Contents

Optimizing Google's PageRank to Rank National Football League Teams

James Venzor

April 30, 2019

Abstract

The PageRank ranking algorithm was originally created by Google in 1998 in order to compare the relative importance of web-pages according to their link structure. In this paper, the PageRank algorithm is generalized to rank National Football League (NFL) teams at the conclusion of each season with data collected from the 2009–2018 seasons. The PageRank rankings were then optimized by tuning the relevant mathematical parameters of the model in order to maximize the association of the PageRank rankings of teams in year N with the win percentages of teams in year $N+1$. It was found that the optimized PageRank rankings offered a significant improvement over the final winning percentages of teams in terms of predicting the results of the subsequent season.

1 PageRank Algorithm

1.1 Google's PageRank

Google developed its PageRank method to distinguish between the importance of web-pages. For Google, when a user searches for a set of words, it is easy to find a myriad of web-pages containing similar (if not identical) words. The difficulty lies in determining which of these matching web-pages is the most relevant to the searcher. In an attempt to distinguish relevant pages from more esoteric ones, Google designed a system of scores called PageRank. The key feature of PageRank is its capability of using the link structure of the web to determine which pages are important. Google made the assumption that a web-page's importance can be related to the number of other web-pages linking to it. In this way, if a fledgling blog devoid of followers contained the words that a user queried, it will not rank of high importance since no other web pages would have linked to it. However, if an impactful and well-known article was published containing the words queried, it is very likely this web-page will be linked in many other pages containing the words queried.

This idea can be translated into mathematics by considering a "hypothetical random web surfer" [1]. When visiting a web-page, the hypothetical surfer randomly decides to either to click on any link present in the current page, or teleport to another page independent of the current page's identity. Web-pages where the random surfer appears more often based on the link structure of the web are then given higher PageRank scores. The astute reader might complain that where the random surfer appears more frequently depends on the probability with which the random surfer decides to either click on a link or randomly teleport, as well as the manner in which the random surfer teleports (is it uniformly distributed over all web-pages, or weighted based on a web-pages visits?). As a result, there arise a natural set of parameters that influence the output of the PageRank algorithm, which will be discussed later in this paper.

For sake of illustration, consider the following over-simplification. Assume that you query "Did Baker Mayfield break the NFL rookie passing touchdown record?" Further assume that, in light of the recent mediocrity of the Cleveland Browns, only 4 web pages contain that search, and are related by the link-structure shown in Figure 1 (where an arrow pointing from a web-page to another indicates the web-page at the tail of the arrow links to the web-page at the head of the arrow).

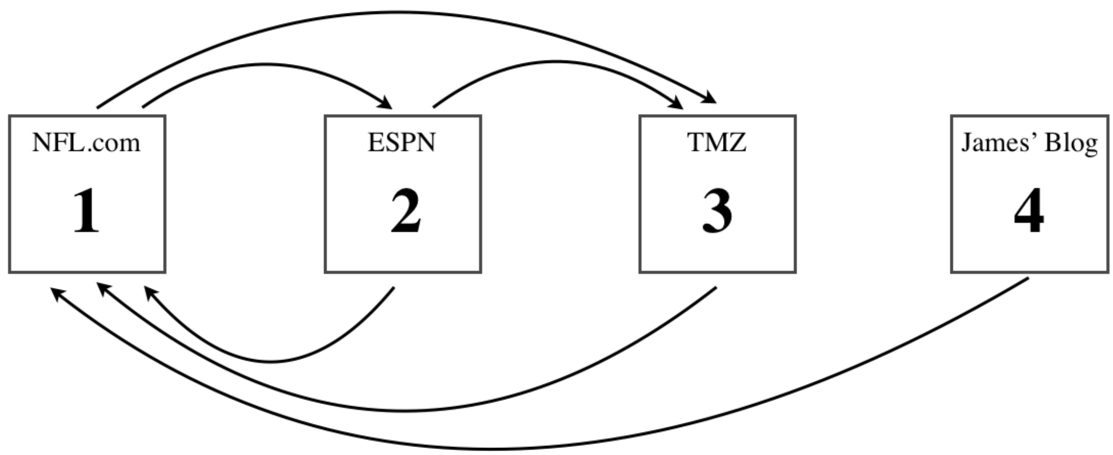

Figure 1: Example of a Simple Web-Structure

To build intuition, first consider the case where the random surfer has no probability of teleporting between web-pages (to get out of a web-page, she must follow a link). The only chance the random surfer has at ever reaching James' Blog is if she began on that web-page, and even then once she leaves it she can never return. As a result, we expect web-page 4 to have the lowest PageRank score. Conversely, since web-page 1 can be reached from any of the other web-pages, we expect it to get the highest PageRank score. It is left to the reader to consider how the rankings of 2 and 3 may vary.

Now consider the case where the random surfer has a positive probability of randomly teleporting. This is great news for James' Blog, as now it is possible for that web-page to be reached by the random surfer at anytime, so long as she is randomly teleporting. So as the probability of teleportation is increased, the ranking associated with James' Blog will go up, while the ranking associated with web-page 1 will go down. While so far only qualitative statements have been made with respect to the rankings, the exact mathematics of how this system can be solved for any probability of teleportation is discussed in the next section, but it is important to first gain some intuition. For further practice with examples, see this collection of examples or go to the next url: `http://www.cs.princeton.edu/~chazelle/courses/BIB/pagerank.htm`

1.2 Generalized PageRank

The mathematics of PageRank can be best understood by generalizing the random surfer idea to a Markov process. Consider a random surfer that, with probability α randomly transitions according to the link structure of the web, and with probability $1 - \alpha$ teleports according to a *teleportation distribution vector* $\mathbf{v}$. A common choice for $\mathbf{v}$ is a uniform distribution over all web-pages. This process can be generalized by replacing the concept of "transitioning according to the link structure of the web" with the stochastic matrix $\mathbf{P}$. Such a mathematical description forms the basis for the applications of PageRank, and will be used extensively in this paper.

We can now give a mathematical description for the PageRank algorithm. Let P_{ij} be the probability of transitioning from "state j" to "state i". Note that using the link-structure shown in Figure 1 as a clarifying example, and assuming the weights of each link to be equal,

$$P_{ij} = \begin{bmatrix} 0 & \frac{1}{2} & 1 & 1 \\ \frac{1}{2} & 0 & 0 & 0 \\ \frac{1}{2} & \frac{1}{2} & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

Additionally, note given the definition of P_{ij}, the sum of each column of P_{ij} must be equal to 1.

Let $\mathbf{v}$ be the *teleportation distribution vector* as before. Hence, if $\mathbf{v}_i = \beta$ for $0 \leq \beta \leq 1$, then the probability of teleporting to"state i" given that a teleportation is occurring is exactly equal to β.

As a result, if there are N distinct states in a system, $\sum_{n=1}^{N} \mathbf{v}_n = 1$.

Let $\mathbf{x}$ be the PageRank vector, where the score for "state i" is equal to $\mathbf{x}_i$. Finally, let $\mathbf{e}$ be a column vector of all ones, with the length of $\mathbf{e}$ being equal to the number of columns (or rows) in P_{ij}. Then the PageRank vector $\mathbf{x}$ is the solution to the eigenvalue problem:

$$\left(\alpha\mathbf{P} + (1-\alpha)\mathbf{v}\mathbf{e}^{\mathrm{T}}\right)\mathbf{x} = \mathbf{x} \tag{1}$$

The PageRank vector $\mathbf{x}$ can be computed using the iteration shown below.

$$\begin{cases} \mathbf{x}^{(0)} = \mathbf{v} \\ \mathbf{x}^{(\mathrm{k}+1)} = \alpha\mathbf{P}\mathbf{x}^{(\mathrm{k})} + (1-\alpha)\mathbf{v} \quad \text{for } k \geq 0 \end{cases} \tag{2}$$

Some nice properties of this iteration technique is that the solution $\mathbf{x}$ is guaranteed to exist and be unique, and that as few as 3656 iterations are necessary in order to converge to a solution with global 1-norm error of $\approx 10^{-16}$ [1]. As a result, we know that we will get a unique solution, and get a solution relatively quickly by solving (1) according to the iteration given in (2). This is favorable computationally.

Returning to our example in Figure 1, for $\alpha = 1$, $\mathbf{x}$ can be calculated to be $\mathbf{x} = \left[\frac{4}{9}, \frac{2}{9}, \frac{3}{9}, 0\right]$ according to (2) with 2000 iterations. With $\alpha = 0.5$, $\mathbf{x}$ can be calculated to be $\mathbf{x} = \left[0.38, 0.22, 0.275, 0.125\right]$ according to (2) with 2000 iterations. Note that these results align with our intuition from before.

1.3 Parameters in the PageRank Model

A basis of this paper is tuning the parameters of the PageRank Model to get optimal results. As such, a fundamental understanding of the parameters in the PageRank model is necessary. The key parameters in the PageRank model that can be tuned are P_{ij}, α, and $\mathbf{v}$. Choosing these correctly is critical in order to successfully model a given situation.

1.3.1 Hyperlink Matrix P_{ij}

The hyperlink matrix P_{ij} is of clear importance. This parameter entirely defines the "link structure" between the nodes of a graph. Some decisions that will personalize P_{ij} for a given application are:

1. *Should the links between nodes be weighted?*

   Placing different weights on the links between states in a graph, as opposed to having all links be of equal weight, will result in a different P_{ij}. For Google, it was straightforward to not weight the links, as there either was or was not a link from one web page to another (there is no concept of a web-page more or less "heavily" linking to another web-page). In many other applications, weighting the links between nodes is appropriate.

2. *How should dangling nodes be handled?*

   A dangling node is a node that does not link to any other nodes. In the example of web-pages, this is a web-page with no out-links present. Dangling nodes often occur in application, and there are various ways of handling them. As noted early, the sum of each column in P_{ij} must equal 1, but if "state j" is a dangling node, then clearly $\sum_{i=1}^{N} P_{ij} = 0$. Hence, P_{ij} must be adjusted to fix this issue. One approach, called *weekly preferential PageRank*, advocates transitioning uniformly from dangling nodes. Another approach, called *strongly preferential PageRank*, advocates transitioning from dangling nodes according to the teleportation vector $\mathbf{v}$. Still another approach, called *sink preferential PageRank*, advocates that a random walk remain at dangling nodes until it moves away in a teleportation step [1]. Which method to chose depends on the application.

3. *Is any data cleaning necessary?*

   In any given data set, many different forms of data cleaning may be applicable. This can encompass removing or adjusting outliers, removing high-leverage points, and altering the metric of interest, among many other possibilities. It is not always obvious what metric to use as the weights for the links in a graph. The choice of this metric is central to the performance of the model.

1.3.2 The α Factor

Recalling our analogy to a random surfer, α is the probability that the random surfer will transition according to the natural hyperlink structure present in P_{ij}, as opposed to the artificial teleportation matrix $\mathbf{v}$. As a result, as $\alpha \to 1$, the PageRank rankings become much more related to P_{ij}, and much less related to $\mathbf{v}$. In words, this means that the PageRank rankings will be much more related to the natural hyperlink structure of the phenomena studied, as opposed to the artificial teleportation probability. This causes the model to have less bias as $\alpha \to 1$, since the rankings better reflect the phenomena studied, but it also increases the variance of the model. This is because as $\alpha \to 1$, slight variations in P_{ij} will lead to larger variations in the rankings of the model [2]. As a result, it can be expected that there is some optimal value of α falling between 0 and 1, and such an optimal value of α will depend on the application.

1.3.3 The Teleportation Vector

The central question surrounding the teleportation vector, $\mathbf{v}$, is whether to make it uniform or non-uniform. While making $\mathbf{v}$ non-uniform can lead to dramatic improvements in the PageRank model's performance, it also dramatically increases the complexity of the model, and makes it much more calculation-laden [2]. The choice of a non-uniform $\mathbf{v}$ is dependent on the specific application.

2 Application to NFL Rankings

2.1 Purpose of NFL Rankings

The ranking of sports teams is a very common and familiar occurrence to any sports fan. Sports writers frequently publish their subjective rankings of teams. Additionally, there are many algorithm driven models to rank teams. The NFL is no exception to these trends. While such "power rankings" of NFL teams are often published throughout the season, a major point of contention is determining the rank of teams at the end of the season. The NFL itself determines the draft order according the their own system for ranking teams, which is based mainly on winning percentage.

Beyond simply ranking teams, a major concern of NFL fans (and bettors) is to predict how a team will fare in an upcoming season. Las Vegas casinos publish over-under odds for teams' wins in a season, and it would be of monetary value to have an algorithm that could correctly predict a teams wins total at the start of a season. The starting point for an over-under prediction is how a given team fared in the previous season (i.e, the teams winning percentage in a previous year). Other factors considered include off-season talent acquisition through the draft and free-agency, and strength of schedule in the upcoming season, in addition to a myriad of other factors.

The aim of this paper is to use the PageRank algorithm to assign scores to teams in year N that are indicative of success (or failure) in year $N + 1$. Furthermore, the goal is to have the PageRank score assigned to a team in year N be a better predictor of that team's winning percentage in year $N + 1$ than the team's winning percentage in year N. As a result, the determined PageRank scores could be used as a simple and efficient predictor of teams success in year $N + 1$ that outperforms making wagers based on a teams winning percentage in year N alone.

It should be noted that there are many factors limiting the extent to which the PageRank method can predict wins in the subsequent year. On the one hand, there are so many variables that are not accounted for using the PageRank model, since it only takes into account performance on the field in year N. For example, it does not take into account key injuries or off-season acquisitions. On the other hand, football is so complicated that there will always be a significant amount of intrinsic error. At the least, in more complicated algorithms for predicting teams' win totals in year $N + 1$, the PageRank scores computed for teams in year N could replace teams' winning percentages in year N in the predictor set to obtain more accurate results. This would alleviate concerns that the PageRank model leaves out many key variables in the prediction of a given team's wins.

2.2 NFL PageRank Formulation

A natural network construction for sports is termed the *winner network* [1]. Each team can be represented by a node in the network, and node i points to node j if j outperformed i in the quantity of interest with a weight given by the amount j outperformed i by. Traditionally, the quantity of interest is the score in a game, however there are many different quantities that can be used, as discussed later in this section.

The application of PageRank to NFL rankings can be thought of in light of a "random fan" as opposed to our "random surfer" from before. In this case, a fan starts with a given team, and either transitions to a team that defeated her current team of fandom, or randomly picks a new team (teleports to a new team).

The principal advantage of PageRank to winning percentage is that PageRank takes into account **strength of schedule**. Since PageRank takes into account the link structure of a graph, if a team beats a stronger team it benefits more than if it beats a weaker team. Conversely, if a team loses to a stronger team, it is penalized less than if it loses to a weaker team.

Govan et al. used the PageRank algorithm with uniform teleportation and $\alpha = 0.85$ to rank football teams with the aforementioned winner network (where the quantity of interest was the score in the game) [3]. For this paper, score, yardage, and first down statistics from each game from the 2009-2018 seasons were pulled from NFL play-by-play data [4]. This paper aims to extend upon the work by Govan et al. by exploring *tuning the various parameters* present in the PageRank algorithm. An explanation is now given as to how this parameter tuning was accomplished, mirroring the structure of subsection 1.3.

2.2.1 Hyperlink Matrix P_{ij}

The key choice that influences P_{ij} for producing NFL rankings is which quantity of interest to use. Separate models using **score, yardage, and first down differential** were built to determine which quantity produced the best ranking.

1. *Should the links between nodes be weighted?*

   To investigate this question, distinct models were built with both weighted and unweighted links between nodes for each quantity of interest tested, and their results were compared.

2. *How should dangling nodes be handled?*

   A dangling node represents a team that did not record a loss. As a result, when adjusting this node in the P_{ij} matrix, the undefeated team should retain an advantage. For this reason, the *sink preferential PageRank* model, discussed in section 1.3.1, was used for this analysis.

3. *Is any further data cleaning necessary?*

   It was first necessary to handle inter-division games, since these occur twice per year. The methodology used in this paper was to average the results of both games, and treat the averaged result as a single game. While there are other potential ways to handle this, such alternatives were not explored.

   Various additional data cleaning methods were investigated. These include:

   - Adjusting outlier results to the 10th and 90th percentile accordingly
   - Adjusting for home field advantage
   - Computing statistics on a per-drive basis
   - Eliminating the first 4 weeks of the season

Of these adjustments, only adjusting for home field advantage was found to be beneficial. For example, over the 10 years of data analyzed, home teams won by an average of 2.5 points per game. As a result, when computing the score differential between two teams, the away team's score was given an additional 2.5 points to account for this. The benefit of this in terms of the model's predictive power is displayed in Figure 11 located in Appendix C (the specifics of the plotting axes are discussed in detail later in this paper).

2.2.2 The α Factor

To account for the effect of α, PageRank models for each distinct pair of P_{ij} and $\mathbf{v}$ tested were constructed at varying values of α with $0.05 \leq \alpha \leq 0.95$ in general. Most plots in this paper will show the predictive value of the model vs. α (the metric used for predictive value is discussed in the section 3).

2.2.3 The Teleportation Vector

Both uniform and non-uniform teleportation vectors $\mathbf{v}$ were tested. The non-uniform teleportation vector used was the winning percentages of teams at the end of year N. This can be reasoned from the perspective of a "bandwagon fan" as opposed to our "random fan" from earlier. Here, when the fan teleports to a new team, she chooses her new team proportionally to each team's winning percentage.

3 Comparison Measure for Rankings

3.1 Adjustment of the Kendall Correlation

The aim of this paper is to determine if the PageRank model can produce a ranking system for teams after year N that, in comparison with team's winning percentages in year N, better predicts teams' winning percentages in year $N+1$. In order to determine if this is acheived, it is first necessary to define what it means for one set of rankings to be more closely related to each other than another set of rankings.

One common statistic used to measure the ordinal association between two measured quantities is the Kendall rank correlation coefficient [5]. The basis for the Kendall rank correlation coefficient is as follows: Let $(x_1, y_1), (x_2, y_2), ..., (x_n, y_n)$ be a set of of observations of joint random variables X and Y, such that all of the x_i and y_i are unique (that is, $x_i = x_j \Rightarrow i = j$). Then any pair of observations (x_i, y_i) and (x_j, y_j) with $i < j$, are said to be concordant if $x_i > x_j$ and $y_i > y_j$ or $x_i < x_j$ and $y_i < y_j$. In words, this means that in both ranking systems, the relation of the ith state to the jth state is the same. Any such pair is said to be discordant if $x_i > x_j$ and $y_i < y_j$ or $x_i < x_j$ and $y_i > y_j$. The Kendall rank correlation coefficient τ is defined as:

$$\tau = \frac{(\text{number of concordant pairs}) - (\text{number of discordant pairs})}{n(n-1)/2} \tag{3}$$

where the denominator is simply $\binom{n}{2}$, the number of ways to choose a unique unordered pair from n states.

As an example, consider the data displayed below in Table 1, taken from 2017 and 2018 NFL seasons. It is clear that according to our above definition, Kansas City (KC) and Philadelphia (PHI)

Table 1: Illustration of Concordance and Discordance

Team	2017 Win %	2018 Win %
KC	0.625	0.75
PHI	0.8125	0.5625
ATL	0.625	0.4375

form a discordant pair, because in 2017 PHI was better than KC, but in 2018 KC was better than PHI. Conversely, we see that Atlanta (ATL) and PHI form a concordant pair, as PHI was better than ATL in both years.

What about KC and ATL? Here we see that the uniqueness condition preceding (3) is violated, since both KC and ATL had the same winning percentage in 2017. This is quite common, and as such, the Kendall rank correlation coefficient is not an applicable metric to ranking NFL teams. However, a slight adjustment can be made to allow a similar ranking coefficient. Consider the case where $x_i = x_j$ for $i \neq j$. Then any pair of observations (x_i, y_i) and (x_j, y_j) is now said to be concordant if $y_i = y_j$. Now, we define a new metric $\bar{\tau}$ such that:

$$\bar{\tau} = \frac{\text{(number of concordant pairs)}}{n(n-1)/2} \tag{4}$$

where the number of discordant pairs is dropped from the definition. Note that $\tau \in [-1, 1]$, while $\bar{\tau} \in [0, 1]$. As a result, $\bar{\tau}$ has the advantage of being interpreted as the percentage of concordant pairs present between two ranking systems.

The chief aim of this paper can now be rehashed as determining how to tune the parameters of the PageRank problem in order to maximize $\bar{\tau}$ for the PageRank rankings in year N with the win percentage rankings in year $N+1$ for the data from the 2009–2018 NFL seasons.

3.2 Computing $\bar{\tau}$ for the NFL Data

The data collected spans from 2009–2018. As a result, it is possible to compute $\bar{\tau}$ between 9 different sets consisting of 2 consecutive years of data. This methodology is detailed below (for fixed α, P_{ij}, and $\mathbf{v}$):

1. Compute the $\hat{\tau}_i$

$$\begin{aligned}
\hat{\tau}_1 &= \bar{\tau} \text{ for 2009 PageRank ranking and 2010 winning percentage} \\
\hat{\tau}_2 &= \bar{\tau} \text{ for 2010 PageRank ranking and 2011 winning percentage} \\
&\vdots \\
\hat{\tau}_9 &= \bar{\tau} \text{ for 2017 PageRank ranking and 2018 winning percentage}
\end{aligned}$$

2. Compute $\bar{\tau}$ Summary Statistics

$$\hat{\mathbb{E}}(\bar{\tau}) = \frac{1}{9}\sum_{i=1}^{9} \hat{\tau}_i \tag{5}$$

$$\hat{\sigma}_{\bar{\tau}} = \frac{\sqrt{\frac{1}{8}\sum_{i=1}^{9}\left(\hat{\tau}_i - \hat{\mathbb{E}}(\bar{\tau})\right)^2}}{3} \tag{6}$$

4 NFL PageRank

4.1 Preliminary Results

As described in subsection 2.2, various P_{ij} matrices were constructed depending on the quantity of interest used. These quantities of interest were game score differential, yardage differential, and first down differential. Additionally, for each choice of P_{ij}, both a weighted and unweighted node structure, in addition to both a uniform and non-uniform $\mathbf{v}$, were tested.

The success of these models ($\bar{\tau}$ for the PageRank rankings with the following year's winning percentages) is plotted vs. α for each of the different quantities of interest in Figures 2–4 on the following page. In the figures, a blue line indicates the use of a uniform $\mathbf{v}$ while a red line indicates the use of a non-uniform $\mathbf{v}$. Additionally, solid lines indicate weighted links, whereas dashed lines indicate unweighted links. The baseline $\bar{\tau}$ value for winning percentage in successive years is plotted in a black dashed line.

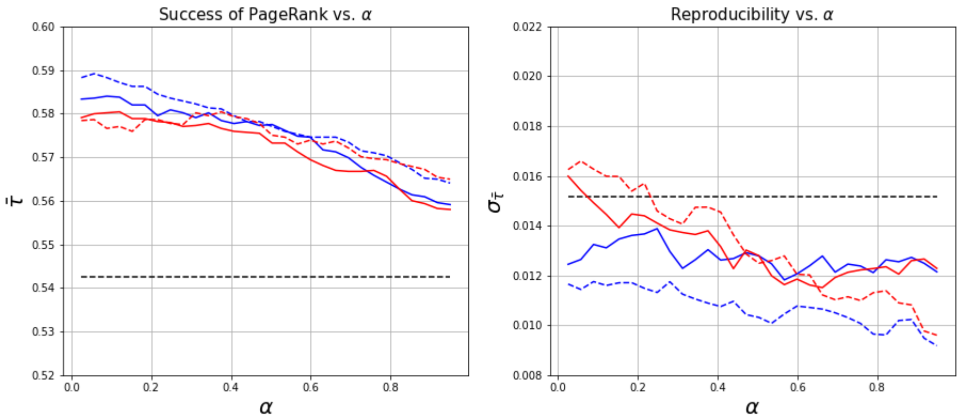

Figure 2: **Score Differential**

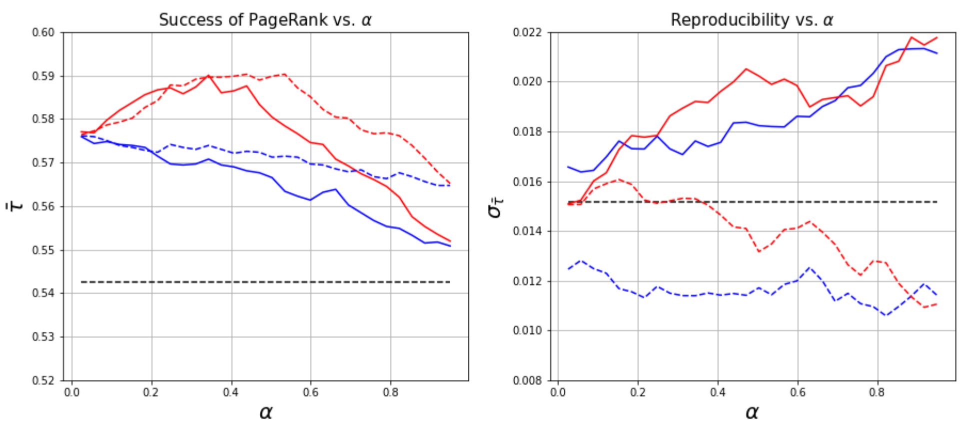

Figure 3: **Yardage Differential**

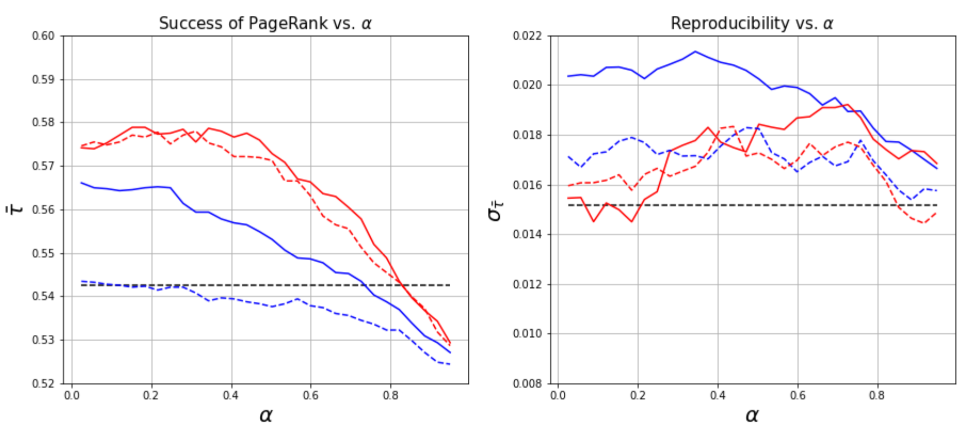

Figure 4: **First Down Differential**

Over the 9 seasons tested between 2009 and 2018, $\bar{\tau}$ for winning percentage in year N with winning percentage in year $N+1$ was $\bar{\tau} \approx 0.5425$[1]. The standard error for the estimate of $\bar{\tau}$, computed by (6), for a given α is plotted on the right halves of Figures 2–4.

Looking at the comparative performance of the three quantities of interest, it is simple to rule out first down differential, on the basis of it having the lowest values of $\bar{\tau}$ in addition to the least reproducibility of the three metrics. An interesting comparison can be made between yard and score differential. The author recommends that yard differential is a better metric to use than score differential, on the basis that score differential attains it maximal $\bar{\tau}$ as $\alpha \to 0$. As a result, the more that the nature of the link structure between teams is considered, the less accurate the model. This signals that score differential is not the correct metric to use.

On the other hand, for the PageRank model with yards, $\bar{\tau}$ reaches a maximum at $\alpha \approx 0.5$, for the model with non-uniform $\mathbf{v}$ and non-weighted links. For these parameters, $\bar{\tau} \approx \mathbf{0.59}$, signaling a *5% improvement* over $\bar{\tau}$ for winning percentage in year N with winning percentage in year $N+1$. Additionally, note that at $\alpha \approx 0.5$, the PageRank model has greater reproducibility than the model with winning percentage in year N. Thus, the PageRank model is not only more accurate, but also more precise than winning percentage with respect to $\bar{\tau}$. In summary, the recommended model displayed below in Table 2.

Table 2: PageRank Model Optimal Parameters

Parameter	Method
Quantity of Interest	Yardage Differential, Unweighted
α	0.5
$\mathbf{v}$	non-uniform

While it is evident that the PageRank model is an improved ranking system over simply ranking teams based off of winning percentage, there are some shortcomings to the model which are now considered.

4.2 Variance in PageRank Model

A major source of variance in the model is the sensitivity of a team's score to one individual game during the season. This is particularly true in the instance where a team defeated one of the better teams in the NFL by a wide margin during the course of the season. While such a victory deserves to be rewarded with an increase in ranking, often this increase is too dramatic a swing.

For illustration, consider the 2017 NFL season. In week 1 of the season, the Kansas City Chiefs defeated the New England Patriots 42-27. To determine the affect that this game had on Kansas City's score, it is reasonable to look at what Kansas City's score *would have been* had they never have played New England. One way to do this is to construct the same P_{ij} matrix as before, except exclude the row and column associated with New England[2]. Taking this a step further, it is possible to look at how Kansas City's score would be impacted by the removal of any given team. This is displayed at the top of the next page Figure 5.

By inspection of of Figure 5, we see that when New England is included in the model, Kansas City's PageRank score remains relatively constant. However, upon their removal, Kansas City's score drops to about 65% of its original value. The result of this behavior is that Kansas City's PageRank scores is dictated almost exclusively by their game against New England. This introduces variance to the model, since a slight variation in that one single game (for example, if Kansas City had only won by 5 instead of 15, or lost instead of won) would lead to an amplified change in Kansas City's PageRank score.

[1] In some sense this is a remarkable result — if "Team A" finished with a better record than "Team B" in the previous year, only *slightly* over half the time will they finish better again in the following year! Note that this percentage is likely to be higher than the reported value for $\bar{\tau}$, since $\bar{\tau}$ also takes into account ties as discussed in section 3 of this paper.

[2] When doing this, dangling nodes must again be accounted for.

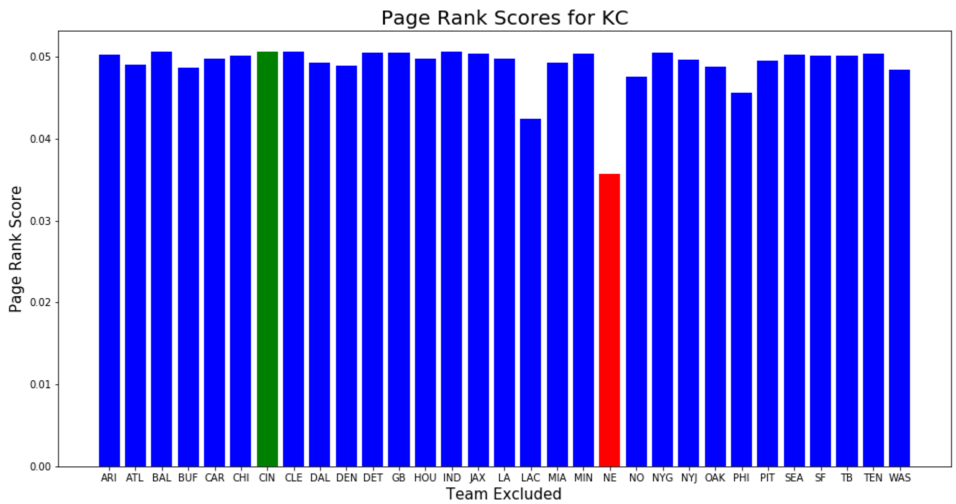

Figure 5: **KC's PageRank Score Dependency on NE**

4.3 Bias in PageRank Model

As mentioned before, the main source of bias in the PageRank model is that it excludes factors such as key injuries and offseason acquisitions. Its realm of information is limited to adjusting for strength of schedule, but it is expected to get this almost exactly correct. As a result, any bias surrounding PageRank's estimation of strength of schedule is significant. These are now discussed.

One source of bias is that the PageRank model does not have any time dependency. A game that occurred in week 1 is treated equivalently to a game that occurred in week 17. This introduces error, because a team's ranking (or abilities) are not constant over the course of a season. In many cases, a team that was easy to play at the beginning of the season becomes a very difficult team to play at the end of the season (or vice-versa).

A second source of error in the PageRank model is that it cannot take into account when a team is resting its players. If a team has secured their playoff position, it is commonplace for that team to rest its players in week 17. Hence, if a team beats another team that is resting its players, it really gives no information as to the comparative strengths of those teams. The PageRank model does not account for this. As an example, Figure 6 shows Carolina's PageRank score as a function of the week of the season[3].

In week 17 of the 2018 NFL season, Carolina played the New Orleans Saints. In this game, the exact scenario laid out above is observed — the New Orlean's saints were resting all of their players in anticipation of the playoffs. Figure 6 shows the dramatic increase in the PageRank score for Carolina, which was an entirely undeserved boost. Additionally, since New Orleans was one of the strongest teams in the NFL, the variance issue raised in the previous section exacerbates the change in Carolina's ranking! Such a scenario is not limited to Carolina in 2018, and as a result this phenomena introduces bias into the PageRank model.

[3] Seperate P_{ij} matrices were constructed including only data from games through the week shown on the x-axis of the barplot in **Figure 6**. For each P_{ij} constructed, a PageRank model according to the optimally determined parameters was implemented to compute Carolina's PageRank score.

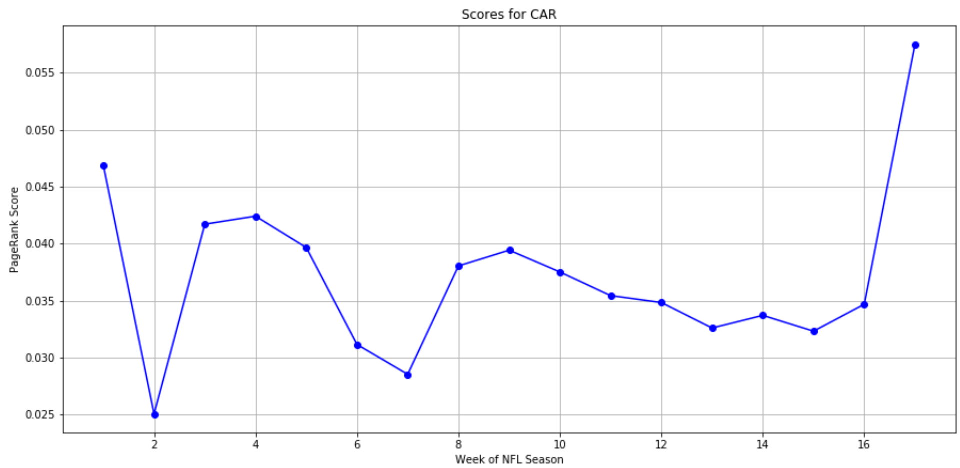

Figure 6: **Carolina's PageRank Score Computed After Each Week of the Season**

5 Modifications to PageRank

5.1 Accounting for Variance

In order to account for the variance issue discussed in subsection 4.2, a modification to PageRank was explored. This modification was implemented according to the following algorithm.

ALGORITHM 1

1. Determine P_{ij}, α, and $\mathbf{v}$ for a fixed quantity of interest.

2. Compute $\mathbf{x}$ according to (1) with these parameters according to the iteration given in (2).

3. Pick a number of teams, N, to decide to remove from the model on each iteration. Let S be the set of all possible unordered groups containing N teams[a].

4. On the kth iteration of this step, adjust the original data set to exclude the N teams contained in the kth element of S. Recompute P_{ij} and $\mathbf{v}$ from this adjusted data set (α remains the same each time). Then compute $\hat{\mathbf{x}}_k$ according to the iteration given in (2), with $(\hat{\mathbf{x}}_k)_j = 1$[b], for $j \in S_k$.

5. Compute $\mathbf{x}^*_\beta = (1-\beta)\mathbf{x} + \beta\big(min\{\hat{\mathbf{x}}_1, \hat{\mathbf{x}}_2, ..., \hat{\mathbf{x}}_{\binom{32}{N}}\}\big)$ for $0 \leq \beta \leq 1$ for various values of β[c], where the minimum operator is defined to act element-wise.

6. Define $\bar{\tau}_\beta \equiv \bar{\tau}$ for $\mathbf{x}^*_\beta$ and the winning percentages of teams in the following year. Then for the value $\beta = \beta^*$ that yields the largest $\bar{\tau}_\beta$ out of all of the values of β tested, assign scores to the 32 teams according to $\mathbf{x}^*_{\beta^*}$ normalized to have a 1-norm equal to 1.

[a] Note that $|S| = \binom{32}{N}$, since there are 32 teams in the NFL.

[b] Here $(\hat{\mathbf{x}}_k)_j$ is set to 1 so that it would not impact the computation of $\mathbf{x}^*_\beta$ in the subsequent step. In this sense, we could have $(\hat{\mathbf{x}}_k)_j = c$ for some $c \geq 1$, and it would make no difference to the output of the algorithm.

[c] How many values of β to test depends on computational feasibility, the author sampled 30 values of β between 0 and 1.

So what does `ALGORITHM 1` mean in words? Steps 1 and 2 simply carry out the same PageRank algorithm as discussed in section 4. To gain a better understanding of the rest of the algorithm, we will first interpret it with the parameter $N = 1$. As stated in step 3, this entails that we will be excluding one team from the analysis upon each iteration.

Notice that this is exactly what was done to construct Figure 5, where we were looking at Kansas City's scores as a function of the team excluded. Step 4 entails excluding one team at a time, and computing scores for the other 31 remaining teams (and setting the score of the one team excluded equal to 1). Then, for each team, we find the *smallest* score it was assigned over all iterations (this is precisely why when a team was excluded it was assigned a score of 1 — a score of 1 cannot possibly be the minimum of the collection of a team's scores unless it received a perfect score of 1 each time!). For Kansas City in Figure 5, this minimum score was when NE was removed, and was found to be ≈ 0.034.

In step 5, all we are doing is taking a weighted average of a team's initial PageRank score, and the worst score they received when removing teams. In this way, we are limiting the variance of the model by accounting for the team's worst score. We do this computation for a host of β values, and find that β which enables the model to have the most predictive value (step 6 is just the mathematical way of sating exactly this).

What was found empirically was that this optimal value of β was $\beta = 1$. This entails that the model operates optimally if each team gets assigned the lowest score it received upon the removal of other teams (so in our example, Kansas City would be assigned a score of ≈ 0.034). In Figure 7 below, this behavior is shown. The fact that $\beta = 1$ is the optimal value of β shows that accounting for high-leverage games lowers the variance of the model enough to improve its results.

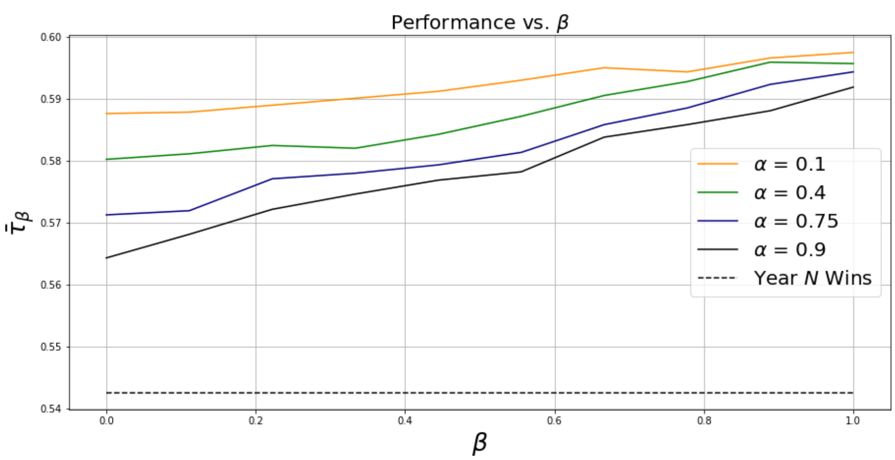

Figure 7: **Effect of β on Predictiveness of Model for Various Values of α**

This analysis can be expanded to include higher values for N. Notice that just increasing N from 1 to 2 requires increasing the number of groups analyzed from 32 to 496! As a result, it is only computationally feasible to test $N = 2$ in addition to $N = 1$. The optimal parameters for this model are displayed below in Table 3.

Table 3: `ALGORITHM 1` Optimal Parameters

Parameter	Method
Quantity of Interest	Score Differential, Unweighted
α	0.4
$\mathbf{v}$	uniform
N	2
β	1

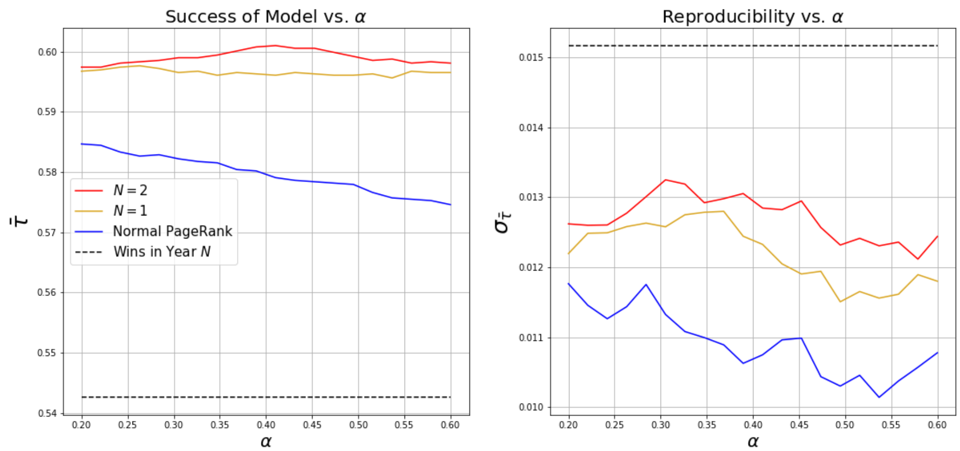

Figure 8: **Effect of N on success of PageRank Model**

Using this setting, as can be seen in Figure 8 above, a value of $\bar{\tau} \approx \mathbf{0.601}$ was obtained, an improvement over the PageRank model alone.

5.2 Accounting for Bias

As discussed in subsection 4.2, one source of bias present in the PageRank model is that it does not take into account teams resting in the final week of the season (week 17) One straightforward way to evaluate the extent to which this factor is introducing bias into the PageRank model is by implementing a PageRank model omitting outcomes from week 17. That is, the P_{ij} matrix will only include results from weeks 1–16. Using game score differential, unweighted nodes, and a uniform $\mathbf{v}$, the results shown in Figure 9 below were observed.

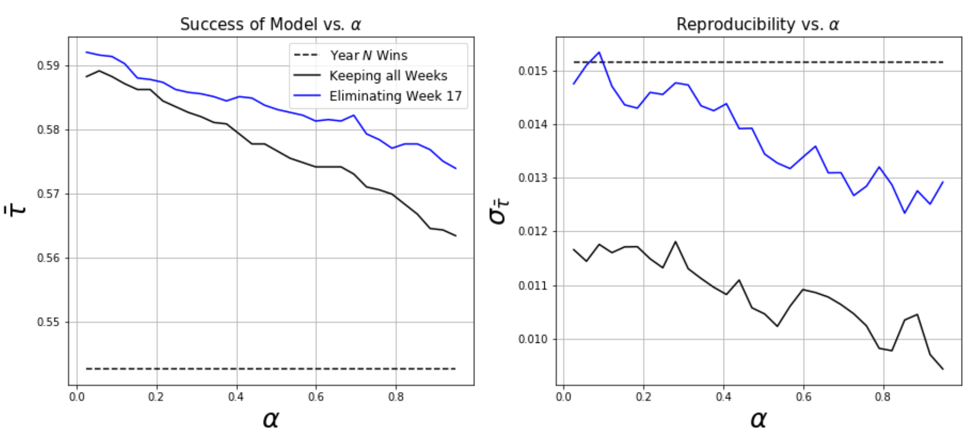

Figure 9: **Effect of Removing the Last Week of Games**

Evidently, the PageRank model is better associated with following year's winning percentages when the last week of the season is excluded from the analysis! It should also be noted that the standard error increases as well. Nevertheless, it is evident from Figure 9 that there is value in eliminating week 17 games from the analysis.

The second issue with the PageRank model discussed in subsection 4.2 that leads to bias was the lack of any time dependency. Given that teams change over the course of the season, it would be desirable if the PageRank model could be adjusted to include a temporal component. This imbues a guiding question:

Is there a way to modify the PageRank algorithm to account for the week of the season a game was played?

One way to take this temporal component of a season into account would be to split the season up into a certain number of blocks. The PageRank method can initially be applied to the first block, starting with a uniform teleportation vector. Then, we can *update the teleportation vector* to be equal to the solution to the Page Rank problem on the first block. From there, we can solve the next block, and continue this process. The solution to the PageRank problem after repeating the process for the final block of games would then be the scores to assign to each team.

In this construction, the parameter α is critical. For $\alpha = 1$, this procedure devolves into only being affected by the last block analyzed, and as such would be a quite worthless algorithm. However, for small α, it allows $\mathbf{v}$ to be *slowly* updated, taking into account when games were played, and teams' strengths at the time of each game played. In this sense, this construction would enable the PageRank model to slowly learn as it moves from one block to the next. The formal algorithm that implements this construction is given below.

ALGORITHM 2

1. Split the first 16[a] weeks of the season into N, preferably evenly spaced, blocks. Let these N blocks be denoted B_1,B_2,...B_N.
2. Let P_{ij}^k be the P_{ij} matrix constructed from the data exclusively from the weeks in B_k.
3. Set $\mathbf{v}^{(0)}$ equal to the uniform teleportation vector, and fix a value for α.
4. Then on step k, set $\mathbf{x}^{(k)}$ to be the solution[b] to the usual PageRank problem with parameters P_{ij}^k, $\mathbf{v}^{(k-1)}$, and α. Then set $\mathbf{v}^{(k)} = \mathbf{x}^{(k)}$.
5. Begin the above iteration at $k = 1$ and end it after $k = N$.
6. Assign scores to the NFL teams according to $\mathbf{v}^{(N)}$ or $\mathbf{x}^{(N)}$ (they are equivalent).

[a] Here, we choose 16 as opposed to 17 weeks because of the findings at the beginning of this section.
[b] This solution can be obtained according to the iteration in (2).

The results of ALGORITHM 2 applied to the 2009–2018 NFL data are displayed in Figure 10 at the top of the next page. This was accomplished using score differential as the quantity of interest, and using *weighted* links. As can be seen in Figure 10, a value of $\bar{\tau} \approx \mathbf{0.61}$ was obtained with $\alpha = 0.06$. Such a small value of α was expected, as it enables slow learning of the model as it moves from each quarter of the season to the next quarter.

A summary of the optimal parameters to use with ALGORITHM 2 is given in Table 4 below.

Table 4: ALGORITHM 2 Optimal Parameters

Parameter	Method
Quantity of Interest	Score Differential, Weighted
α	0.06
Blocks	Each Quarter of the First 16 Weeks[4]

[4] Here, "each quarter of the season" is another way of saying $B_1 = \{\text{week 1, week 2, week 3, week 4}\}, ...,$ $B_4 = \{\text{week 13, week 14, week 15, week 16}\}$.

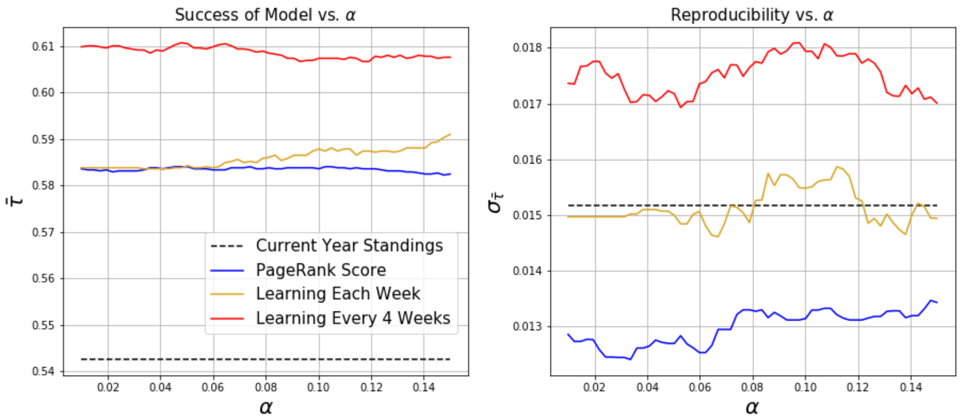

Figure 10: **Effect of B_1,B_2,...B_N on success of PageRank Model**

6 Conclusion

We have shown that the normal PageRank model can be improved for the application to ranking NFL teams by accounting for sources of bias and variance. Additionally, it is clear that these PageRank rankings, in comparison with teams' winning percentages, associate better with teams' subsequent year winning percentages. These improvements are summarized below in Table 5, where the reported quantities are for the ranking outputted by the reported model in year N with teams' winning percentages in year $N+1$. The **optimal** and <u>worst</u> values obtained for each quantity are bolded and underlined, respectively.

Table 5: Comparison of Methods Discussed

Model	$\bar{\tau}$	$\sigma_{\bar{\tau}}$
Year N Winning Percentages	<u>0.542</u>	0.015
PageRank	0.590	0.013
ALGORITHM 1	0.601	**0.013**
ALGORITHM 2	**0.610**	<u>0.017</u>

The results listed in Table 5 were obtained using the optimal parameters determined for each of the models given in Tables 2–4. The cross-validated results are given in Appendix D, and are comparable to the results listed in Table 5 above. Using ALGORITHM 2, it is clear that we were able to achieve just shy of a **7%** improvement with respect to $\bar{\tau}$ over the baseline method of using teams' winning percentages after a season. This can be stated in words as:

If Team A is ranked higher than Team B according to the output of ALGORITHM 2*, and Team C is ranked higher than Team D according to their respective winning percentages, then at the conclusion of the subsequent season, it is approximately* 7% *more likely that Team A will have a higher winning percentage than team B, than it is for Team C to have a higher winning percentage than Team D.*

This comes with the added cost of increasing the variance of the output of the model by a slight margin. However, if one wants both more accurate and precise results, the author recommends the implementation of ALGORITHM 1. The results of these algorithms applied to the previous NFL season (the 2018 NFL season) is given in Table 6 located in Appendix A. Additionally, in Table 7 of Appendix B, the performance of all three models are compared for the 2017 NFL season.

7 Topics for Further Works

There are many more applications of the PageRank algorithm to the NFL that could be explored. A non-exhaustive list is given below.

- Can the parameters of PageRank be optimized to predict the results of NFL games *during the season*? After how many weeks does the model become sufficiently predictive?
- Can the outputted scores be integrated in a more complicated regression analysis to predict NFL teams' winning percentages prior to a season? Is it beneficial for these scores to replace previous year's winning percentages in the predictor set of the more complicated model?
- Can the PageRank model be used to objectively determine a team's strength of schedule? Could this be valuable for use by the NFL in tiebreaker situations? How does it compare to common strength of schedule metrics that are publicly available?
- Can `ALGORITHM 1` and `ALGORITHM 2` be combined into a more powerful algorithm?
- There should be some sense that a larger difference in PageRank scores between teams is more indicative of future winning percentages than smaller differences in PageRank scores. Is there some threshold value ϵ such that if the difference in PageRank scores of two teams is greater than ϵ, the PageRank model performs better compared to win percentages than the PageRank models that use no such threshold?
- Is there any relation between the trend of a team's PageRank score over the course of the season with their subsequent year's winning percentages? If a team's PageRank score steadily increases over the second half of the season, does that give evidence that the team is improving in actuality, and cause them to do better than expected the next year? If a team tails off in PageRank score over the latter half of the season, is it indicative of decline?
- Is there another (better?) natural way to deal with dangling nodes present in the data set? How do the results compare with using *sink preferential PageRank*, as was done in this paper?

References

[1] David F Gleich. Pagerank beyond the web. *SIAM Review*, 57(3):321–363, 2015.

[2] Amy N Langville and Carl D Meyer. *Google's PageRank and beyond: The science of search engine rankings*. Princeton University Press, 2011.

[3] Anjela Y Govan, Carl D Meyer, and Russell Albright. Generalizing google's pagerank to rank national football league teams. In *Proceedings of the SAS Global Forum*, volume 2008, 2008.

[4] nflscrapr-data. https://github.com/ryurko/nflscrapR-data/tree/master/play_by_play_data/regular_season. Accessed: 2018-12-30.

[5] Maurice George Kendall. Rank correlation methods. 1948.

Appendices

A 2018 NFL Season Predictions

Table 6: 2018 NFL Predictions

Team	ALGORITHM 2	ALGORITHM 1	PageRank	Winning %
NO	1	1	3	1.5
CHI	2	4	1	4
KC	3	3	5	4
NE	4	2	6	6.5
TEN	5	6	11	13.5
PIT	6	13	9	12
HOU	7	5	14	6.5
IND	8	12	10	9.5
LA	9	11	2	1.5
SEA	10	7	15	9.5
BAL	11	8	4	9.5
LAC	12	9	8	4
DEN	13	27	25	23.5
DAL	14	14	12	9.5
MIN	15	16	17	15
ATL	16	25	23	18.5
GB	17	19	22	21
CAR	18	10	7	18.5
TB	19	21	13	27
WAS	20	24	19	18.5
PHI	21	17	16	13.5
CIN	22	23	26	23.5
JAX	23	22	27	27
DET	24	18	28	23.5
NYG	25	20	18	27
NYJ	26	29	31	30
MIA	27	28	29	18.5
BUF	28	26	20	23.5
CLE	29	15	21	16
OAK	30	31	30	30
ARI	31	32	32	32
SF	32	30	24	30

Each model used in Table 6 above was run with the parameters displayed in Tables 2–4. Teams are colored by the relationship between the algorithm's ranking and the winning percentage ranking of the team.

B How the Different Models Fared in 2017

Table 7: 2017 NFL Season Model Performance

Team	2018 Win %	2017 Win %	ALGORITHM 2	ALGORITHM 1	PageRank
NO	1.5	6	3	4	7
LA	1.5	6	6	7	10
CHI	4	26.5	19	21	8
KC	4	9	7	8	11
LAC	4	14	23	11	16
NE	6.5	2.5	2	1	4
HOU	6.5	29.5	20	24	25
IND	9.5	29.5	30	27	31
SEA	9.5	14	11	17	15
BAL	9.5	14	10	12	21
DAL	9.5	14	13	15	13
PIT	12	2.5	8	5	1
TEN	13.5	14	12	13	19
PHI	13.5	2.5	4	2	3
MIN	15	2.5	1	3	2
CLE	16	32	32	32	32
ATL	18.5	9	14	9	6
MIA	18.5	23	16	25	29
CAR	18.5	6	5	6	9
WAS	18.5	20	18	18	23
GB	21	20	21	20	17
DEN	23.5	26.5	27	30	19
CIN	23.5	20	24	28	28
DET	23.5	14	17	22	5
BUF	23.5	14	15	16	24
TB	27	26.5	29	29	30
JAX	27	9	9	10	12
NYG	27	31	31	31	22
NYJ	30	26.5	26	23	26
OAK	30	23	28	26	14
SF	30	23	25	14	27
ARI	32	18	22	19	17
$\bar{\tau}$	—	**0.554**	**0.645**	**0.621**	**0.579**
Avg. Error	—	**8.31**	**6.78**	**7.88**	**8.63**
% Improved	—	—	$\mathbf{\frac{19}{32}}$	$\mathbf{\frac{15}{32}}$	$\mathbf{\frac{9}{32}}$
% Worse	—	—	$\mathbf{\frac{9}{32}}$	$\mathbf{\frac{11}{32}}$	$\mathbf{\frac{22}{32}}$

In Table 7 above, the cell entries are colored according to whether or not the model of interest better predicted a given team's ranking in 2018 in comparison to that team's win total in 2017. Notice that it is clear that ALGORITHM 2 performs significantly better than the other models for this single year of data. Some alternative summary statistics to $\bar{\tau}$ are shown to illustrate how the PageRank rankings compare in other measures of ranking association.

C Accounting for Home Field Advantage

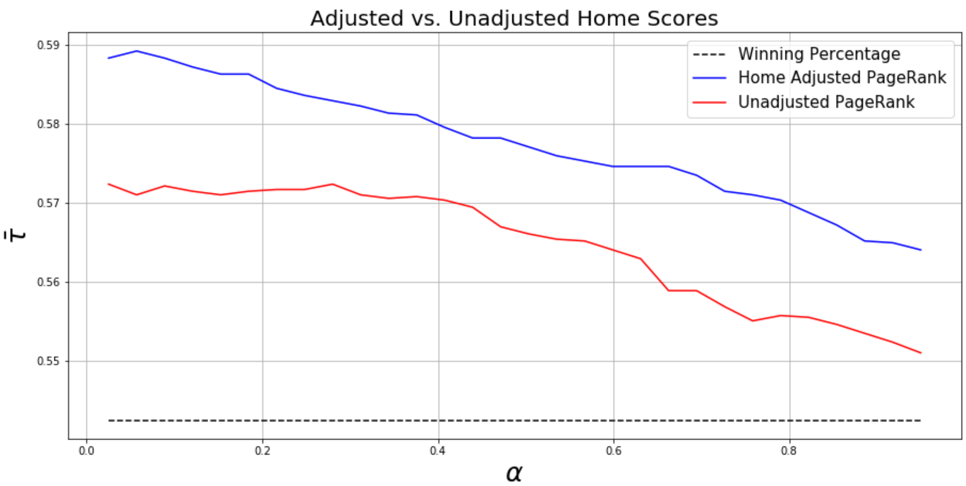

Figure 11: **Accounting for Home Field Advantage**

The models compared in Figure 11 above used unweighted links and a uniform teleportation vector, but this trend is not limited to just this case. It was observed for all quantities of interest tested.

D Cross Validated Results

Table 8, below, displays the cross validated results for each algorithm. These were obtained by sequentially removing one year of data at a time to serve as the test set. Then, the optimal value for α was determined among the remaining years of data. This value of α was then applied to the test set data, and $\bar{\tau}$ was computed. The average of all of the $\bar{\tau}$ values obtained was used as the cross validated estimate for the true value of $\bar{\tau}$.

Table 8: Comparison of Methods Discussed

Model	$\bar{\tau}$	Cross Validated $\bar{\tau}$
PageRank	0.590	0.586
`ALGORITHM 1`	0.601	0.598
`ALGORITHM 2`	0.610	0.602