

How to Play Strategically in Fantasy Sports (and Win)

Paper Track: Other Sports

Paper ID: 5558

1. Introduction

Daily Fantasy Sports (DFS) has become a multi-billion dollar industry [24, 20, 28, 1, 30] with millions of annual users [14, 28]. The pervasiveness of fantasy sports in modern popular culture is reflected by the regular appearance of articles discussing fantasy sports issues in the mainstream media. Moreover, major industry developments and scandals are now capable of making headline news as evidenced in Figures 1(a) and 1(b) below. The two major DFS websites are FanDuel and DraftKings and together they control approximately 95% of the U.S. market [24, 20]. Approximately 80% of DFS players have been classified as *minnows* [25] as they are not believed to use sophisticated techniques for decision-making and portfolio construction. Accordingly, these users provide financial opportunities to the so-called *sharks* who do use sophisticated techniques [25, 29, 22, 17] when constructing their fantasy sports portfolios. The goal of this paper is to provide a coherent framework for constructing fantasy sports portfolios where we explicitly model the behavior of other players. Our approach is therefore strategic and to the best of our knowledge, we are the first academic work to develop such an approach in the context of fantasy sports.

The number of competitors in a typical DFS contest might range from thousands to hundreds of thousands with each competitor constructing a fantasy team of real world players, e.g. National Football League (NFL) players in a fantasy football contest, with each portfolio being subject to budget and possibly other constraints. The performance of each portfolio is determined by the performances of the real world players in a series of actual games, e.g. the series of NFL games in a given week. The competitors with the best performing portfolios then earn a monetary reward, which depends on the specific payoff structure, e.g. *double-up* or *top-heavy*, of the DFS contest.

Several papers have already been written on the topic of fantasy sports. For example, Fry, Lundberg, and Ohlmann [13] and Becker and Sun [4] develop models for season-long fantasy contests while Bergman and Imbrogno [5] propose strategies for the survivor pool contest, which is also a season long event. Several papers have been written of course on so-called office pools (which pre-date fantasy sports contests), where the goal is to predict the maximum number of game winners in an upcoming elimination tournament such as the *March Madness* college basketball tournament. Examples of this work include Kaplan and Garstka [19] and Clair and Letscher [6]. There has been relatively little work, however, on the problem of constructing portfolios for daily fantasy sports. One notable exception is the recent work of Hunter et al. [18], which is closest to the work we present in this paper. They consider a winner-takes-all payoff structure and aim to maximize the probability that one of their portfolios (out of a total of N) wins. Their approach is a greedy heuristic that maximizes their portfolio means, i.e. expected number of fantasy points, subject to constraints that lower bound their portfolio variances and upper bound

their inter-portfolio correlations. Technically, their framework requires the solution of linear integer programs and they apply their methodology to fantasy sports contests, which are top-heavy in their payoff structure as opposed to winner-takes-all. Their work has received considerable attention, e.g. [8], and the authors report earning¹ significant sums in real fantasy sports contests based on the National Hockey League (NHL) and baseball.

There are several directions for potential improvement, however, and they are the focus of the work in this paper. First, Hunter et al. [18] do not consider their opponents' behavior. In particular, they do not account for the fact that the payoff thresholds are stochastic and depend on both the performances of the real-world players as well as the unknown team selections of their fellow fantasy sports competitors. Second, their framework is only suitable for contests with the top-heavy payoff structure and is in general not suitable for the double-up payoff structure. Third, their approach is based on (approximately) optimizing for the winner-takes-all payoff, which is only a rough approximation to the top-heavy contests they ultimately target. In contrast, we directly model the true payoff structure (top-heavy or double-up) and seek to optimize our portfolios with this objective in mind.

Scandal Erupts in Unregulated World of Fantasy Sports

By JOE DRAPE and JACQUELINE WILLIAMS OCT. 5, 2015



(a) New York Times headline, 5 October 2015

NEWS AUG 3 2016, 11:44 PM ET

After Ferocious Battle, New York Legalizes DraftKings, FanDuel

by ALEX JOHNSON



(b) NBC News headline, 3 August 2016

Figure 1: Fantasy Sports in the News.

Our work makes several contributions to the DFS literature. First, we formulate an optimization problem that accurately describes the DFS problem for a risk-neutral decision-maker in both double-up and top-heavy settings. Our formulation seeks to maximize the expected reward subject to portfolio feasibility constraints and we explicitly account for our opponents' unknown portfolio choices in our formulation. Second, we connect our problem formulation to the finance literature on

¹ They donated all earnings to charity and we intend to do the same at the end of the 2017-18 NFL season, which is the basis for the real-world contests that we are currently playing using our methodology. Further details are available in Section 6.

mean-variance optimization and in particular, the mean-variance literature on outperforming stochastic benchmarks. Using this connection, we show how our problem can be reduced (via some simple assumptions and approximations) to the problem of solving binary quadratic programs. Third, we can use our model to estimate the value of “insider trading” where an insider, i.e. an employee of the DFS contest organizers, gets to see information on opponents’ portfolio choices before making his own team selections. This has been a topic of considerable recent media interest [9, 10]; see also Figure 1(a), which refers to the case of a DraftKings employee using data from DraftKings contests to enter a FanDuel DFS contest in the same week and win \$350,000. This problem of insider trading is of course also related to the well-known value-of-information concept from decision tree analysis. A further contribution of our work is the introduction of a Dirichlet-multinomial data generating process for modeling opponents’ team selections. We estimate the parameters of this model via Dirichlet regressions and we demonstrate its value in predicting opponents’ portfolio choices. Finally, we demonstrate the value of our framework by applying it to both double-up and top-heavy DFS contests in the 2017-2018 NFL season. While the season is still ongoing (as of writing), our results to date are positive particularly in the top-heavy contests, which are our main focus.

The remainder of this paper is organized as follows. In Section 2, we formulate both the double-up and top-heavy versions of the problem while we outline our Dirichlet regression approach to modeling our opponents’ team selections in Section 3. In Section 4, we use results from mean-variance optimization (that relate to maximizing the probability of outperforming a stochastic benchmark) to solve the double-up problem. We then extend this approach to solve the top-heavy problem in Section 5 and we present numerical results based on the 2017-18 NFL season for both problem formulations in Section 6. In Section 7, we briefly discuss the value of information and in particular, how much an insider can profit from having advance knowledge of his opponents’ team selections. We conclude in Section 8, where we also discuss some directions for ongoing and future research. Some technical details and results are deferred to the appendix.

2. Problem Formulation

We assume there are a total of P players whose performance, $\delta \in \mathbb{R}^P$, in a given round of games is random. We will assume² that

$$\delta \sim N(\mu_\delta, \Sigma_\delta),$$

so that δ is multivariate normally distributed with mean vector μ_δ and variance-covariance matrix Σ_δ . Our goal in the fantasy sports competition is to choose a portfolio $\mathbf{w} \in \{0,1\}^P$ of athletes. Typically, there are many constraints on \mathbf{w} . For example, in a typical NFL fantasy sports contest, we will only be allowed to select $C = 9$ players out of a total of $P \approx 100$ to 300 NFL players. Each

² While the formal results of Sections 4 and 5 require a normality assumption, we note that the mean-variance framework we develop can still be used even if normality is not justified. Moreover, we are free in lines 10 and 4 of Algorithms 2 and 3, respectively, to use a more accurate probability distribution if we so desire.



player also has a certain “cost” and our portfolio cannot exceed a given budget B . These constraints on \mathbf{w} can then be formulated as

$$\sum_{p=1}^P w_p = C$$

$$\sum_{p=1}^P c_p w_p \leq B$$

$$w_p \in \{0,1\}, p = 1, \dots, P,$$

where c_p denotes the cost of the p^{th} player. Other constraints are also typically imposed by the context organizers. These constraints include positional constraints, e.g. exactly one quarterback can be chosen, diversity constraints, e.g. you can not select more than 4 players from any single NFL team, etc. These constraints can generally be modeled as linear constraints and we use \mathbb{W} to denote the set of binary vectors $\mathbf{w} \in \{0,1\}^P$ that satisfy these constraints.

A key aspect of our approach to constructing fantasy sports portfolios is in modeling our opponents, i.e. other players who also enter the same fantasy sports competition. We assume there are O such opponents and we use $\mathbf{W}_{op} := \{\mathbf{w}_o\}_{o=1}^O$ to denote their portfolios with each $\mathbf{w}_o \in \mathbb{W}$. Once the round of NFL games has taken place, we get to observe the realized performances $\boldsymbol{\delta}$ of the P NFL players. Our portfolio then realizes a points total of $F := \mathbf{w}^T \boldsymbol{\delta}$ whereas our opponents' realized point totals are $G_o = \mathbf{w}_o^T \boldsymbol{\delta}$ for $o = 1, \dots, O$. All portfolios are then ranked according to their points total and the cash payoffs are determined. These payoffs take difference forms, depending on the structure of the contest. There are two contest structures that dominate in practice and we consider both of them. They are the so-called double-up and top-heavy structures.

2.1. The Double-Up Problem Formulation

Under the double-up payoff structure, the top r portfolios (according to the ranking based on realized points total) each earn a payoff of R dollars. Suppose now that we enter $N \ll O$ portfolios³ to the contest. Then, typical values of r are $r = (O + N)/2$ and $r = (O + N)/5$ with corresponding payoffs of $R = 2$ and $R = 5$ assuming an entry fee of 1 per portfolio. The ($r = (O + N)/2, R = 2$) case is then a double-up competition whereas the ($r = (O + N)/5, R = 5$) is a *quintuple-up* contest. We will refer to all such contests as “double-up” contests except when we wish to draw a distinction between different types of double-up contests, e.g. (true) double-up versus quintuple-up. In practice of course, the contest organizers take a cut and keep approximately 15% of the entry fees for themselves. This is reflected by reducing r appropriately and we note that this is easily accounted for in our problem formulation below. We also note that this means the average DFS player loses approximately 15% of her initial entry. In contrast to financial investments then, DFS

³ There is usually a cap imposed by the contest organizer, however. Typical cap sizes we have observed can range from $N = 1$ to $N = 150$.





investments are on average NPV-negative and so some skill is required in portfolio construction to overcome this handicap.

While it is possible and quite common for a fantasy sports player to submit multiple entries, i.e. multiple portfolios, to a given contest, we will consider initially the case where we submit just $N = 1$ entry. Given the double-up payoff structure, our fantasy-sports portfolio optimization problem is to solve

$$\max_{\mathbf{w} \in \mathcal{W}} \mathbb{P}\{\mathbf{w}^T \boldsymbol{\delta} > G^{(r')}(\mathbf{W}_{\text{op}}, \boldsymbol{\delta})\}, \quad (1)$$

where we use $G^{(r)}$ to denote the r^{th} order statistic of $\{G_o\}_{o=1}^O$ and we define $r' := O + 1 - r$. Note that we explicitly recognize the dependence of $G^{(r)}$ on the portfolio selections \mathbf{W}_{op} of our O opponents and the performance vector $\boldsymbol{\delta}$ of the NFL players.

2.2. The Top-Heavy Problem Formulation

The top-heavy payoff structure is more complicated than the double-up structure as the size of the cash payoff generally increases with the portfolio ranking. In particular, we first define payoffs

$$R_1 > \dots > R_D > R_{D+1} := 0$$

and corresponding percentiles

$$0 := r_0 > r_1 > \dots > r_D.$$

Then a portfolio whose rank lies in $(r_{d-1}, r_d] \times O$ wins R_d for $d = 1, \dots, D$. In contrast to the double-up structure, we now account⁴ for the possibility of submitting $N > 1$ entries to the contest. We use $\mathbf{W} := \{\mathbf{w}_i\}_{i=1}^N$ to denote these entries and $F_i := \mathbf{w}_i^T \boldsymbol{\delta}$ to denote the realized fantasy points total of our i^{th} entry.

It is then easy to see that our fantasy-sports portfolio optimization problem is to solve

$$\max_{\mathbf{W} \in \mathcal{W}^N} \sum_{i=1}^N \sum_{d=1}^D (R_d - R_{d+1}) \mathbb{P}\{\mathbf{w}_i^T \boldsymbol{\delta} > G_{-i}^{(r'_d)}(\mathbf{W}_{-i}, \mathbf{W}_{\text{op}}, \boldsymbol{\delta})\}, \quad (2)$$

where $r'_d = O + N - r_d O$, $G_{-i}^{(r)}$ is the r^{th} order statistic of $\{G_o\}_{o=1}^O \cup \{F_j\}_{j=1}^N \setminus F_i$ and $\mathbf{W}_{-i} := \mathbf{W} \setminus \mathbf{w}_i$. We note again that the top-heavy payoff structure is our main concern in this paper. That said, it should be clear that the double-up formulation of (1) is a special case of the top-heavy formulation in (2). We will therefore address the double-up problem before taking on the top-heavy problem. Before doing this, however, we must discuss the modeling of our opponents' portfolios \mathbf{W}_{op} .

⁴ Our reasons for this will be clear later in Section 5.



3. Modeling Opponents' Team Selections

A key aspect of our modeling approach is that there is value to modeling our opponents' portfolio choices, \mathbf{W}_{op} . This is in direct contrast to the work of Hunter et al. [18], who ignore this aspect of the problem and focus instead on constructing portfolios that maximize the expected number of fantasy points, subject to possible constraints⁵ that encourage high-variance portfolios. In numerical simulations, we obtained significant gains in expected dollar payoffs by explicitly modeling \mathbf{W}_{op} . This followed because of the well-known fact that some players are (often considerably) more popular than other players and because there is some predictability in the team selections of players who may be responding to weekly developments that contain more noise than genuine information. To the best of our knowledge, we are the first to explicitly model \mathbf{W}_{op} and embed it in our portfolio construction process. That said, we certainly acknowledge that some members of the fantasy sports community also attempt to be strategic in their attempted selection of less popular players and avoidance of more popular players, other things being equal; see for example Gibbs [15].

If we are to exploit our opponents' team selections, then we must be able to *predict* \mathbf{W}_{op} reasonably accurately. Indeed it is worth emphasizing that \mathbf{W}_{op} is *not observed* before the contest and so we must make do with predicting it. In most fantasy sports contests, however, it is possible to learn about \mathbf{W}_{op} after the contest is over and the winners have been announced. We therefore assume we have access to partial data on \mathbf{W}_{op} from previous contests. In particular, we make the reasonable assumption that we have access to sufficient data to estimate the 1-dimensional marginal distributions⁶ of $\mathbf{w}_o \in \{0,1\}^P$, a random opponent's portfolio. We will also assume we have access to other observable features, e.g. expected NFL player performance $\boldsymbol{\mu}_\delta$, home or away indicators, quality of opposing teams etc. from these previous contests. We will use this information to build a Dirichlet regression model for estimating the marginal distributions of \mathbf{w}_o . Towards this end, we must first describe our regression model and then in Section 3.2, we put everything together to (implicitly) define the distribution of \mathbf{W}_{op} .

3.1. Dirichlet Regression

To make things clear, we will focus on the specific case of DFS in the NFL setting. Specifically, consider for example the following NFL contest organized by FanDuel [11]. Each fantasy team has $C = 9$ positions which must consist of 1 quarterback (QB), 2 running backs (RB), 3 wide receivers (WR), 1 tight end (TE), 1 kicker (K) and 1 defense (D). Consider now the marginal distribution of

⁵ They included constraints that encouraged high-variance portfolios because they too were focused on top-heavy contests where very few contestants earn substantial payoffs. It is intuitively clear that high-variance portfolios are desirable for such contests. We discuss this property in further detail in Sections 4 and 5 in light of the results from mean-variance optimization that we bring to bear on the problem.

⁶ Sometimes we can only get to observe \mathbf{W}_{op} after a contest via web-scraping, something that is not very desirable and that is frowned upon by some sites.



the QB selections⁷ in \mathbf{w}_o . To begin, we assume there is a total of N_{QB} QB's available for selection and we number them from $k = 1$ to $k = N_{QB}$. We assume a Dirichlet-multinomial data generating process for QB selection. Specifically, we assume:

- $\mathbf{p}_{QB} \sim \text{Dir}(\boldsymbol{\alpha}_{QB})$ where $\text{Dir}(\boldsymbol{\alpha}_{QB})$ denotes the Dirichlet distribution with parameter vector $\boldsymbol{\alpha}_{QB}$.

Note $\mathbf{p}_{QB} := \{p_{QB}^k\}_{k=1}^{N_{QB}}$ lies on the unit simplex in $\mathbb{R}^{N_{QB}}$ and therefore defines a probability distribution over the available quarterbacks. We can therefore

- Select QB k with probability p_{QB}^k for $k = 1, \dots, N_{QB}$.

It is important to note that the selection probabilities \mathbf{p}_{QB} are not known in advance of the DFS contest. Moreover, they do not appear to be perfectly predictable and so we have to explicitly model⁸ their randomness. We do this by assuming instead that the parameter vector $\boldsymbol{\alpha}_{QB}$ in $\mathbb{R}^{N_{QB}}$ is predictable. In particular, we assume

$$\boldsymbol{\alpha}_{QB} = \exp(\mathbf{X}_{QB}\boldsymbol{\beta}_{QB}), \quad (3)$$

where $\boldsymbol{\beta}_{QB}$ is a vector of parameters that we must estimate and \mathbf{X}_{QB} is a matrix (containing N_{QB} rows) of observable independent variables that are related to the specific features of the NFL games and QB's underlying the DFS contest. To be clear, the exponential function in the r.h.s. of (3) is actually an $N_{QB} \times 1$ vector of exponentials.

For example, in a DFS contest for week t , we might assume

$$\boldsymbol{\alpha}_{QB}^t = \exp(\beta_0^{QB}\mathbf{1} + \beta_1^{QB}\mathbf{f}_{QB}^t + \beta_2^{QB}\mathbf{c}_{QB}^t + \beta_3^{QB}\boldsymbol{\mu}_{QB}^t), \quad (4)$$

where \mathbf{f}_{QB}^t is an estimate of \mathbf{p}_{QB} that we can obtain from the FantasyPros website [12], \mathbf{c}_{QB}^t are the (appropriately scaled) costs of the QB's in the contest, and $\boldsymbol{\mu}_{QB}^t$ is an (appropriately scaled) sub-vector of $\boldsymbol{\mu}_\delta$ whose components correspond to the QB positions in $\boldsymbol{\mu}_\delta$. Other features are of course also possible. For example, we might also want to include expected returns $\boldsymbol{\mu}_{QB}^t/\mathbf{c}_{QB}^t$ (where

⁷ Recall that $\mathbf{w}_o \in \{0,1\}^P$ where P is the total number of NFL players available for selection. We can assume then that the first $N_{QB} < P$ components of \mathbf{w}_o correspond to the N_{QB} QB's and so a feasible portfolio will set exactly one of these components to 1 with the remaining $N_{QB} - 1$ components set to 0. The QB component set to 1 corresponds of course to the QB that is selected in the portfolio.

⁸ In initial unreported experiments we assumed \mathbf{p}_{QB} was fixed and known but this led to over-certainty and poor performance of the resulting portfolios.



division is understood to be component-wise), home-away indicators, quality of opponents etc. as features. That said, these latter features should already be accounted for by the features included in (4).

We can estimate the β_{QB} vector by fitting a Bayesian Dirichlet regression. Assuming we have data from weeks $t = 1$ to $t = T - 1$ and a flat prior on β_{QB} , then the posterior satisfies

$$\begin{aligned} p(\beta_{QB} | \{p_{QB}^t\}_{t=1}^{T-1}) &\propto p(\beta_{QB})p(\{p_{QB}^t\}_{t=1}^{T-1} | \beta_{QB}) \\ &\propto \prod_{t=1}^{T-1} \text{Dir}(p_{QB}^t | \exp(X_{QB}^t \beta_{QB})) \\ &\propto \prod_{t=1}^{T-1} \frac{1}{B(\alpha_{QB}^t)} \prod_{k=1}^{N_{QB}} (p_{QB}^t(k))^{\alpha_{QB}^t(k)-1}, \end{aligned}$$

where $B(\alpha_{QB}^t)$ is the normalization factor for the Dirichlet distribution. We fit this model using Bayesian software package STAN [27].

3.2. Generating Random Opponents' Portfolio

Suppose now that the Dirichlet regression model has been fit for each position, i.e. the α_{QB} , α_{RB} , α_{TE} etc. vectors have all been estimated. The question arises as to how we can use them to define a random opponent's portfolio w_o . We note that the fitted Dirichlet regression models provide a straightforward mechanism for defining the *positional marginals* of w_o . For example, let w_o^{QB} denote the $N_{QB} \times 1$ sub-vector of w_o corresponding to the QB's available for selection. We can easily generate a sample of this vector by:

1. First drawing a sample p_{QB} from the $\text{Dir}(\alpha_{QB})$ distribution.
2. Then drawing a single sample from the $\text{Mult}(p_{QB})$ distribution, i.e. multinomial distribution with parameter vector p_{QB} .
3. This draw then defines our chosen QB, i.e. it sets one component of w_o^{QB} to 1 with the others being set to 0.

The obvious approach to defining the distribution of w_o is to assume that each positional marginal vector is independent and therefore drawn according to the 3 steps outlined above with the obvious understanding that α_{RB} would be used for the RB positions, α_{TE} for the TE position etc. This marginal independence assumption then defines the distribution of a *potential* w_o . There is no guarantee, however, that the resulting portfolio is feasible, i.e. that $w_o \in \mathbb{W}$. We therefore use an accept-reject approach whereby potential portfolios w_o are generated according to the steps outlined above and are only accepted if they are feasible. In fact, we impose one further condition: we insist that an accepted w_o uses up most of the available budget. We impose this condition because it is very unlikely in practice that a player in a DFS contest would leave much of his budget unspent. This is purely a behavioral requirement and so we insist the cost of an accepted w_o satisfy $c^T w_o \geq B_{lb}$ for some lower bound $B_{lb} \leq B$ that we get to choose. Algorithm 1 below describes how



a random opponent's \mathbf{w}_o can be generated and therefore, it (implicitly) defines the distribution of \mathbf{w}_o . (We use $\text{Mult}(\cdot, 1)$ to denote a single draw from the multinomial distribution.)

Algorithm 1 describes the case where the positional marginals are independent. It is well known, however, that some sophisticated DFS players intentionally select their portfolios so that the positional marginals of \mathbf{w}_o are *not* independent. We are referring here to the phenomenon of so-called *stacking*, where a portfolio is chosen with a view to maximizing its points variance (as well as its points mean). For example, if a DFS player chooses a QB from a particular team, then it is quite common for the DFS player to also choose the WR from the same team [2]. This is because the realized points of a QB and WR from the same team tend to be strongly correlated and picking the two together will result in a fantasy portfolio with a higher variance (than would otherwise be the case if the chosen QB and WR were from different NFL teams).

We mention here that it is easy to adapt Algorithm 1 to model this stacking behavior and indeed we do this in the numerical experiments of Section 6. For example, once the QB has been chosen we could assume that the WR from the same team is then chosen with some known probability q . Otherwise the WR is chosen as before, i.e. as a draw from the relevant multinomial distribution. Note that in this latter case, stacking might still occur by chance since the positional marginals are assumed to be independent.

Algorithm 1 Sampling O Opponent Portfolios

Require: $(\boldsymbol{\beta}_{\text{QB}}, \dots, \boldsymbol{\beta}_{\text{D}})$, $(\mathbf{X}_{\text{QB}}, \dots, \mathbf{X}_{\text{D}})$, \mathbf{c} , B_{lb}

- 1: $(\boldsymbol{\alpha}_{\text{QB}}, \dots, \boldsymbol{\alpha}_{\text{D}}) = (\exp(\mathbf{X}_{\text{QB}}\boldsymbol{\beta}_{\text{QB}}), \dots, \exp(\mathbf{X}_{\text{D}}\boldsymbol{\beta}_{\text{D}}))$
 - 2: $(\mathbf{p}_{\text{QB}}, \dots, \mathbf{p}_{\text{D}}) \sim (\text{Dir}(\boldsymbol{\alpha}_{\text{QB}}), \dots, \text{Dir}(\boldsymbol{\alpha}_{\text{D}}))$
 - 3: **for** $o = 1:O$ **do**
 - 4: reject = true
 - 5: **while** reject **do**
 - 6: $(k_{\text{QB}}, \dots, k_{\text{D}}) \sim (\text{Mult}(\mathbf{p}_{\text{QB}}, 1), \dots, \text{Mult}(\mathbf{p}_{\text{D}}, 1))$
 - 7: \mathbf{w}_o denotes the portfolio corresponding to $(k_{\text{QB}}, \dots, k_{\text{D}})$
 - 8: **if** $\mathbf{w}_o \in \mathbb{W}$ and $\mathbf{c}^T \mathbf{w}_o \geq B_{\text{lb}}$ **then**
 - 9: reject = false
 - 10: **end if**
 - 11: **end while**
 - 12: **return** $\mathbf{W}_{\text{op}} = \{\mathbf{w}_o\}_{o=1}^O$
-

4. Solving the Double-Up Problem

As mentioned earlier, we first tackle the double-up problem since our solution to this problem will help inform how we approach the top-heavy problem. We begin first by recalling some results from mean-variance optimization and in particular, the problem of maximizing the probability of exceeding a stochastic benchmark.



4.1. Mean Variance Optimization and Outperforming Stochastic Benchmarks

We consider⁹ a one-period problem where at time $t = 0$, there are P financial securities available to invest in. At time $t = 1$, the corresponding random return vector $\xi = (\xi_1, \dots, \xi_P)$ is realized. Let μ_ξ and Σ_ξ denote the mean return vector and variance-covariance matrix, respectively, of ξ . The goal is then to construct a portfolio $\mathbf{w} = (w_1, \dots, w_P)$ with random return $R_{\mathbf{w}} = \mathbf{w}^T \xi$ that maximizes the probability of exceeding a random benchmark r_0 . Mathematically, we wish to solve

$$\max_{\mathbf{w} \in \mathbb{W}} \mathbb{P}\{R_{\mathbf{w}} - r_0 > 0\}, \quad (5)$$

where \mathbb{W} includes the budget constraint $\mathbf{w}^T \mathbf{1} = 1$ as well as any other linear constraints we wish to impose. We assume that $\xi - r_0$ has a multivariate normal distribution so that

$$R_{\mathbf{w}} - r_0 \sim N(\mu_{\mathbf{w}}, \sigma_{\mathbf{w}}^2),$$

for some¹⁰ $\mu_{\mathbf{w}}$ and $\sigma_{\mathbf{w}}^2$ that depend on the portfolio \mathbf{w} . Let \mathbf{w}^* be the optimal solution to (5). Then, the problem in (5) amounts to solving

$$\max_{\mathbf{w} \in \mathbb{W}} 1 - \Phi\left(-\frac{\mu_{\mathbf{w}}}{\sigma_{\mathbf{w}}}\right), \quad (6)$$

where $\Phi(\cdot)$ denotes the standard normal CDF. The following result is adapted from Morton et al. [21] and follows from the representation in (6).

Proposition 4.1. (i) Suppose $\mu_{\mathbf{w}} < 0$ for all $\mathbf{w} \in \mathbb{W}$. Then,

$$\mathbf{w}^* \in \left\{ \mathbf{w}(\lambda): \mathbf{w}(\lambda) \in \underset{\mathbf{w} \in \mathbb{W}}{\operatorname{argmax}} (\mu_{\mathbf{w}} + \lambda \sigma_{\mathbf{w}}^2), \lambda \geq 0 \right\}. \quad (7)$$

(ii) Suppose $\mu_{\mathbf{w}} \geq 0$ for some $\mathbf{w} \in \mathbb{W}$. Then,

$$\mathbf{w}^* \in \left\{ \mathbf{w}(\lambda): \mathbf{w}(\lambda) \in \underset{\mathbf{w} \in \mathbb{W}, \mu_{\mathbf{w}} \geq 0}{\operatorname{argmax}} (\mu_{\mathbf{w}} - \lambda \sigma_{\mathbf{w}}^2), \lambda \geq 0 \right\} \quad (8)$$

so that \mathbf{w}^* is mean-variance efficient.

⁹ The material and results in this subsection follow Morton et al. [21] and they should be consulted for further details and related results. In this subsection, we will sometimes use the same notation from earlier sections to make the connections between the financial problem of this subsection and the DFS problem more apparent.

¹⁰ If the benchmark r_0 is deterministic, then $\mu_{\mathbf{w}} := \mathbf{w}^T \mu_\xi - r_0$ and $\sigma_{\mathbf{w}}^2 := \mathbf{w}^T \Sigma_\xi \mathbf{w}$.



Proposition 4.1 is useful because it allows us to solve the problem in (5) (or equivalently in (6)). In particular, we determine which of the two cases from the proposition applies. This can be done when \mathbb{W} is polyhedral by simply solving a linear program that maximizes $\mu_{\mathbf{w}}$ over $\mathbf{w} \in \mathbb{W}$. If the optimal is negative then we are in case (i); otherwise we are in case (ii). We then form a grid Λ of possible λ values and for each $\lambda \in \Lambda$, we solve the appropriate quadratic optimization problem (defining $\mathbf{w}(\lambda)$) from (7) or (8) and then choose the value of λ that yields the largest objective in (5) or (6). See Algorithm 2 in Section 4.2 below for when we apply these results to our double-up problem.

4.2. The Double-Up Problem

Recall now the double-up problem as formulated in (1). We define $Y_{\mathbf{w}} := \mathbf{w}^T \boldsymbol{\delta} - G(r')$ and assume¹¹ that $Y_{\mathbf{w}} \sim N(\mu_{Y_{\mathbf{w}}}, \sigma_{Y_{\mathbf{w}}}^2)$ where

$$\begin{aligned}\mu_{Y_{\mathbf{w}}} &:= \mathbf{w}^T \boldsymbol{\mu}_{\boldsymbol{\delta}} - \mu_{G(r')} \\ \sigma_{Y_{\mathbf{w}}}^2 &:= \mathbf{w}^T \boldsymbol{\Sigma}_{\boldsymbol{\delta}} \mathbf{w} + \sigma_{G(r')}^2 - 2\mathbf{w}^T \boldsymbol{\sigma}_{\boldsymbol{\delta}, G(r')}$$

where $\mu_{G(r')} = E[G(r')]$, $\sigma_{G(r')}^2 = \text{Var}(G(r'))$ and $\boldsymbol{\sigma}_{\boldsymbol{\delta}, G(r')}$ is a $P \times 1$ vector with p^{th} component equal to $\text{Cov}(\delta_p, G(r'))$. The solution of our double-up problem now follows immediately from Proposition 4.1 and the following discussion. Our procedure is stated formally in Algorithm 2. Note that \mathbf{w}^* in line 10 can be computed using the Monte-Carlo samples $\boldsymbol{\delta}$ and \mathbf{w}_o that are inputs to the algorithm. Alternatively, \mathbf{w}^* could also be computed using the normal approximation assumption.

Generating Monte-Carlo Samples

In order to execute Algorithm 2, we must first compute the inputs $\mu_{G(r')}$, $\sigma_{G(r')}^2$, $\boldsymbol{\sigma}_{\boldsymbol{\delta}, G(r')}$ as defined above. These quantities can be estimated off-line via Monte-Carlo simulation as they do not depend on our portfolio choice \mathbf{w} . We simply note here that the Monte-Carlo can be performed efficiently¹² using results (see [7]) from the theory of order statistics. These results rely on letting $O \rightarrow \infty$ and so our Monte-Carlo algorithm is most suited for large DFS contests (which happen to be the contests of most interest anyway). Nonetheless, we can still use the Monte-Carlo algorithm as an approximation for medium-sized contests, and if O is small, then we can simply simulate directly without relying on these results.

Solving the Binary Quadratic Program

The optimization problems in lines 3 and 7 of Algorithm 2 require the solution of binary quadratic programs (BQP's). We solve them using Gurobi's [16] default BQP solver. In our experiments, all such problem instances were successfully solved to optimality with required computation time

¹¹ We justified the normal approximation for $Y_{\mathbf{w}}$ via Monte-Carlo simulation. Specifically, we generated many samples of $Y_{\mathbf{w}}$ corresponding to various portfolios \mathbf{w} . These portfolios included \mathbf{w}^* as well as highly ranked portfolios from DFS contests in the current NFL season. In each case, we found the distribution of $Y_{\mathbf{w}}$ to be unimodal and approximately symmetric about its mean.

¹² The specific details are described in Appendix A.



varying with P (number of real-world players), λ and the contest structure (double-up, quintuple-up or top-heavy). A typical BQP problem instance took anywhere from a fraction of a second to a few hundred seconds to solve on a shared high-performance computing (HPC) cluster.

Algorithm 2 Optimization Algorithm for the Double-Up Problem with a Single Entry

Require: \mathbb{W} , Λ , μ_δ , Σ_δ , $\mu_{G(r')}$, $\sigma_{G(r')}^2$, $\sigma_{\delta, G(r')}$ and Monte-Carlo samples of δ and \mathbf{w}_o

```

1: if  $\exists \mathbf{w} \in \mathbb{W}$  with  $\mu_{Y_{\mathbf{w}}} \geq 0$  then
2:   for all  $\lambda \in \Lambda$  do
3:      $\mathbf{w}_\lambda = \operatorname{argmax}_{\mathbf{w} \in \mathbb{W}, \mu_{Y_{\mathbf{w}}} \geq 0} (\mu_{Y_{\mathbf{w}}} - \lambda \sigma_{Y_{\mathbf{w}}}^2)$ 
4:   end for
5: else
6:   for all  $\lambda \in \Lambda$  do
7:      $\mathbf{w}_\lambda = \operatorname{argmax}_{\mathbf{w} \in \mathbb{W}} (\mu_{Y_{\mathbf{w}}} + \lambda \sigma_{Y_{\mathbf{w}}}^2)$ 
8:   end for
9: end if
10:  $\lambda^* = \operatorname{argmax}_{\lambda \in \Lambda} \mathbb{P}\{Y_{\mathbf{w}_\lambda} > 0\}$ 
11: return  $\mathbf{w}_{\lambda^*}$ 

```

4.3. The Double-Up Problem with Multiple Entries

We briefly consider now the case where we can submit a fixed but finite number of $N > 1$ entries to the double-up contest. In the case of a risk-neutral DFS player, it should be intuitively clear¹³ that if $O \rightarrow \infty$ so that $N/O \rightarrow 0$, then the optimal strategy is a *replication* strategy. In particular, the DFS player should solve for the $N = 1$ optimal portfolio and then submit N copies of this portfolio to the contest. Even when O is not large then we suspect the replication strategy may still be close to optimal. The key issue would be the probability of having a portfolio exactly at the cutoff $G(r')$ between receiving and not receiving the cash payoff.

Finally, we note that a risk-averse DFS player would never (ignoring certain pathological cases) want to use a replication strategy and would prefer instead a more diversified portfolio of entries. Such a diversified portfolio can easily be constructed using the heuristics we outline in Section 5.1 for the case of top-heavy contests.

5. Solving the Top-Heavy Problem

¹³ Certainly we would need to impose some technical conditions on $\{\mathbf{w}_o\}_{o=1}^O$ to make this claim rigorous. We will not discuss such technical conditions, however, since top-heavy contests are our main focus in this paper and such a replication result does not hold for these contests. See Section 5 for further discussion on this issue.



We can now extend the analysis we developed for the double-up problem in Section 4 to tackle the more interesting top-heavy problem. We consider first the single-entry case where $N = 1$. In that case, the problem in (2) simplifies to solving

$$\max_{\mathbf{w} \in \mathbb{W}} \sum_{d=1}^D (R_d - R_{d+1}) \mathbb{P}\{\mathbf{w}^T \boldsymbol{\delta} > G^{(r'_d)}(\mathbf{W}_{\text{op}}, \boldsymbol{\delta})\}, \quad (9)$$

where $r'_d = 0 + 1 - r_d 0$. Following the development in Section 4.2, we can define $Y_w^d := \mathbf{w}^T \boldsymbol{\delta} - G^{(r'_d)}$ and assume $Y_w^d \sim N(\mu_{Y_w^d}, \sigma_{Y_w^d}^2)$ with

$$\mu_{Y_w^d} := \mathbf{w}^T \boldsymbol{\mu}_{\boldsymbol{\delta}} - \mu_{G^{(r'_d)}} \quad (10)$$

$$\sigma_{Y_w^d}^2 := \mathbf{w}^T \boldsymbol{\Sigma}_{\boldsymbol{\delta}} \mathbf{w} + \sigma_{G^{(r'_d)}}^2 - 2\mathbf{w}^T \boldsymbol{\sigma}_{\boldsymbol{\delta}, G^{(r'_d)}}, \quad (11)$$

where $\mu_{G^{(r'_d)}} = E[G^{(r'_d)}]$, $\sigma_{G^{(r'_d)}}^2 = \text{Var}(G^{(r'_d)})$ and $\boldsymbol{\sigma}_{\boldsymbol{\delta}, G^{(r'_d)}}$ is a $P \times 1$ vector with p^{th} component equal to $\text{Cov}(\delta_p, G^{(r'_d)})$. We can now write (9) as

$$\max_{\mathbf{w} \in \mathbb{W}} \sum_{d=1}^D (R_d - R_{d+1}) \left(1 - \Phi \left(-\frac{\mu_{Y_w^d}}{\sigma_{Y_w^d}} \right) \right). \quad (12)$$

Before proceeding, we need to make two additional assumptions, which we will state formally.

Assumption 5.1. $\mu_{Y_w^d} < 0$ for $d = 1, \dots, D$ and for all $\mathbf{w} \in \mathbb{W}$.

Assumption 5.1 can be interpreted as stating that, in expectation, the points total of our optimal portfolio will not be sufficient to achieve the minimum payout R_D . In option-pricing terminology, we are therefore assuming our optimal portfolio is “out-of-the-money”. This is a very reasonable assumption to make for top-heavy contests where it is often the case that only the top 20% or so of entries earn a cash payout. In numerical experiments, our model often predicts that our optimal portfolio will (in expectation) be at or around the 20th percentile. The assumption therefore may break down if payoffs extend beyond the top 20% of entries. Nonetheless, the payoff sizes around the 20th percentile are very small and almost negligible. Indeed within our model most of the expected P&L comes from the top few percentiles and $\mu_{Y_w^d} < 0$ is certainly true for these values of d . Finally, we note the well-known general tendency of models to over-estimate the performance of an optimally chosen quantity (in this case our portfolio). We therefore anticipate that our optimal portfolio will not quite achieve (in expectation) the 20th percentile and may well be out of the money for all payoff percentiles as assumed in Assumption 5.1.

Given Assumption 5.1, it follows that each of the arguments $-\mu_{Y_w^d}/\sigma_{Y_w^d}$ to the normal CDF terms in (12) are *positive*. Given the objective in (12) is to maximize, it is also clear that for a fixed value of



$\mathbf{w}^T \boldsymbol{\mu}_\delta$ in (10), we would like the standard deviation σ_{Y_d} to be as large as possible. Unfortunately, the third term, $2\mathbf{w}^T \boldsymbol{\sigma}_{\delta, G^{(r'_d)}}$, on the r.h.s. of (11) suggests that \mathbf{w} impacts the variance by a quantity that depends on d . The following assumption enables us to circumvent this problem.

Assumption 5.2. $Cov(\delta_p, G^{(r'_d)}) = Cov(\delta_p, G^{(r'_{d'})})$ for all $d, d' = 1, \dots, D$.

Assumption 5.2 seems very reasonable based on our numerical experiments with real-world DFS top-heavy contests. In these contests, we found these covariance terms to be very close to each other for values of d corresponding to the top 20 percentiles and in particular for the top few percentiles. We therefore proceed to take Assumption 5.2 as given. It is then clear from (11) that the impact of \mathbf{w} on $\sigma_{Y_d}^2$ does not depend on d .

Given the preceding arguments, it should be clear that for any fixed value of $\mathbf{w}^T \boldsymbol{\mu}_\delta$, we would like to make $\mathbf{w}^T \boldsymbol{\Sigma}_\delta \mathbf{w} - 2\mathbf{w}^T \boldsymbol{\sigma}_{\delta, G^{(r'_d)}}$ (the terms from (11) that depend on \mathbf{w}) as large as possible. We are therefore in the situation of part (i) of Proposition 4.1 and we have a simple algorithm for solving the top-heavy problem. This is given in Algorithm 3 below where we omit the dependence on d of those terms that are assumed (by Assumption 5.2) to not vary with d .

Algorithm 3 Optimization Algorithm for the Top-Heavy Problem with a Single Entry

Require: \mathbb{W} , Λ , $\boldsymbol{\mu}_\delta$, $\boldsymbol{\Sigma}_\delta$, $\boldsymbol{\sigma}_{\delta, G^{(r'_)}}$ and Monte-Carlo samples of $\boldsymbol{\delta}$ and \mathbf{w}_o

1: **for all** $\lambda \in \Lambda$ **do**

2: $\mathbf{w}_\lambda = \operatorname{argmax}_{\mathbf{w} \in \mathbb{W}} \left(\mathbf{w}^T \boldsymbol{\mu}_\delta + \lambda \left(\mathbf{w}^T \boldsymbol{\Sigma}_\delta \mathbf{w} - 2\mathbf{w}^T \boldsymbol{\sigma}_{\delta, G^{(r'_)}} \right) \right)$

3: **end for**

4: $\lambda^* = \operatorname{argmax}_{\lambda \in \Lambda} \sum_{d=1}^D (R_d - R_{d+1}) \mathbb{P} \left\{ \mathbf{w}^T \boldsymbol{\delta} > G^{(r'_d)}(\mathbf{W}_{\text{op}}, \boldsymbol{\delta}) \right\}$

5: **return** \mathbf{w}_{λ^*}

As with Algorithm 2, the \mathbf{w}_λ 's are computed by solving BQP's and the optimal \mathbf{w}^* from line 4 can then be determined via the Monte-Carlo samples.

5.1. The Top-Heavy Problem with Multiple Entries

Consider now the more general top-heavy problem where we must submit N entries or portfolios. We repeat again the problem formulation from Section 2.2:

$$\max_{\mathbf{w} \in \mathbb{W}^N} \sum_{i=1}^N \sum_{d=1}^D (R_d - R_{d+1}) \mathbb{P} \left\{ \mathbf{w}_i^T \boldsymbol{\delta} > G_{-i}^{(r'_d)}(\mathbf{W}_{-i}, \mathbf{W}_{\text{op}}, \boldsymbol{\delta}) \right\}, \quad (13)$$

where $r'_d = O + N - r_d O$, $G_{-i}^{(r)}$ is the r^{th} order statistic of $\{G_o\}_{o=1}^O \cup \{F_j\}_{j=1}^N \setminus F_i$ and $\mathbf{W}_{-i} := \mathbf{W} \setminus \mathbf{w}_i$. The first thing to note is that, unlike the double-up format, there is no reason to believe that a



portfolio replication strategy should ever be optimal here. To see this, consider the winner-takes-all contest where only the entry with the most fantasy points wins. This is clearly a special-case¹⁴ of the top-heavy format. Under some mild assumptions, e.g. the probability of a winning tie being very small, it is clear that the goal is to maximize the probability of having a winning entry. This probability can only be increased by diversifying and so replication is (in general) strictly sub-optimal.

For more general top-heavy contests that are not winner-takes-all contests, it is not clear whether or not some form of replication should take place. For example, consider the case where $N = 2$ and we are extremely skilled relative to all of our opponents. If the contest pays out to the top 20% of entries say, and O is very large, then it may well be the case that submitting the same optimal entry twice is the best strategy. In general, solving the multiple entry top-heavy problem is difficult and so we propose using a greedy approach to tackle it. In particular, we will follow¹⁵ the greedy approach of Hunter et al. [18] and also impose constraints on our entries that ensure a diversified portfolio of entries. In particular, we:

1. Solve for the optimal $N = 1$ entry. Call this entry \mathbf{w}_1^* .
2. For $i = 2, \dots, N$, we again solve for the $N = 1$ problem but with a new constraint set \mathbb{W}_i defined as

$$\mathbb{W}_i := \mathbb{W} \cup \{\mathbf{w}_i^T \mathbf{w}_j^* \leq \gamma \quad \forall j = 1, \dots, i - 1\}$$

where γ is a pre-specified parameter satisfying $\gamma \leq C$. Let \mathbf{w}_i^* denote the optimal solution.

Note that the effect of γ is to ensure that portfolio i can not have more than γ players in common with each of the previous $i - 1$ portfolios. We mention in passing that in our numerical results of Section 6, we found this diversification heuristic for top-heavy contests to have a higher expected P&L than the replication strategy, which simply submits N copies of the optimal portfolio for the single-entry problem. (This, not surprisingly, was not the case for our double-up contests.)

Remark 5.1. *While we have taken N as given up to this point, it is perhaps worth mentioning that one can always use this heuristic to determine an “optimal” value of N . Specifically, we can continue to increase N until the expected P&L contribution from the next entry (determined by our heuristic above) goes negative. We note that it is easy to estimate the P&L of any portfolio of entries via Monte-Carlo simulation.*

6. Numerical Experiments

¹⁴ The winner-takes-all contest can also be viewed as a pathological case of the double-up format where replication is definitely not an optimal strategy for the reasons outlined in this subsection.

¹⁵ Hunter et al. [18] provide some performance guarantees in their setting for the greedy approach in the winner-takes all setting. Those guarantees would also apply here.

We participated in real-world DFS contests on FanDuel during the 2017-18 NFL season. As of writing, the season is still ongoing and so we present results for the first twelve weeks. Each week, we participated in three contests: top-heavy, quintuple-up and double-up. The cost per entry was \$1 in top-heavy and \$2 in both quintuple-up and double-up contests. The number of opponents O was approximately 200,000, 10,000 and 30,000 for the three contests, respectively, with these numbers varying by around 10% from week-to-week. In the top-heavy contest, the payoff¹⁶ for rank 1 was around \$5,000, for rank 2 was around \$2,500, declining quickly to around \$100 for rank 30. The lowest winning rank was around 50,000, with a payoff of \$2.

We used two different models for each contest. The first model (which we used across all contests) is the *strategic* model that we introduced in Section 4 for double-up / quintuple-up contests and in Section 5 for top-heavy contests. For all contests, we used the diversification heuristic¹⁷ instead of the replication strategy for $N > 1$. The second model is a *benchmark* model that does not model opponents and hence, is not strategic. For each model, we submitted $N = 50, 25$ and 10 entries to top-heavy, quintuple-up and double-up contests, respectively each week.

6.1. Benchmark Models

Our two benchmark models do not model opponents and in fact, they (implicitly) assume the benchmarks $G^{(r')}$ or $G^{(r'_d)}$ are deterministic and simply maximize the expected points total in the objective. First, we describe the benchmark model for double-up and then, the benchmark model for top-heavy.

Benchmark Model 1

To optimize in the $N = 1$ case, our first benchmark model simply maximizes the expected points total subject to the feasibility constraints on the portfolio. The resulting optimization model is a binary program (BP):

$$\max_{w \in W} w^T \mu_\delta. \quad (14)$$

For $N > 1$ (which is the case in our numerical experiments), we employ the greedy heuristic discussed in Section 5.1 but suitably adapted for the case where we do not model opponents. We use this benchmark model for the double-up contest because, according to our calibrated model, we

¹⁶ We note that there are other top-heavy contests with even more competitors and payoff structures that are even more "top-heavy". For example, a regular contest on FanDuel often has approximately 400,000 entries with a top payoff of \$250,000 to \$1,000,000, which declines quickly to a payoff of \approx \$500 for the 50th rank. Top-heavy contests are therefore extremely popular and hence are our principal focus in this paper.

¹⁷ This makes sense as previously discussed for top-heavy contests. The reason for doing so in the double-up / quintuple-up contests was simply to reduce the variance of P&L. This is discussed further in Section 6.3.

are comfortably in the case (ii) scenario of Proposition 4.1 where, other things being equal, we prefer less variance to more variance.

Benchmark Model 2

The second benchmark model is similar to the first and indeed the objective functions are identical. The only difference is that we add a stacking constraint to force the model to pick the QB and main WR¹⁸ from the same team. We denote this constraint as “QB-WR”. Mathematically, the resulting BP for $N = 1$ is:

$$\max_{w \in W, \text{QB-WR}} w^T \mu_\delta. \quad (15)$$

Again for $N > 1$, we employ a suitably adapted version of the greedy heuristic from Section 5.1. As discussed in Section 3.2, the purpose of the stacking constraint is to increase the portfolio's variance. This is because we are invariably “out-of-the-money” in these contests as we noted in Section 5 and so variance is preferred all other things, i.e. expected number of points, being equal. We note this model is very similar to the model proposed by Hunter et al. [18] for hockey contests. They presented several variations of their model typically along the lines of including more stacking (or anti-stacking¹⁹) constraints, e.g. choosing players from exactly 3 teams to increase portfolio variance. We note that we could easily construct and back-test other similar benchmark strategies as well but for the purposes of our experiments, the two benchmarks above seemed reasonable points of comparison.

6.2. Parameters

Our models rely on the following four input parameters²⁰: the expected fantasy points of the real-world players μ_δ , the corresponding variance-covariance matrix Σ_δ , the stacking probability q from Section 3.2 and the diversification parameter γ from Section 5.1.

¹⁸ By “main” WR of a team, we refer to the WR with the highest μ_δ value among all the WRs in the same team.

¹⁹ An example of an anti-stacking constraint in hockey is that the goalie of team A cannot be selected if the attacker of team B was selected and teams A and B are playing each other in the series of games underlying the DFS contest in question. Such an anti-stacking constraint is also designed to increase variance by avoiding players whose fantasy points would naturally be negatively correlated.

²⁰ In addition to these four input parameters, we need the input features X and the realized p values for the Dirichlet regression. Such data is available on the internet. For example, the f feature (point estimate of p) is available at the FantasyPros website and FanDuel contains the cost vector c (before a contest starts) and the realized positional marginals p (after a contest is over). We note that accessing the positional marginals data at FantasyPros required us to create an account and pay for a six-month subscription costing \$65.94.



We obtain the estimate of μ_δ from FantasyPros [12]. This estimate is specific to each week's games and we normally obtained it a day before the NFL games were played. We decompose the variance-covariance matrix Σ_δ into the correlation matrix ρ_δ and the standard deviations of the individual players $\sigma_\delta \in \mathbb{R}^P$. The estimate of ρ_δ is obtained from RotoViz [26] and σ_δ is estimated using the realized δ values from the 2016-17 and the ongoing 2017-18 seasons. Note that ρ_δ does not change from week to week whereas σ_δ is updated weekly using the realized δ from the previous week.

For the stacking probability q , we first note that we expect it to be contest-specific as we anticipate more stacking to occur in top-heavy style contests where variance is relatively more important than in double-up contests. Accordingly, we empirically checked the proportion of opponents who stacked using data from the 2016-17 season for each contest-type. We then calibrated q to ensure that our Dirichlet-multinomial model for generating opponents implied the same proportion (on average). We estimated q to be 0.35, 0.25 and 0.20 for top-heavy, quintuple-up and double-up contests, respectively. Finally, we set²¹ $\gamma = 6$ for the strategic and benchmark models across all contests.

6.3. Main Results

We now discuss the P&L-related results for the strategic and benchmark models across the three contest structures for the first twelve weeks of the FanDuel DFS contests in the ongoing 2017-18 NFL season. Figure 2 and Table 1 display the cumulative realized P&L for both models across the three contest structures during the season. The strategic portfolio has out-performed the benchmark portfolio since inception in the top-heavy series of contests. The strategic portfolio has earned a cumulative profit of \$384.74, which is ≈ 2.5 times the realized P&L of the benchmark portfolio. Moreover, the maximum cumulative loss, i.e. the max shortfall, for the strategic portfolio is just \$18.5. In addition, the small initial investment of \$50 has been sufficient to fund the strategic portfolio throughout the twelve weeks. This suggests a profit of \$384.74 on an initial investment of \$50, i.e. a return of over 750% in just 12 weeks. In contrast, the benchmark portfolio needed more capital than the initial investment of \$50. If we account for this additional required capital, then the benchmark portfolio has earned a return of less than 100% in 12 weeks. Note that given the so-called house-edge of approximately 15%, both models have performed considerably better than the average portfolio which would have lost $\approx 12 \times 15\% \times 50 = \90 across the 12 weeks.

With regards to the quintuple-up series, the strategic model was better until the end of week 6 but since then the benchmark portfolio has out-performed it. We note, however, that the current difference in the cumulative P&L between the two models is almost the same as the difference at the end of week 6, implying that one week is enough to flip the story. A similar situation applies to the double-up series.

We believe the realized P&L to-date for each contest series is actually conservative and that superior performance (in expectation) could easily be attained. There are at least three reasons for this belief. First, we used off-the-shelf estimates of the input parameters μ_δ and Σ_δ , which are clearly vital to the optimization model. Moreover, we obtained the μ_δ estimate a day before the

²¹ We found this value of γ to produce a near maximum within-model expected P&L.



actual NFL games started and mostly ignored the developments in the last few hours preceding the games, which can be very important in football. For example, in week 7, the main RB of the Jacksonville Jaguars (Leonard Fournette) was questionable to play. Accordingly, their second main RB (Chris Ivory) was expected to play more time on the field. However, our μ_δ estimate did not reflect this information. Our estimate projected 17.27 and 6.78 fantasy points for Fournette and Ivory, respectively. Moreover, since FanDuel sets the price of the players a few days before the games take place, Fournette was priced at 9000 and Ivory at 5900. There was a clear benefit of leveraging this information as Fournette was over-priced and Ivory was under-priced. In fact, our opponents exploited this opportunity as around 60% of them (in double-up) picked Ivory. A proactive user would have updated his μ_δ estimate following such news. In fact, the sharks do react to such last-minute information [23], meaning that we were at a disadvantage by not doing so.

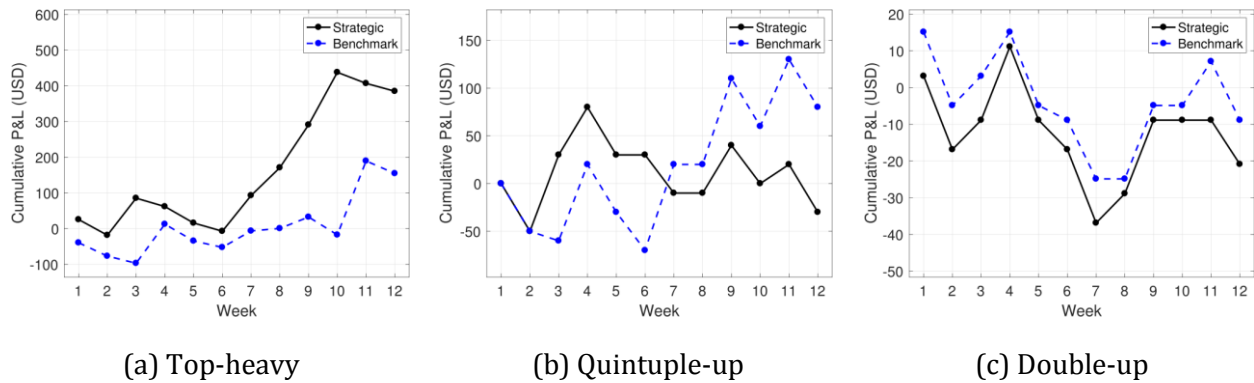


Figure 2: Cumulative realized dollar P&L for the strategic and benchmark models across the three contest structures for the first twelve weeks of the FanDuel DFS contests in the 2017-18 NFL season.

Table 1: **Cumulative realized dollar P&L** for the strategic and benchmark models across the three contest structures for the first twelve weeks of the FanDuel DFS contests in the 2017-18 NFL season. We invested \$50 per week per model in top-heavy series with each entry costing \$1. In quintuple-up the numbers were \$50 per week per model with each entry costing \$2 and in double-up we invested \$20 per week per model with each entry costing \$2. (We did not participate in the quintuple-up contest in week 1 due to logistical reasons.)

Week	Top-heavy		Quintuple-up		Double-up	
	Strategic	Benchmark	Strategic	Benchmark	Strategic	Benchmark
1	25.5	-39.5	-	-	3.13	15.13
2	-18.5	-77	-50	-50	-16.87	-4.87
3	85.24	-97	30	-60	-8.87	3.13
4	61.74	12.5	80	20	11.13	15.13
5	15.74	-34.5	30	-30	-8.87	-4.87
6	-7.26	-52.5	30	-70	-16.87	-8.87
7	92.74	-6.5	-10	20	-36.87	-24.87
8	170.74	0.5	-10	20	-28.87	-24.87
9	290.74	32.5	40	110	-8.87	-4.87
10	437.74	-17.5	0	60	-8.87	-4.87
11	406.74	189.5	20	130	-8.87	7.13
12	384.74	154.5	-30	80	-20.87	-8.87



The second reason is simply a variance issue in that a large number of DFS contests (and certainly much greater than 12) will be required to fully establish the out-performance of the strategic model in general. In fact, we believe the variance of the cumulative P&L is particularly high for NFL DFS contests. There are several reasons for this. Certainly the individual performance of an NFL player in a given week will have quite a high variance due to the large roster size²² as well as the relatively high probability of injury. This is in contrast to other DFS sports where there is considerably more certainty over the playing time of each athlete. To give but one example, in week 5 we witnessed a series of injuries that impacted many of our submitted portfolios (both strategic and benchmark). Devante Parker (Miami Dolphins) was injured in the first quarter but was picked by 56 of our entries. Charles Clay (Buffalo Bills) and Sterling Shepard (NY Giants) were injured before halftime, affecting 70 and 4 entries, respectively. Bilal Powell (NY Jets) and Travis Kelce (Kansas City Chiefs) left the field close to the halftime, impacting 44 and 25 entries, respectively. Furthermore, the NFL season consists of just 16 games per team whereas teams in sports such as basketball, ice hockey and baseball play 82, 82 and 162 games, respectively, per season. As a result, we believe the cumulative P&L from playing DFS contests over the course of an NFL season will have a very high variance relative to these other sports.

The third reason applies specifically to the quintuple up contests and is quite an interesting case. In our strategic model for quintuple-up, there is a possibility of incorrectly minimizing portfolio variance when we should in fact be maximizing it (along with expected number of points of course). Proposition 4.1 leads us to try and increase variance if $\mu_w < 0$ for all $w \in \mathbb{W}$ and to try and decrease variance otherwise. But μ_w must be estimated via Monte-Carlo and is of course also model-dependent. As such, if we estimate a value of $\mu_w \approx 0$, it is quite possible we will err and increase variance when we should decrease it and vice versa. We suspect this occurred on regular basis with the quintuple-up contests, where we often obtained an estimate of μ_w that was close to zero. We note that the benchmark portfolio is always long expected points and variance of points. This problem should be easy to fix by introducing a threshold value of μ_w (presumably greater than 0) below which we are led to increase variance. One could back-test to find a good choice of this threshold.

Figure 3 displays the P&L distribution for the diversification strategy from Section 5.1 for both strategic and benchmark portfolios in week 10²³ contests. We note this P&L distribution is as determined by our model and therefore assumes the Gaussian model for NFL players and the Dirichlet-multinomial model for opponents portfolios. The strategic model dominates the benchmark model in terms of expected profit. In top-heavy, the expected profit of the strategic portfolio is over 5 times that of the benchmark portfolio. The gain is not as drastic in quintuple-up and double-up contests. The substantial gain in top-heavy seems to come from the fact that the

²² There are more than 45 players on a roster but only 11 on the field at any one time.

²³ Other weeks have similar results as shown in Figure 5.

strategic portfolio has a considerably more mass in the right-tail. Note this leads to the higher standard deviation of the top-heavy strategic portfolio²⁴.

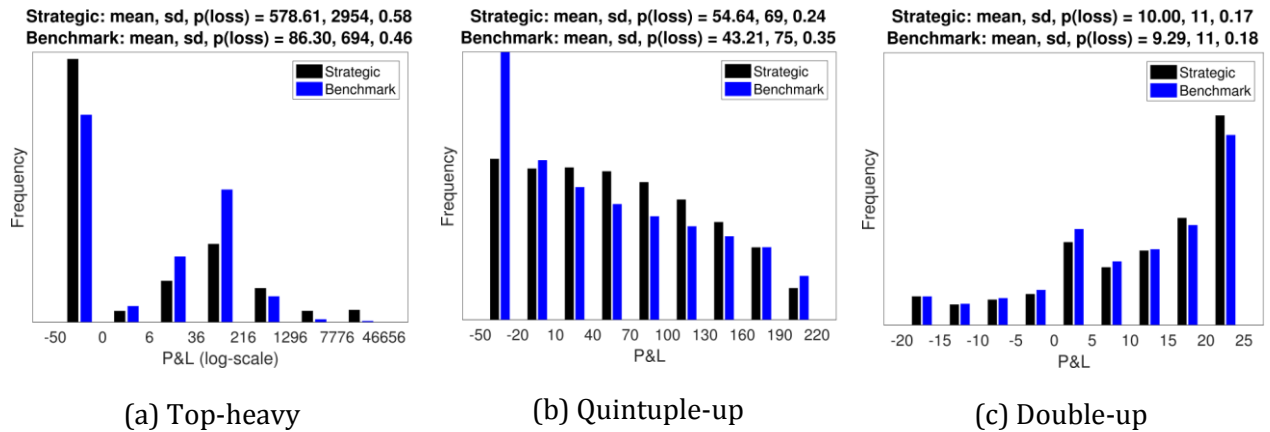


Figure 3: P&L distribution for the diversification strategy for the strategic and benchmark portfolios for week 10 contests of the 2017-18 NFL season. Recall $N = 50, 25$ and 10 for top-heavy, quintuple-up and double-up, respectively. The three metrics at the top of each image are the expected P&L, the standard deviation of the P&L and the probability of loss, i.e. $\mathbb{P}(P\&L < 0)$.

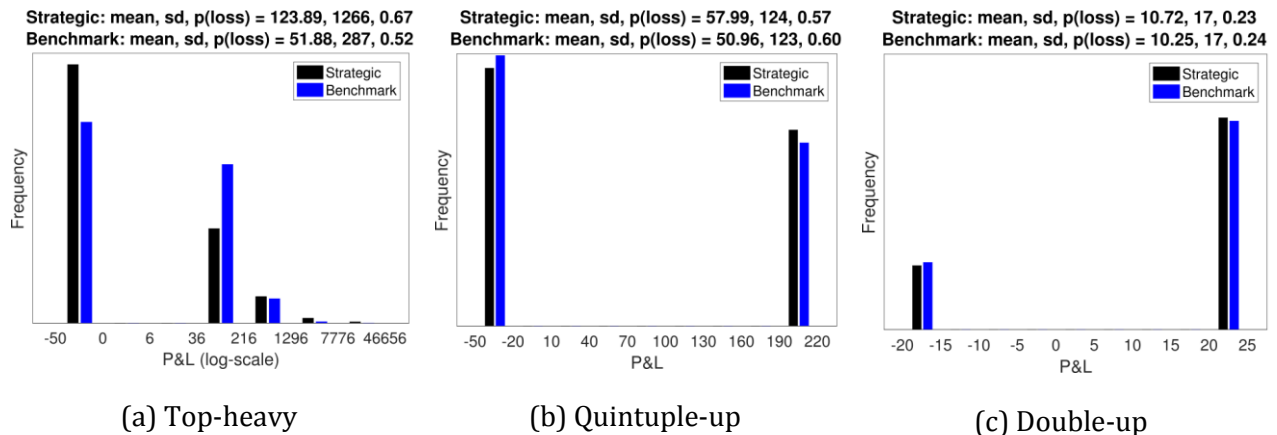


Figure 4: P&L distribution for the replication strategy for the strategic and benchmark portfolios for week 10 contests of the 2017-18 NFL season. Recall $N = 50, 25$ and 10 for top-heavy, quintuple-up and double-up, respectively. The three metrics at the top of each image are the expected P&L, the standard deviation of the P&L and the probability of loss, i.e. $\mathbb{P}(P\&L < 0)$.

Figure 4 is similar to Figure 3 except it is based upon using the replication strategy from Section 4.3 instead of the diversification strategy. We note the strategic model continues to have a higher

²⁴ The high standard deviation in the top-heavy strategic portfolio should be seen as a pro instead of a con, since it is mostly coming from the right-tail of the P&L distribution.

expected P&L than the benchmark model. The main observation here is that the expected P&L drops considerably when we go from the diversification heuristic to the replication strategy for top-heavy. This is consistent with the winner-takes-all story we discussed in Section 5.1, where we argued that replication was in general clearly suboptimal. In contrast, the P&L increases for both quintuple-up and double-up when we employ the replication strategy. Again, this is consistent with our earlier argument in favor of replication for double-up style contests. In our numerical experiments, however, we used the diversification strategy for both double-up and quintuple-up contests. This was only because of the variance issue highlighted earlier and our desire to use a strategy, which had a considerably smaller standard deviation (while ceding only a small amount of expected P&L). As can be seen from Figures 3 and 4, the diversification strategy has (as expected) a smaller expected P&L as well as a smaller probability of loss.

Figure 5 displays the realized and expected P&L's. For both strategic and benchmark models and all three contests, the expected profit is greater than the realized profit. This is perhaps not too surprising given the bias that results from optimizing within a model. In top-heavy, however, the realized P&L is within one standard deviation of the expected P&L although this is not the case for the quintuple- and double-up contests. As discussed above, we believe our realized results are conservative and that a more proactive user of these strategies who makes a more determined effort to estimate μ_δ and Σ_δ and responds to relevant news breaking just before the games can do even better. Despite this potential for improvement, the strategic model has performed very well overall. The small losses from the double-up and quintuple-up contests have been comfortably offset by the gains in the top-heavy contests. As we noted earlier, the return on investment in top-heavy is over 750% for a twelve week period although we do acknowledge there is considerable variance in this number as evidenced by Figure 5(a).

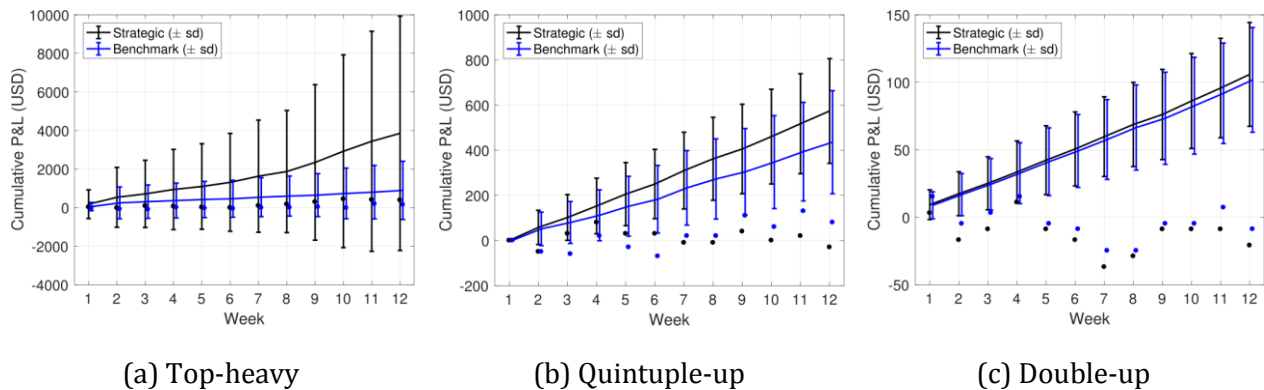


Figure 5: Predicted and realized cumulative P&L for the strategic and benchmark models across the three contest structures for the first twelve weeks of the FanDuel DFS contests in the 2017-18 NFL season. The realized cumulative P&L's are displayed as points

7. The Value of Modeling Opponents and Insider Trading



In unreported numerical simulations as well as the numerical results of Section 6, we found that modeling opponents' behavior can significantly increase the expected P&L from participating in DFS contests. But many interesting questions remain open and we briefly describe some of them here.

1. How much value is there in modeling opponents? Clearly this is contest-dependent and we have seen in Section 6 that our framework, which explicitly models opponents, clearly does much better (in terms of expected P&L) than the benchmark models that do not consider opponents. But the value in modeling opponents also depends on how *accurately* we model their behavior. On this latter point it would be of interest to consider:
 - a. How much do we gain if we use a deterministic $(\mathbf{p}_{QB}, \dots, \mathbf{p}_D)$? For example, in the NFL contests, we could set \mathbf{p} equal to the values predicted by the FantasyPros website.
 - b. How much additional value is there if instead we assume $(\mathbf{p}_{QB}, \dots, \mathbf{p}_D) \sim (\text{Dir}(\boldsymbol{\alpha}_{QB}), \dots, \text{Dir}(\boldsymbol{\alpha}_D))$ as in Algorithm 1 but now $\boldsymbol{\alpha}_{QB}, \dots, \boldsymbol{\alpha}_D$ are constant vectors and do not vary with the features $\mathbf{X}_{QB}, \dots, \mathbf{X}_D$ as assumed in line 1 of the algorithm?
 - c. Finally, how much additional value is there to be gained by assuming the model of Algorithm 1 where $\boldsymbol{\alpha}_{QB}, \dots, \boldsymbol{\alpha}_D$ depend on the features $\mathbf{X}_{QB}, \dots, \mathbf{X}_D$? And just how well can we do if we could find a set of "optimal" features?
2. A question that is somewhat dual to the first question concerns the issue of insider trading and the value of information. This question received considerable attention in 2015 [9, 10] when a DraftKings employee used data from DraftKings contests to enter a FanDuel DFS contest in the same week and win \$350,000. Without addressing the specific nature of insider trading in that case, we can pose several questions:
 - a. How much does the insider gain if he knows the entries of all contestants, i.e. \mathbf{W}_{op} ? In that case, the only uncertainty in the system is the performance vector $\boldsymbol{\delta}$ of the real-world players. (Note that the problem of computing an optimal portfolio given full knowledge of \mathbf{W}_{op} is straightforward in our framework.)
 - b. How much does the insider gain if he only knows the true positional marginals $\mathbf{p} = (\mathbf{p}_{QB}, \dots, \mathbf{p}_D)$ as opposed to knowing \mathbf{W}_{op} ?

We note the answers to the second question depend on how well the non-insider can predict opponents' behavior. For example, the more sophisticated the non-insider is, then the less value there is to having inside information. We are currently investigating these issues and will present our results in an extended version of this paper.

8. Conclusions and Further Research

In this paper, we have developed a new framework for constructing portfolios for both double-up and top-heavy DFS contests. Our methodology explicitly accounts for the behavior of DFS opponents and leverages mean-variance theory (for the out-performance of stochastic benchmarks) to develop a tractable algorithm that requires solving a series of binary quadratic

programs. Following Hunter et al. [18], we also provide a tractable greedy heuristic for handling the multiple entry, i.e. $N > 1$, case for top-heavy style contests. This is in contrast to the replication approach we advocate for double-up style contests.

There are many potential directions for future research. First, the numerical results of Section 6 are preliminary as they are based on the ongoing 2017-18 NFL season. They must therefore be updated at the end of the season. The data we are currently gathering for these numerical experiments can be used to back-test other benchmark strategies as well as refining our own preferred strategies. For example, in Section 6.3, we discussed the possibility our strategic model might be incorrectly minimizing portfolio variance in quintuple-up. A more robust approach is therefore called for and it would be interesting to optimize over the threshold value of μ_w below which we are led to increase variance.

While NFL contests are among the most popular DFS contests, the season is quite short with only 17 rounds of games. Moreover, as mentioned in Section 6.3, the individual performance of an NFL player in a given week has quite a high variance, potentially causing the cumulative P&L in football DFS contests to be high relative to other sports such as basketball, ice hockey and baseball. For these reasons and others, it would be interesting to apply our modeling framework to DFS contests in these other sports. It would also be interesting to use domain knowledge of these other sports to actively update estimates of μ_δ and Σ_δ as the round of games approaches. This is something we did not do in the current NFL season. Indeed we recorded many instances when it would have been possible to avoid certain players in our DFS entries had we used up-to-date information that was available before the games in question and before our entries needed to be submitted. As a result, we believe the net positive P&L achieved by our models is very encouraging and can easily be improved (in expectation) by more active monitoring of the players.

It would also be interesting to further develop our modeling and estimation approach for a random opponent's portfolio w_o . We assumed in Section 3 that we had sufficient data to estimate the positional marginals of w_o . We would like to explore other features that might be useful in the Dirichlet regression to better estimate these marginals. We would also like to explore other approaches to splicing these marginals together to construct w_o . (We only discussed splicing them together independently or using a somewhat ad-hoc stacking approach but to do so more systematically we would need to estimate the relevant *copula*. It is not clear, however, whether we could ever obtain a rich enough data-set to estimate such a copula accurately.)

Other directions for future research include a more systematic study to estimate the value of insider information. We have already done some (unreported) work on this problem and intend to develop it further. For example, how much gain in expected P&L is there if we have full access to all the positional marginals in W_{op} . How much if we know W_{op} entirely? And to what extent can these gains be offset by superior modeling and estimation of W_{op} ? Certainly our Dirichlet regression approach has reduced the value of this information but interesting questions still remain.

Finally, we briefly mention the area of mean-field games. In our modeling of opponents, we did not assume they were strategic although we did note how some strategic modeling along the lines of



stacking to increase portfolio variance could be accommodated. If we allowed some opponents to be fully strategic, then we are in a game-theoretic setting. Such games would most likely be impossible to solve. Even refinements such as mean-field games (where we let $O \rightarrow \infty$ in some appropriate fashion) would still likely be intractable, especially given the discreteness of the problem (binary decision variables) and portfolio constraints. But it may be possible to solve very stylized versions of these DFS games where it is possible to purchase or sell short fractional amounts of players. There has been some success in solving mean-field games in the literature on parimutuel betting [3] in horse racing and it may be possible to do likewise here for very stylized versions of DFS contests.

We are pursuing some of these directions in current research.



References

1. Anderton, K. 2016. FanDuel And DraftKings Are Dominating The Daily Fantasy Sports Market. <https://www.forbes.com/sites/kevinanderton/2016/11/30/fanduel-and-draftkings-are-dominating-the-daily-fantasy-sports-market-infographic/#2979acb87c4f>. [Accessed: 26-Feb-2017].
2. Bales, J. 2016. Pairing a QB with his Receiver(s). <https://rotogrinders.com/articles/pairing-a-qb-with-his-receiver-s-481544>. [Accessed: 26-Feb-2017].
3. Bayraktar, E. and A. Munk. 2017. High-Roller Impact: A Large Generalized Game Model of Parimutuel Wagering. *Market Microstructure and Liquidity*. 3(01) 1750006.
4. Becker, A. and X.A. Sun. 2016. An analytical approach for fantasy football draft and lineup management. *Journal of Quantitative Analysis in Sports*. 12(1) 17-30.
5. Bergman, D. and J. Imbrogno. 2017. Surviving a National Football League Survivor Pool. *Operations Research*. 65(5) 1343-1354.
6. Clair, B. and D. Letscher. 2007. Optimal strategies for sports betting pools. *Operations Research*. 55(6) 1163-1177.
7. David, H.A. and H.N. Nagaraja. 1981. *Order statistics*. Wiley Online Library.
8. Davis, A. 2017. Data-Driven Portfolios Power 'Home-Run' Exits in MIT Study. <https://www.wsj.com/articles/data-driven-portfolios-power-home-run-exits-in-mit-study-1502105402>. [Accessed: 1-Dec-2017].
9. Drape, J. and J. Williams. 2015. In Fantasy Sports, Signs of Insiders Edge. <https://www.nytimes.com/2015/10/12/sports/fantasy-sports-draftkings-fanduel-insiders-edge-football.html>. [Accessed: 26-Feb-2017].
10. Drape, J. and J. Williams. 2015. Scandal Erupts in Unregulated World of Fantasy Sports. <https://www.nytimes.com/2015/10/06/sports/fanduel-draftkings-fantasy-employees-bet-rivals.html>. [Accessed: 26-Feb-2017].
11. FanDuel. 2016. Rules & Scoring. <https://www.fanduel.com/rules>. [Accessed: 26-Feb-2017].
12. FantasyPros. 2017. <https://www.fantasypros.com>. [Accessed: 30-Aug-2017].
13. Fry, M.J., A.W. Lundberg, and J.W. Ohlmann. 2007. A player selection heuristic for a sports league draft. *Journal of Quantitative Analysis in Sports*. 3(2).



14. FSTA. 2015. Industry demographics. <http://fsta.org/research/industry-demographics/>. [Accessed: 26-Feb-2017].
15. Gibbs, J. 2017. Week 1 FanDuel NFL Tournament Pivots. <https://www.numberfire.com/nfl/lists/16195/week-1-fanduel-nfl-tournament-pivots>. [Accessed: 1-Dec-2017].
16. Gurobi Optimization, Inc. 2016. Gurobi Optimizer Reference Manual. <http://www.gurobi.com>.
17. Harwell, D. 2015. Why you (probably) won't win money playing DraftKings, FanDuel. <http://www.dailyherald.com/article/20151012/business/151019683/>. [Accessed: 26-July-2016].
18. Hunter, D.S., J.P. Vielma, and T. Zaman. 2016. Picking winners using integer programming. *arXiv preprint arXiv:1604.01455*.
19. Kaplan, E.H. and S.J. Garstka. 2001. March madness and the office pool. *Management Science*. 47(3) 369-382.
20. Kolodny, L. 2015. Fantasy Sports Create Billion-Dollar Startups. <https://www.wsj.com/articles/fantasy-sports-create-billion-dollar-startups-1436846402>. [Accessed: 26-Feb-2017].
21. Morton, D.P., E. Popova, I. Popova, M. Zhong. 2003. Optimizing benchmark-based utility functions. *Bulletin of the Czech Econometric Society*. 10(18).
22. Mulshine, M. 2015. How one man made hundreds of thousands of dollars playing daily fantasy sports. <http://www.businessinsider.com/draft-kings-jonathan-bales-a-day-in-the-life-2015-10>. [Accessed: 26-Feb-2017].
23. Nickish, K. 2015. Meet A Bostonian Who's Made \$3 Million This Year Playing Daily Fantasy Sports. <http://www.wbur.org/news/2015/11/23/dfs-power-player-profile>. [Accessed: 26-Feb-2017].
24. O'Keeffe, K. 2015. Daily Fantasy-Sports Operators Await Reality Check. <https://www.wsj.com/articles/daily-fantasy-sports-operators-await-reality-check-1441835630>. [Accessed: 26-Feb-2017].
25. Pramuk, J. 2015. Former pro poker player makes a living on fantasy sports. <https://www.cnbc.com/2015/10/06/former-pro-poker-player-makes-a-living-on-fantasy-sports.html>. [Accessed: 26-Feb-2017].
26. RotoViz. 2017. <http://rotoviz.com>. [Accessed: 30-Aug-2017].
27. Stan Development Team. 2017. *RStan: the R interface to Stan*. R package version 2.16.2. <http://mc-stan.org>.



28. Wong, K. 2015. The Fantasy Sports Industry, by the Numbers.
<http://www.nbcnews.com/business/business-news/fantasy-sports-industry-numbers-n439536>. [Accessed: 26-Feb-2017].
29. Woodward, C. 2015. Top fantasy sports player uses software, analytics to reap millions.
<https://www.bostonglobe.com/business/2015/12/23/from-boston-penthouse-world-best-fantasy-player-plunges-into-startup-world/QHNpLh003QMyUDqTd4t27N/story.html>.
[Accessed: 26-Feb-2017].
30. Woodward, C. 2016. DraftKings, FanDuel collected \$3b in entry fees last year, analyst says.
<https://www.bostonglobe.com/business/2016/02/29/fantasy-sports-industry-hits-amid-legal-questions-analyst-says/NKw364kiLjv8XcD54vRr4H/story.html>. [Accessed: 26-Feb-2017].



A. Appendix: Efficient Sampling of Order Statistic Moments

In order to execute Algorithms 2 and 3, we must first estimate the inputs $\mu_{G^{(r')}}$, $\sigma_{G^{(r')}}^2$ and $\sigma_{\delta, G^{(r')}}$. These quantities can be estimated off-line via Monte-Carlo simulation as they do not depend on our portfolio choice \mathbf{w} . Here, we describe how the Monte-Carlo can be performed efficiently using results from the theory of order statistics assuming O is large. Recalling $G_o = \mathbf{w}_o^T \boldsymbol{\delta}$ is the fantasy points score of the o^{th} opponent, we first note the G_o 's for $o = 1, \dots, O$ are IID given $(\boldsymbol{\delta}, \mathbf{p})$, where \mathbf{p} denotes the multinomial probability vectors (for the positional marginals) from Section 3. This suggests the following algorithm for obtaining independent samples of $\boldsymbol{\delta}$ and $G^{(r')}$:

1. Generate $\boldsymbol{\delta} \sim N(\boldsymbol{\mu}_\delta, \boldsymbol{\Sigma}_\delta)$ and $(\mathbf{p}, \mathbf{W}_{\text{op}})$ using Algorithm 1 (or the stacking variant of it) where $\mathbf{p} := (\mathbf{p}_{\text{QB}}, \dots, \mathbf{p}_{\text{D}})$ and $\mathbf{W}_{\text{op}} = \{\mathbf{w}_o\}_{o=1}^O$.
2. Compute $G_o = \mathbf{w}_o^T \boldsymbol{\delta}$ for $o = 1, \dots, O$.
3. Order the G_o 's.
4. Return $(\boldsymbol{\delta}, G^{(r')})$.

We note other algorithms are also available. For example, a standard result on order statistics yields

$$F_{G^{(r')} | (\boldsymbol{\delta}, \mathbf{p})}(x) = \sum_{i=r'}^O \frac{O!}{(O-i)! i!} (F_{G | (\boldsymbol{\delta}, \mathbf{p})}(x))^i (1 - F_{G | (\boldsymbol{\delta}, \mathbf{p})}(x))^{O-i}, \quad (16)$$

where $F_{G^{(r')} | (\boldsymbol{\delta}, \mathbf{p})}$ and $F_{G | (\boldsymbol{\delta}, \mathbf{p})}$ denote the CDFs of $G^{(r')}$ and a random opponent G_o , respectively, given $(\boldsymbol{\delta}, \mathbf{p})$. Instead of generating \mathbf{W}_{op} and the G_o 's, we could use the inverse transform approach and (16) to generate $G^{(r')}$ directly. Unfortunately, both algorithms are quite computationally expensive since we will need many samples of $(\boldsymbol{\delta}, G^{(r')})$ to estimate $\mu_{G^{(r')}}$, $\sigma_{G^{(r')}}^2$ and $\sigma_{\delta, G^{(r')}}$. Moreover, we would also need to estimate $F_{G | (\boldsymbol{\delta}, \mathbf{p})}$ and so to do this, many samples from $G | (\boldsymbol{\delta}, \mathbf{p})$ would be required for each value of $(\boldsymbol{\delta}, \mathbf{p})$.

We can circumvent these problems when O is large (the typical case of interest) by noting²⁵ that (see [7])

$$G^{(r')} | (\boldsymbol{\delta}, \mathbf{p}) \xrightarrow{p} F_{G | (\boldsymbol{\delta}, \mathbf{p})}^{-1} \left(\frac{r'}{O} \right) \quad \text{as } O \rightarrow \infty,$$

where \xrightarrow{p} denotes convergence in probability. Therefore in the limit of large O , there is no uncertainty given $(\boldsymbol{\delta}, \mathbf{p})$ and so we can simply set $G^{(r')} = F_{G | (\boldsymbol{\delta}, \mathbf{p})}^{-1} \left(\frac{r'}{O} \right)$. Of course, we still need to estimate $F_{G | (\boldsymbol{\delta}, \mathbf{p})}$ and so, many samples of $G | (\boldsymbol{\delta}, \mathbf{p})$ would be required. The algorithm therefore proceeds as:

²⁵ In fact the result states that $G^{(r')} | (\boldsymbol{\delta}, \mathbf{p})$ is normal with mean $F_{G | (\boldsymbol{\delta}, \mathbf{p})}^{-1} \left(\frac{r'}{O} \right)$ and a variance that vanishes as $O \rightarrow \infty$.



1. Generate $\boldsymbol{\delta} \sim N(\boldsymbol{\mu}_{\boldsymbol{\delta}}, \boldsymbol{\Sigma}_{\boldsymbol{\delta}})$ and $(\boldsymbol{p}, \boldsymbol{W}_{\text{op}})$ using Algorithm 1 (or the stacking variant of it) where $\boldsymbol{p} := (\boldsymbol{p}_{\text{QB}}, \dots, \boldsymbol{p}_{\text{D}})$ and $\boldsymbol{W}_{\text{op}} = \{\boldsymbol{w}_o\}_{o=1}^O$.
2. Compute $G_o = \boldsymbol{w}_o^T \boldsymbol{\delta}$ for $o = 1, \dots, O$.
3. Use the G 's to construct $\hat{F}_{G|(\boldsymbol{\delta}, \boldsymbol{p})}(\cdot)$.
4. Set $G^{(r')} = \hat{F}_{G|(\boldsymbol{\delta}, \boldsymbol{p})}^{-1}\left(\frac{r'}{O}\right)$.
5. Return $(\boldsymbol{\delta}, G^{(r')})$.

We note that the value of O is no longer assumed to be the number of DFS competitors (which is assumed to be infinite here) but instead is simply set large enough so that we can estimate $F_{G|(\boldsymbol{\delta}, \boldsymbol{p})}$ sufficiently accurately.