Winning at Daily Fantasy Hockey Using Analytics

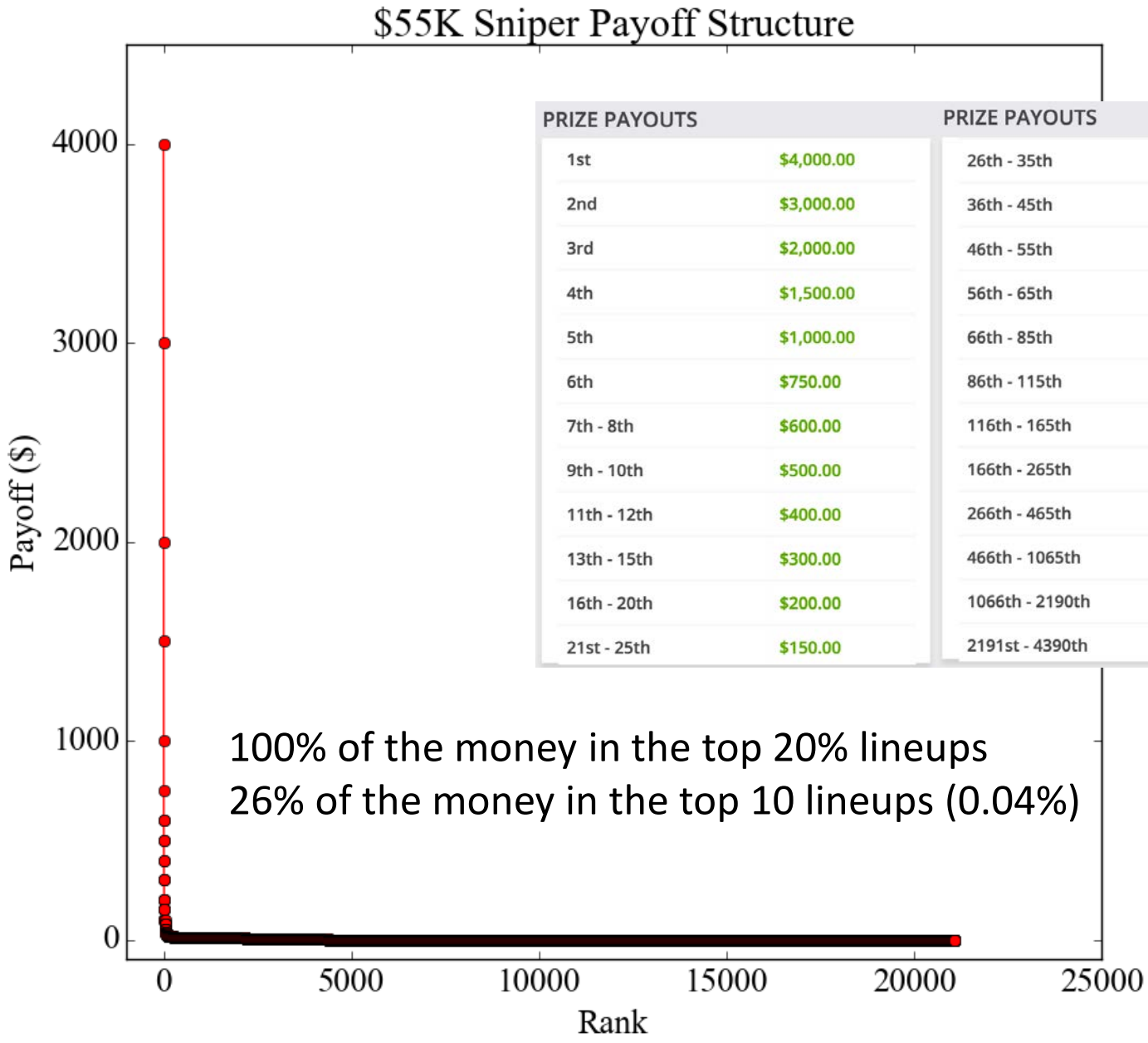
David Hunter,
Juan Pablo Vielma (@J_P_Vielma), and
Tauhid Zaman (@zlisto)

Example Entry

LINEUP

Avg. Rem. / Player: $0
Rem. Salary: $0

POS	PLAYER	OPP	FPPG	SALARY	
C	Jussi Jokinen	**Fla**@Anh	3.1	$5,300	✖
C	Brandon Sutter	Pit@**Van**	3.0	$4,400	✖
W	Nikolaj Ehlers	**Wpg**@Tor	3.9	$4,800	✖
W	Daniel Sedin	Pit@**Van**	3.8	$6,400	✖
W	Radim Vrbata	Pit@**Van**	3.4	$5,800	✖
D	Brian Campbell	**Fla**@Anh	2.6	$4,100	✖
D	Morgan Rielly	Wpg@**Tor**	3.5	$4,200	✖
G	Corey Crawford P	StL@**Chi**	6.3	$7,800	✖
UTIL	Blake Wheeler	**Wpg**@Tor	4.8	$7,200	✖

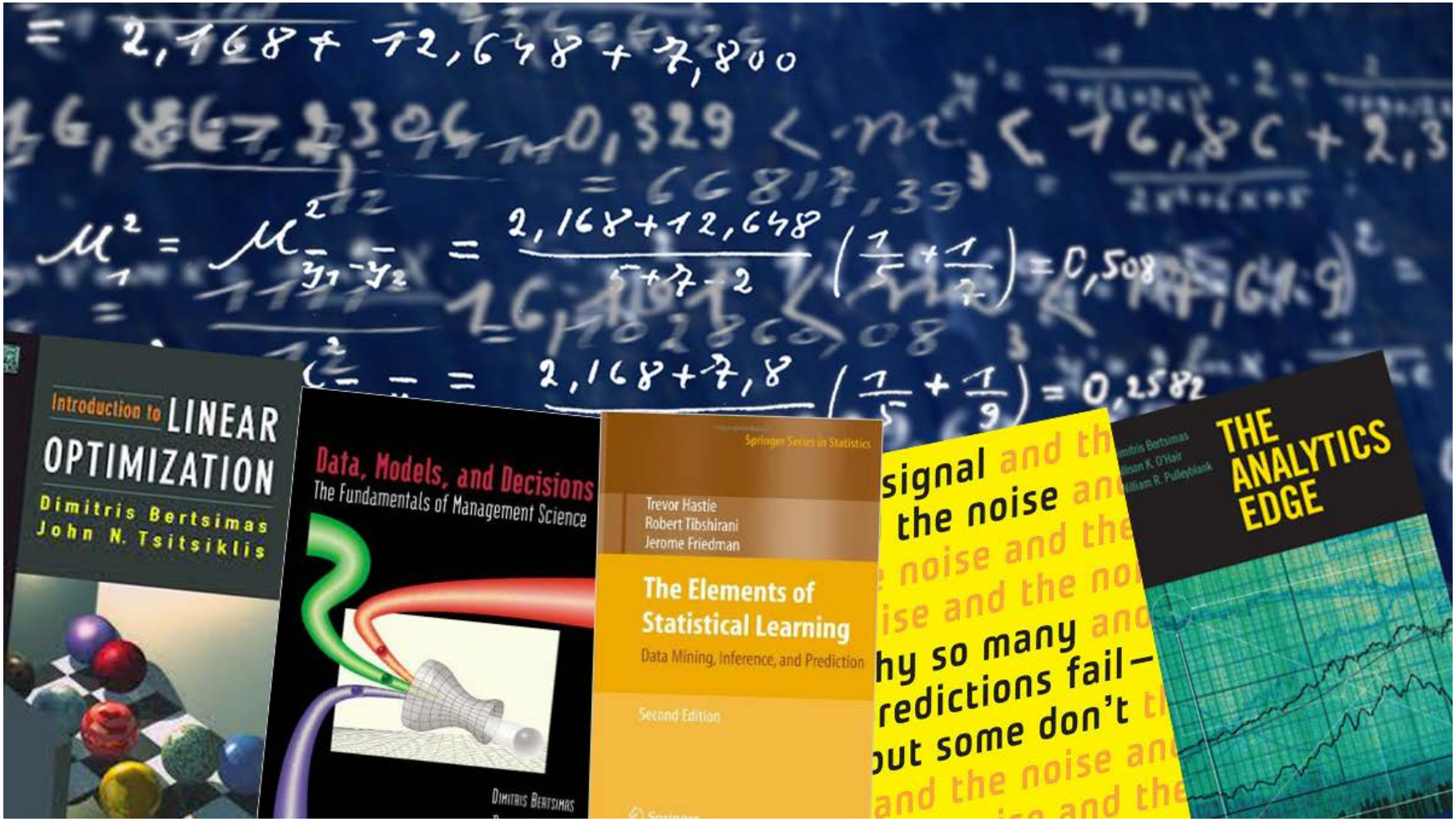


PRIZE PAYOUTS	
1st	$4,000.00
2nd	$3,000.00
3rd	$2,000.00
4th	$1,500.00
5th	$1,000.00
6th	$750.00
7th - 8th	$600.00
9th - 10th	$500.00
11th - 12th	$400.00
13th - 15th	$300.00
16th - 20th	$200.00
21st - 25th	$150.00

PRIZE PAYOUTS	
26th - 35th	$100.00
36th - 45th	$75.00
46th - 55th	$50.00
56th - 65th	$40.00
66th - 85th	$30.00
86th - 115th	$25.00
116th - 165th	$20.00
166th - 265th	$15.00
266th - 465th	$12.00
466th - 1065th	$10.00
1066th - 2190th	$8.00
2191st - 4390th	$6.00

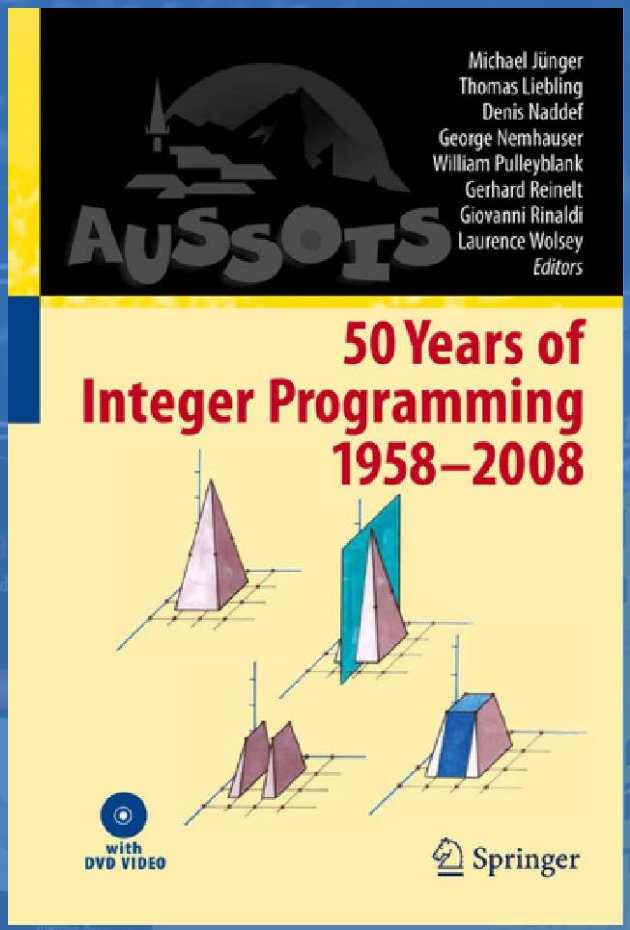
100% of the money in the top 20% lineups
26% of the money in the top 10 lineups (0.04%)

Previous Knowledge: Analytics



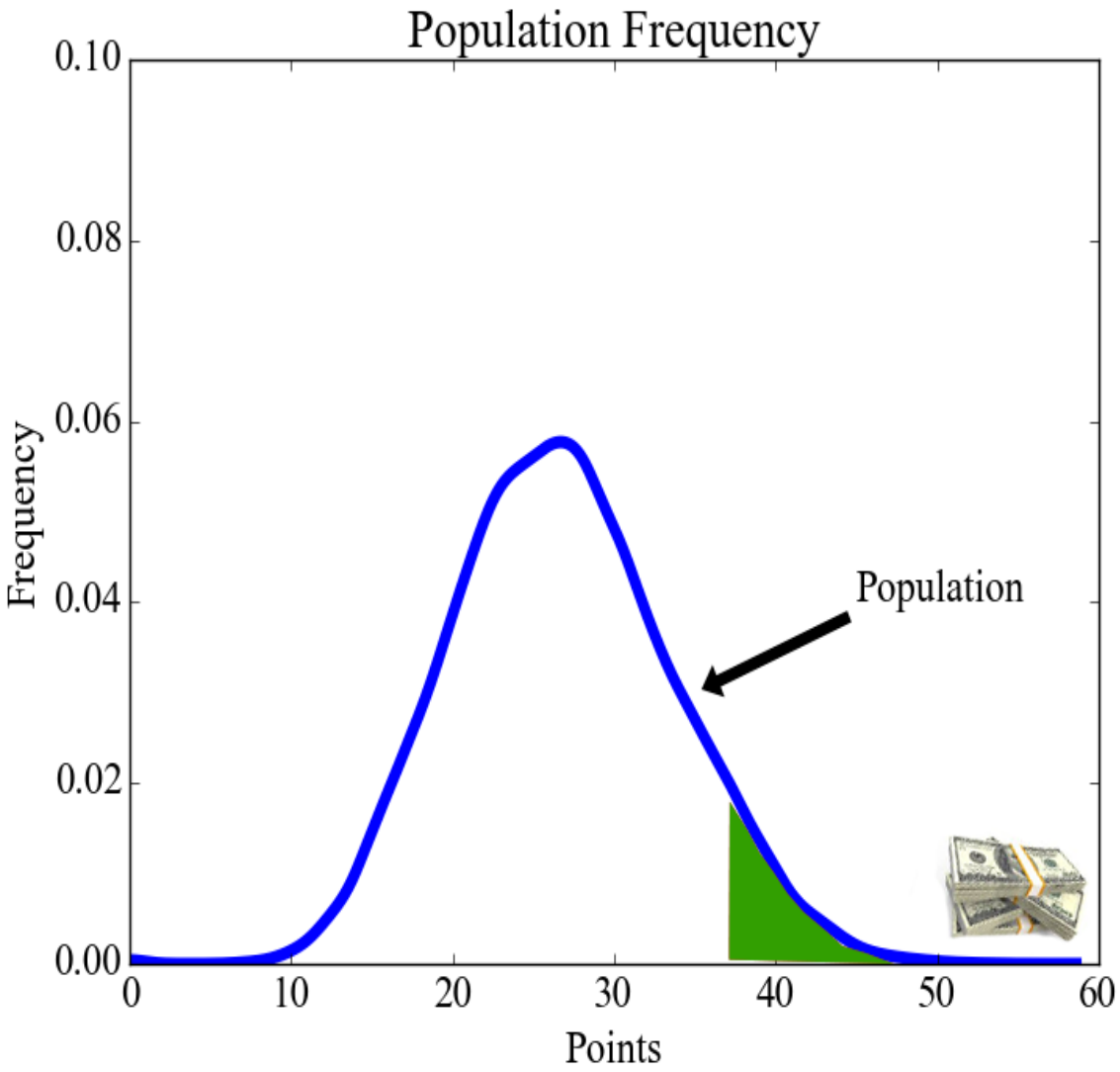
Previous Knowledge: Analytics

#KnightsOfMIP

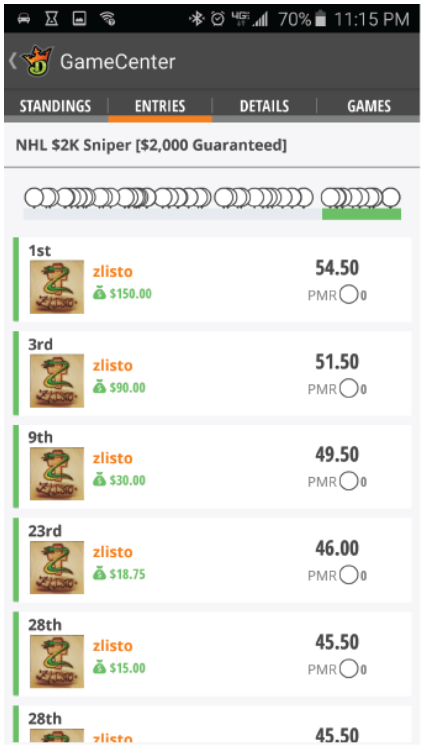


Building a Lineup

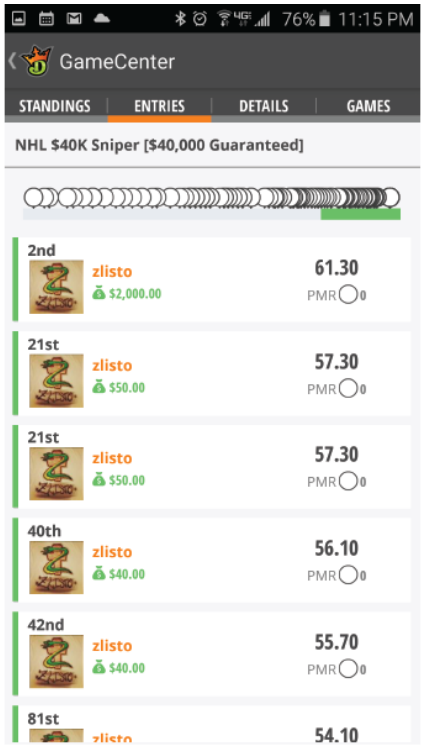
Using this knowledge…



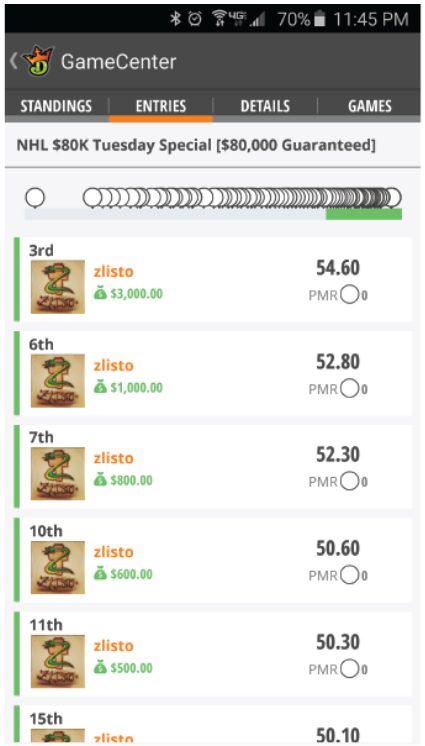
Were we able to do it?



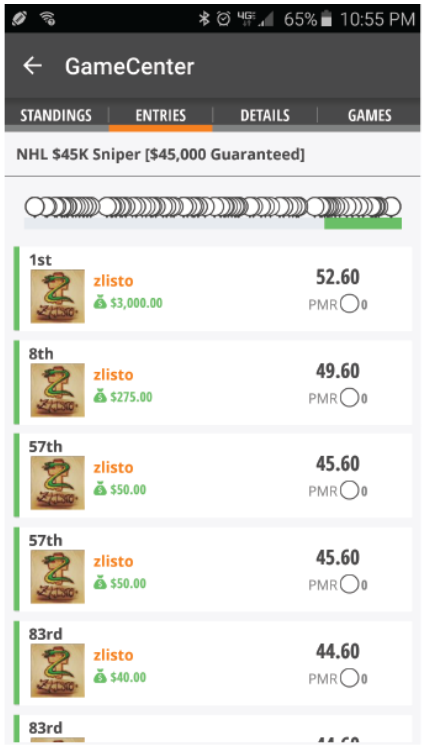
November 15, 2015



November 16, 2015



November 17, 2015



November 23, 2015

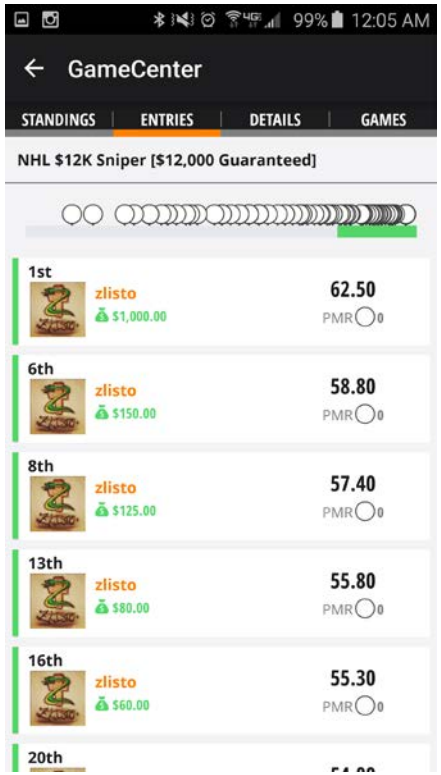
200 lineups

Policy Change

Policy Change

200 lineups -> 100 lineups

Were we able to continue it?



December 12, 2015

100 lineups

Legal Disclaimer:
All profits are in the process of being donated to charity.

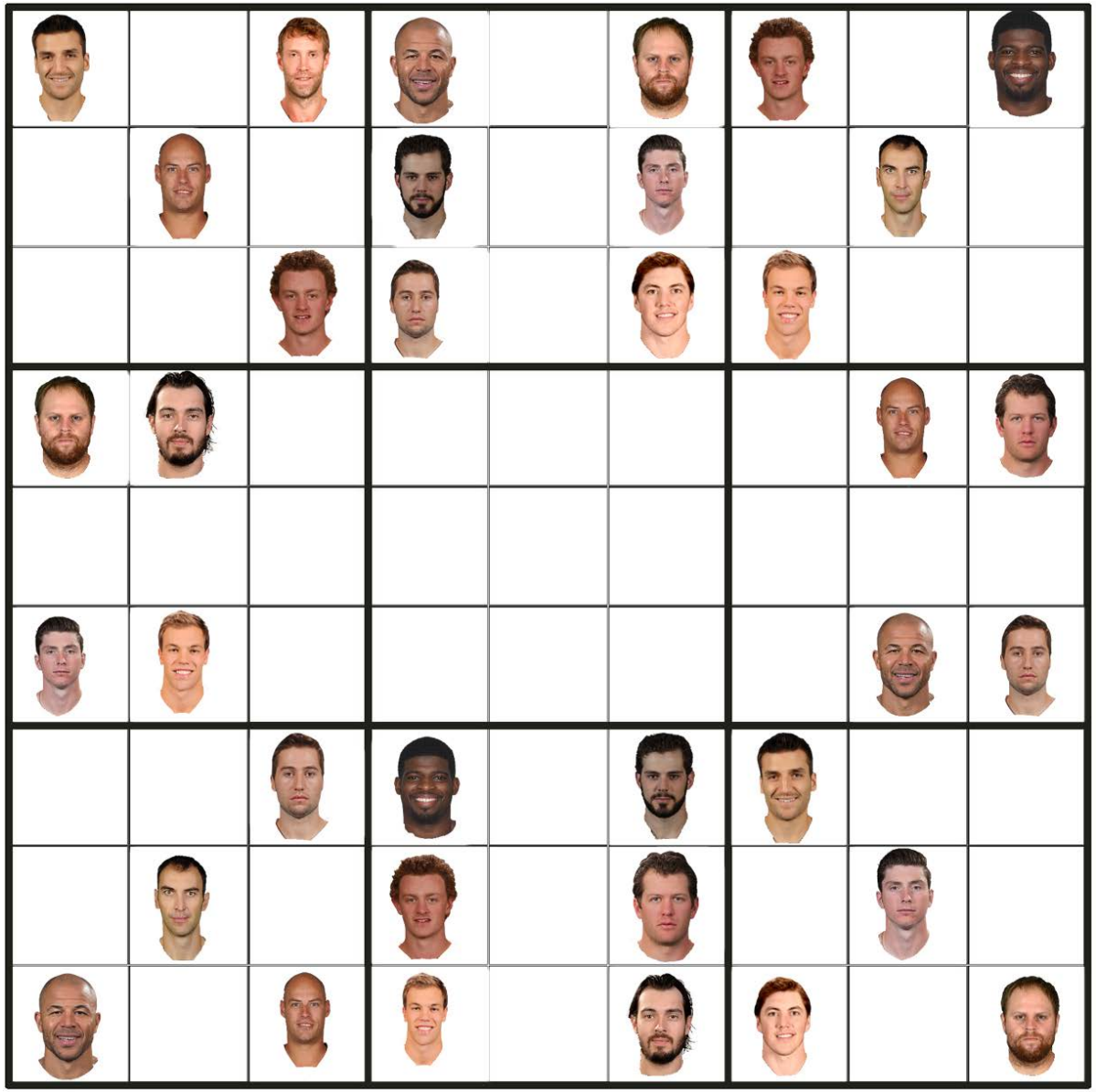
Integer Programming Formulation

- We will make a bunch of lineups consisting of 9 players each
- Use an integer programming approach to find these lineups

Decision variables

$$x_{pl} = \begin{cases} 1, & \text{if player } p \text{ in lineup } l \\ 0, & \text{otherwise} \end{cases}$$

First Attempt…



Basic Feasibility

- 9 different players
- Salary less than $50,000

Basic constraints

$$\sum_{p=1}^{N} c_p x_{pl} \leq \$50,000, \quad \text{(budget constraint)}$$

$$\sum_{p=1}^{N} x_{pl} = 9, \quad \text{(lineup size constraint)}$$

$$x_{pl} \in \{0,1\}, \quad 1 \leq p \leq N.$$

Position Feasibility

- Between 2 and 3 centers
- Between 3 and 4 wingers
- Between 2 and 3 defensemen
- 1 goalie

Position constraints

$$2 \leq \sum_{p \in C} x_{pl} \leq 3, , \quad \text{(center constraint)}$$

$$3 \leq \sum_{p \in W} x_{pl} \leq 4, \quad \text{(winger constraint)}$$

$$2 \leq \sum_{p \in D} x_{pl} \leq 3, \quad \text{(defensemen constraint)}$$

$$\sum_{u \in G} x_{pl} = 1 \quad \text{(goalie constraint)}$$

Team Feasibility

- At least 3 different NHL teams

Team constraints

$$t_i \leq \sum_{p \in T_i} x_{pl}, \quad \forall\, i \in \{1, \ldots, N_T\}$$

$$\sum_{i=1}^{N_T} t_i \geq 3,$$

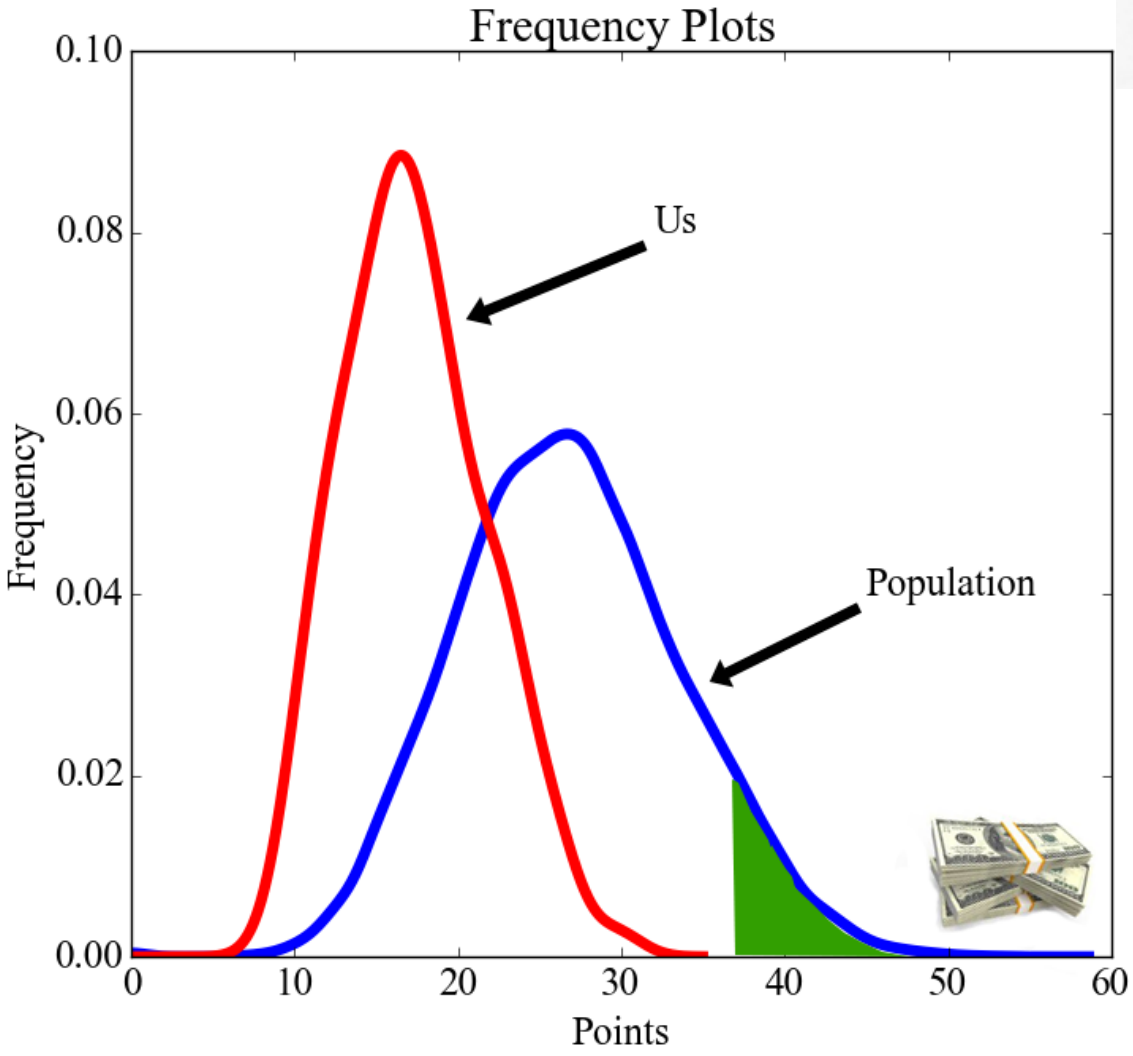
$$t_i \in \{0, 1\}, \quad \forall\, i \in \{1, \ldots, N_T\}.$$

First Attempt…

$6400	$7200	$4200	$4100	$5300	$4400	$4800	$5800	$7800
W	UTIL	D	D	C	C	W	W	G

> 3 Different Teams

First Attempt…

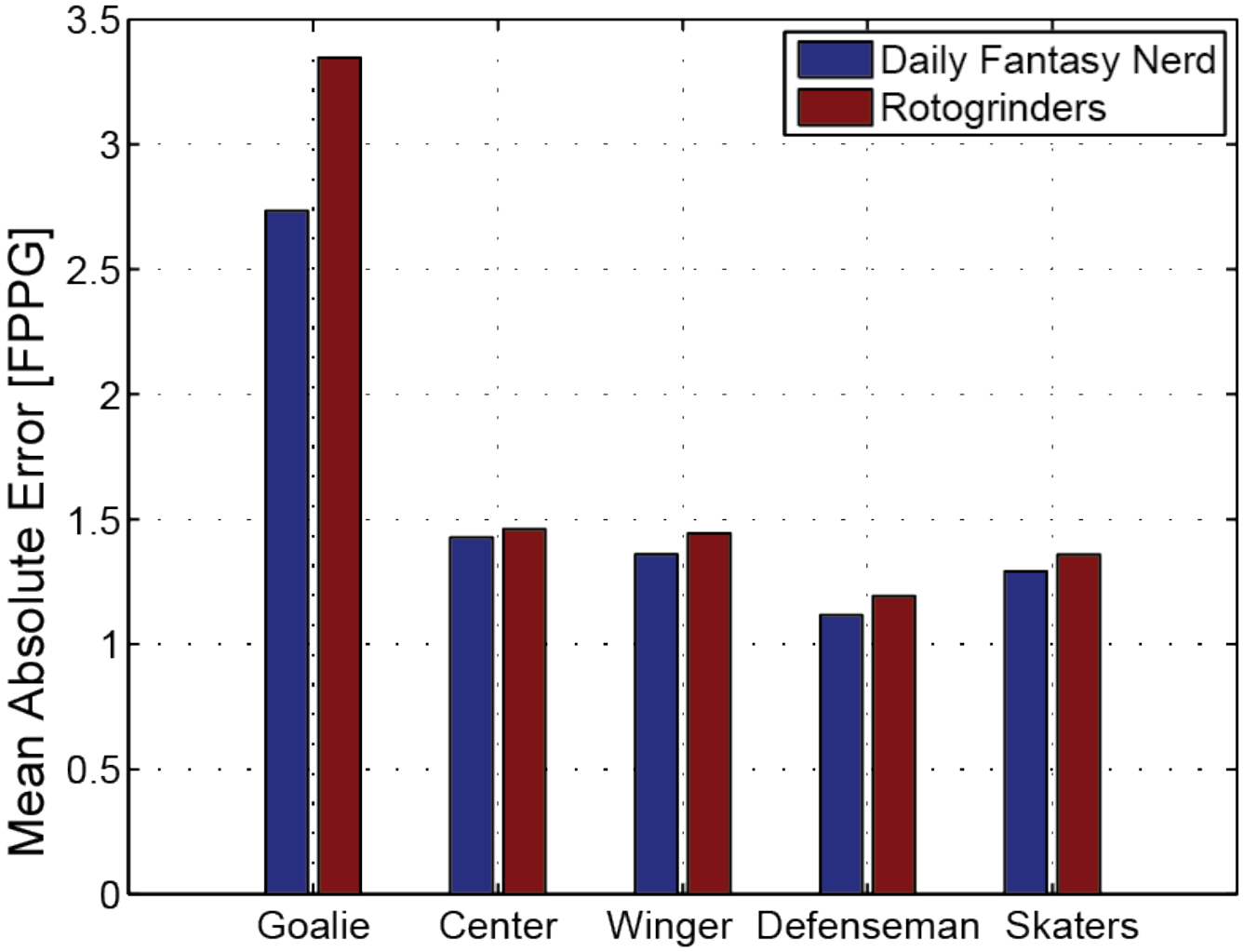


Second Attempt…

- Must increase our mean points…
- Solution:

USE EXPERT PREDICTIONS

Prediction Errors



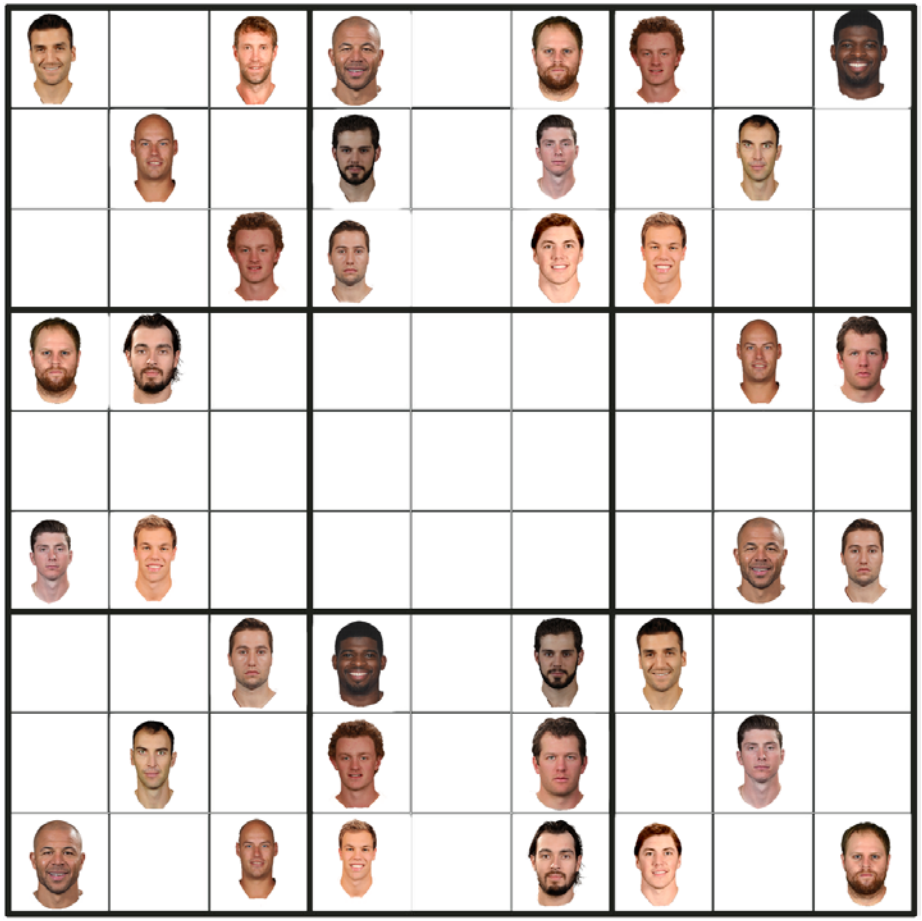
Maximize Points

- Forecasted points for player p: f_p
- You get to choose what the forecasts are

Points Objective Function

$$\sum_{p=1}^{N} f_p x_{pl}$$

Second Attempt…



+ Maximize points

$$\max_{x} \sum_{p=1}^{N} f_p x_{pl}$$

Old Lineup

$6400	$7200	$4200	$4100	$5300	$4400	$4800	$5800	$7800
W	UTIL	D	D	C	C	W	W	G

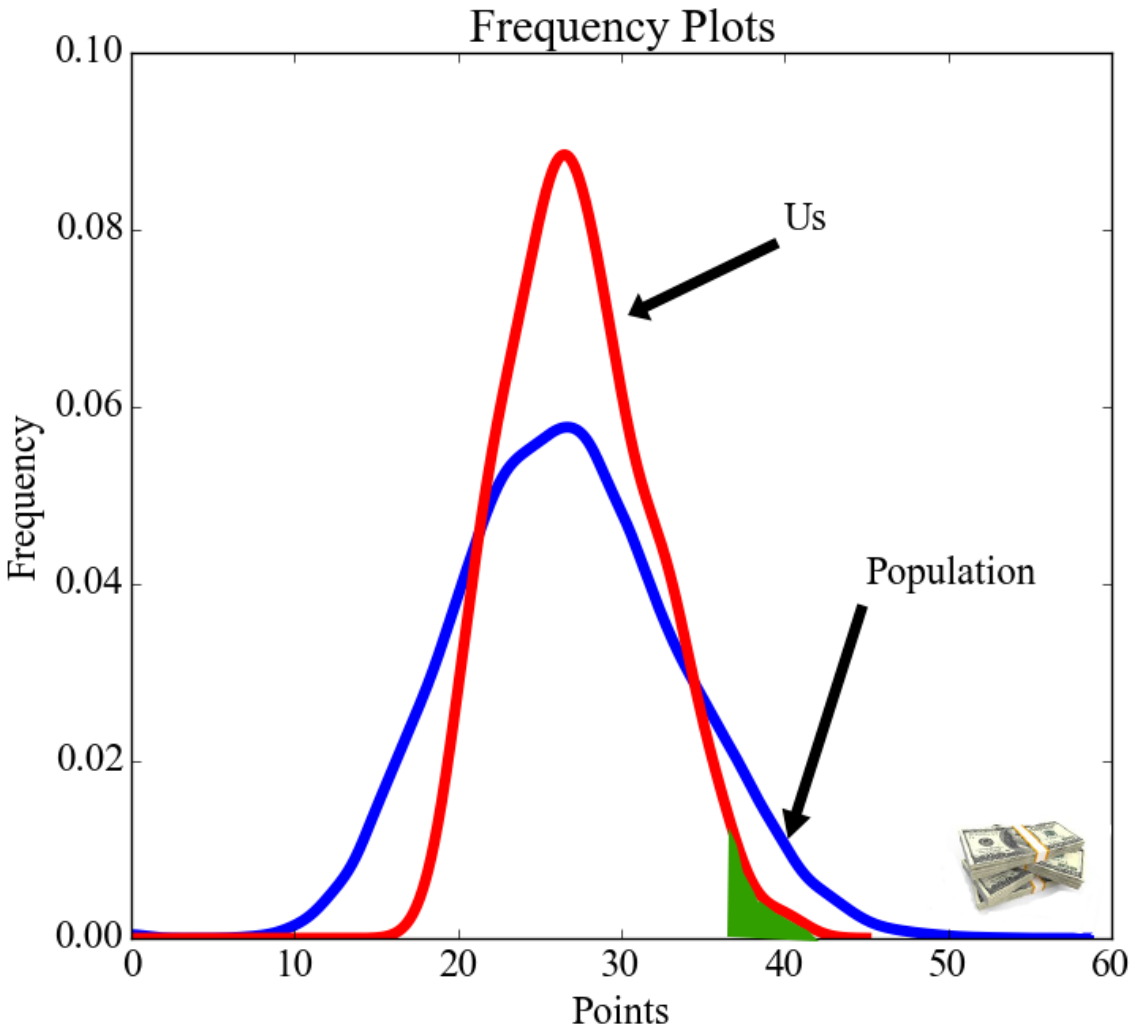
12 points on average

New Lineup

Projections:	5.4	2.5	3.4	3.0	3.2	4.2	3.5	3.4	5.7
	$9500	$2700	$4600	$3800	$4600	$6400	$5200	$5100	$8000
	W	UTIL	D	D	C	C	W	W	G

23 points on average

Second Attempt…

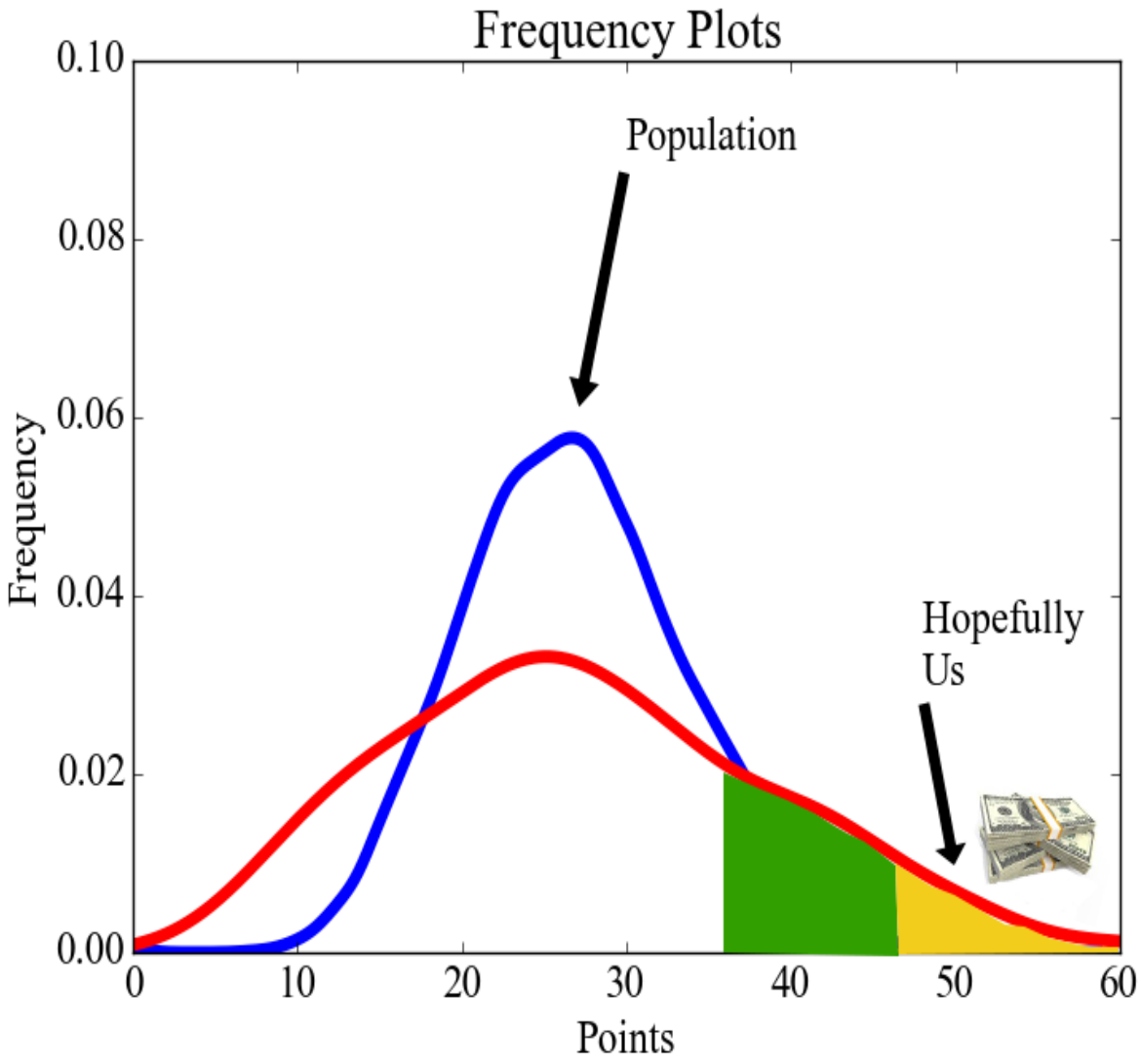


How can we do better?

- Three D's of Finance:

So what do we do?

By doing this…

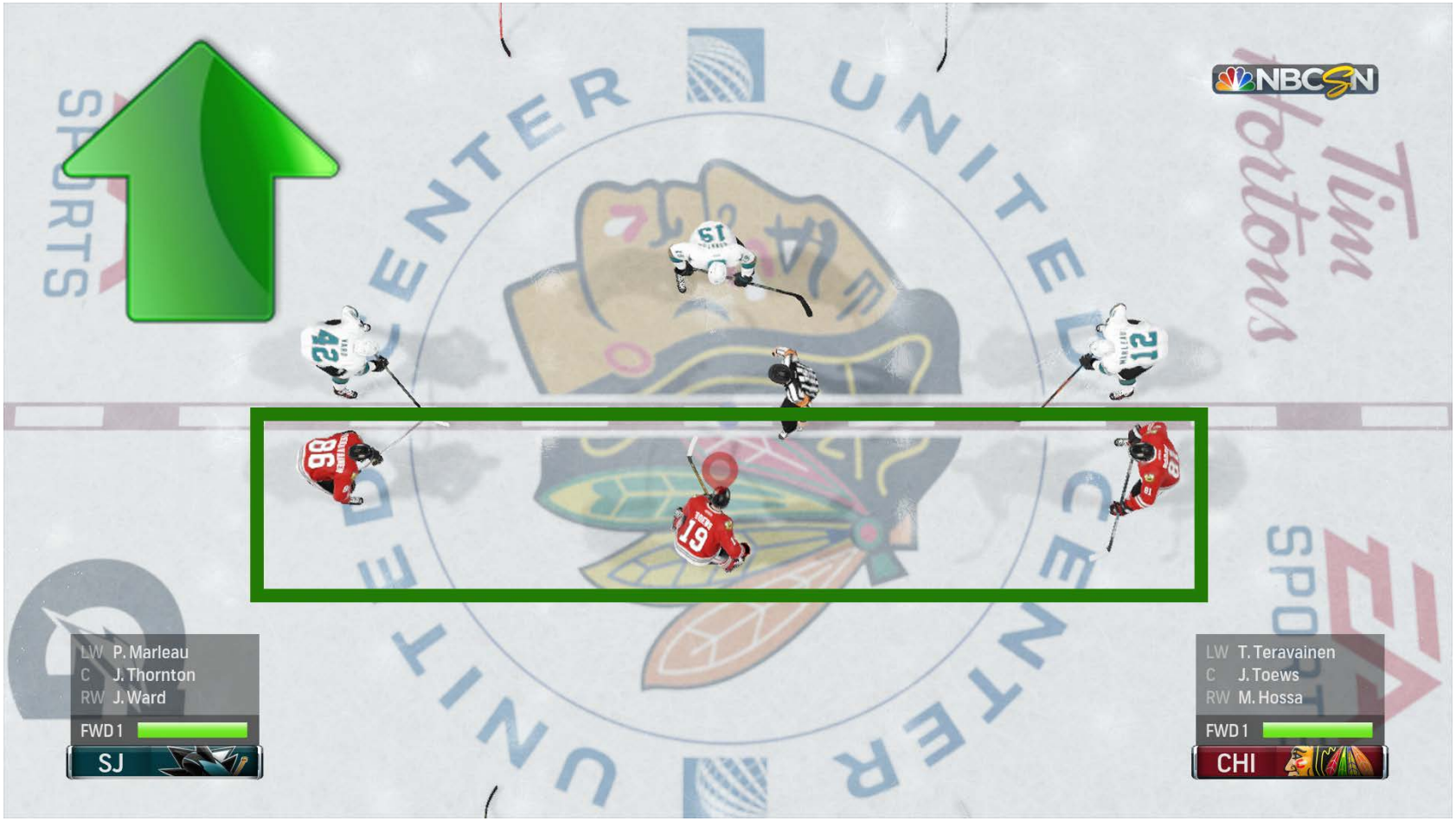


Stacking Lineups

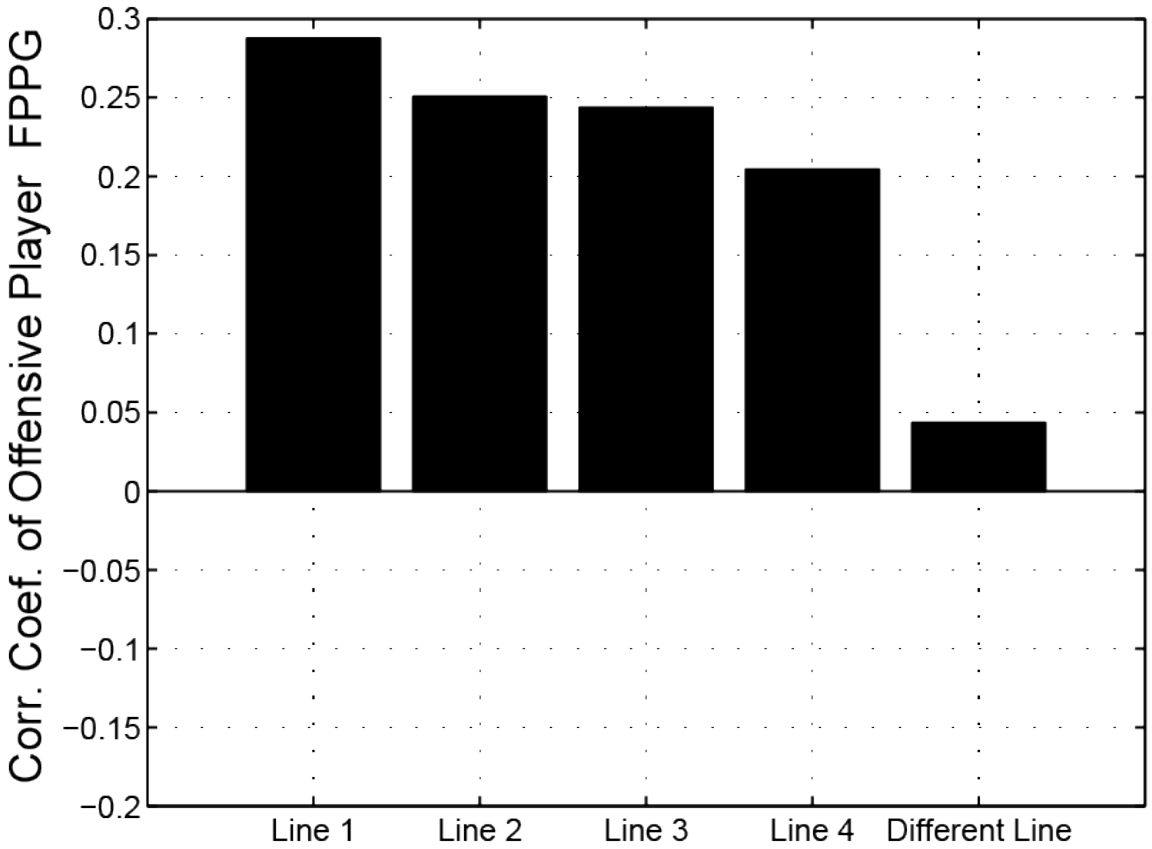
- Stacking means putting players on a single lineup that have a positive correlation
- Either the players pop off together -> tons of points
- Or the players crap out -> few points
- We stack using “structural correlations”

Structural Correlations - Teams

Structural Correlations - Lines



Structural Correlations - Lines



Structural Correlations - Lines

- At least 1 complete line (3 players per line)
- At least 2 partial lines (at least 2 players per line)

1 complete line constraint

$$3v_i \leq \sum_{p \in L_i} x_{pl}, \qquad \forall i \in \{1, \ldots, N_L\}$$

$$\sum_{i=1}^{N_L} v_i \geq 1$$

$$v_i \in \{0, 1\}, \qquad \forall i \in \{1, \ldots, N_L\}.$$

2 partial lines constraint

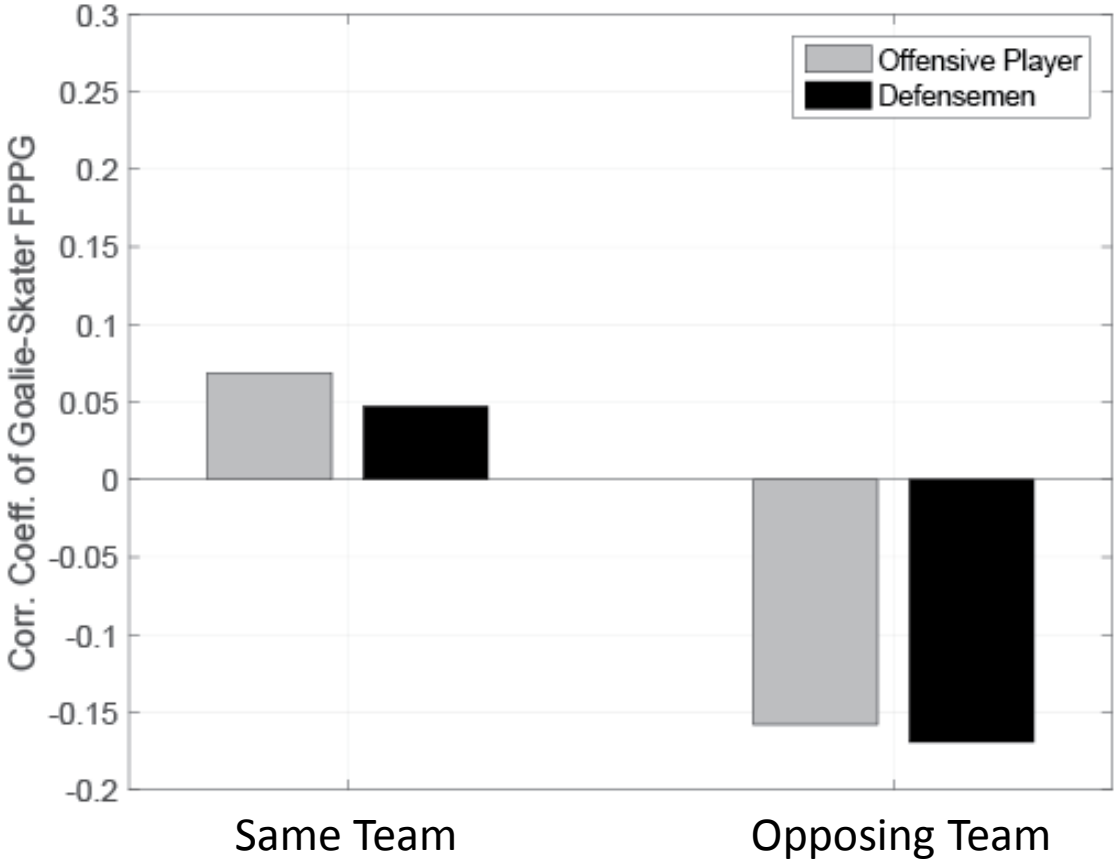
$$2w_i \leq \sum_{p \in L_i} x_{pl}, \qquad \forall i \in \{1, \ldots, N_L\}$$

$$\sum_{i=1}^{N_L} w_i \geq 2$$

$$w_i \in \{0, 1\}, \qquad \forall i \in \{1, \ldots, N_L\}.$$

Structural Correlations – Goalie Against Opposing Players

Structural Correlations – Goalie Against Opposing Players

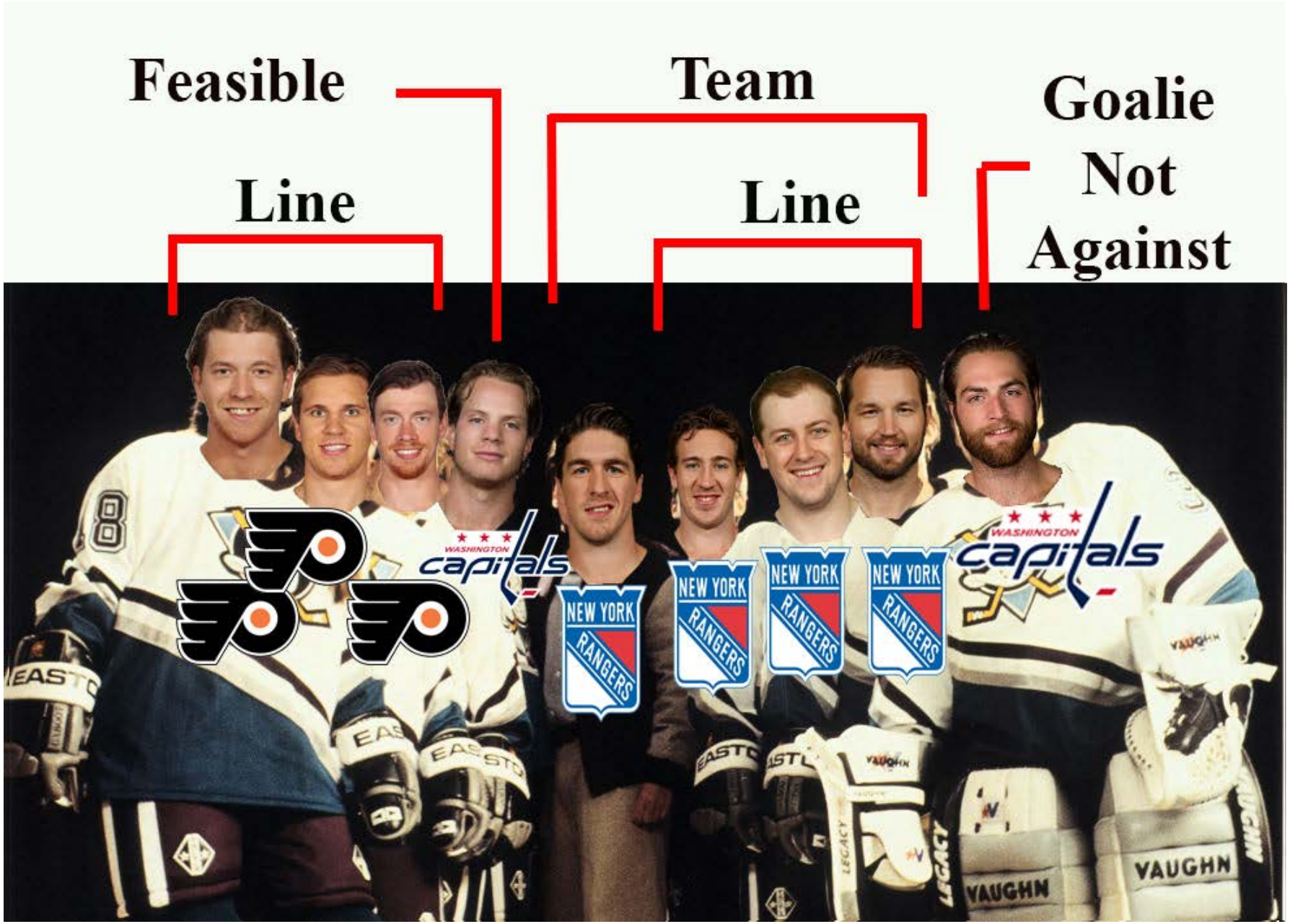


Structural Correlations – Goalie Against Skaters

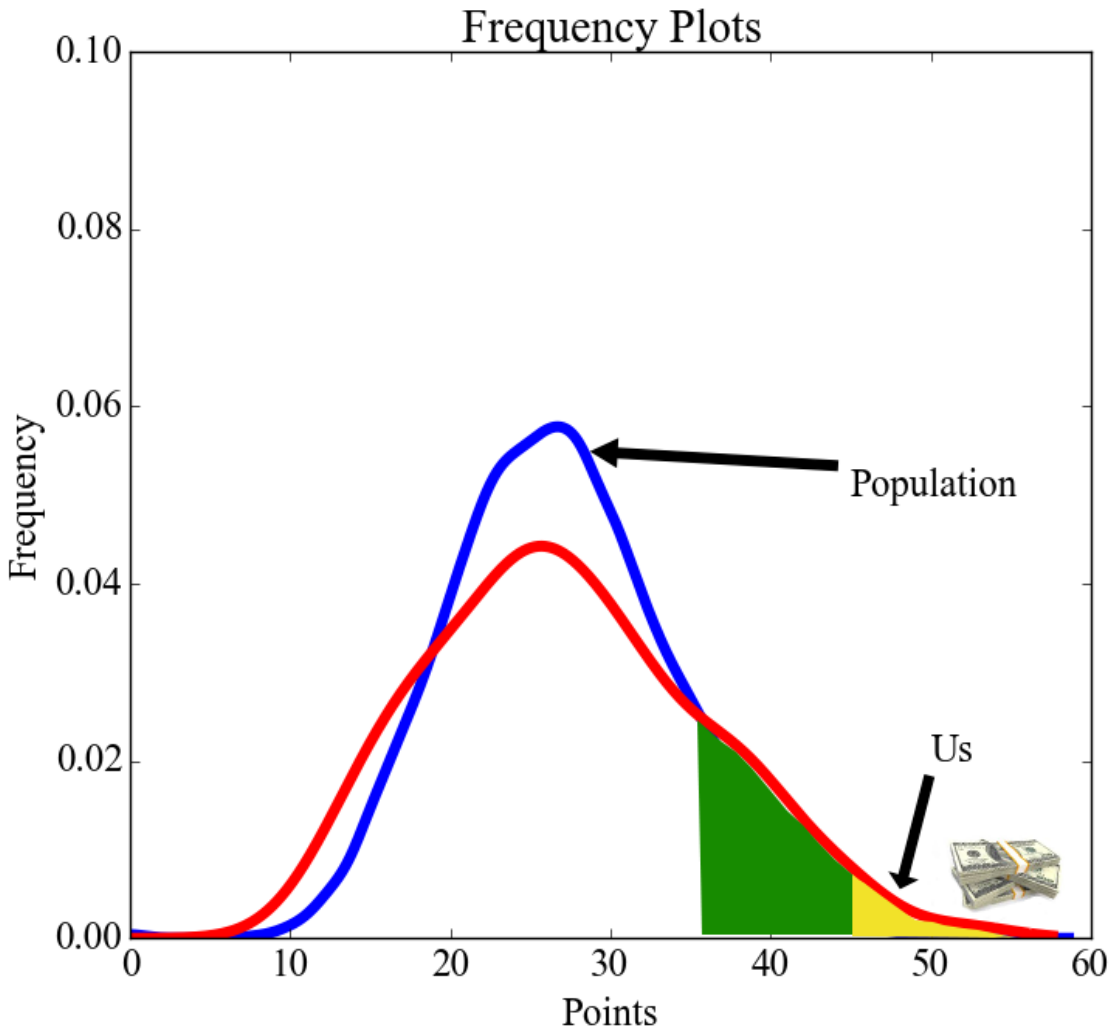
- No skater against goalie

No skater against goalie constraint

$$6x_{pl} + \sum_{q \in Opponents_p} x_{ql} \leq 6, \quad \forall p \in G$$


Feasible
Line
Team
Line
Goalie
Not
Against
WASHINGTON
capitals
NEW YORK
RANGERS
NEW YORK
RANGERS
NEW YORK
RANGERS
NEW YORK
RANGERS
WASHINGTON
capitals
VAUGHN
VAUGHN

Second Attempt…



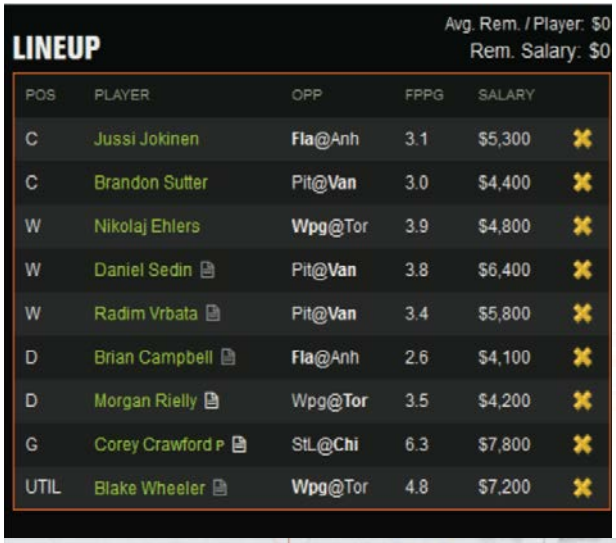
Lineup Diversity

- Make sure lineup l has no more than γ players in common with lineups 1 to l-1

Diversity constraint

$$\sum_{p=1}^{N} x_{pk}^{*} x_{pl} \leq \gamma, k = 1, \ldots, l-1$$

To Review…



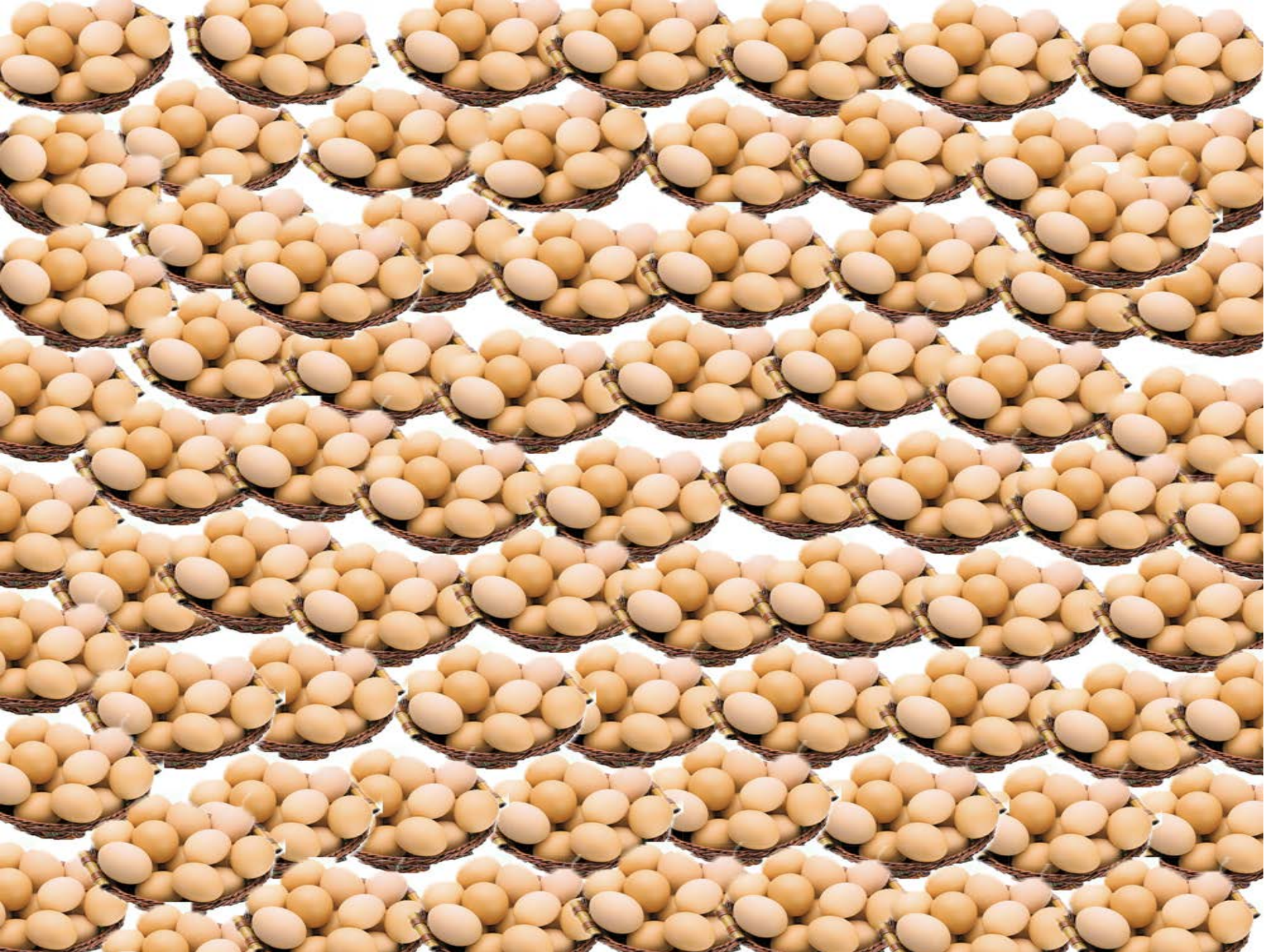
LINEUP

Avg. Rem. / Player: $0
Rem. Salary: $0

POS	PLAYER	OPP	FPPG	SALARY	
C	Jussi Jokinen	Fla@Anh	3.1	$5,300	✖
C	Brandon Sutter	Pit@Van	3.0	$4,400	✖
W	Nikolaj Ehlers	Wpg@Tor	3.9	$4,800	✖
W	Daniel Sedin	Pit@Van	3.8	$6,400	✖
W	Radim Vrbata	Pit@Van	3.4	$5,800	✖
D	Brian Campbell	Fla@Anh	2.6	$4,100	✖
D	Morgan Rielly	Wpg@Tor	3.5	$4,200	✖
G	Corey Crawford P	StL@Chi	6.3	$7,800	✖
UTIL	Blake Wheeler	Wpg@Tor	4.8	$7,200	✖

RotoGrinders.com

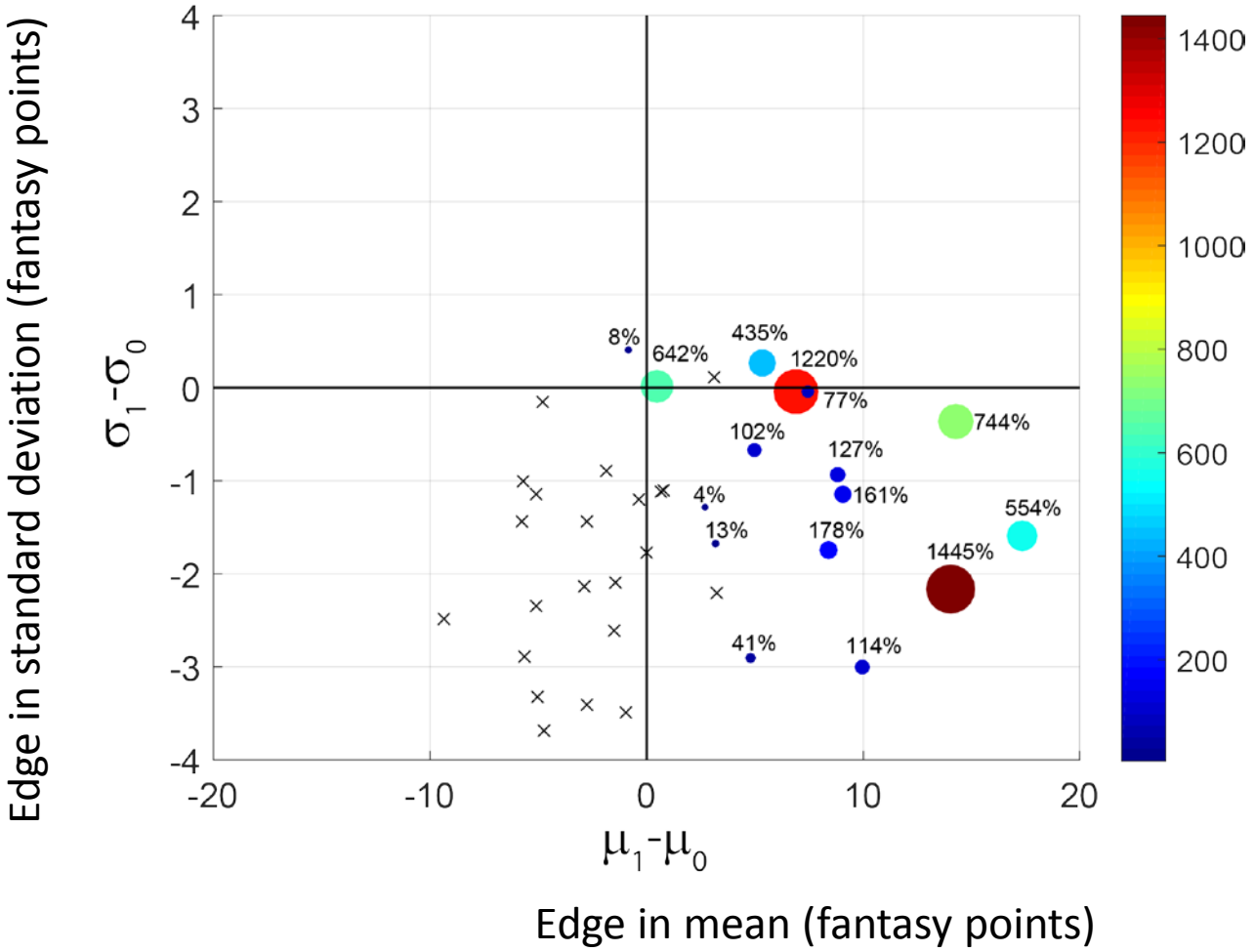
dfnerd



PERFORMANCE ON REAL CONTESTS

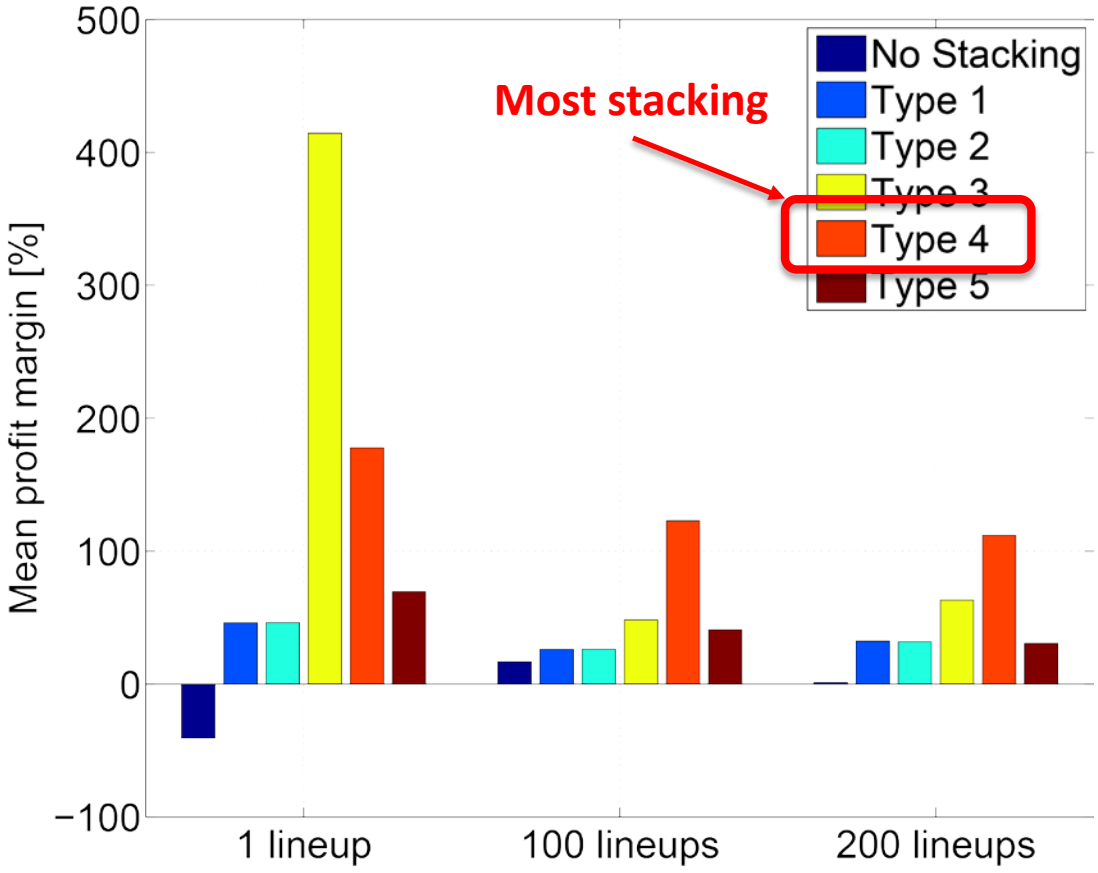
Performance on Real Contests

- Each point is a contest, with profit margin shown
- Used all stacking, a maximum overlap of 7, and 200 lineups



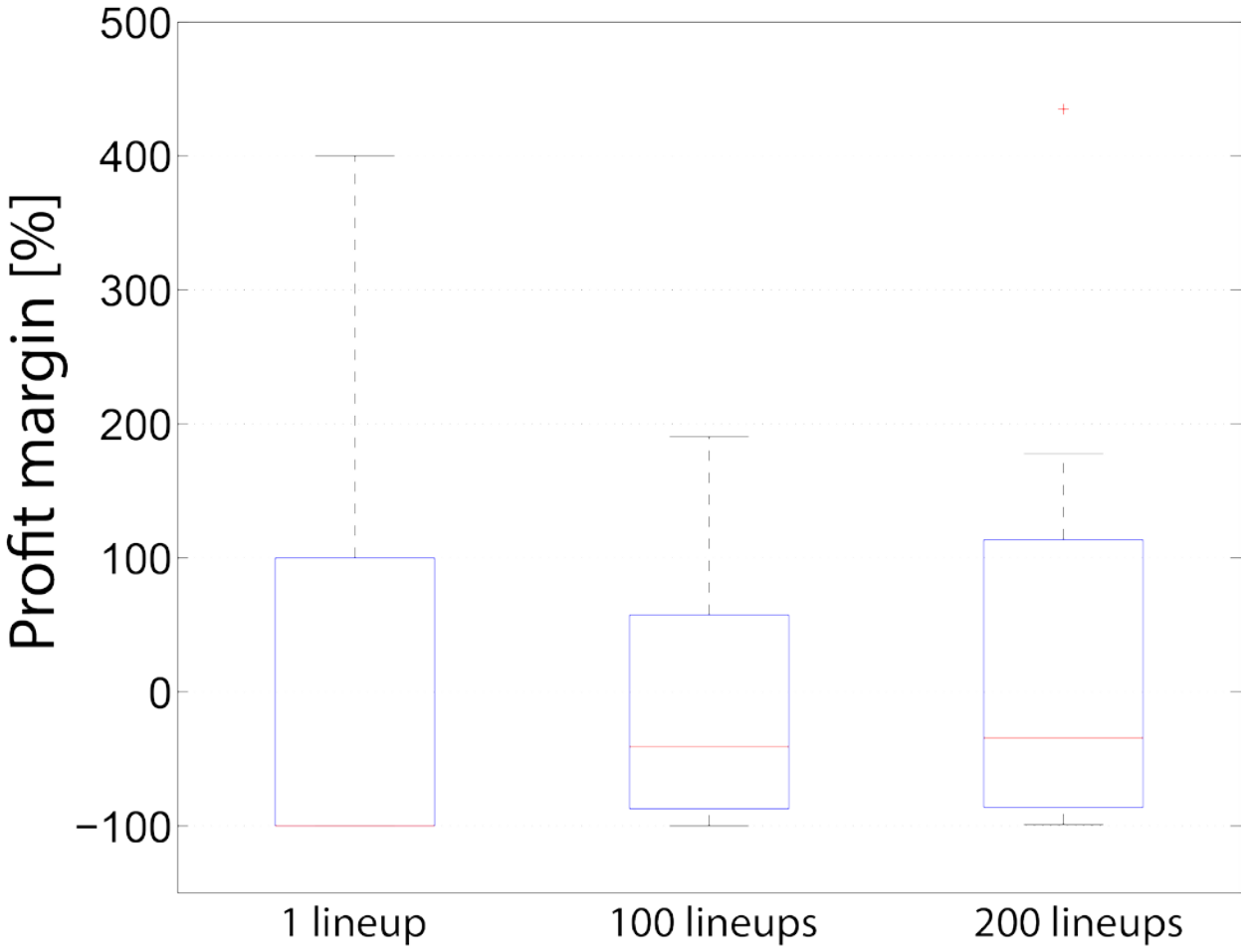
Impact of Stacking

- Used a maximum overlap of 7, and 200 lineups



More stacking is better

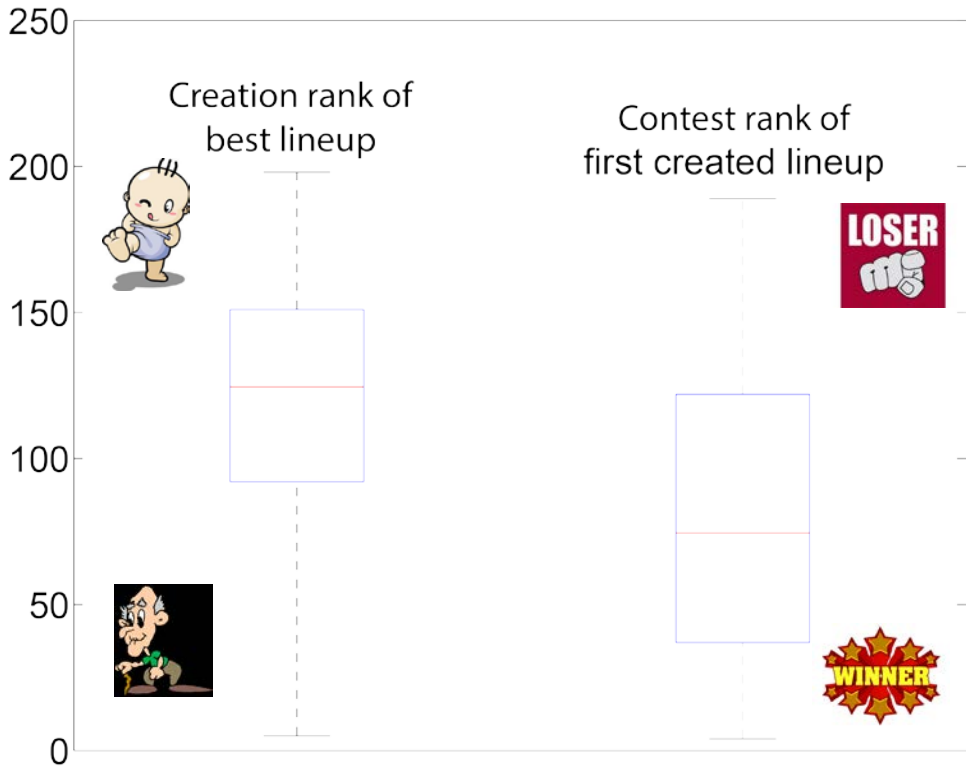
Impact of Number of Lineups



More lineups is better

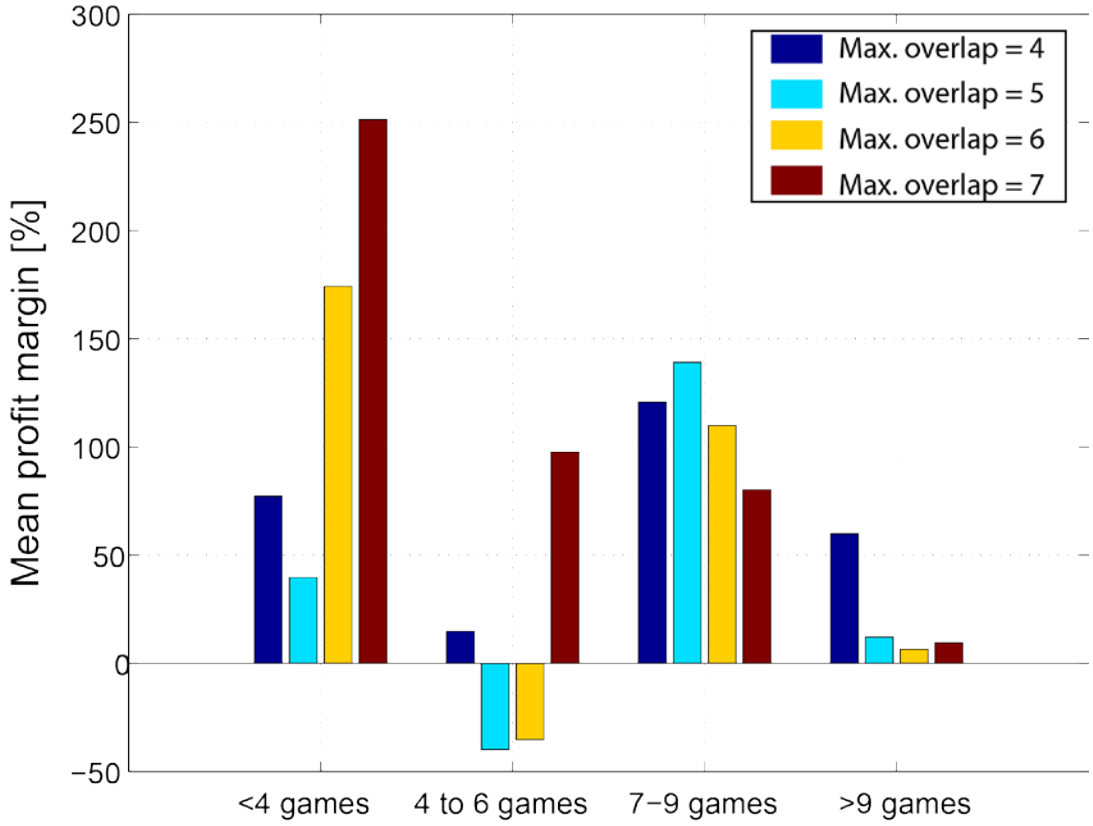
Impact of Lineup Birth Order

- We create lineups sequentially
- Are the best lineups the “oldest” lineups?



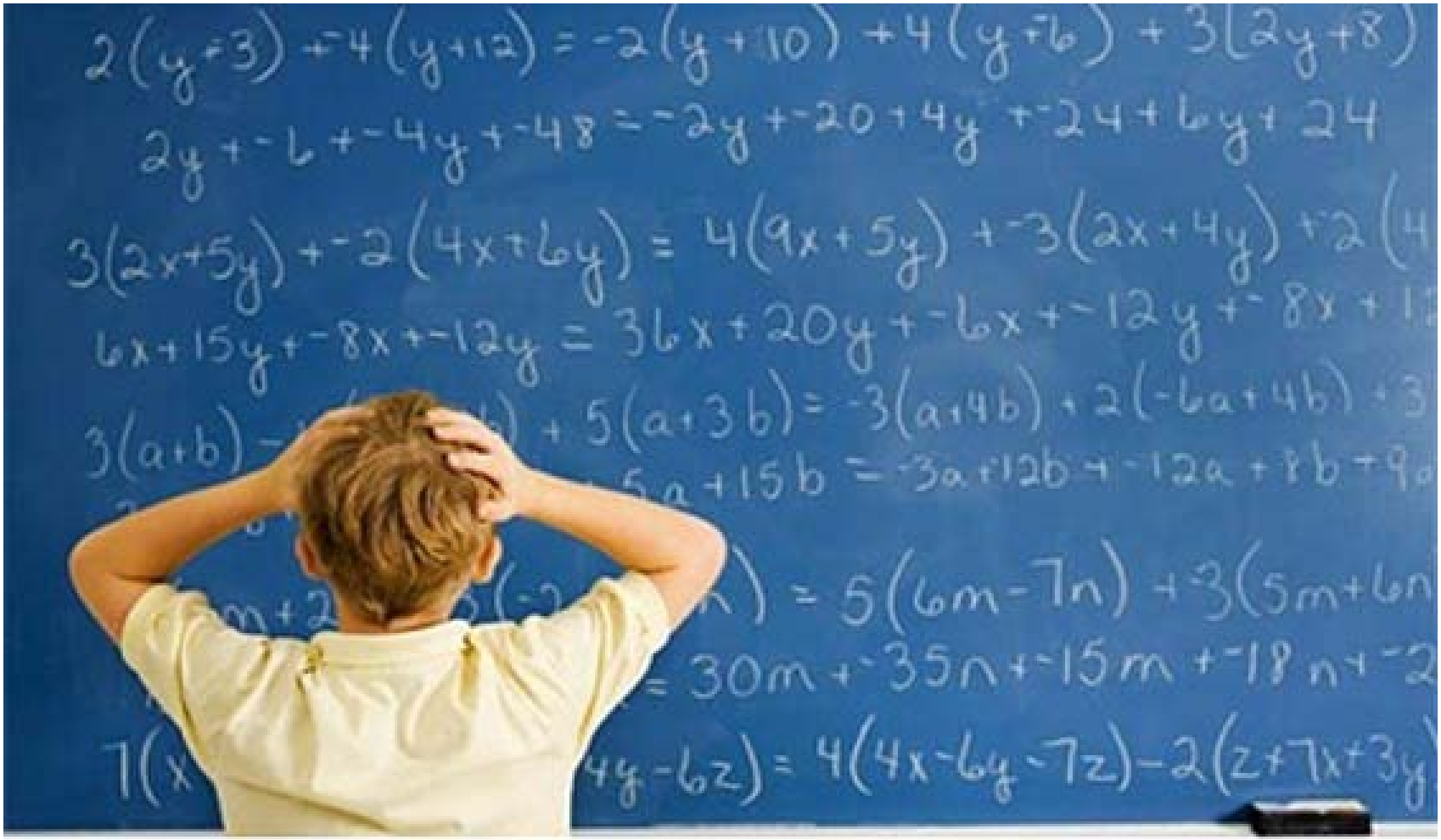
First lineup isn’t usually the best lineup

Impact of Diversity



More games -> Use more diversity

How can you do it?



Lineup Construction Procedure

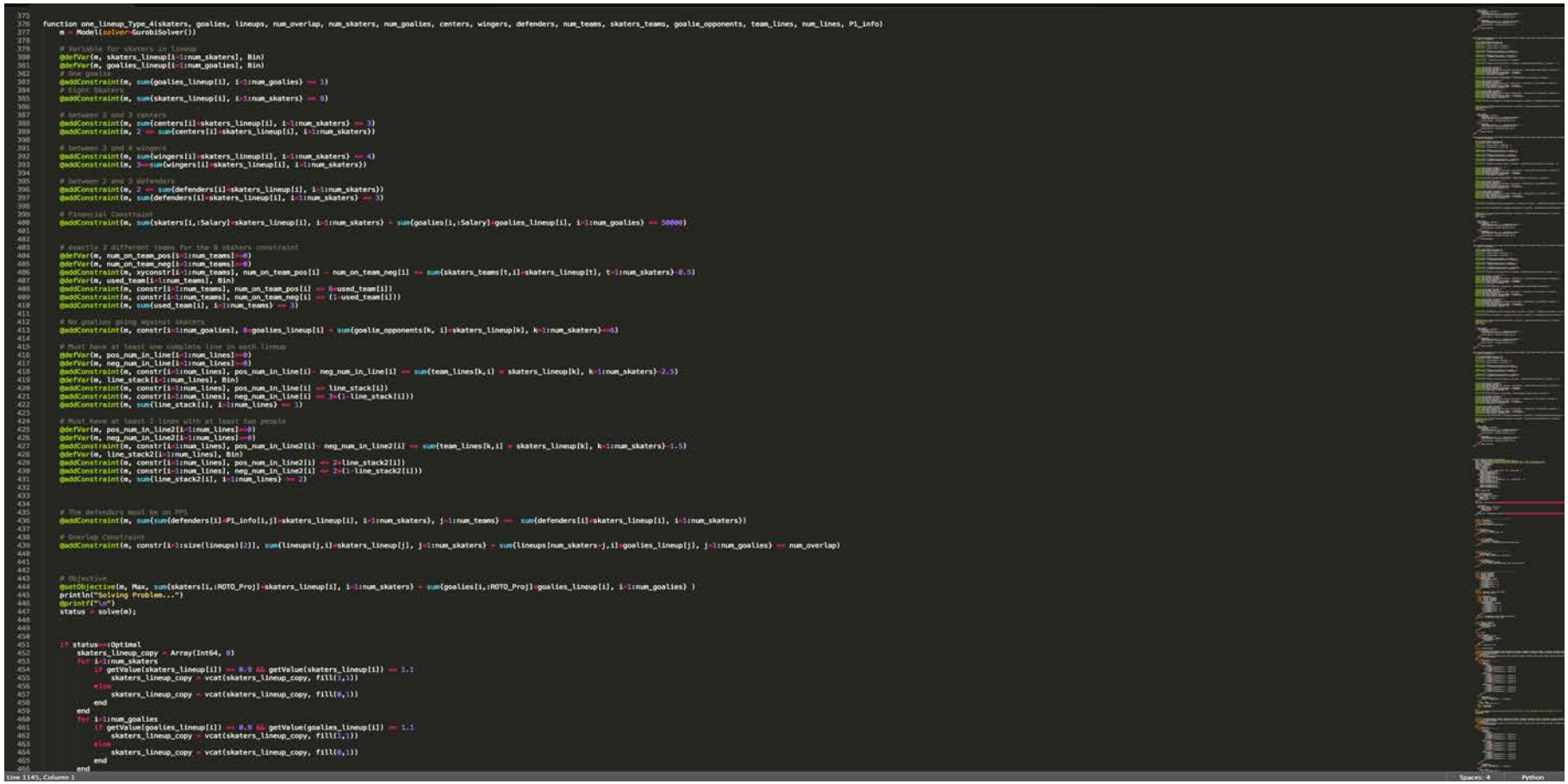
- Get projection data
  - Make sure you wait until the starting goalies are announced
- Solve integer program for each lineup one at a time
  - But add in the new diversity constraints for each new lineup

< 30 Minutes

How can you do it?

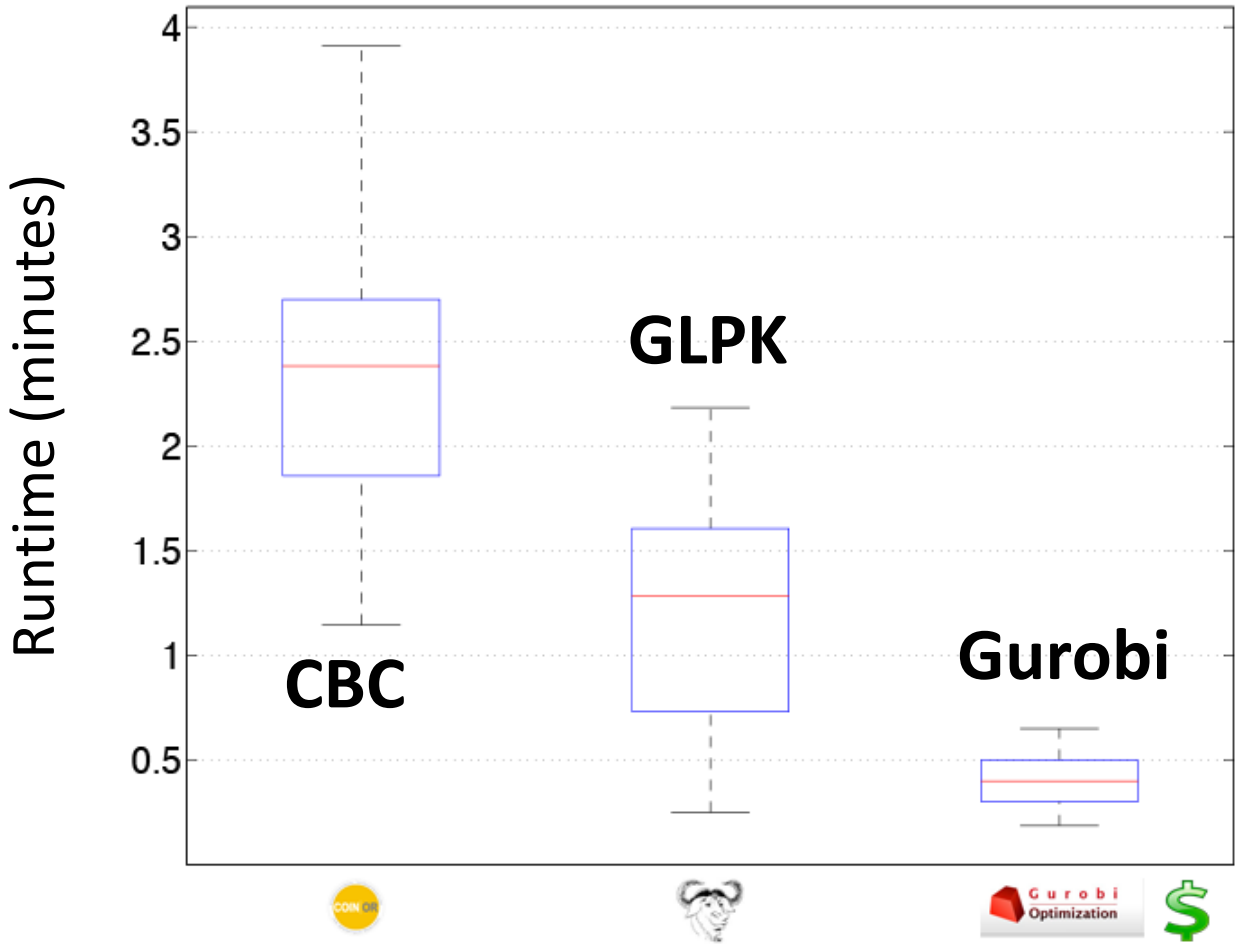
Download Code from Github:

https://github.com/dscotthunter/Fantasy-Hockey-IP-Code



```
function one_lineup_Type_4(skaters, goalies, lineups, num_overlap, num_skaters, num_goalies, centers, wingers, defenders, num_teams, skaters_teams, goalie_opponents, team_lines, num_lines, P1_info)
    m = Model(solver=GurobiSolver())

    # Variable for skaters in lineup
    @defVar(m, skaters_lineup[i=1:num_skaters], Bin)
    @defVar(m, goalies_lineup[i=1:num_goalies], Bin)
    # One goalie
    @addConstraint(m, sum(goalies_lineup[i], i=1:num_goalies) == 1)
    # Eight skaters
    @addConstraint(m, sum(skaters_lineup[i], i=1:num_skaters) == 8)

    # between 2 and 3 centers
    @addConstraint(m, sum(centers[i]*skaters_lineup[i], i=1:num_skaters) <= 3)
    @addConstraint(m, 2 <= sum(centers[i]*skaters_lineup[i], i=1:num_skaters))

    # between 3 and 4 wingers
    @addConstraint(m, sum(wingers[i]*skaters_lineup[i], i=1:num_skaters) <= 4)
    @addConstraint(m, 3<=sum(wingers[i]*skaters_lineup[i], i=1:num_skaters))

    # between 2 and 3 defenders
    @addConstraint(m, 2 <= sum(defenders[i]*skaters_lineup[i], i=1:num_skaters))
    @addConstraint(m, sum(defenders[i]*skaters_lineup[i], i=1:num_skaters) <= 3)

    # Financial Constraint
    @addConstraint(m, sum(skaters[i,:Salary]*skaters_lineup[i], i=1:num_skaters) + sum(goalies[i,:Salary]*goalies_lineup[i], i=1:num_goalies) <= 50000)


    # exactly 3 different teams for the 8 skaters constraint
    @defVar(m, num_on_team_pos[i=1:num_teams]>=0)
    @defVar(m, num_on_team_neg[i=1:num_teams]>=0)
    @addConstraint(m, xyconstr[i=1:num_teams], num_on_team_pos[i] - num_on_team_neg[i] == sum(skaters_teams[t,i]*skaters_lineup[t], t=1:num_skaters)-0.5)
    @defVar(m, used_team[i=1:num_teams], Bin)
    @addConstraint(m, constr[i=1:num_teams], num_on_team_pos[i] <= 6*used_team[i])
    @addConstraint(m, constr[i=1:num_teams], num_on_team_neg[i] <= (1-used_team[i]))
    @addConstraint(m, sum(used_team[i], i=1:num_teams) >= 3)

    # No goalies going against skaters
    @addConstraint(m, constr[i=1:num_goalies], 8*goalies_lineup[i] + sum(goalie_opponents[k, i]*skaters_lineup[k], k=1:num_skaters)<=8)

    # Must have at least one complete line in each lineup
    @defVar(m, pos_num_in_line[i=1:num_lines]>=0)
    @defVar(m, neg_num_in_line[i=1:num_lines]>=0)
    @addConstraint(m, constr[i=1:num_lines], pos_num_in_line[i] - neg_num_in_line[i] == sum(team_lines[k,i] * skaters_lineup[k], k=1:num_skaters)-2.5)
    @defVar(m, line_stack[i=1:num_lines], Bin)
    @addConstraint(m, constr[i=1:num_lines], pos_num_in_line[i] <= line_stack[i])
    @addConstraint(m, constr[i=1:num_lines], neg_num_in_line[i] <= 3*(1-line_stack[i]))
    @addConstraint(m, sum(line_stack[i], i=1:num_lines) >= 1)

    # Must have at least 2 lines with at least two people
    @defVar(m, pos_num_in_line2[i=1:num_lines]>=0)
    @defVar(m, neg_num_in_line2[i=1:num_lines]>=0)
    @addConstraint(m, constr[i=1:num_lines], pos_num_in_line2[i] - neg_num_in_line2[i] == sum(team_lines[k,i] * skaters_lineup[k], k=1:num_skaters)-1.5)
    @defVar(m, line_stack2[i=1:num_lines], Bin)
    @addConstraint(m, constr[i=1:num_lines], pos_num_in_line2[i] <= 2*line_stack2[i])
    @addConstraint(m, constr[i=1:num_lines], neg_num_in_line2[i] <= 2*(1-line_stack2[i]))
    @addConstraint(m, sum(line_stack2[i], i=1:num_lines) >= 2)


    # The defenders must be on PP1
    @addConstraint(m, sum(sum(defenders[i]*P1_info[i,j]*skaters_lineup[i], i=1:num_skaters), j=1:num_teams) == sum(defenders[i]*skaters_lineup[i], i=1:num_skaters))

    # Overlap Constraint
    @addConstraint(m, constr[i=1:size(lineups)[2]], sum(lineups[j,i]*skaters_lineup[j], j=1:num_skaters) + sum(lineups[num_skaters+j,i]*goalies_lineup[j], j=1:num_goalies) <= num_overlap)


    # Objective
    @setObjective(m, Max, sum(skaters[i,:ROTO_Proj]*skaters_lineup[i], i=1:num_skaters) + sum(goalies[i,:ROTO_Proj]*goalies_lineup[i], i=1:num_goalies) )
    println("Solving Problem...")
    @printf("\n")
    status = solve(m);


    if status==:Optimal
        skaters_lineup_copy = Array(Int64, 0)
        for i=1:num_skaters
            if getValue(skaters_lineup[i]) >= 0.9 && getValue(skaters_lineup[i]) <= 1.1
                skaters_lineup_copy = vcat(skaters_lineup_copy, fill(1,1))
            else
                skaters_lineup_copy = vcat(skaters_lineup_copy, fill(0,1))
            end
        end
        for i=1:num_goalies
            if getValue(goalies_lineup[i]) >= 0.9 && getValue(goalies_lineup[i]) <= 1.1
                skaters_lineup_copy = vcat(skaters_lineup_copy, fill(1,1))
            else
                skaters_lineup_copy = vcat(skaters_lineup_copy, fill(0,1))
            end
        end
```

Performance Time



In the paper…

- Consider several strategies
- Different Integer Programming formulations
- Varying prediction models
- Number of lineups
- http://arxiv.org/pdf/1604.01455v1.pdf


YOU CAN DO IT