or($\mathbf{e}, \mathbf{w}^+$)	60.59%	62.24%	60.18%	62.00%
Or($\mathbf{e}, \mathbf{s}^+$)	61.83%	63.04%	61.69%	63.40%
Or($\mathbf{w}^+, \mathbf{e}$)	61.53%	63.02%	61.81%	64.88%
Or($\mathbf{w}^+, \mathbf{w}^+$)	61.63%	63.05%	61.86%	64.88%
Or($\mathbf{w}^+, \mathbf{s}^+$)	**63.41%**	**65.09%**	**64.10%**	**66.63%**
Or($\mathbf{s}^+, \mathbf{e}$)	60.16%	61.27%	58.36%	61.72%
Or($\mathbf{s}^+, \mathbf{w}^+$)	61.47%	62.39%	60.78%	64.36%
Or($\mathbf{s}^+, \mathbf{s}^+$)	**63.20%**	**64.90%**	**64.15%**	**67.06%**
s-Or($\mathbf{e}, \mathbf{e}$)	60.38%	62.94%	60.43%	61.56%
s-Or($\mathbf{e}, \mathbf{w}^+$)	60.52%	62.98%	60.52%	61.91%
s-Or($\mathbf{e}, \mathbf{s}^+$)	60.66%	62.89%	60.69%	62.09%
s-Or($\mathbf{w}^+, \mathbf{e}$)	61.03%	63.01%	61.56%	62.78%
s-Or($\mathbf{w}^+, \mathbf{w}^+$)	61.04%	62.79%	61.25%	62.26%

Continued on next page

Table 4 – *Continued from previous page*

s-Or($\mathbf{w}^+, \mathbf{s}^+$)	61.31%	62.98%	61.90%	63.22%
s-Or($\mathbf{s}^+, \mathbf{e}$)	62.88%	**64.44%**	**64.07%**	**66.46%**
s-Or($\mathbf{s}^+, \mathbf{w}^+$)	62.05%	63.71%	**62.33%**	64.78%
s-Or($\mathbf{s}^+, \mathbf{s}^+$)	**63.41%**	**64.93%**	**63.68%**	65.76%
m-Or($\mathbf{e}, \mathbf{e}$)	60.94%	63.09%	61.21%	63.13%
m-Or($\mathbf{e}, \mathbf{w}^+$)	60.89%	62.70%	60.82%	62.70%
m-Or($\mathbf{e}, \mathbf{s}^+$)	61.67%	63.33%	61.86%	63.75%
m-Or($\mathbf{w}^+, \mathbf{e}$)	61.92%	63.62%	**62.64%**	64.54%
m-Or($\mathbf{w}^+, \mathbf{w}^+$)	61.90%	63.49%	61.77%	63.83%
m-Or($\mathbf{w}^+, \mathbf{s}^+$)	62.76%	**64.35%**	**63.16%**	65.33%
m-Or($\mathbf{s}^+, \mathbf{e}$)	62.26%	63.65%	61.86%	**66.21%**
m-Or($\mathbf{s}^+, \mathbf{w}^+$)	62.06%	63.30%	62.03%	65.57%
m-Or($\mathbf{s}^+, \mathbf{s}^+$)	**63.13%**	**64.74%**	**63.81%**	**66.45%**
PageRank ($\alpha = 0.65$)	59.56%	61.77%	59.40%	61.66%
PageRank ($\alpha = 0.85$)	59.33%	61.44%	58.67%	60.61%
PageRank($\mathbf{w}^+$) ($\alpha = 0.65$)	60.22%	62.33%	60.13%	62.52%
PageRank($\mathbf{w}^+$) ($\alpha = 0.85$)	59.43%	61.50%	59.10%	61.13%
PageRank($\mathbf{s}^+$) ($\alpha = 0.65$)	61.88%	63.65%	61.43%	63.75%
PageRank($\mathbf{s}^+$) ($\alpha = 0.85$)	60.02%	62.15%	59.61%	62.18%
s-PageRank ($\alpha = 0.65$)	60.61%	62.69%	61.30%	62.43%
s-PageRank ($\alpha = 0.85$)	60.23%	62.84%	60.82%	62.43%
s-PageRank($\mathbf{w}^+$) ($\alpha = 0.65$)	60.71%	62.65%	61.04%	62.17%
s-PageRank($\mathbf{w}^+$) ($\alpha = 0.85$)	60.56%	63.03%	60.91%	62.52%
s-PageRank($\mathbf{s}^+$) ($\alpha = 0.65$)	61.49%	63.37%	61.60%	63.39%
s-PageRank($\mathbf{s}^+$) ($\alpha = 0.85$)	60.47%	62.85%	60.91%	62.52%
m-PageRank ($\alpha = 0.65$)	60.69%	63.07%	61.56%	63.57%
m-PageRank ($\alpha = 0.85$)	60.30%	62.78%	61.04%	63.39%
m-PageRank($\mathbf{w}^+$) ($\alpha = 0.65$)	60.94%	63.13%	61.43%	63.66%
m-PageRank($\mathbf{w}^+$) ($\alpha = 0.85$)	60.25%	62.30%	60.61%	62.87%
m-PageRank($\mathbf{s}^+$) ($\alpha = 0.65$)	61.48%	63.42%	61.95%	63.49%
m-PageRank($\mathbf{s}^+$) ($\alpha = 0.85$)	60.73%	62.96%	61.56%	63.83%

Table 4: Overall prediction percentage of all NFL games and average number of correctly predicted NFL games per season, relative to the WH method. In bold, we indicate methods whose predictions, on average, are better than or equal to the WH method.

In Table 5 and Table 6, for each of the eight prediction methods we give the overall prediction percentage of all NFL games in the seasons ranging from 1966–2013 and 2002–2013, respectively, by considering all possible starting weeks between Week 4 and Week 11. More

importantly, when just comparing the predicting power of the Or($\mathbf{w}^+, \mathbf{s}^+$) model in relation to other Markov methods, the results show that Oracle model provides a viable alternative to a Markovian method to rank sport teams.

NFL Seasons between 1966–2013

Starting Week	Week 4	Week 5	Week 6	Week 7
WH	62.89%	62.72%	62.89%	63.00%
Massey	62.77%	**63.19%**	**63.37%**	**63.71%**
Colley	61.79%	61.83%	62.07%	62.07%
PageRank	59.33%	59.46%	59.51%	59.88%
Keener	60.08%	60.49%	60.87%	61.36%
Biased Voter	60.80%	61.13%	61.56%	61.91%
Bradley-Terry Model	62.49%	**62.82%**	**63.40%**	**63.49%**
Oracle(w+,s+)	**63.41%**	**63.26%**	**63.63%**	**63.57%**
Starting Week	Week 8	Week 9	Week 10	Week 11
WH	63.00%	63.42%	63.61%	63.94%
Massey	**63.96%**	**64.04%**	**64.52%**	**64.99%**
Colley	62.13%	62.31%	62.69%	63.15%
PageRank	59.98%	60.36%	60.70%	61.44%
Keener	61.82%	61.94%	62.28%	62.73%
Biased Voter	61.88%	62.01%	62.36%	62.86%
Bradley-Terry Model	**63.50%**	**63.78%**	**64.01%**	**64.23%**
Oracle(w+,s+)	**63.78%**	**64.22%**	**64.70%**	**65.09%**

Table 5: Average of correct foresight prediction percentage of all NFL games starting in Week 4 up to Week 11. In bold, we indicate methods whose predictions, on average, are better than or equal to the WH method.

In Table 7, for each of the eight prediction methods we give the overall prediction percentage of all NFL games in the seasons ranging from 1966 – 2013 and 2002 – 2013, respectively, using the 10-week fixed prediction model described in Section 4. Comparison with Table 3 shows that the results are similar to the usual foresight prediction.

NFL Seasons between 2002–2013

Starting Week	Week 4	Week 5	Week 6	Week 7
WH	62.25%	62.41%	62.67%	63.06%
ESPN PR	**63.02%**	**63.15%**	**63.37%**	**63.66%**
Massey	**63.80%**	**64.31%**	**64.74%**	**65.20%**
Colley	61.99%	62.31%	62.46%	62.73%
PageRank	58.67%	58.87%	58.58%	58.76%
Keener	61.04%	61.58%	61.52%	62.19%
Biased Voter	61.17%	61.71%	61.96%	62.29%
Bradley-Terry Model	**63.58%**	**64.04%**	**64.54%**	**64.99%**
Oracle(w+,s+)	**64.10%**	**64.22%**	**64.34%**	**64.65%**
Starting Week	Week 8	Week 9	Week 10	Week 11
WH	63.61%	63.98%	64.26%	65.76%
ESPN PR	**64.09%**	63.97%	64.02%	65.48%
Massey	**65.96%**	**65.93%**	**66.07%**	**67.41%**
Colley	63.60%	63.77%	63.96%	65.32%
PageRank	59.06%	59.68%	59.64%	60.61%
Keener	63.01%	63.45%	63.50%	65.07%
Biased Voter	62.94%	63.04%	63.20%	64.19%
Bradley-Terry Model	**65.72%**	**65.53%**	**65.55%**	**66.98%**
Oracle(w+,s+)	**65.42%**	**65.26%**	**65.47%**	**66.63%**

Table 6: Average of correct foresight prediction percentage of all NFL games starting in Week 4 up to Week 11. In bold, we indicate methods whose predictions, on average, are better than or equal to the WH method.

Table 8 is the same as Table 3, except that predictions are made for games played from Week 4 or Week 11 through the the final week of each season (rather than the penultimate week). If comparing the results in these two tables, one would see that the predictive power of the Or($\mathbf{w}^+,\mathbf{s}^+$) model improves relative to the WH method, which is not surprising since the WH method is assumed to not do so well at predicting games in the final week of the season.

	1966–2013	2002–2013
Starting Week	Week 11	Week 11
WH	63.77%	65.31%
ESPN	n/a	64.86%
Massey	**64.30%**	**65.93%**
Colley	63.08%	63.57%
PageRank	61.13%	61.48%
Keener	61.69%	64.30%
Biased Voter	62.51%	62.51%
Bradley-Terry Model	**63.97%**	65.14%
Oracle(w+,s+)	**64.72%**	**65.57%**

Table 7: Average of correct 10-week fixed foresight prediction percentage of all NFL games. In bold, we indicate methods whose predictions, on average, are better than or equal to the WH method.

	1966–2013		2002–2013	
Starting Week	Week 4	Week 11	Week 4	Week 11
WH	62.79%	63.61%	61.98%	64.75%
ESPN PR	n/a	n/a	**62.57%**	**64.29%**
Massey	62.95%	**65.13%**	**63.53%**	**66.39%**
Colley	61.77%	63.00%	61.61%	64.15%
PageRank	59.35%	61.08%	58.87%	60.72%
Keener	60.10%	62.34%	61.06%	64.53%
Biased Voter	60.73%	62.45%	60.94%	63.33%
Bradley-Terry Model	62.63%	**64.32%**	**63.41%**	**66.17%**
Oracle(w+,s+)	**63.45%**	**65.05%**	**63.93%**	**65.95%**

Table 8: Average of correct foresight prediction percentage of all NFL games starting either in Week 4 or in Week 11 up to, and including, the final week of the season. In bold, we indicate methods whose predictions, on average, are better than or equal to the WH method.

References

Agresti, A. (2002): *Categorical Data Analysis*, Wiley Series in Probability and Statistics, Wiley-Interscience, 2nd edition.

Agresti, A. and D. B. Hitchcock (2005): "Bayesian inference for categorical data analysis," *Statistical Methods and Applications*, 14, 297–330.

Bradley, R. A. and M. E. Terry (1952): "Rank Analysis of Incomplete Block Designs: I. The Method of Paired Comparisons," *Biometrika*, 39, 324–345.

Brin, S. and L. Page (1998): "The anatomy of a large-scale hypertextual web search engine," *Computer Networks ISDN Systems*, 30, 107–117.

Callaghan, T., P. J. Mucha, and M. A. Porter (2007): "Random walker ranking for NCAA division I-A football," *American Mathematical Monthly*, 114, 761–777.

Colley, W. (2002): "Colley's bias free college football ranking method: the colley matrix explained," Retrieved Jan. 10, 2014 from `http://www.colleyrankings.com/matrate.pdf`.

Constantine, P. G. and D. F. Gleich (2010): "Random alpha PageRank," *Internet Mathematics*, 6, 189–236.

David, H. A. (1963): *The Method of Paired Comparisons*, Charles Griffin & Co.

Easterbrook, G. (2008): "Time to look back on some horrible predictions," Retrieved Jan. 10, 2014 from `sports.espn.go.com/espn/page2/story?page=easterbrook/090210`.

ESPN (2014): "NFL power rankings," Retrieved Jan. 10, 2014 from `http://espn.go.com/nfl/powerrankings`.

Ford, J., L. R. (1957): "Solution of a ranking problem from binary comparisons," *American Mathematical Monthly*, 64, 28–33.

Gleich, D. F. (2011): "Review of: Numerical algorithms for personalized search in self-organizing information networks by Sep Kamvar, Princeton Univ. Press, 2010," *Linear Algebra and its Applications*, 435, 908–909.

Horn, R. A. and C. R. Johnson (1990): *Matrix Analysis*, Cambridge University Press.

Hunter, D. R. (2004): "MM Algorithms for Generalized Bradley-Terry Models," *The Annals of Statistics*, 32, 384–406.

Keener, J. P. (1993): "The Perron-Frobenius Theorem and the Ranking of Football Teams," *SIAM Review*, 35, 80–93.

Langville, A. N. and C. D. Meyer (2004): "Deeper inside PageRank," *Internet Mathematics*, 1, 2004.

Langville, A. N. and C. D. Meyer (2006): *Google's PageRank and Beyond: The Science of Search Engine Rankings*, Princeton, NJ, USA: Princeton University Press.

Langville, A. N. and C. D. Meyer (2012): *Who's #1?: The Science of Rating and Ranking*, Princeton, NJ, USA: Princeton University Press.

Massey, K. (1997): "Statistical models applied to the rating of sports teams," Bachelor's honors thesis, Bluefield College.

Page, L., S. Brin, R. Motwani, and T. Winograd (1999): "The pagerank citation ranking: Bringing order to the web." Technical Report 1999-66, Stanford InfoLab.

Sports Reference, LLC. (2014): "Pro-Football-Reference," Retrieved Jan. 10, 2014 from `http://www.pro-football-reference.com/`.

Thurstone, L. L. (1927): "A law of comparative judgment," *Psychological Review*, 34, 273–286.

Zermelo, E. (1929): "Die berechnung der turnier-ergebnisse als ein maximumproblem der wahrscheinlichkeitsrechnung," *Mathematische Zeitschrift*, 29, 436–460.