15TH ANNUAL
MIT SLOAN
SPORTS ANALYTICS CONFERENCE
APRIL 8-9, 2021 | VIRTUAL
presented by ESPN 42 ANALYTICS

MAYFIELD: Machine Learning Algorithm for Yearly Forecasting Indicators and Estimation of Long-Run Player Development

Alexander H. Williams[1]
Department of Economics
The Ohio State University

Sethward R. Brugler
Inventory Systems
General Motors

Benjamin A. Clarke
Department of Computer Science & Engineering
The Ohio State University

Accurate statistical prediction of American football player development and performance is an important issue in the sports industry. We propose and implement a novel, fast, approximate k-nearest neighbor regression model utilizing locality-sensitive hashing in highly dimensional spaces for prediction of yearly National Football League player statistics. MAYFIELD accepts quantitative and qualitative input data and can be calibrated according to a variety of parameters. Concurrently, we propose several new computational metrics for empirical player comparison and evaluation in American football, including a weighted inverse-distance similarity score, stadium and league factors, and NCAA-NFL statistical translations. We utilize a training set of comprehensive NFL statistics from 1970-2009, across all player positions and conduct cross-validation on the model with the subset of 2010-18 NFL statistics. Preliminary results indicate the model to significantly improve on current, publicly available predictive methods. Future training with advanced statistical datasets and integration with scouting-based methods could improve MAYFIELD's accuracy even further.

Keywords: k-nearest neighbors, regression, locality-sensitive hashing, time-series forecasting, nonparametric models, American football

1. Introduction

Accurate forecasting of on-field performance in professional sports is a major component of player evaluation by fans, coaches, and sports executives. Due to a variety of factors, including a lack of highly detailed, publicly available data, emphasis on traditional video scouting methods by coaches and executives, and the relative complexity of the sport, American football is considerably less analytically developed than other professional sports, most notably baseball and basketball. Silver (2003, 2015) presents advanced comprehensive forecasting models for the MLB and NBA, although

[1] Corresponding author; can be reached via email at williams.5889@osu.edu.

these algorithms are proprietary and are thus irreproducible. Schatz (2008) presents a similar non-comprehensive[2] model for the NFL, although it is likewise not publicly available.

We present a reproducible, comprehensive, learning-based methodology for year-by-year statistical forecasting of NFL players' careers and implement it on the entire set of post-merger (i.e., after 1970) NFL players. A wide survey of the relevant literature reveals that, to date, no algorithm exists which comprehensively projects NFL player statistics across all positions and utilizes a dataset of MAYFIELD's size and scope. We also propose several important contributions to football analytics for future implementation into MAYFIELD: an Approximate Value metric for collegiate football players, NCAA-NFL statistical translations which adjust for park and league factors, and a Jamesean-style Similarity Scores framework for empirical player comparison.

Our paper proceeds in the following manner. Section 2 describes MAYFIELD's dataset, the operation of the MAYFIELD algorithm, and reviews the relevant previous work which MAYFIELD builds upon. Section 3 gives our initial evaluation of MAYFIELD's accuracy. Section 4 concludes the paper.

2. Methodology

2.1 Data

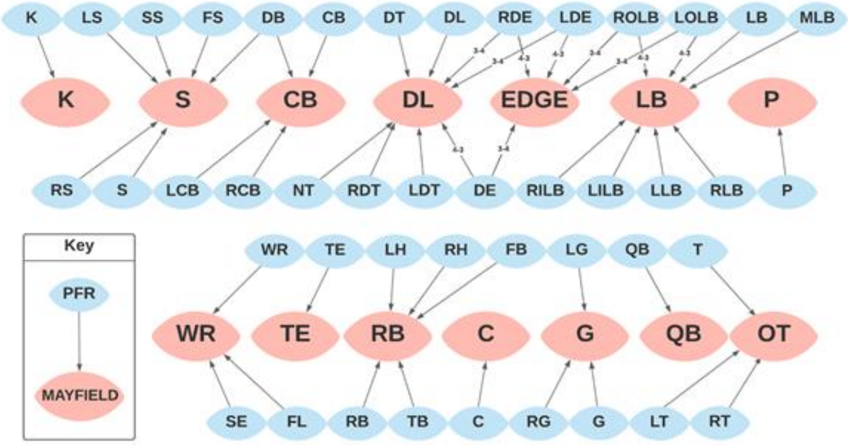

Figure 1- Position Mapping

MAYFIELD utilizes a dataset derived from the Pro Football Reference (PFR) historical database consisting of on-field performance statistics and biographical information for every player, team, and season since 1970, when the NFL and the AFL merged. For purpose of analysis, we

[2] Offensive linemen and punters are not included in Schatz's (2008) forecasts.

segregate the player data into several position groups according to their PFR-listed positions as shown in Figure 1[3].

Our on-field statistics comprise a set of standard NFL box-score statistics such as yards, touchdowns, tackles, field goal attempts, etc. As depicted in Figure 2, we limit the set of variables considered for players at a given position to only those relevant to the on-field role of a typical player at that position. Importantly, we include PFR's "Approximate Value" metric (Drinen 2008b), which places a numerical value on the all-inclusive contributions of a player to his team's success in a given season.

Our biographical data can be split into two categories: static, and dynamic. Static biographical variables are variables for a player whose initial value never changes from year to year, whereas dynamic biographical variables may fluctuate from year to year. We list which biographical variables are used for offensive (i.e., QB, RB, TE, WR, OL) and defensive (i.e., DL, EDGE, LB, CB, S) players in Figure 3[4]. In addition to data collected from PFR, some of the dynamic variables are not taken explicitly from PFR, but instead calculated from the data, including the variables for changes coaching, team, and scheme, consecutive years with a players' current coaches, team, and scheme, and the total AV per game of a team's players at each position during the previous season. One dynamic biographical statistic of note is team ratings from the Simple Rating System (SRS), which is described by Drinen (2006). SRS estimates the strength of a team's offense and defense relative to the league average in terms of points per game- for instance, a team with an offensive SRS of +6.0 would be expected to score 6 more points per game than an average team, all else equal.

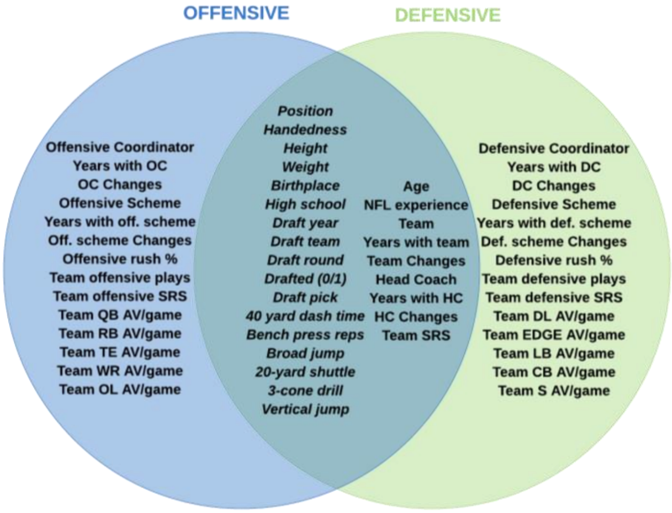

Figure 2- Biographical Variable Assignment

[3] Note that some PFR-listed positions are mapped to MAYFIELD's according to the defensive schemes which the player performed under (i.e., 4-3 or 3-4). This is denoted by the name of the scheme marked on the corresponding arrow.

[4] Special teams players (K, P) only use the statistics listed in the intersection of Figure 3. Note that static biographical statistics are italicized.

When predicting a future performance, MAYFIELD combines players' historical performance over many years into overlapping periods of the respective player's career, which we refer to as segments. Each segment consists of a player's static biographical data and Y consecutive seasons of that player's dynamic biographical and performance data. We find the set S of all of the consecutive Y-year segments and use this S as the basis for our training set.

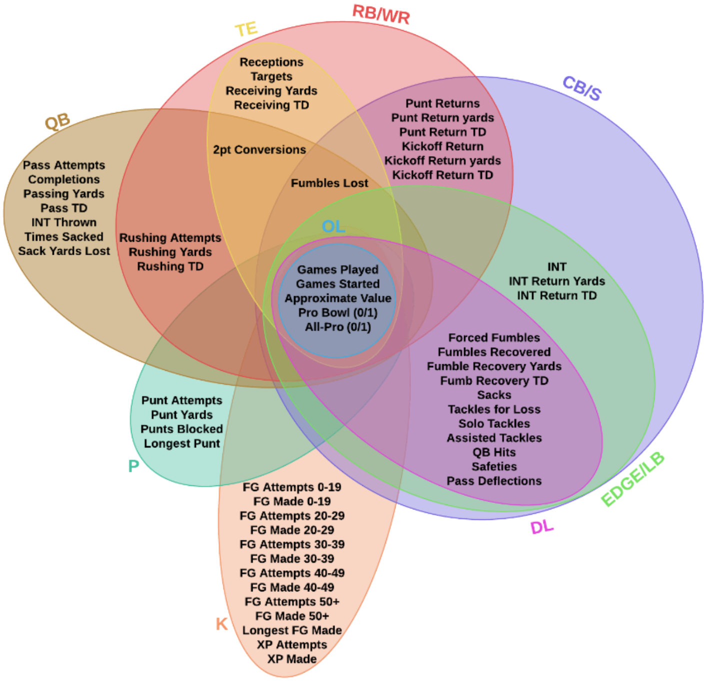

Figure 3- Performance Variable Assignment

2.2 Statistical Translations

2.2.1 NFL Equivalencies

A major hurdle when comparing raw statistics of professional football players is variance in scoring environments. Kickers, for instance, often experience high performance variance due to weather conditions (Pasteur & Cunningham-Rhoads 2014). Similarly, due to differences in rules across leagues, or the quality and styles of play, players of otherwise equal ability may experience nonrandom variance in their observed statistics. Despite that some of the earliest sabermetrics work addresses this issue in professional baseball (e.g., Davenport 1996; James 1985; Thorn & Palmer 1984), no method yet exists for American football.

We adopt a method similar to that of Szymborski (1997) to translate players' collegiate and professional statistics to one NFL-average baseline. First, we calculate stadium factors of each statistic for each team and post-merger season using a modified variant of Thorn & Palmer (1984)'s calculations, which we give below. Our base factor $\Phi^{i,t}$ for team i in year t on a statistic λ, where $\lambda^{i,t}$ is the per-game average of λ which team i recorded in year t, *and* $\lambda'^{i,t}$ is the per-game average of λ which team i allowed in year t, is:

$$\Phi^{i,t} = \frac{\lambda^{i,t}_{home} + \lambda'^{i,t}_{home}}{\lambda^{i,t}_{away} + \lambda'^{i,t}_{away}}$$

However, Φ is not entirely satisfactory since it accounts for neither the caliber of the team in question nor its opponents. To resolve the issue, we introduce Thorn & Palmer's (1984) offensive and defensive team ratings with respect to λ which we call τ. The formulae for τ are as follows, letting n^t be the number of NFL teams playing during the current year:

$$\tau^{i,t}_{off} = \left[\frac{\lambda^{i,t}_{away}(n^t - 1)}{n^t(n^t - 2) + \Phi^{i,t}} + \frac{\lambda^{i,t}_{home}}{\Phi^{i,t}(n^t - \Phi^{i,t})}\right] \cdot \frac{n^t - 2 + \tau^{i,t}_{def}}{2\overline{\lambda^{i,t}}}$$

$$\tau^{i,t}_{def} = \left[\frac{\lambda'^{i,t}_{away}(n^t - 1)}{n^t(n^t - 2) + \Phi^{i,t}} + \frac{\lambda'^{i,t}_{home}}{\Phi^{i,t}(n^t - \Phi^{i,t})}\right] \cdot \frac{n^t - 2 + \tau^{i,t}_{off}}{2\overline{\lambda^{i,t}}}$$

Since τ are codependent, we initialize each $\tau = 1$, and then recompute their values until convergence, typically after 3 iterations. The adjusted stadium factors Ω are then respectively:

$$\Omega^{i,t}_{off} = \frac{n^t - \Phi^{i,t}\frac{2n^t - \Phi^{i,t} - 1}{2n^t - 2}}{n^t - 2 + \tau^{i,t}_{def}} \quad ; \quad \Omega^{i,t}_{def} = \frac{n^t - \Phi^{i,t}\frac{2n^t - \Phi^{i,t} - 1}{2n^t - 2}}{n^t - 2 + \tau^{i,t}_{off}}$$

We maintain Thorn & Palmer's (1984) s segregation of offensive and defensive factors in our calculations; that is, offensive players' statistics are adjusted using Ω_{off} and defensive players' statistics are adjusted using Ω_{def}. However, instead of the typical 1-year park factors used in sabermetrics, we utilize an unweighted 5-year moving average of Ω due to the comparably smaller sample size of games in a given NFL season. Without adjustment to the time horizon of stadium factor calculations, random variance in the data may influence stadium factors more than is optimal.

So far, our stadium factors only differ from those of Thorn & Palmer (1984) in our calculation of Φ. From here, we utilize Szymborski's (1997) method for our translations. The two remaining components of interest are year and league factors. To adjust for these effects, we compute a league-year factor, Θ for λ as follows:

$$\Theta^t = \frac{\overline{\lambda^{i,*}}}{\overline{\lambda^{i,t}}} \cdot \delta$$

Θ is just the average team's λ during some base year and league over the average team's λ in a given league and year, times a deflator δ which measures the relative talent level of a player's league as a compared to the base league. We define our base as the NFL 2018-19 season, although any season or league could theoretically be used. The calculation of δ is not detailed by Szymborski, so where $\chi^{i,t}$ is the cumulative yearly statistics which player *i* recorded, we define δ as the ratio in NFL-average $\chi^{i,t}$ and the average $\chi^{i,t}$ for a given league, among players which played in both the NFL and that league over the course of their careers (note that we adjust $\chi^{i,t}$ with Ω and Θ first before computing averages).

We conduct translations of both players' collegiate and professional statistics. Importantly, δ is measured separately for each NCAA D-IA conference and an “Other" category for all non-D-IA players. The final translated value of a player's $\chi^{i,t}$ is then:

$$\chi\widehat{i,t} = \chi i,t\sqrt{\Theta^{t}/\Omega^{i,t}}$$

Note that these translations are not a prediction of how a player would have performed during that year if they had been in the NFL, rather, they are an estimation of the player's observed performance in the context of the baseline NFL environment. Changes in playing time due to higher levels of team talent, differences in play style, coaching, and strategy of NFL teams as opposed to collegiate teams, and other factors which influence observed player statistics in more nuanced manners are not captured by these translations.

2.2.2 Collegiate Approximate Value

All-inclusive measures of total player contributions to team success are an important component of advanced statistical analysis of sporting performance. The development and spread of such metrics (e.g., Wins Above Replacement in the MLB, Player Efficiency Rating in the NBA, and Real Plus-Minus in the NHL and NBA) have greatly affected both the style of play on the field and the evaluation of player talent off of it. Although less analytically advanced relative to its analogs in other professional sports, Approximate Value (Drinen 2008a) has proven a useful tool for empirically evaluating the performance of NFL athletes. However, Approximate Value is only defined for the NFL, and does not currently support evaluation of collegiate football players. Therefore, we extend Drinen’s (2008a) methodology to NCAA football, and give our formulation of collegiate Approximate Value below.

Approximate Value evaluates player contributions to team success according to the players' primary positional responsibilities with respect to cumulative team success at that position group. Consider the following partition of offensive team performance, where ζ_a is the fraction of total team offensive output by a positional responsibility a, among skill position players (defined as QB, RB, WR, TE):

$$\zeta^{i,t}_{Pass} = 0.11 \cdot \frac{\chi^{i,t}_{PassYards}}{g^{i,t}\lambda^{i,t}_{TotalYards}} \div \frac{\overline{\lambda^{i,t}_{PassYards}}}{\overline{\lambda^{i,t}_{TotalYards}}} + \left(\chi^{i,t}_{AYA} - \overline{\chi^{i,t}_{AYA}}\right) \cdot \begin{cases} \frac{1}{2} & \chi^{i,t}_{AYA} > \overline{\chi^{i,t}_{AYA}} \\ -2 & \chi^{i,t}_{AYA} < \overline{\chi^{i,t}_{AYA}} \end{cases}$$

$$\zeta^{i,t}_{Rush} = 0.12 \cdot \frac{\chi^{i,t}_{RushYards}}{g^{i,t}\lambda^{i,t}_{TotalYards}} \div \frac{\overline{\lambda^{i,t}_{RushYards}}}{\overline{\lambda^{i,t}_{TotalYards}}} + \left(\chi^{i,t}_{YPC} - \overline{\chi^{i,t}_{YPC}}\right) \cdot \begin{cases} \frac{3}{4} & \chi^{i,t}_{YPC} > \overline{\chi^{i,t}_{YPC}} \\ -2 & \chi^{i,t}_{YPC} < \overline{\chi^{i,t}_{YPC}} \end{cases}$$

$$\zeta^{i,t}_{Rec} = 0.31\overline{54} \cdot \frac{\chi^{i,t}_{RecYards}}{g^{i,t}\lambda^{i,t}_{TotalYards}} \div \frac{\overline{\lambda^{i,t}_{RecYards}}}{\overline{\lambda^{i,t}_{TotalYards}}}$$

$$\zeta^{i,t}_{Block} = 0.1\overline{136} \cdot \frac{\chi^{i,t}_{Games}}{5g^{i,t} + \sum_{i \in Team_{i,t}} \chi^{i,t}_{Games}} \cdot \begin{cases} 6 & \text{Top 1 on } Team_i \text{ in } \chi^{i,t}_{Touches} \text{ at TE} \\ 1 & \text{else} \end{cases}$$

Where $g^{i,t}$ is the number of games which player i's team played in year t, $Team_{i,t}$ is the set of players on player i's team in year t, $\chi_{YPC} = \frac{\chi_{RushYards}}{\chi_{RushAtt}}$, $\chi_{AYA} = (\chi_{PassYards} + 20\chi_{PassTD} - 45\chi_{PassINT})/\chi_{PassAtt}$, and $\chi_{Touches} = \chi_{RushAtt} + \chi_{Receptions}$. Note that here (and thereafter) the leading coefficients on each term are due to Drinen (2008a), and as in Section 2.2.1, averages are calculated within-conference. Furthermore, ζ_{Block} is used exclusively for the TE position and takes a value of 0 otherwise. So, letting $\lambda_{PPD} = \lambda_{Points}/\lambda_{Drives}$, skill position Approximate Value is thus:

$$AV_{skill}^{i,t} = 100\frac{\lambda_{PPD}^{i,t}}{\overline{\lambda_{PPD}^{i,t}}} \cdot (\zeta_{Pass}^{i,t} + \zeta_{Rush}^{i,t} + \zeta_{Rec}^{i,t} + \zeta_{Block}^{i,t})$$

So far, we use Drinen's (2008a) method without modification. However, we necessarily deviate for calculation of Approximate Value at the offensive line positions (OT, OG, C) due to Drinen's (2008a) use of starting lineup and Pro Bowl / All-Pro[5] data, which is unavailable and/or nonexistent at the collegiate level. Therefore, we adopt the following procedure, which substitutes All-America and All-Conference awards for the Pro Bowl and All-Pro dummy variables respectively, and does not utilize starting lineup data:

$$AV_{lineman}^{i,t} = \frac{\chi_{Games}^{i,t}}{g^{i,t}} \begin{cases} 6 & \text{Top 5 on } Team_i \text{ in } \chi_{Games}^{i,t} \text{ at OL} \\ 1 & \text{else} \end{cases} \cdot \begin{cases} 2 & \text{All} - \text{America Selection} \\ 1.65 & \text{All} - \text{Conference Slection} \\ 1 & \text{else} \end{cases}$$

We now define defensive Approximate Value:

$$\begin{aligned} AV_{defense}^{i,t} = {} & (200 + 100\frac{\overline{\lambda'^{i,t}_{PPD}}}{\lambda'^{i,t}_{PPD}} - 50\frac{\lambda'^{i,t}_{PPD}}{\overline{\lambda'^{i,t}_{PPD}}}) \cdot (\frac{\chi_{Sack}^{i,t}}{g^{i,t}\lambda_{Sack}^{i,t}} + 4\frac{\chi_{INT}^{i,t}}{g^{i,t}\lambda_{INT}^{i,t}} + 4\frac{\chi_{FumblesForced}^{i,t}}{g^{i,t}\lambda_{FumblesForced}^{i,t}} \\ & +4\frac{\chi_{FumbleRec}^{i,t}}{g^{i,t}\lambda_{FumblesRec}^{i,t}} + 5\frac{\chi_{DefTD}^{i,t}}{g^{i,t}\lambda_{DefTD}^{i,t}} + \psi^i\frac{\chi_{Tackles}^{i,t}}{g^{i,t}\lambda_{Tackles}^{i,t}} + \frac{\chi_{Games}^{i,t}}{5g^{i,t} + \sum_{i \in Team_{i,t}} \chi_{Games}^{i,t}} \\ & \cdot \begin{Bmatrix} 6 & \text{Top 11 on } Team_i \text{ in } \chi_{Tackles}^{i,t} \\ 1 & else \end{Bmatrix} + \frac{6.\overline{6}\chi_{Games}^{i,t}}{43 + 6.\overline{6}g^{i,t}} \begin{Bmatrix} 1.55 & \text{All-America Selection} \\ 0.85 & \text{All-conference Selection} \\ 0 & \text{else} \end{Bmatrix}) \end{aligned}$$

Where $\psi^i = 0.6$ when player i is a DL or EGDE, 0.3 when an LB, and 0.1 when a CB or S. This formulation of defensive AV is identical to that of Drinen (2008a), except for its modification on account of All-Americans and a lack of starting lineups, and that unlike Drinen, we utilize proportions of team-total defensive statistics (i.e., $\frac{\chi}{g\lambda}$) rather than pooling by position group. Special teams Approximate Value is given as follows:

$$\begin{aligned} AV_{special}^{i,t} = {} & \chi_{ReturnTD}^{i,t} + 1.65\frac{\chi_{Punts}^{i,t}}{g^{i,t}\lambda_{Punts}^{i,t}} + 0.06\frac{\chi_{PuntYards}^{i,t} - \chi_{Punts}^{i,t}\overline{\chi_{YPP}^{i,t}}}{g^{i,t}} \\ & +2.35\frac{\chi_{XPAtt}^{i,t} + 3\chi_{FGAtt}^{i,t}}{g^{i,t}(\lambda_{XPAtt}^{i,t} + 3\lambda_{FGAtt}^{i,t})} + 2.4\frac{\chi_{XP}^{i,t} - \chi_{XPAtt}^{i,t}\overline{\chi_{XPP}^{i,t}} + 3(\chi_{FG}^{i,t} - \chi_{FGAtt}^{i,t}\overline{\chi_{FGP}^{i,t}})}{g^{i,t}} \end{aligned}$$

[5] For the unfamiliar reader, these are designations given to the top players at each position in the NFL.

Where $\chi_{YPP} = \frac{\chi_{PuntYards}}{\chi_{Punts}}, \chi_{XPP} = \frac{\chi_{XP}}{\chi_{XPAtt}}$, and $\chi_{FGP} = \frac{\chi_{FG}}{\chi_{FGAtt}}$. Finally, the cumulative Approximate Value for a player is thus:

$$AV^{i,t} = AV^{i,t}_{skill} + AV^{i,t}_{lineman} + AV^{i,t}_{defense} + AV^{i,t}_{special}$$

This formulation of collegiate Approximate Value borrows heavily from Drinen's (2008a) methodology and is designed to be as similar as feasibly possible, as to make player comparisons of the variety described in Section 2.3 more meaningful, as comparing these "Approximate Values" for one player across his collegiate and professional career yields less useful information the more dissimilar the two metrics are. We now proceed to describe MAYFIELD's algorithmic structure and operation.

2.3 The kNN Algorithm

K-nearest neighbor (kNN) algorithms are among the oldest and most widely utilized methods in machine learning. Since Cover (1968) originally put forth kNN[6], nearest-neighbor algorithms have proliferated in usage for prediction problems in both classification and regression. The nearest-neighbor principle from which kNN derives is essentially just that objects which are highly similar in their observable characteristics (i.e., are "nearest neighbors") are likely to be similar in their unknown characteristics as well. Due to the simplicity of this premise and of nearest-neighbor algorithms in general, kNN has been shown as a versatile, but surprisingly effective method for prediction in problems as disparate as credit scoring (Mukid et al. 2018), cardiovascular medicine (Shouman, Turner, & Stocker 2012), and encryption (Wong et al. 2009). Importantly, several advanced player projection techniques similar to MAYFIELD (Silver 2003, 2015; Schatz 2008) utilize nearest-neighbor comparisons, although their proprietary nature prevents analysis of their algorithmic structure (kNN or otherwise). MAYFIELD follows in the tradition of these algorithms, albeit with more formalized and reproducible methods.

MAYFIELD can be characterized as an approximate kNN regression method for predicting the on-field statistics of National Football League players. In this respect, MAYFIELD is distinct from both kNN classification techniques, which predict categorical variables rather than continuous ones, and exact kNN methods, which search the entire training set for neighbors, as opposed to MAYFIELD's only partial search as controlled by locality-sensitive hashing (see Section 2.3.2). We provide the specification of MAYFIELD's exact structure and operation in the following subsections.

2.3.1 Model Training Procedure

Supervised learning methods such as kNN principally depend on parameters whose values are learned via training the specified model on the dataset. This process is achieved by generation of parameter combinations, measurement of the model's accuracy under the given parameter values, and repetition until errors are minimized. Hence, supervised learning is effectively equivalent to the optimization problem posed by finding the parameters values which minimize the model's error.

MAYFIELD learns the optimal values for 6 + Y distinct weighting parameters, whose respective roles in the MAYFIELD algorithm are detailed in Sections 2.3.2-2.3.6. These weighting

[6] For more recent surveys of modern kNN methods, see Bhatia & Vandana (2010) and Altman (1992).

parameters should be distinguished from hyperparameters, whose values are not learned via optimization, but instead heuristically selected based upon results of cross-validation. More specifically, we specify 5 hyperparameters: K, the number of “nearest neighbors" used to predict players' future performance vectors; Y, the length in years of player career segments; u, the number of years into the future which MAYFIELD predicts performance; T, the number of randomly generated hash functions used during locality-sensitive hashing (see Section 2.3.2); and B, the proportion of the data which is used for calculating each local regression in our LOESS correction (see Section 2.3.5). However, we only select values for K and Y during cross-validation; B and T have little direct relationship to MAYFIELD's accuracy, and we set these ex-ante. u is effectively a hyperparameter in name only, as we train MAYFIELD for all values of $u \in \{1,2,3\}$, since different u represent different sets of performance vectors we seek to predict.

We select the Covariance Matrix Adaptation-Evolution Strategy (CMA-ES) solver for optimizing MAYFIELD's parameters. For an excellent summary of CMA-ES, see Hansen (2016); we describe its basic method and desirable properties here. CMA-ES is a stochastic, evolutionary strategy solver which is especially useful for optimization on ill-conditioned problems (i.e., when fitness functions lack continuity, convexity, linearity, existence of derivatives, separability, low dimensionality, or other simplifying properties), and has been shown to be a highly efficient and reliable method for global optimization (Hansen 2009; Hansen & Kern 2004). As opposed to methods which attempt to directly estimate gradients (e.g., quasi-Newton), CMA-ES estimates a fitness function in several generations, selecting the most optimal parameter value combinations to serve as “parents" for the following generation of parameter estimates, repeating until the fitness function value converges. Since the landscape of player performance vectors is highly rugged, noisy, and highly dimensional, CMA-ES is an ideal optimization algorithm for use in MAYFIELD compared to other existing methods (e.g., BOBYQA, BFGS, Nelder-Mead, Powell, etc.).

We train MAYFIELD using the CMA-ES algorithm in the following manner. Given the entire training set of Y-length segments S and values for each hyperparameter, we specify a subset $R = \{r^{i,t} \in S: \exists r^{i,t+u} \in S\}$ of segments which have corresponding future performance vectors in the training data. After applying the statistical translations of Section 2.2 and converting the quantitative data to percentile ranks, we then apply locality-sensitive hashing on R which generates a reduced set $H_{r^{i,t}}$ of nonequivalent and arbitrarily similar segments for each member of R. Given an initial CMA-ES generation of parameter value combinations, the K most similar segments in $H_{r^{i,t}}$ to each $r^{i,t}$ are then used to predict the future performance vectors $\{v^{i,t+u}\}$ which correspond to R and whose value is a function of the parameters. The root-mean squared error across each dimension of the performance vector space V on these predictions is then calculated for each parameter combination. CMA-ES then re-estimates a generation of new parameter combinations based upon the most accurate members of the current generation and continues to repeat the parameters-predictions cycle until errors have converged. Once the solver terminates and returns the trained parameter values, we move on to a cross-validation step where we select the most optimal model across each different combination of hyperparameters, which we describe in Section 2.5.

2.3.2 Locality Sensitive Hashing

Locality-sensitive hashing (LSH) is a dimensionality reduction technique commonly employed in nearest neighbor search problems when distance measurements are computationally costly. LSH was first proposed by Gionis, Indyk & Motwani (1999) and has acquired a considerable history of usage in statistical similarity measurement and related applications (e.g., Datar et al. 2004; Andoni & Indyk 2008; Andoni et al. 2015; Das et al. 2007; Koga, Ishibashi, & Watanabe 2007; Cochez & Mou 2015; Brizna et al. 2010).

For any given player segment, MAYFIELD's initial N-order set of comparables may be in excess of 10,000 segments comprising up to q = 49*Y+17 variable dimensions, so some form of dimensionality reduction is necessary to make training MAYFIELD's parameters practically feasible. As a consequence of employing LSH, MAYFIELD is not an exact nearest neighbor search algorithm since some potential neighbors are excluded from distance measurement. However, the manner in which LSH operates ensures with high probability that these excluded segments are unlikely to be of practical interest when K is small relative to N (Leskovec 2001). We describe our LSH procedure below.

When predicting future performance vectors $v^{t+u} \in V = [0,1]^p$ of a segment r^0, we begin with a set $R = r^{i,t}$ of segments which might be compared to r^0. Each segment $r^{i,t}$ can thus be viewed as corresponding to a vector in the q-dimensional feature space, F. Due to our choice of distance function (see Section 2.3.3 below for discussion), we apply a linear transformation to F, which yields a modified feature space F':

$$F' = |C|^{-\frac{1}{q}} C\Sigma^{-\frac{1}{2}} F$$

$\Sigma^{-\frac{1}{2}}$ is the principal root of the inverse of the auto-covariance matrix Σ of F, and C is a $q \ \times q$ diagonal matrix of nonnegative weighting coefficients (Section 2.3.3 explains further). Letting r' be the vector in F' corresponding to $r \in F$, we first construct $[log\_2(N)]$ hyperplanes. For each hyperplane, we first generate q random vectors, which are distributed under the following multivariate uniform distribution:

$$w \sim \Pi_{j=1}^{q} \mathcal{U}^{j} \left(min(r_j' \in F'), max(r_j' \in F') \right)$$

Multiple linear regression on the set $\{w\}$ of these random vectors yields some affine function $f^{k\in\{1,2\ldots q\}}$: $\Pi_{j=1}^{q-1} F_j' \rightarrow F_q'$ which describes a q-1-dimensional hyperplane that intersects every point in $\{w\}$. If we consider r' as just σ, r_q', the resulting hash function h will be:

$$h: F' \rightarrow \{0,1\}^q, h(r'^{i,t}) \coloneqq ([f^1(\sigma^{i,t}) \leq r_q'^{i,t}], \ldots [f^q(\sigma^{i,t}) \leq r_q'^{i,t}])$$

Notice that the jth bit of the hash code is just the value of the indicator function for that r'^{it}'s residual on f^k is nonnegative. These hash functions naturally imply a set of segments arbitrarily similar to r^0, namely:

$$H_{r^0}^{k} = \{r^{i,t} \in R: h(r'^{i,t}) = h(r^0) \wedge r^{i,t} \neq r^0\}$$

Statistical noise during the hyperplane construction process may result in the exclusion of low-distance comparables due to hyperplanes which cut closer to r^0 than optimal. Therefore, we repeat the LSH process T times and take the union $H_{r^0} = \cup_{k=1}^{T} H_{r^0}^{k}$ across the H-sets or repeat until $|H_{r^0}| \geq K$, whichever occurs last. We set the hyperparameter T = 20, although any positive integer value could potentially be used instead; however, one should keep in mind that too low of a T will result in a narrow H_{r^0}, while too high of T may not give the desired reduction in runtime.

2.3.3 A Novel Weighted Mahalanobis Distance Metric

The canonical nearest neighbor problem for some object of interest a, is to find the most similar object b to a within a search set A, where $a \in A$ and $a \neq b$, and to then make inferences about a based upon the observable characteristics of b. kNN is a generalization of nearest neighbor search which instead of mapping a to the most similar element b, returns the set $B \subseteq A$ of the K elements most similar to a (i.e., $|B| = K$ where $a \notin B$). For MAYFIELD, our object of interest is r^0, and the search set is the LSH output H_r^0, which is a set of segments arbitrarily similar to r^0.

The principal question when designing a nearest-neighbor search algorithm is that of similarity measurement. The standard approach (which we adopt) is to specify some distance metric on the search set, and to identify the objects with lower distances to the object of interest as monotonically more similar. While many distance metrics exist which could potentially be utilized, the Mahalanobis distance (MD) is especially useful in similarity learning algorithms such as kNN (Mahalanobis 1936; De Maesschalck, Jouan-Rimbaud, & Massart 2000; Xing et al. 2003; Schultz & Joachims 2004; Woefel & Ekenel 2005). We give the classical formulation of MD below, where a and b are vectors in the feature space A, and Σ is the auto-covariance matrix of the features identified in A:

$$d_A(a, b) = \sqrt{(b - a)^\top \Sigma^{-1} (b - a)}$$

MD offers several advantages over more common metrics, such as Euclidean distance. Chiefly, the insertion of the inverse auto-covariance matrix Σ^{-1} simultaneously standardizes the scale of each axis (as measured by the feature variance) and adjusts for the presence of correlation between dimensions of the data, which may result in a non-orthonormal basis of A if the issue is left uncorrected (as in Euclidean distance). Moreover, if we set Σ to be just I_q, then d_A is the Euclidean norm on A. Similarly, any diagonal Σ (for instance, when each feature is pairwise independent) will yield a standardized Euclidean distance. Note that throughout, the distance component for a categorical variable between two segments would be the Jaccard distance (Jaccard 1901), rather than the numerical distance between the pair of vectors.

However, some authors identify issues with using this formulation of the MD metric. Σ is highly sensitive to the presence of outliers, which may bias estimates of MD (Woefel & Ekenel 2005). Additionally, Σ may perform suboptimally if MD is used for inference on the object of interest, as it does not adjust for the relation between the measured features and those which are being predicted (Schultz & Joachims 2004; Xing et al. 2003). As MAYFIELD is foremost a predictive algorithm, these issues are of serious concern.

To address the salient problems with classical MD, we make two modifications which should result in a more useful distance metric. Firstly, instead of estimating Σ as the classical auto-covariance matrix, we compute Rousseuw's (1984) Minimum Covariance Determinant (MCD) estimator Σ. MCD has been shown to be highly robust to outliers (Rousseuw & Van Drissen 1992; Hubert & Debruyne 2010), and previous authors' results demonstrate MCD to be a quite clearly superior estimator of covariance when used in MD calculations. Secondly, we introduce a diagonal matrix C^2 of weighting coefficients into the MD equation in the reparameterization described by Schultz & Joachims (2004). Their "weighted" MD metric replaces Σ with a matrix $A\,W\,A^\top$, where W is some $q\ \times q$ diagonal real matrix with nonnegative entries, and A is any $q\ \times q$ real matrix such that $A\,W\,A^\top$ is positive semidefinite. Since Σ is positive semidefinite and symmetric, assuming Σ is nonsingular[7], it necessarily has a unique positive semidefinite inverse square root (i.e., some A such that $A^2 = \Sigma^{-1}$) which is also symmetric (hence $\Sigma^{-\frac{1}{2}} = \Sigma^{-\frac{1}{2}^\top}$). Therefore, we set Schultz & Joachims' (2004) A equal to the square root of the inverse of Σ as estimated by the MCD (denoted by $\Sigma^{-\frac{1}{2}}$), and W to a matrix $|C|^{-\frac{2}{q}}C^2$, which we discuss below. This weighted MD is now:

$$d_F\left(r^{i,t}, r^0\right) = |C|^{-\frac{1}{q}}\sqrt{(r^{i,t} - r^0)^\top \Sigma^{-\frac{1}{2}} C^2 \Sigma^{-\frac{1}{2}} (r^{i,t} - r^0)}$$

Conveniently, d_F can be interpreted as simply the Euclidean distance between $r'^{i,t}$ and r'^0 in the linearly transformed space F'. Note that since $\Sigma^{-\frac{1}{2}}$ is positive semidefinite, and C^2 has strictly nonnegative entries independent of the specification of a diagonal C and is thus also positive semidefinite, the product matrix will satisfy Schultz & Joachims (2004) criteria of the positive semidefiniteness of $A\,W\,A^\top$. Additionally, since we utilize $\Sigma^{-\frac{1}{2}}$ and C in our implementation of LSH, our estimations of Σ and C are performed on the entire population of segments, rather than r^0-dependent subsets thereof such as R or H_{r^0}.

This generalized parameterization of MD is considerably more flexible than the classical variant. Considering each entry $C_{j,j}^2$ as simply the relative weight placed on the jth feature of $F_{j\in\{1,2,\ldots q\}}$, we are able to link each feature's influence on d_F to its predictive power. For instance, supposing each feature to have equivalent predictive power, we set $C = I_q$, which yields the classical MD, since $\Sigma^{-\frac{1}{2}}C^2\Sigma^{-\frac{1}{2}} = \Sigma^{-1}$ by definition.

The optimal measure of a feature's predictive power, especially when predicting future components of a vector as opposed to a singular scalar value, is less clear. While Schultz & Joachims (2004) suggest supervised learning of each coefficient in C, this is impractical for our values of q due to a lack of computing power. We therefore adopt the following approach to estimating C, which yields a computationally more efficient solution.

The principal goal of setting C is to maximize similarity between the arbitrary segments $r^{i_1,t}$ and $r^{i_2,t}$ contingent upon the similarity of future performance vectors $v^{i_1,t+u}$ and $v^{i_2,t+u}$, respectively. Therefore, we seek to identify the features that correlate most strongly with highly similar future performance of two segments. Consider the random variables η_j and κ, which are respectively the standardized Euclidean (i.e., the classical 1-dimensional Mahalanobis) pairwise distance between $r_j^{i_1,t}$ and $r_j^{i_2,t}$, and the classical MD $d_A\left(v^{i_1,t+u}, v^{i_2,t+u}\right)$, for any given segments $r^{i_1,t}$ and $r^{i_2,t}$:

[7] In practice, it is statistically unlikely to observe singular estimates of Σ. In such cases, we utilize the Moore-Penrose pseudoinverse of Σ, which exists for any real matrix Σ (Moore 1920; Penrose 1955; Bjerhammar 1951). Additionally, since the pseudoinverse of nonsingular Σ are just Σ^{-1}, we may be regarded as simply computing the pseudoinverse of Σ in all cases.

$$\eta_j = \frac{1}{\Sigma_{F_{j,j}}} \sqrt{\left(r_j^{i_2,t} - r_j^{i_1,t}\right)^2} \quad ; \quad \kappa = \sqrt{(v^{i_2,t+u} - v^{i_1,t+u})^\top \Sigma_V^{-1} (v^{i_2,t+u} - v^{i_1,t+u})}$$

To calculate C, we first identify the $\binom{N}{2}$ nonidentical segments in the population of the data and compute the values of η_j and κ for each pair under their respective MCD auto-covariance matrices. Letting $\rho_j = corr(\eta_j, \kappa)$ we appear to have found a suitable candidate for each coefficient on the diagonal of C. However, simply setting each $C_{j,j} = \rho_j$ is not quite satisfactory for two reasons. Firstly, if any $\rho_j = 0$, then C will be singular and result in indeterminate values of d_F, making any estimation of v impossible. Secondly, features which have negative correlations with κ make estimations of v based on d_F less accurate. These "malignant" features are such that the more similar any two segments are in that dimension of the data, the less similar their corresponding future performance vectors are. Consequently, weighting a malignant feature such that it comprises a higher proportion of d_F will result in higher selection rates of "near" neighbors which are similar in terms of r_j, but imply predicted v dissimilar to the actual "true" v, resulting in highly imprecise estimates. While the easiest course of action might be to simply remove such features from the calculation of d_F by setting $C_{j,j} = 0 \ \forall \rho_j \leq 0$, this runs into the former issue of a singular C. Therefore, to ensure strictly positive $C_{j,j}$ which appropriately upweight features with large, positive ρ, we adopt the following exponential weighting scheme based on a hyperparameter $\alpha \in [1, \infty)$:

$$C = \bigoplus_{j=1}^{q} \alpha^{\rho_j} \widehat{e_j}$$

This C in some sense still makes d_F a "learned" metric on F in the same manner which Schultz & Joachims's (2004) original specification of their "learned" MD; however, our approach is feasible when q is large without any loss of resolution in the data via principal components analysis or similar methods.

2.3.4 Inverse Distance-Weighted Similarity Scores

It is often useful to compute indices of similarity as an inverse distance function. Whereas a distance metric measures the dissimilarity of two objects on the interval $[0, \infty)$, where the larger the distance, the more dissimilar the two objects are, a similarity measure indexes the similarity of two objects on the interval $(0,1]$, with a larger similarity index indicating, as the name suggests, highly similar objects. The key property of any similarity measure is a strictly decreasing image with respect to its corresponding distance metric. Often, similarity measures can be expressed in closed form as kernel functions that map directly from the feature space, which are commonly used in learning-based algorithms (Schoelkopf, Tsuda, & Vert, 2004).

Similarity measures appear in some sabermetrics and other sports analytics work, albeit in a more informal manner. Bill James (1994) first introduced the concept of "similarity scores" for empirical comparison of Baseball Hall of Fame candidates to its already inducted members, which spawned a variety of methods for sports performance comparison (e.g. Silver, 2015; Kubatko 2004; Hollinger 2003; Pelton 2003; Drinen 2008b). Some forecasting models (Silver 2003, 2015; Schatz 2008) incorporate similarity scores into their respective methodologies as well. Similarity scores are one of the few areas of advanced empirical analysis of sports in which American football is wellrepresented. Schatz (2010) and Drinen's (2008b) methodologies are well-known, and more importantly, reproducible. However, almost all work on sports performance similarity scores have used strictly first or second order polynomial models[8] with no consideration for interactions between features. Moreover, the weighting coefficients utilized in such models appear to be completely arbitrary, as their respective authors give no treatment to the procedure used to estimate weights[9]. We seek to rectify these shortcomings by returning to more formally established methods in the non-sporting literature.

MD has the property that, if features are normally distributed, $d_A^2 \sim \chi^2(q)$. While our data is certainly nonnormal (and moreover, are highly nonindependent), making this property less useful, we percentilize the data, which conforms features to a strictly uniform distribution on $(0,1)$. Linear transformation of the feature space from F into F' modifies the range of the underlying feature distributions, but not their uniformity. Our distance function d_F is therefore principally a summation of uniformly distributed random variables, which for large q, converges to a normal distribution. Since our q certainly qualify as large ($q > 60$ in all cases), we may expect the distribution of d_F to be sufficiently approximated by $\chi^2(q)$.

We define our similarity score as the probability, given r^0, of observing a segment at least as distant from r^0 as a given segment $r^{i,t}$, which under the assumed $d_F \sim \chi^2(q)$:

$$SS(r^{i,t},r^0=1-\frac{1}{\Gamma(\frac{1}{2}q)}\gamma(\frac{1}{2}q,\frac{1}{2}d_F(r^{i,t},r^0))$$

Γ and γ are the complete and lower incomplete gamma functions, respectively. Notice that SS is just the complement of the $\chi^2(q)$ cumulative distribution function over d_F. For every segment $r^{i,t} \in H_{r^0}$, we compute $SS(r^{i,t}, r^0)$ and identify the following set of r^0's "nearest neighbors":

$$M_{r^0}=\{r^{i_1,t}\in H_{r^0}:|\{r^{i_2,t}\in H_{r^0}:SS(r^{i_2,t},r^0)\geq SS(r^{i_1,t},r^0)\}|\leq K\}$$

Less formally, M_{r^0} is just the set of the K most similar segments to r^0 which reside in H_{r^0}, or equivalently, the set of the K segments in H_{r^0} which are least distant from r^0. We now proceed to the following steps in the MAYFIELD algorithm, where we describe MAYFIELD's further utilization of M_{r^0} in its predictions of $v^{0,t+u}$.

2.3.5 The Regression Equation

Given the set M_{r^0}, our remaining task is to form an estimate of $v^{0,t+u}$. Consider the set L_{r^0}, the set of future performance vectors which correspond to the members of M_{r^0}:

$$L_{r^0}=\{v^{i,t+u}\in V:r^{i,t}\in M_{r^0}\}$$

[8] The methods we review exclusively use affine equations over either absolute or squared differences in observed feature values to compute their similarity scores.

[9] We speculate that the weighting coefficients of these similarity scores were set a priori by their respective authors, rather than via analytical methods.

Classical kNN regression techniques (e.g., Benedetti 1977; Altman 1992) typically compute a weighted average of vectors in L_{r^0} as the predicted value of $v^{0,t+u}$. However, more recent authors (e.g., Mehdizadeh 2020; Al-Qhatani & Crone 2013; Wen, Song, & Wang 2016) find that incorporating traditional time-series techniques such as autoregressive moving-average (ARMA) terms into the kNN regression equation greatly improves model accuracy and fit in some applications. We therefore take an approach which builds autoregressive terms and a directional component of L_{r^0} into the classical kNN regression equation:

$$\hat{v^0} = \varphi_0 + \varphi_1 \frac{\sum_{v^{i,t+u} \in L_{r_0}} SS(r^{i,t}, r^0)^{\beta_1} v^{i,t+u}}{\sum_{v^{i,t+u} \in L_{r_0}} SS(r^{i,t}, r^0)^{\beta_1}} + \varphi_2 \frac{\sum_{v^{i,t+u} \in L_{r_0}} SS(r^{i,t}, r^0)^{\beta_2} (v^{i,t+u} - v^{i,t})}{\sum_{v^{i,t+u} \in L_{r_0}} SS(r^{i,t}, r^0)^{\beta_2}} + \sum_{w=1}^{Y} \varphi_{w+2} v^{0,t-w+1}$$

The predicted value for $v^{0,t+u}$ is a linear combination of an inverse-distance weighted interpolation of L_{r^0}, a similar interpolation on the net change in $v^{i,t}$[10], and lagged values of $v^{0,t}$, plus a constant. The first pair of terms is the classical kNN regression value, which is a first-order polynomial on the interpolated $v^{i,t+u}$ weighted by similarity and a parameter β_1. The third term is the interpolated change in $v^{i,t}$ weighted by similarity and a parameter β_2. The final term is a Y-order autoregressive series on $v^{0,t}$, which are the single-year components of the performance vector within r^0. Note that the φ and β are weighting parameters whose values are learned through the training process.

We favor this hybrid approach to regression, much like other parts of MAYFIELD, due to its flexibility and computational efficiency. Beyond the aforementioned improvement in accuracy resulting from integrating AR into the kNN model, there are a few other advantageous attributes to the above formulation of the regression equation. One common pitfall of algorithms similar in aim to MAYFIELD is their failure to consistently predict statistics with accuracy better than even a naive model. Simply letting $\varphi_3 = 1$ and the remaining $\varphi_{w \neq 3} = 0$, the above just becomes a naive prediction of $v^{0,t+u}$. MAYFIELD is thus guaranteed accuracy no worse than either a naive or the classical kNN predicted value, since both methods are subsumed by our model.

However, we are not quite done. Two potential issues may arise in the above regression, dependent on the local structure of the data in the training set. Firstly, our formulation of the kNN regression may predict values of $\widehat{v^{0,t+u}}$ which has components $\widehat{v_J^{0,t+u}} \notin [0,1]$. Since percentile ranks not contained on this interval are impossible, we need a method for dealing with such instances in the data. Standard methods in least-squares regression for prediction of bounded dependent variables include transforming the data using an asymptotically bounded function (e.g., probit and logit models), or censoring fitted values which fall outside the interval (e.g., Tobit models). We take a slightly different route due to the second potential issue we identify: systematic bias in $v^{0,t+u}$, which may arise among any or all of the components of $v^{0,t+u}$. Since CMA-ES optimization attempts to minimize a scalar error (see Sections 2.3.1 and 2.3.6), each individual dimension of V may be suboptimally predicted by $v^{0,t+u}$. We may regard predictions of $v_j^{0,t+u} \notin [0,1]$ as a special case of such systemic bias, since any $\widehat{v_J^{0,t+u}} > 1$ are inherently biased upwards, and any $\widehat{v_J^{0,t+u}} < 0$ are similarly biased downwards. Additionally, there may be cases where for of some range of $\widehat{v_J^{0,t+u}} \in$

[10] This term can be unambiguously interpreted as the expected residual of a naive prediction of $v^{i,t+u}$.

$[a, b] \subseteq [0,1]$ the expected residuals $E\left(v_j^{0,t+u} - \widehat{v_j^{0,t+u}}\right) \notin [a, b]$. For instance, $v_j^{0,t+u}$ may be nonmonotonic with respect to $\widehat{v_j^{0,t+u}}$, or may increase at a rate different from unity. kNN regressions such as MAYFIELD are particularly suspect in this respect, since interpolations on the data as performed in nearest-neighbor algorithms may over-predict regression to the mean. Although we expect such cases to be limited, robustness to such issues is a desirable feature for regression models.

Given the possible estimation bias that may enter into our regression, we take the additional step of fitting a LOESS model for each feature in V, i.e., regressing $v_j^{0,t+u}$ on $\widehat{v_j^{0,t+u}}$. LOESS, or locally estimated scatterplot smoothing, was independently discovered by Savitzky & Golay (1964) and Cleveland (1979), and has since become widely utilized in several fields of application, including learning-based algorithms (e.g., Cleveland & Devlin 1988; Cleveland, Grosse, & Shyu, 1992; Jacoby 2000; Trexler & Travis 1993; Berger et al., 2004; Howarth & McArthur 1997; McArthur & Howarth 2001). LOESS is a nonparametric method which estimates weighted low-order polynomial regressions on overlapping subsets of the independent variable and constructs a smoothed function on the local polynomials. The size of these subsets depends on the bandwidth hyperparameter B, which is the ratio of the order of the subsets to N, the number of total players in R. Larger values of B result in smoother results and make the locally estimated polynomials more robust to outliers. We set $B = \frac{log_2(N)}{N}$, which results in a relatively large bandwidth, as our data is both highly noisy and highly dense. So, where g_j is the LOESS-constructed function for the jth feature of V, our final predicted value of the performance vector $v^{0,t+u}$ is:

$$\widehat{\widehat{v^{0,t+u}}} = \sum_{j=1}^{p} g_j\left(\widehat{v_j^{0,t+u}}\right)\widehat{e_j}$$

Our utilization of this LOESS correction to $\widehat{v^{0,t+u}}$ yields several advantages. Firstly, the aforementioned issues of ill-defined and biased values of $\widehat{v^{0,t+u}}$ are resolved. Secondly, the nonparametric nature of LOESS admits a far more flexible bias correction than the standard parametric models; cases of non-monotonically increasing $v^{0,t+u}$ with respect to $\widehat{v^{0,t+u}}$ are likely to benefit. Thirdly, LOESS's confidence intervals are calculated based on the local structure of the data, as opposed to aggregate measures of variance (as in most least-squares regressions) and give a meaningful and ready-made confidence bounds on the expected range of $v^{0,t+u}$. These may be especially useful when more than just point-estimates of $v^{0,t+u}$ are required. Finally, investigation on the shape of the LOESS-generated functions g_j may reveal various characteristics of $\widehat{v^{0,t+u}}$ such as over or under-prediction of regression to the mean, artificially induced clustering, or lack of predictive power by the training data.

Note that while it is possible to absorb the linear regression in the above formula into the LOESS correction by simply computing a multiple LOESS regression of $v^{0,t+u}$ on the respective terms in the former regression stage, there are good reasons against doing so. Since we are interested in the values of φ, which offer a measure of the relative importance of the respective components in the above formula, LOESS' lack of a functional form with directly interpretable coefficients would mean that combining the two regression stages would result in a loss of this information. Additionally, we intend LOESS as a minor correction to the prediction of $\widehat{v^{0,t+u}}$, not the prediction itself; due to LOESS' flexibility, overfitting, especially in a multiple regression setting, is a concern for the out-of-sample robustness of $\widehat{v^{0,t+u}}$. Moreover, LOESS is computationally expensive, especially when performed on multivariate data, so letting LOESS estimate $v^{0,t+u}$ from $SS(r^0, r^{i,t})$, $v^{0,t-w-1}$, $v^{i,t+u}$, and $v^{i,t}$ would result in far greater runtimes than with our linear first stage and a univariate LOESS correction.

2.3.6 RMSE Fitness Function

After predicting $\widehat{v^{i,t+u}}$ for all $r^{i,t} \in R$, we need a mechanism for computing the cumulative error of $\widehat{v^{0,t+u}}$ across all dimensions of V. Standard methods, such as computing root-mean squared error (RMSE) or mean absolute scaled error (MASE), may work for predicting univariate data, but for estimating V, do not have readily available formulations for comparing accuracy across the entire performance vector space V. Hence, we require an extra step to convert $\widehat{v^{0,t+u}} - v^{0,t+u}$ from a p-dimensional vector to a scalar value.

Fortunately, we have already specified a metric which has this capacity: the classical Mahalanobis distance. In fact, classical MD is especially useful since it adjusts for covariance between the dimensions of V, and thus constitutes a measure of the general lack of information on $v^{0,t+u}$ with respect to $\widehat{v^{0,t+u}}$, rather than simply the cumulative observed errors across each dimension. After computing the classical MD between the predicted and realized values of $v^{0,t+u}$, errors can be interpreted as univariate, and methods such as RMSE and MASE may then be applied to measure the cumulative error across the set of all $\widehat{v^{0,t+u}}$. Therefore, we specify our fitness function, whose value the CMA-ES optimizer attempts to minimize, as follows:

$$\epsilon(R) = \sqrt{\frac{\sum_{r^{i,t}\in R}\left(\widehat{v^{i,t+u}} - v^{i,t+u}\right)^{\top} \Sigma^{-1} \left(\widehat{v^{i,t+u}} - v^{i,t+u}\right)}{N}}$$

ε is thus the RMSE of the classical MD between $v^{i,t+u}$ and $\widehat{v^{i,t+u}}$ for all segments in R. While ε will depend on the local structure of the data in R, it is principally a function of the weighting parameters ε, β_1, and β_2, and the set of φ, which MAYFIELD learns via CMA-ES minimization of ε.

Position	***U = 1***	***U = 2***	***U = 3***
QB	15	15	15
RB	20	40	35
WR	15	45	50
TE	45	50	45
OL	50	40	50
DL	30	30	25
EDGE	15	15	25
LB	20	20	40
CB	25	40	45
S	20	45	50
K	10	10	45
P	45	50	50

Table 1: Optimal K

2.4 Cross-Validation Procedure

To select the optimal hyperparameters for MAYFIELD, we first train (via the CMA-ES optimizer) MAYFIELD's parameters on a 1970-2009 subset of the data, and then evaluate MAYFIELD's out-of-sample accuracy on a 2010-2018 subset of the data in a cross-validation step. We test values of $K \in \{5,10,15,20,25,30,35,40,45,50\}$ and $u \in \{1,2,3\}$ with Y= 3. We present the results of our cross-validation above in Table 1. Note that for player-seasons for which the player in question has played less than Y years, we use his predicted stats for $Y = y$, where y is the length in years of the player's career so far, and the same value of K as is Y were unchanged. RMSE are stable between the training set and the out-of-sample results, indicating that MAYFIELD is not accuracy due to spurious variation in the data, but rather genuine predictive power (see Table 2 for example)

Table 2: MAYFIELD Tight End RMSEs (u=2) In-sample vs. out-of-sample

	Games	Games Started	AV	All pro	Pro bowl	2pt conversions	Fumbles	Receptions	Targets	Rec yards	Rec td
In-sample	2.63	3.37	1.33	0.13	0.20	0.15	0.62	11.34	15.08	137.32	1.46
Out-of-sample	2.67	3.38	1.46	0.16	0.25	0.23	0.44	11.88	18.95	137.43	1.51

3. Results

To evaluate the effectiveness of MAYFIELD, we compared the predicted season results of our model to the 2010-2017 predictions of KUBIAK (Schatz, 2008), the foremost football statistics prediction model. Similar to MAYFIELD, KUBIAK considers historical performance of each player over multiple seasons, biographical statistics, and comparisons to similar players. We compare the standardized RMSEs (i.e., where 1.0 represents an RMSE of 1 standard deviation, lower is better) for all positions below:

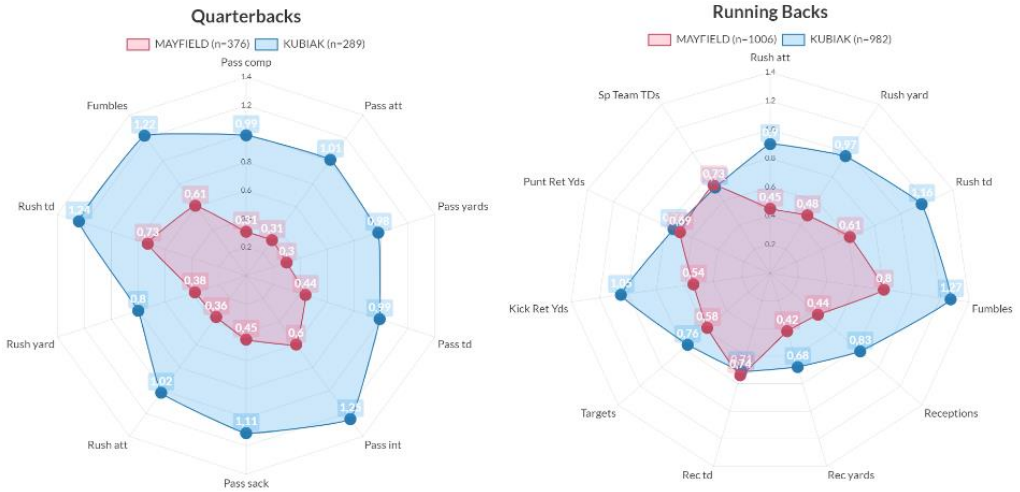

Figure 4- MAYFIELD vs. KUBIAK Standardized RMSEs

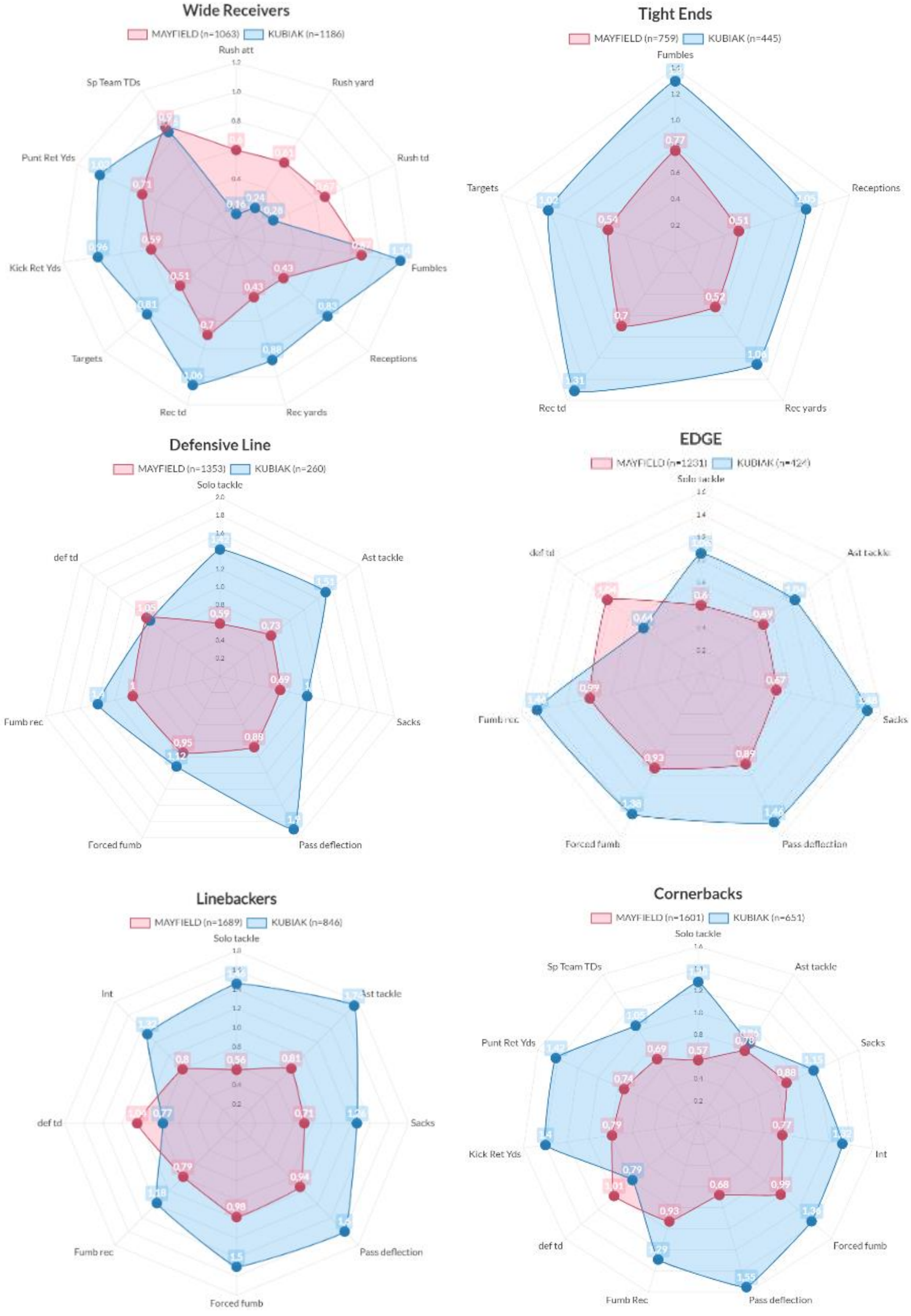

Figure 5 (contd.)- MAYFIELD vs. KUBIAK Standardized RMSEs

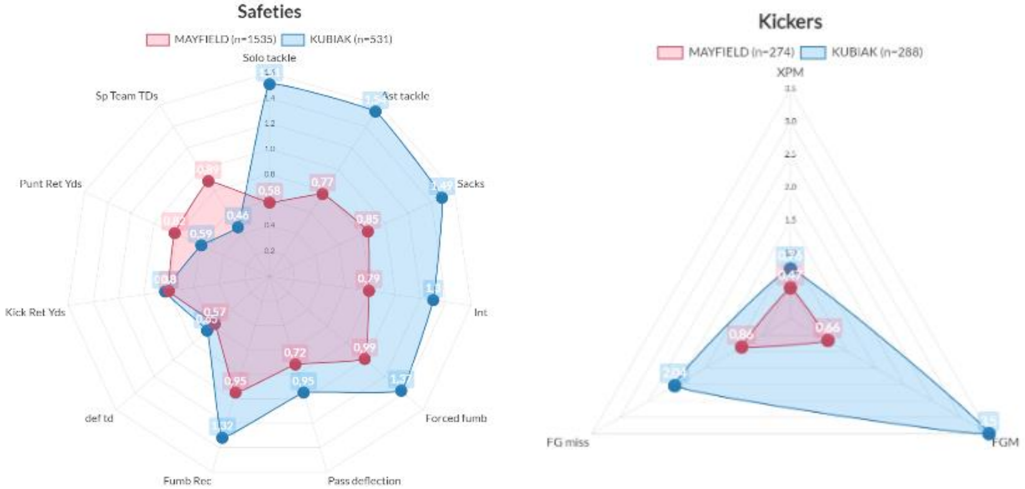

Figure 6 (contd.)- MAYFIELD vs. KUBIAK Standardized RMSEs

Overall, MAYFIELD offers a substantial improvement in accuracy over KUBIAK's methods in every position as shown by the graphs above. Furthermore, MAYFIELD's standardized RMSEs are more balanced across each statistic as compared to KUBIAK. MAYFIELD's main weaknesses appear to be the prediction of defensive/special teams touchdowns and wide receiver rushing statistics. For almost every position that includes the defensive/special teams TD statistic, MAYFIELD's standardized RMSE’s for these variables were among the largest among all statistics at the respective positions and were also usually larger than the KUBIAK values. In addition to these categories, the only other case where KUBIAK significantly (at 90% confidence) outperformed MAYFIELD was for wide receiver rushing statistics (rushing attempts, yards, touchdowns). While these previously mentioned variables, are better estimated by KUBIAK, they are not truly indicative of player performance for that respective position, as defensive touchdowns, special teams touchdowns, and wide receiver rushes are currently rare occurrences in the NFL and not stable. MAYFIELD is significantly better than KUBIAK for all other statistics. This includes all quarterback, tight end, and kicker statistics. This also includes all statistics that are very relevant to their respective positions, including running back rushing attempts, yards, touchdowns and fumbles, wide receiver targets, receptions, receiving yards and touchdowns, and defensive solo tackles, assisted tackles, tackles for loss, sacks, forced fumbles, fumble recoveries, pass deflections, and interceptions. These statistics are better indicative of player talent level and of a team’s performance as a whole.

Table 3: MAYFIELD RMSEs (u=1) 2010-2017 out-of-sample

	QB	RB	WR	TE	OL	DL	EDGE	LB	CB	S	K	P
Games	2.78	2.98	3.31	2.73	4.30	2.44	3.06	2.68	2.54	2.45	2.35	2.32
Games Started	2.01	3.07	3.28	3.09	5.14	2.97	3.04	3.39	2.81	3.38	1.62	1.16
AV	2.36	1.91	1.67	1.34	3.05	1.58	1.67	1.54	1.36	1.32	1.47	2.80
All pro	0.15	0.17	0.16	0.14	0.20	0.15	0.18	0.15	0.18	0.18	0.18	0.19
Pro bowl	0.25	0.23	0.23	0.22	0.28	0.21	0.25	0.20	0.23	0.24	0.24	0.22
2pt conversions	0.17	0.29	0.30	0.22								
Pass comp	44.77											
Pass att	70.07											
Pass yards	508.44											
Pass td	5.13											
Pass int	3.56											
Pass sack	6.89											
Pass sack yards	48.28											
Rush att	8.73	34.64	16.31									
Rush yard	51.69	161.23	77.93									
Rush td	1.08	1.60	0.66									
Fumbles	2.28	1.09	0.94	0.45					0.35	0.34		
Receptions		8.85	12.02	10.55								
Targets		17.04	22.80	16.82								
Rec yards		92.92	157.14	123.76								
Rec td		1.14	1.94	1.45								
Punt ret		3.35	5.21						2.48	2.39		
Punt ret yards		35.90	53.86						25.61	25.35		
Punt ret TD		0.11	0.18						0.08	0.08		
Kick ret		3.53	4.10						1.86	1.89		
Kick ret yards		93.12	108.70						48.66	49.44		
Kick ret TD		0.11	0.12						0.02	0.02		
Forced fumb						0.86	1.06	0.89	0.77	0.77		
Fumb rec						0.61	0.68	0.65	0.60	0.62		
Fumb rec yards						5.14	9.04	7.94	9.30	9.29		
Fumb rec TD						0.15	0.20	0.18	0.18	0.18		
Sacks						2.12	2.34	1.81	0.83	0.82		
Tfl						2.82	3.09	2.72	1.56	1.61		
Solo tackle						7.94	11.56	14.45	13.17	13.59		
Ast tackle						5.14	6.21	8.45	6.18	6.26		
QB hits						5.02	5.32	3.83	1.80	1.85		
Safeties						0.13	0.15	0.13	0.06	0.07		
Pass deflection						1.58	1.93	2.21	3.33	3.34		
Int							0.47	0.70	1.07	1.10		
Int yards							8.68	13.62	22.65	22.85		
Int TD							0.18	0.22	0.32	0.32		
FGA 0-19											0.38	
FGM 0-19											0.38	
FGA 20-29											2.35	
FGM 20-29											2.23	
FGA 30-39											2.35	
FGM 30-39											2.25	
FGA 40-49											2.59	
FGM 40-49											2.41	
FGA 50+											1.92	
FGM 50+											1.56	
FGM Long											5.24	
XPA											8.06	
XPM											8.06	
Punt att												8.62
Punt yards												403.88
Punt block												0.53
Punt Long												5.81

4. Conclusion

MAYFIELD displays excellent predictive accuracy across all positions and appears to be very balanced across the performance variables, rather than targeting a few specific variables. This is likely due to the endogenous nature of MAYFIELD's parameter weighting scheme (as discussed in Section 2.3), its large sample size of player comparisons to draw upon, and its algorithmic architecture that builds in state-of-the-art computational tools from the computer science, statistics, and sports analytics literature. Given the results shown here, MAYFIELD could be successfully applied for numerous sports forecasting problems, such as in player scouting, sports gambling, and prediction of team-level performance and results.

The MAYFIELD algorithm we propose is designed in a manner which supports integration with possible future advances in football analytics. For instance, while our dataset consists of only standard box-score performance variables, newer metrics reliant[11] upon player-tracking or other advanced techniques are easily absorbed into MAYFIELD's feature space, since our distance function is robust to missing data (a major concern for any newer metrics which are unable to be calculated for historical data). Similarly, integration with other contextual data, such as player positions on depth chart, scouting report grades, and medical or salary information, would likewise increase MAYFIELD's accuracy by their addition to the feature space without any modification needed to make full use of the inserted variables.

Most American sports, such as baseball and basketball, have experienced a boom in analytics while football has fallen behind. Factors such as the complexity of the sport, emphasis on traditional scouting methods, and lack of high-quality public data have led to this gap. Our MAYFIELD algorithm is an effort to close that gap. Similar methods exist in other sports such as those proposed by Silver (2003, 2015), but the algorithm details are not as transparent as MAYFIELD. Attempts to produce a similar model for football (Schatz 2008) focus only on fantasy-relevant players and have methods which are likewise largely not publicly available. We present a reproducible, comprehensive, learning-based methodology for year-by-year statistical forecasting of NFL players' careers and implement it on the entire set of post-merger NFL players. The initial results we present here indicate MAYFIELD to be an improvement over currently existing methods. Based on a wide survey of the relevant literature, MAYFIELD is unprecedented in size and scope of application. We also propose several important contributions to football analytics for future implementation into MAYFIELD: an Approximate Value metric for collegiate football players, NCAA-NFL statistical translations which adjust for park and league factors, and a Jamesean-style Similarity Scores framework for empirical player comparison. These advancements represent substantial progress in updating football analytics methods to the state-of-the-art as compared to other professional sports and demonstrate MAYFIELD's potential for utilization by football decision-makers, statisticians, and fans alike.

[11] E.g., pass blocking and pass rushing win rate (Burke 2018), air yards, completion percentage allowed, nflWAR (Yurko, Ventura, & Horowitz 2019), etc.

Acknowledgements

We thank Dr. John Draper and Dr. Ryan Ruddy for their guidance on the project. Their advice, assistance, and encouragement proved invaluable. Additionally, we thank Dr. Chris Knoester and the Ohio State University Sports and Society Initiative for their support, Jesse Chick, Nick Sarkauskas and Dr. Dhabaleswar Panda for their technical assistance, and Josh Krassenstein for his contributions to the project's initial stages. Finally, we thank Aaron Schatz and Football Outsiders for generously sharing KUBIAK's historical data with us.

References

Al-Qhatani, F., Crone, S. 2013. Multivariate k-nearest neighbour regression for time series data — a novel algorithm for forecasting UK electricity demand. Proceedings of the 2013 International Joint Conference on Neural Networks, 228-235.

Altman, N.S. 1992. An introduction to kernel and nearest neighbor nonparametric regression. The American Statistician, 4(3): 175-185.

Andoni, A., Indyk, P. 2008. Near-optimal hashing algorithms for approximate nearest neighbor in high dimensions. Communications of the ACM, 51(1): 117-122.

Andoni, A., Indyk, P., Laarhoven, T., Razenshteyn, I., Schmidt, L. 2015. Practical and optimal LSH for angular distance. Proceedings of the 28th International Conference on Neural Information Processing Systems, 1225–1233.

Benedetti, J. K. 1977. On the Nonparametric Estimation of Regression Functions. Journal of the Royal Statistical Society Series B, 39(2): 248-253.

Berger, J., Hautaniemi, S, Jaervinen, A.K., Edgren, H. Mitra S.K., Astola, J. 2004. Optimized LOWESS normalization parameter selection for DNA microarray data. BMC Bioinformatics, 5(194): 1-13.

Bhatia, N., Vandana. 2010. Survey of nearest neighbor techniques. International Journal of Computer Science and Information Security, 8(2): 302-305.

Bjerhammar, A. 1951. Application of calculus of matrices to method of least squares; with special references to geodetic calculations. Transactions of the Royal Institute of Technology, 49.

Brizna, D., Schultz, M., Tesler, G., Bafna, V. 2010. RAPID detection of gene-gene interactions in genome-wide association studies. Bioinformatics, 26(22): 2856-2862.

Burke, B. 2018. We created better pass-rusher and pass-blocker stats: How they work. ESPN. Accessed at https://www.espn.com/nfl/story/_/id/24892208/creating-better-nfl-pass-blocking-pass-rushing-stats-analytics-explainer-faq-how-work.

Cleveland, W.S. 1979. Robust locally weighted regression and smoothing scatterplots. Journal of the American Statistical Association, 74(368): 829-836.

Cleveland, W.S., Devlin, S. 1988. Locally weighted regression: an approach to regression analysis by local fitting. Journal of the American Statistical Association, 83(403): 596-610.

Cleveland, W.S., Grosse, E., Shyu, W.M. Local regression models; 309-376 in Chambers, J.M., Hastie, T. 1992. Statistical models in S. Chapman & Hall / CRC Press. Print.

Cochez, M., Mou, H. 2015. Twister tries: approximate hierarchical agglomerative clustering for average distance in linear time. Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data, 505-517.

Cover, T. 1968. Estimation by the nearest neighbor rule. IEEE Transactions on Information Theory, 14(1): 50-55.

Das, A., Datar, M., Garg, A., Rajaram, S.S. 2007. Google news personalization: scalable online collaborative filtering. Proceedings of the 16th International Conference on World Wide Web, 271-280.

Datar, M., Immorlica, N., Indyk, P., Mirrokni., V. 2004. Locality-sensitive hashing scheme based on p-stable distributions. Proceedings of the twentieth annual symposium on Computational geometry, 253-262.

Davenport, C. Davenport Translations; in Huckaby, G., Davenport, C., Jazayerli, R., Kahrl, C., Sheehan, J. 1996. Baseball Prospectus 1996. Baseball Prospectus LLC. Accessed at https://legacy.baseballprospectus.com/other/bp1996/dtessay.html.

De Maesschalck, R. Jouan-Rimbaud, D. Massart, D.L. 2000. The Mahalanobis distance. Chemometrics and Intelligent Laboratory Systems, 50(1): 1-18.

Drinen, D. 2006. A very simple ranking system. Pro Football Reference. Accessed at https://www.pro-football-reference.com/blog/index4837.html?p=37.

Drinen, D. 2008. Approximate value in the NFL. Pro Football Reference. Accessed at https://www.pro-football-reference.com/blog/index6b92.html?p=465.

Drinen, D. 2008. Who is the current Dave Duerson?. Pro Football Reference. Accessed at https://www.pro-football-reference.com/blog/indexa215.html?p=556.

Gionis, A., Indyk, P., Motwani, R. 1999. Similarity search in high dimensions via hashing. Proceedings of the 25th VLDB Conference, 518-529.

Hansen, N. 2009. Benchmarking a bi-population CMA-ES on the BBOB-2009 function testbed. Workshop Proceedings of the GECCO Genetic and Evolutionary Computation Conference, 2389-2395.

Hansen, N. 2016. The CMA Evolution Strategy: a tutorial. ArXiv Preprint: 1604.0077. Accessed at https://arxiv.org/abs/1604.00772.

Hansen, N., Kern, S. 2004. Evaluating the CMA Evolution Strategy on multimodal test functions. Proceedings of the Eighth International Conference on Parallel Problem Solving from Nature PPSN VIII, 282-291.

Hollinger, J. 2003. Pro Basketball Prospectus: 2003 Edition. University of Nebraska Press. Print.

Howarth, R.J., McArthur, J.M. 1997. Statistics For strontium isotope stratigraphy: a robust Lowess fit to the marine sr-isotope curve for 0 to 206 Ma, with look-up table for derivation of numeric age. The Journal of Geology, 105(4): 441-456.

Hubert, M., Debruyne, M. 2010. Minimum covariance determinant. Computational Statistics, 2(1): 36-43.

Jaccard, P. 1901. Etude comparative de la distribution florale dans une portion des Alpes et du Jura., Bulletin de la Soci'et'e Vaudoise des Sciences Naturelles, 37(1): 547–579.

Jacoby, W. 2000. Loess: a nonparametric, graphical tool for depicting relationships between variables. Electoral Studies, 19(4): 577-613.

James, B. 1985. The Bill James baseball abstract, 1985. Ballantine Books. Print.

James, B. 1994. The politics of glory. Macmillan Publishers. Print.

Kerhet, A. Small, C. Quon, H. Riauka, T. Schrader, L. Greiner, R. Yee, D. McEwan, A. Roa, W. 2010. Application of machine learning methodology for PET-based definition of lung cancer. Current Oncology, 17(1): 41–47.

Koga, H., Ishibashi, T., Watanabe, T. 2007. Fast agglomerative hierarchical clustering algorithm using locality-sensitive hashing. Knowledge and Information Systems, 12(1): 25-53.

Kubatko, J. 2004. Similarity scores. Basketball Reference. Accessed at https://www.basketball-reference.com/about/similar.html.

Leskovec, J., Rajaraman, A., Ullman, J. 2011. Mining of massive datasets. Cambridge University Press. Print.

Mahalanobis, P.C. 1936. On the generalized distance in statistics. Proceedings of National Institute of Sciences, 2(1): 49-55.

McArthur, J.M., Howarth, R.J., Bailey, T.R. 2001. Strontium isotope stratigraphy: LOWESS version 3: best fit to the marine sr-isotope curve for 0–509 Ma and accompanying look-up table for deriving numerical age. The Journal of Geology, 109(2): 155-170.

Mehdizadeh, S. 2020. Using AR, MA, and ARMA time series models to improve the performance of MARS and KNN approaches in monthly precipitation modeling under limited climatic data. Water Resources Management, 34(1): 263–282.

Moore, E. H. 1920. On the reciprocal of the general algebraic matrix. Bulletin of the American Mathematical Society, 26(9): 394–95.

Mukid, M.A., Widiharih, T., Rusgiyono, A., Prahutama, A. 2018. Credit scoring analysis using weighted k nearest neighbor. Journal of Physics: Conference Series, 1025(1): 012114.

Pasteur, R., Cunningham-Rhoads, K. 2014. An expectation-based metric for NFL field goal kickers. Journal of Quantitative Analysis in Sports, 10(1): 49-66.

Pelton, K. 2003. Review: Pro Basketball Prospectus: 2003-04 Edition. Hoopsworld. Accessed at http://www.hoopsworld.com/article_5978.shtml.

Penrose, R. 1955. A generalized inverse for matrices. Proceedings of the Cambridge Philosophical Society, 51(3): 406–13.

Rezvani, M., Hashemi, S.M. 2012. Enhancing accuracy of topic sensitive PageRank using Jaccard Index and cosine similarity. Proceedings of the 2012 IEEE/WIC/ACM International Conferences on Web Intelligence and Intelligent Agent Technology, 620-624.

Rousseeuw, P. 1984. Least median of squares regression. Journal of the American Statistical Association, 79(338): 871-880.

Rousseeuw, P., Van Driessen, K. 1999. A fast algorithm for the minimum covariance determinant estimator. Technometrics, 41(3): 212-223.

Savitzky, A., Golay, M.J.E. 1964. Smoothing and differentiation of data by simplified least squares procedures. Analytical Chemistry, 36(8): 1627–1639.

Schatz, A. 2008. Pro Football Prospectus 2008. Plume. Print.

Schatz, A. 2010. Football Outsiders similarity scores. Football Outsiders. Accessed at https://www.footballoutsiders.com/stats/similarity.

Schoelkopf, B., Tsuda, K., Vert, J.P. 2004. Primer on kernel methods in computational biology. MIT Press. Print.

Schultz, M., Joachims, T. 2004. Learning a distance metric from relative comparisons. Proceedings of the 16th International Conference on Neural Information Processing Systems, 41-48.

Shouman, M., Turner, T., Stocker, R. 2012. Applying k-Nearest Neighbour in diagnosing heart disease patients. International Journal of Information and Education Technology, 2(3): 220-223.

Silver, N. Introducing PECOTA; 507-514 in Huckaby, G., Kahrl, C., Pease, D. 2003. Baseball Prospectus: 2003 Edition. Potomac Books. Print.

Silver, N. 2015. We're predicting the career of every NBA player. Here's how. FiveThirtyEight. Accessed at https://fivethirtyeight.com/features/how-were-predicting-nba-player-career/.

Szymborski, D. 1997. How to calculate MLEs. Baseball Think Factory. Accessed at https://www.baseballthinkfactory.org/btf/scholars/czerny/articles/calculatingMLEs.htm.

Thorn, J., Palmer, P. 1984. The hidden game of baseball. Knopf Doubleday Publishing Group. Print.

Trexler, J., Travis, J. 1993. Nontraditional regression analyses. Ecology, 74(6): 1629-1637.

Wen Y., Song M., Wang, J. 2016. A combined AR-kNN model for short-term wind speed forecasting. Proceedings of the 2016 IEEE 55th Conference on Decision and Control, online.

Woelfel, M., Ekenel, H.K. 2005. Feature weighted mahalanobis distance: improved robustness for Gaussian classifiers. Proceedings of the 2005 13th European Signal Processing Conference, online.

Wong, W.K., Cheung, D.W., Kao, B., Mamoulis, N. 2009. Secure kNN computation on encrypted databases. Proceedings of the 2009 ACM SIGMOD International Conference on Management of data, 139–152.

Xing, E., Ng, A., Jordan, M., Russell, S. 2003. Distance metric learning with application to clustering with side-information. Proceedings of the 15th International Conference on Neural Information Processing Systems, 521-528.

Yurko, R., Ventura, S., Horowitz, M. nflWAR: a reproducible method for offensive player evaluation in football. Journal of Quantitative Analysis in Sports, 15(3): 163-183.