

Winning at Daily Fantasy Hockey Using Analytics

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



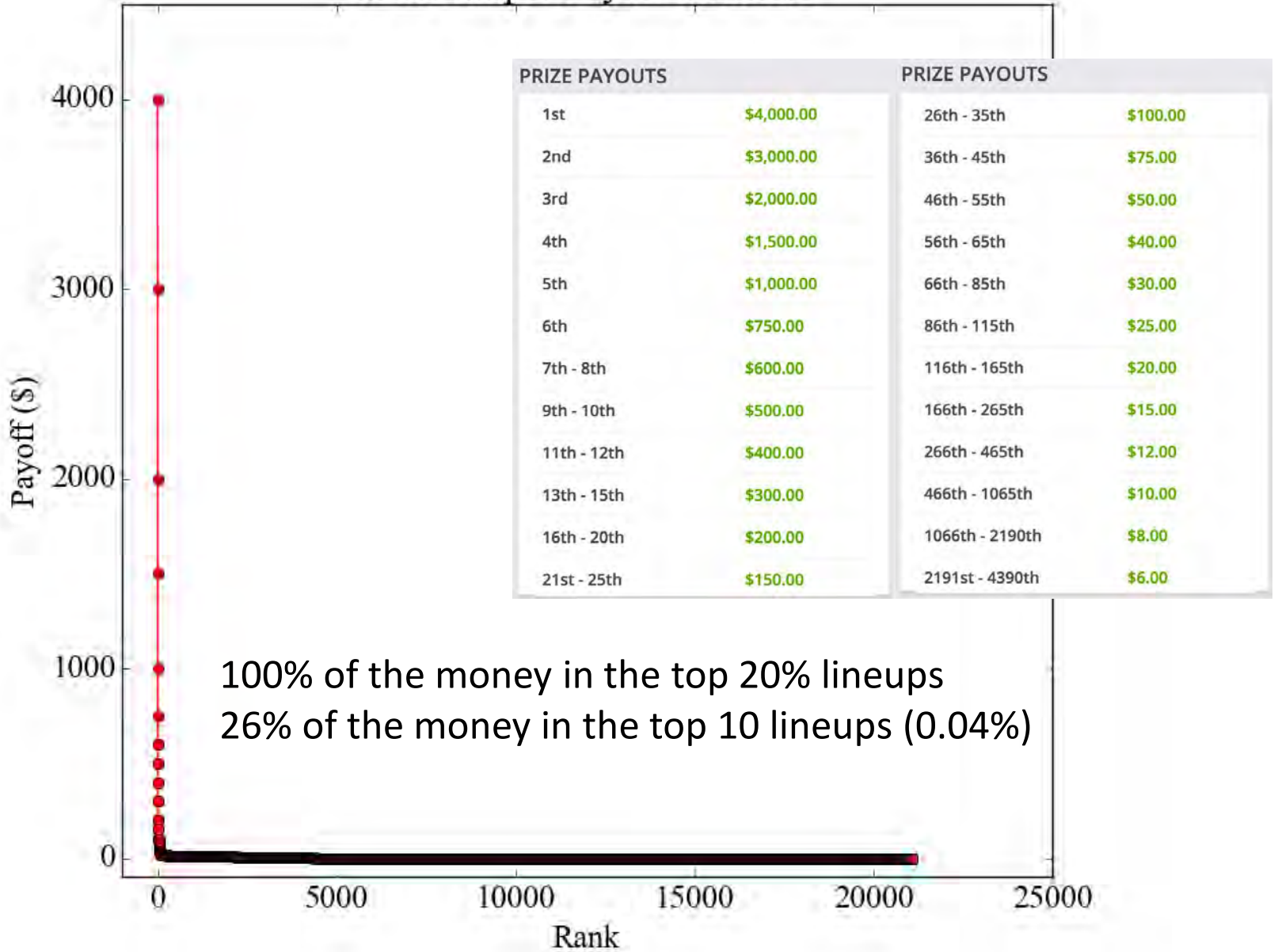
Example Entry



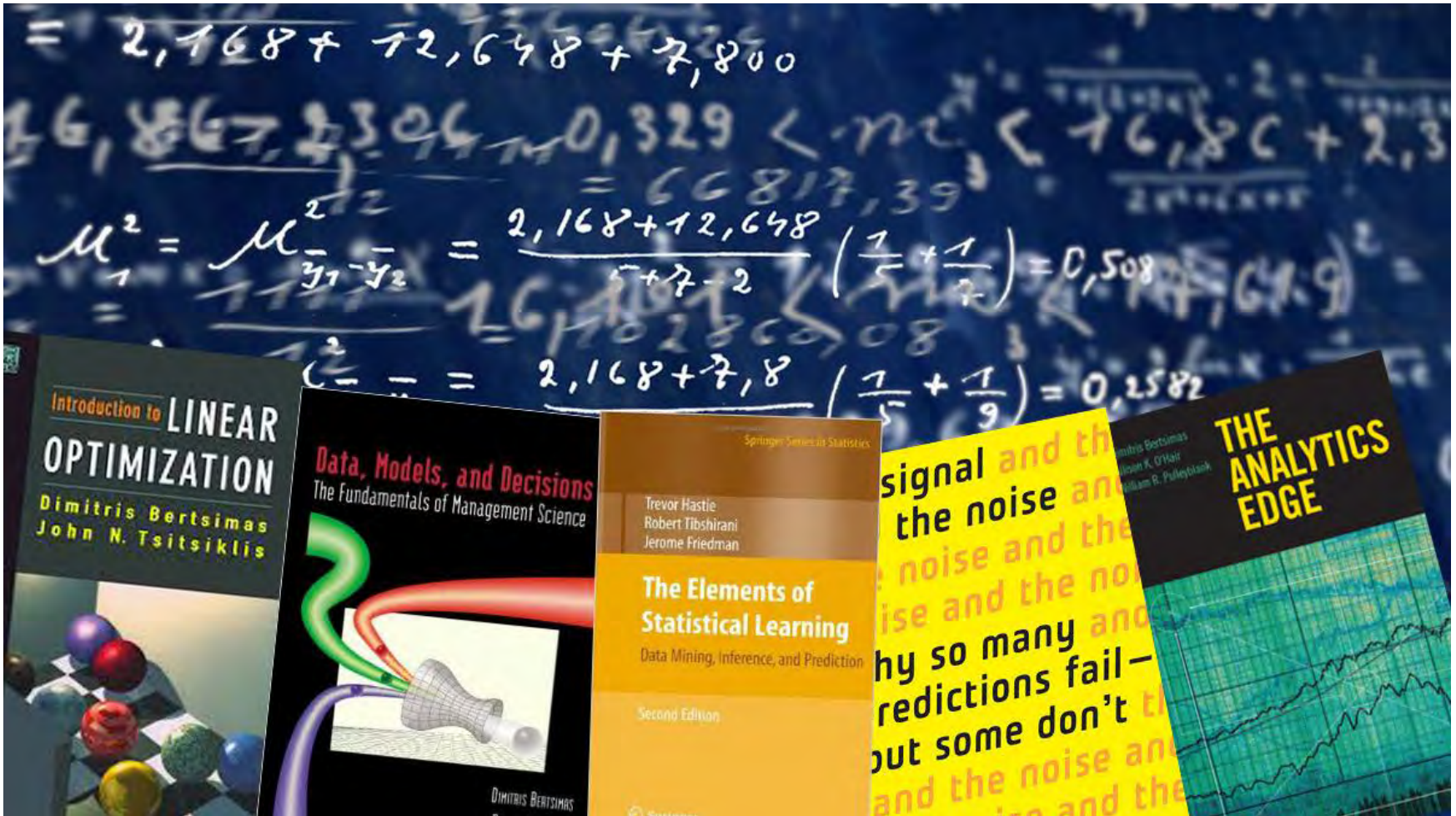
Avg. Rem. / Player: \$0
Rem. Salary: \$0

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	StL@Chi	6.3	\$7,800	✘
UTIL	Blake Wheeler	Wpg@Tor	4.8	\$7,200	✘

\$55K Sniper Payoff Structure

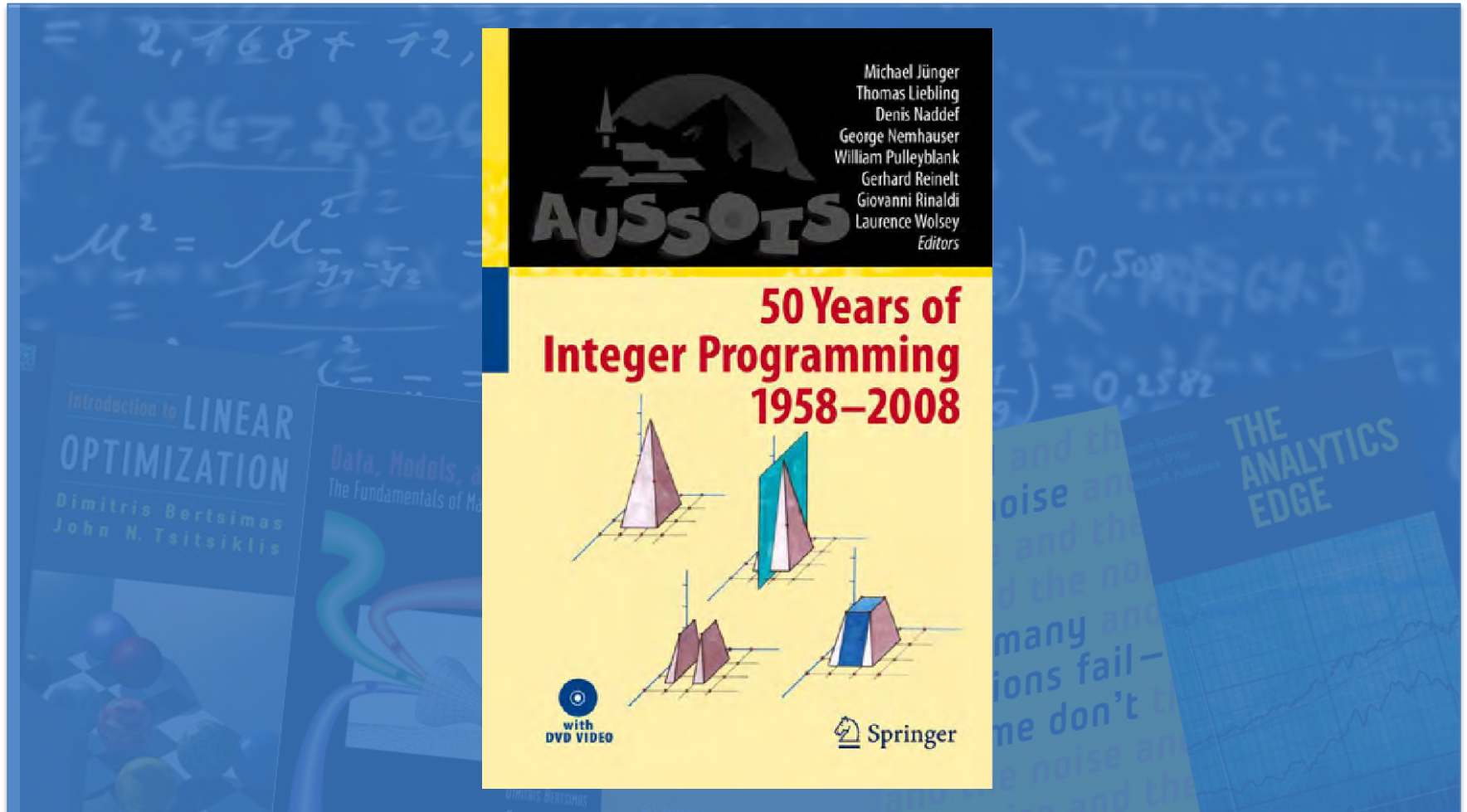


Previous Knowledge: Analytics



Previous Knowledge: Analytics

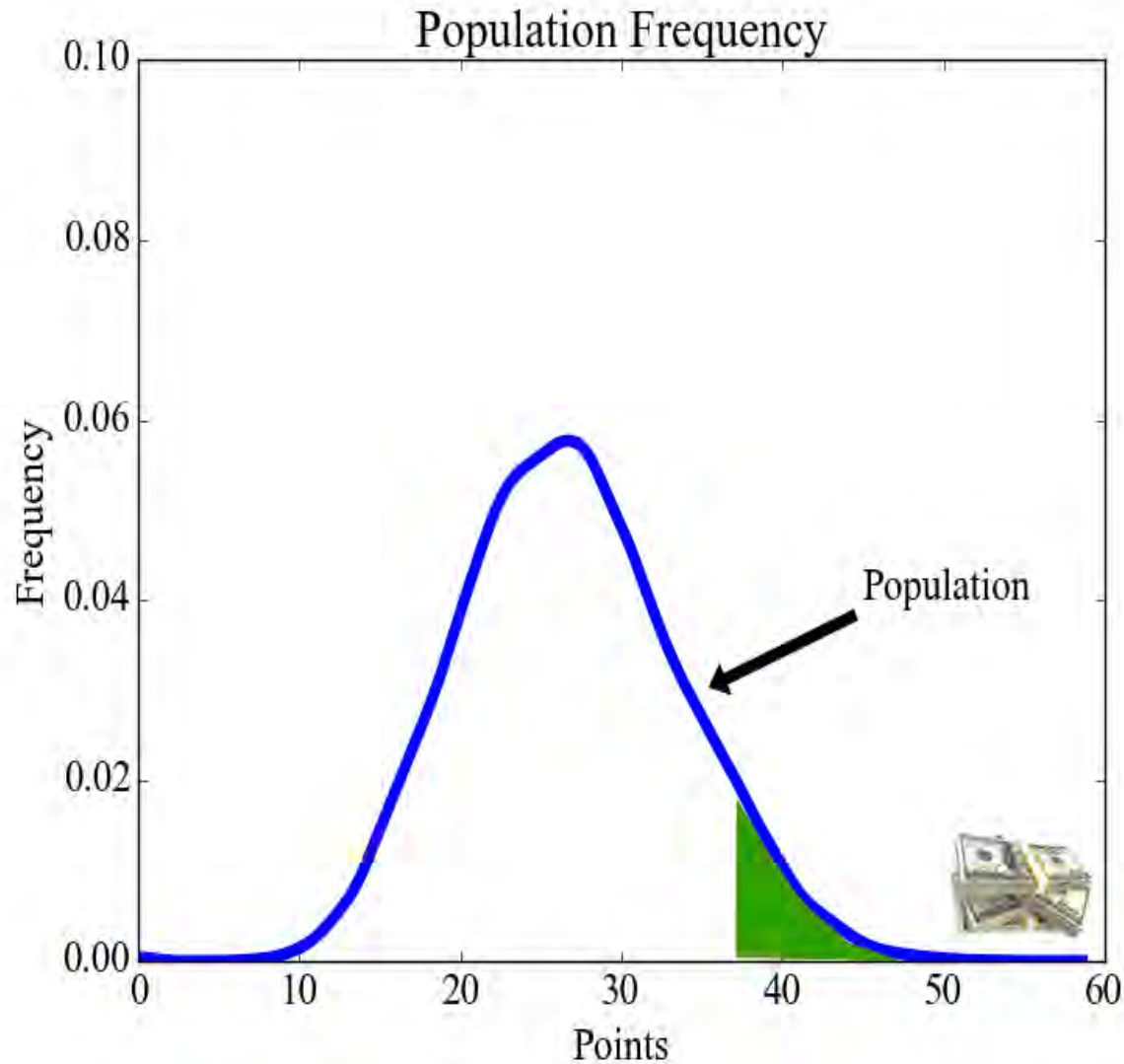
#KnightsOfMIP



Building a Lineup



Using this knowledge...



Were we able to do it?

NHL \$2K Sniper [\$2,000 Guaranteed]

Rank	Player	Score	Prize
1st	zlisto	54.50	\$150.00
3rd	zlisto	51.50	\$90.00
9th	zlisto	49.50	\$30.00
23rd	zlisto	46.00	\$12.75
28th	zlisto	45.50	\$15.00
28th	zlisto	45.50	\$15.00

November 15, 2015

NHL \$40K Sniper [\$40,000 Guaranteed]

Rank	Player	Score	Prize
2nd	zlisto	61.30	\$2,000.00
21st	zlisto	57.30	\$50.00
21st	zlisto	57.30	\$50.00
40th	zlisto	56.10	\$40.00
42nd	zlisto	55.70	\$40.00
81st	zlisto	54.10	\$40.00

November 16, 2015

NHL \$80K Tuesday Special [\$80,000 Guaranteed]

Rank	Player	Score	Prize
3rd	zlisto	54.60	\$3,000.00
6th	zlisto	52.80	\$1,000.00
7th	zlisto	52.30	\$800.00
10th	zlisto	50.60	\$600.00
11th	zlisto	50.30	\$500.00
15th	zlisto	50.10	\$500.00

November 17, 2015

NHL \$45K Sniper [\$45,000 Guaranteed]

Rank	Player	Score	Prize
1st	zlisto	52.60	\$3,000.00
8th	zlisto	49.60	\$275.00
57th	zlisto	45.60	\$50.00
57th	zlisto	45.60	\$50.00
83rd	zlisto	44.60	\$40.00
83rd	zlisto	44.60	\$40.00

November 23, 2015

200 lineups

Policy Change

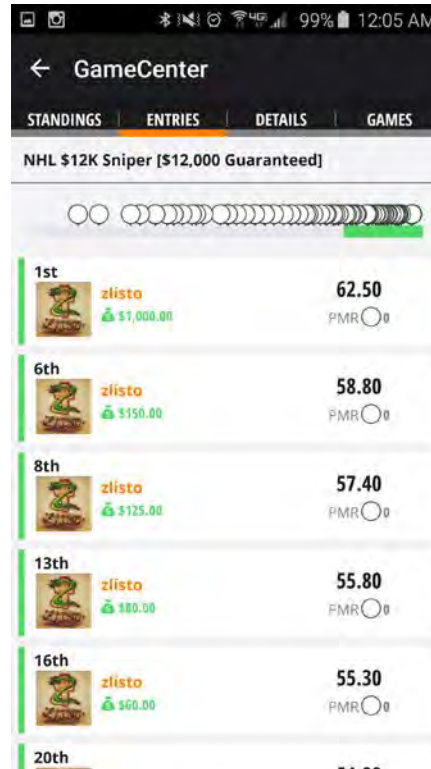


Policy Change



200 lineups -> 100 lineups

Were we able to continue it?



The screenshot shows the GameCenter interface for the 'NHL \$12K Sniper [\$12,000 Guaranteed]' tournament. At the top, there are navigation tabs for STANDINGS, ENTRIES, DETAILS, and GAMES. Below the tabs is a progress indicator consisting of a series of small circles, with the first one filled. The main content area displays a list of players, all named 'zlisto', with their scores and prize amounts. The 1st place player has a score of 62.50 and a prize of \$1,000.00. The 6th place player has a score of 58.80 and a prize of \$150.00. The 8th place player has a score of 57.40 and a prize of \$125.00. The 13th place player has a score of 55.80 and a prize of \$80.00. The 16th place player has a score of 55.30 and a prize of \$60.00. The 20th place player has a score of 54.00 and a prize of \$40.00. Each player's entry includes a small icon of a snake and the text 'zlisto'.

Rank	Player	Score	Prize
1st	zlisto	62.50	\$1,000.00
6th	zlisto	58.80	\$150.00
8th	zlisto	57.40	\$125.00
13th	zlisto	55.80	\$80.00
16th	zlisto	55.30	\$60.00
20th	zlisto	54.00	\$40.00

December 12, 2015

100 lineups

Legal Disclaimer:

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

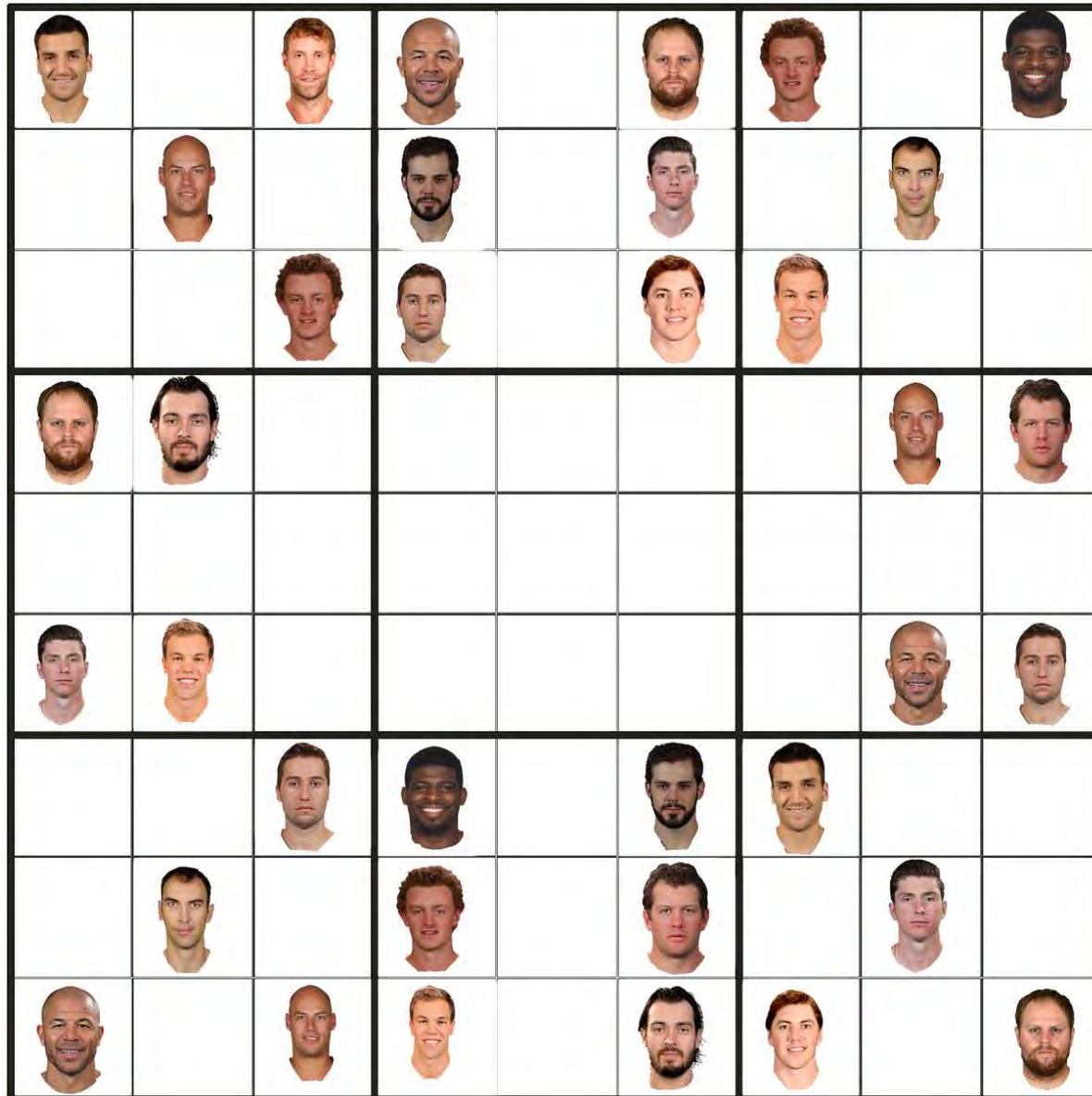
Integer Programming Formulation

- 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

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

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

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

First Attempt...

\$6400 \$7200 \$4200 \$4100 \$5300 \$4400 \$4800 \$5800 \$7800

W UTIL D D C C W W G

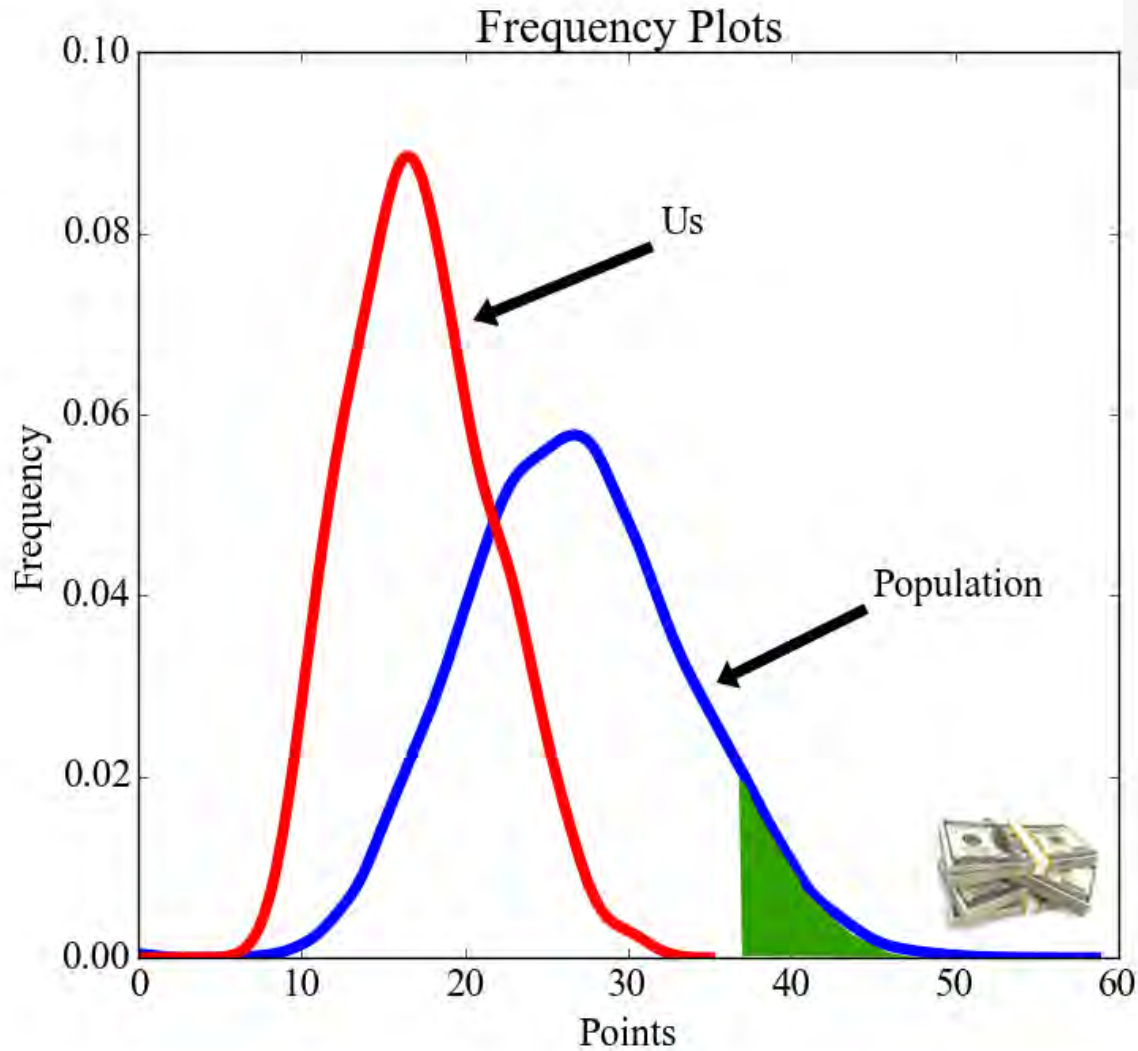


> 3 Different Teams

First Attempt...



BROKE



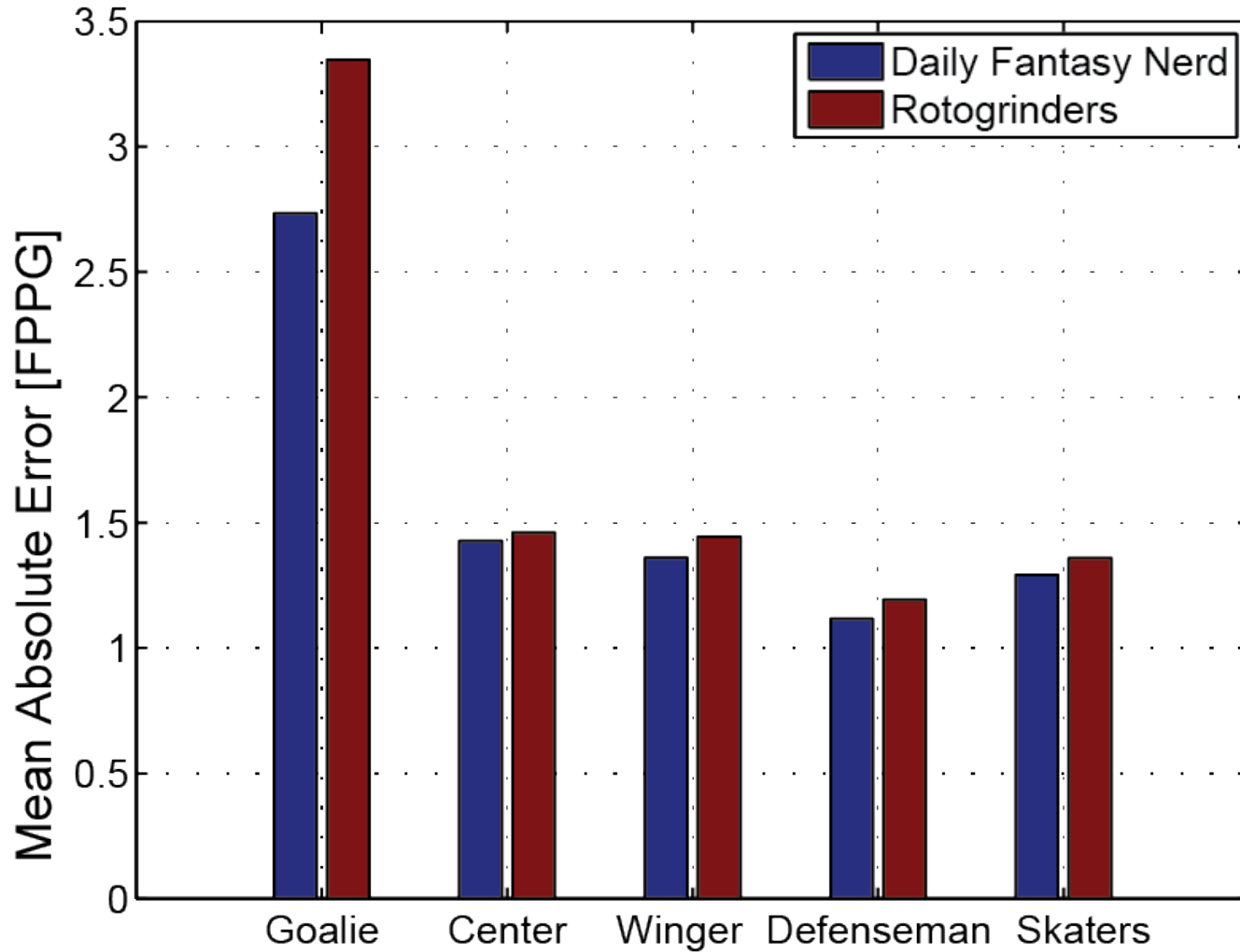
Second Attempt...

- Must increase our mean points...
- Solution:

USE EXPERT PREDICTIONS



Prediction Errors



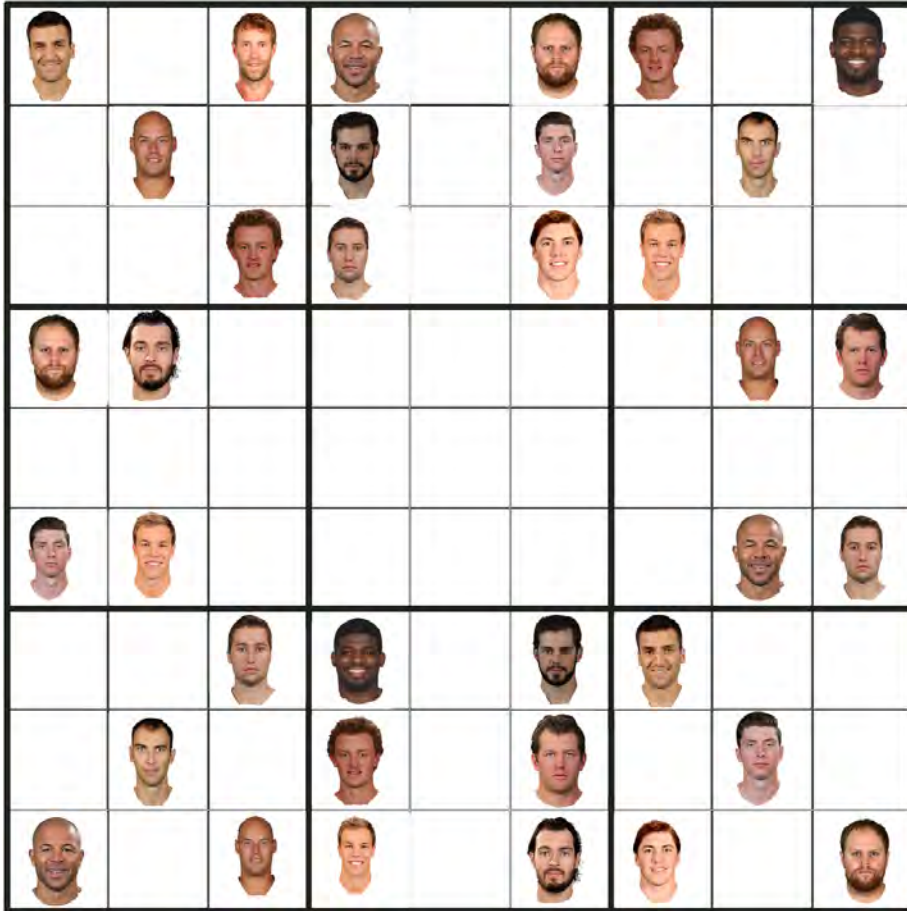
Maximize Points

- 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

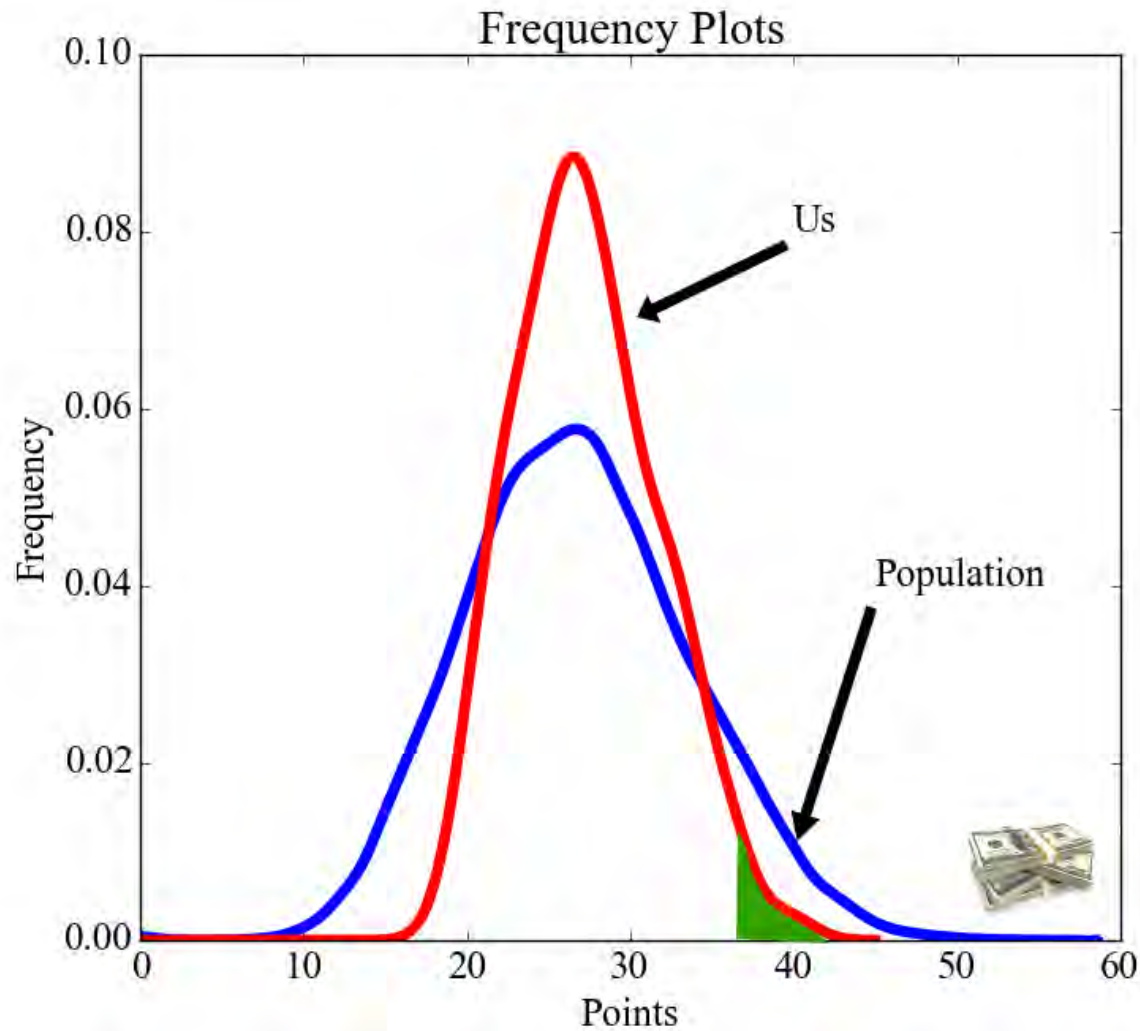
New Lineup

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?

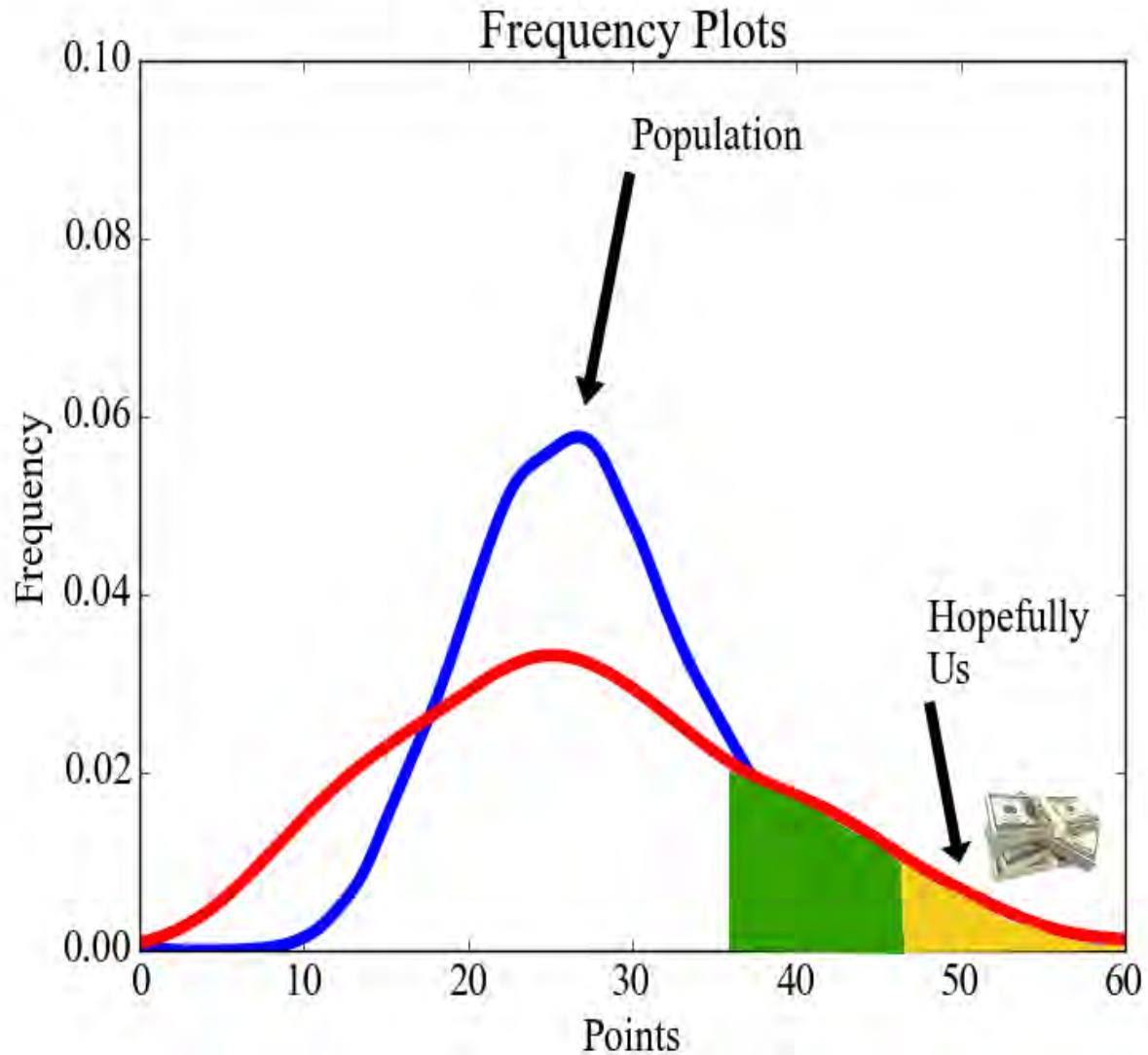
- Three D's of Finance:



So what do we do?



By doing this...



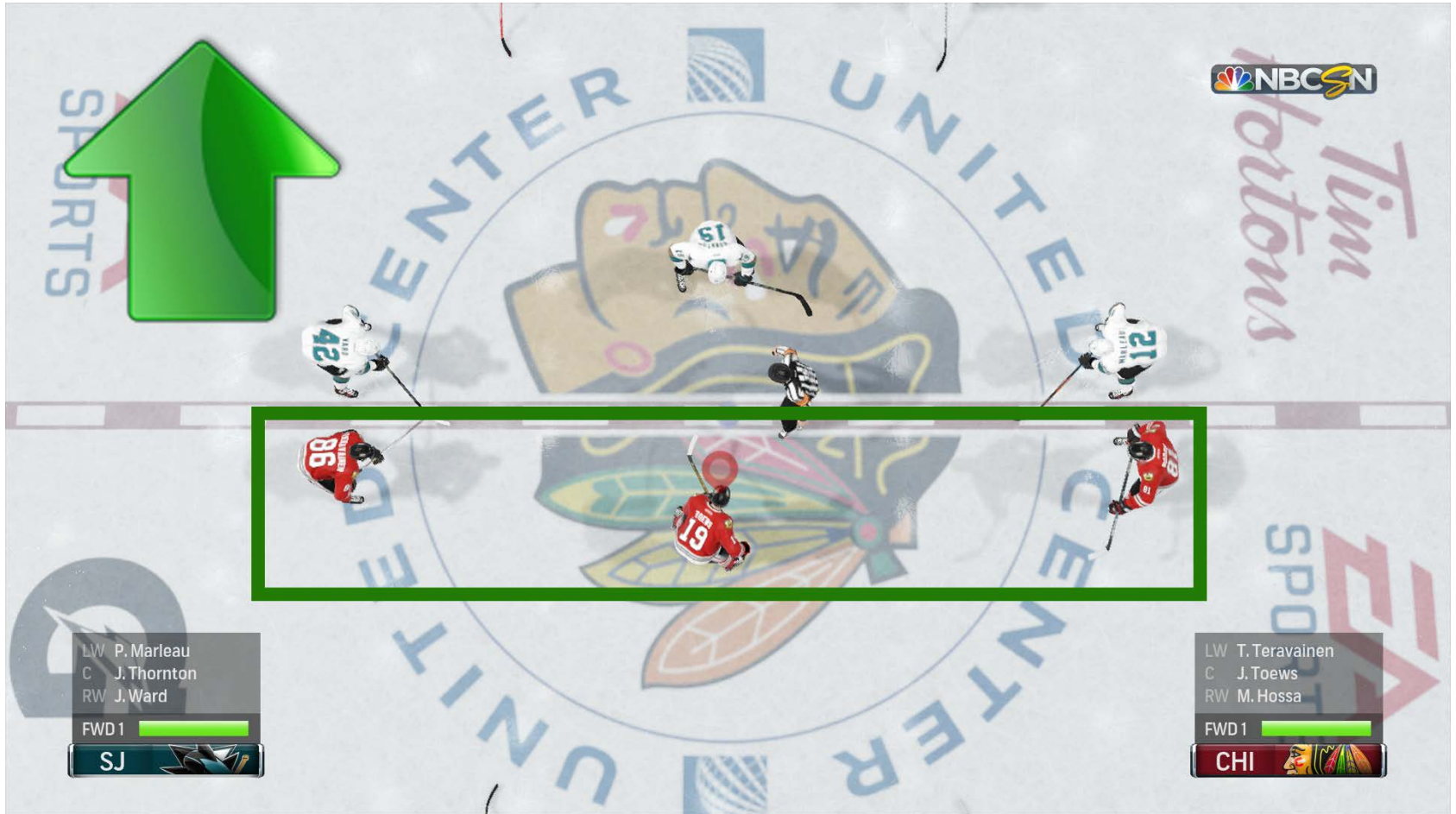
Stacking Lineups

- 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”

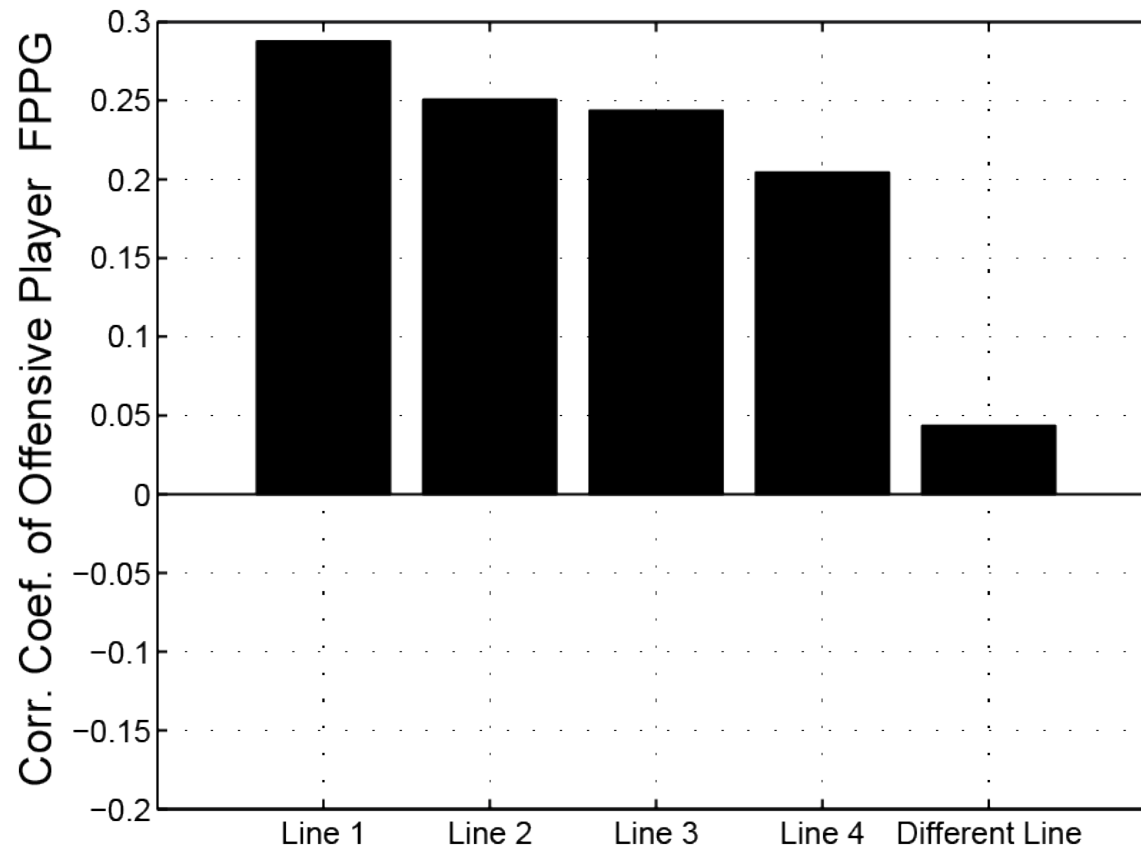
Structural Correlations - Teams



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

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

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

$$v_i \in \{0, 1\}, \quad \forall i \in \{1, \dots, N_L\}.$$

2 partial lines constraint

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

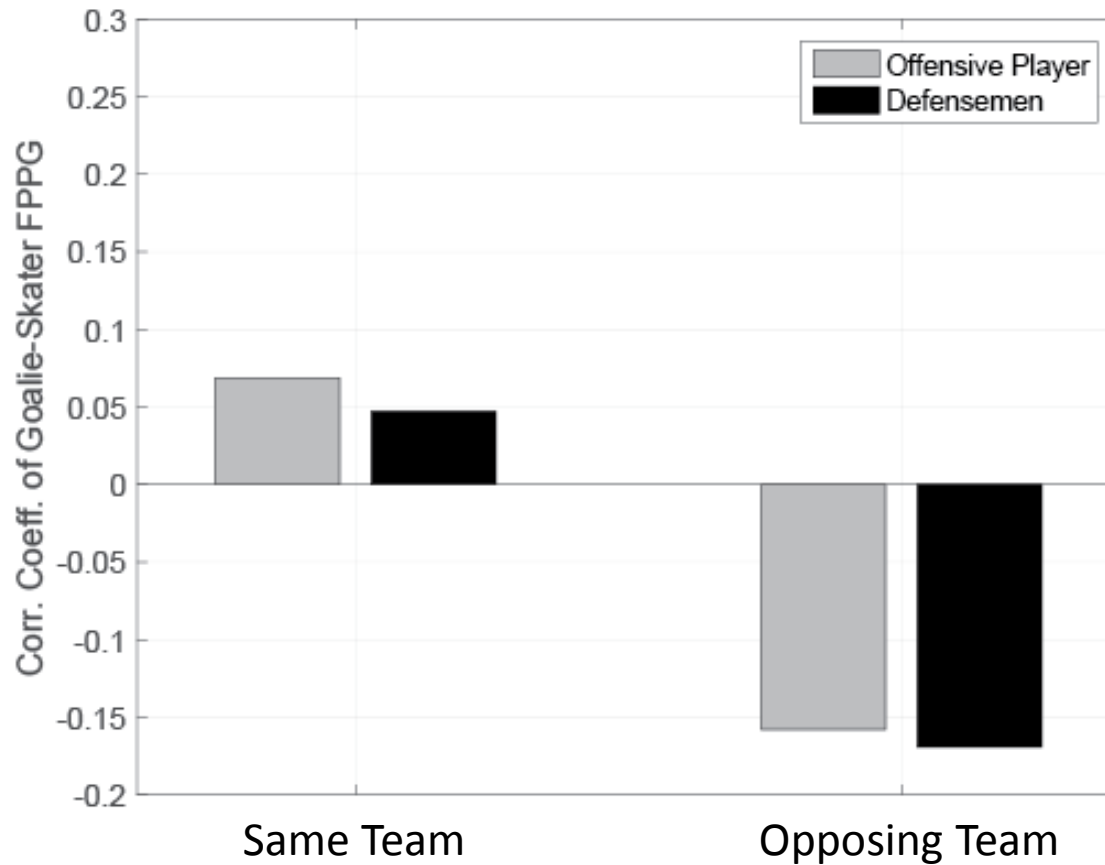
$$\sum_{i=1}^{N_L} w_i \geq 2$$

$$w_i \in \{0, 1\}, \quad \forall i \in \{1, \dots, N_L\}.$$

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

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

Feasible

Team

Goalie

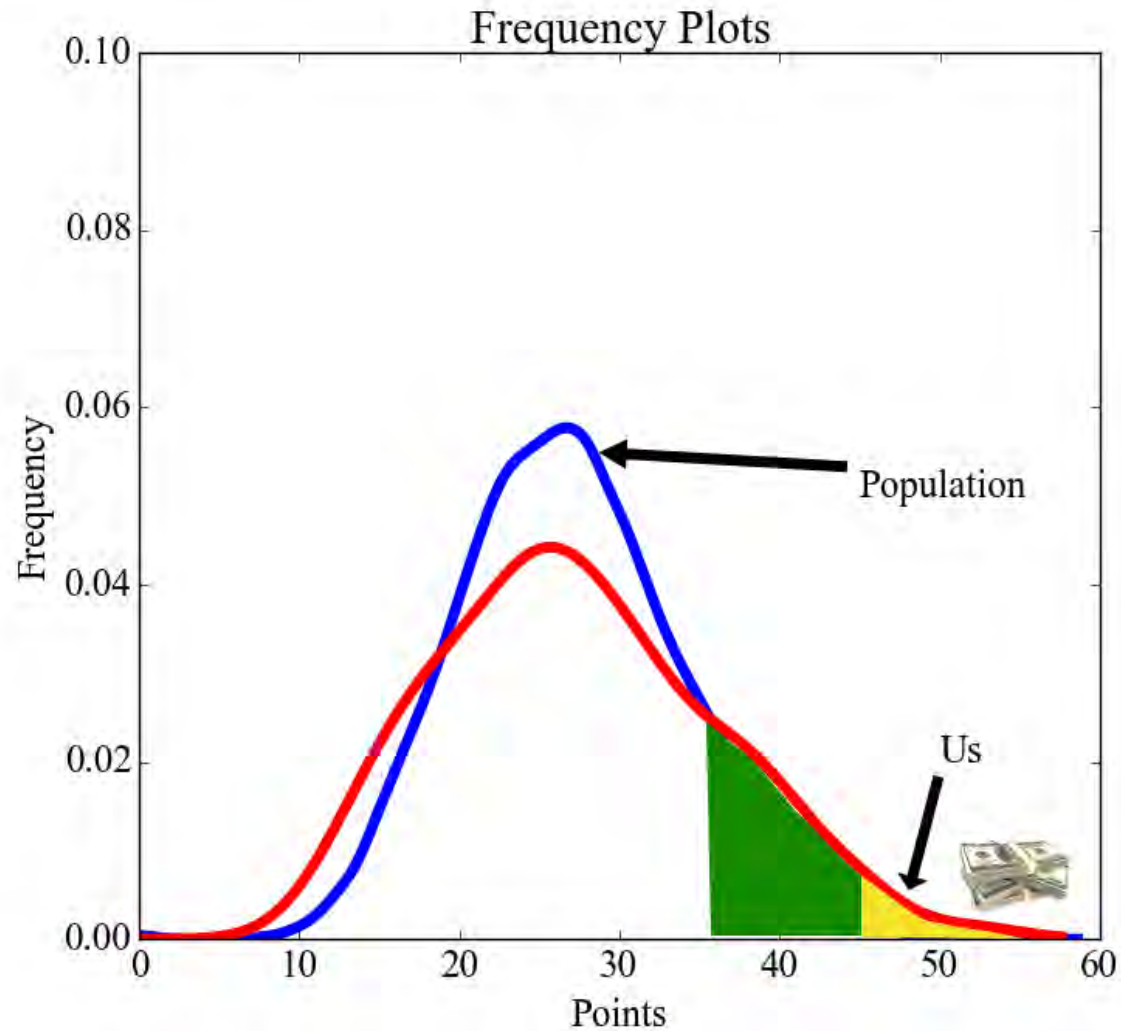
Line

Line

Not
Against



Second Attempt...





Lineup Diversity

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

Diversity constraint

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

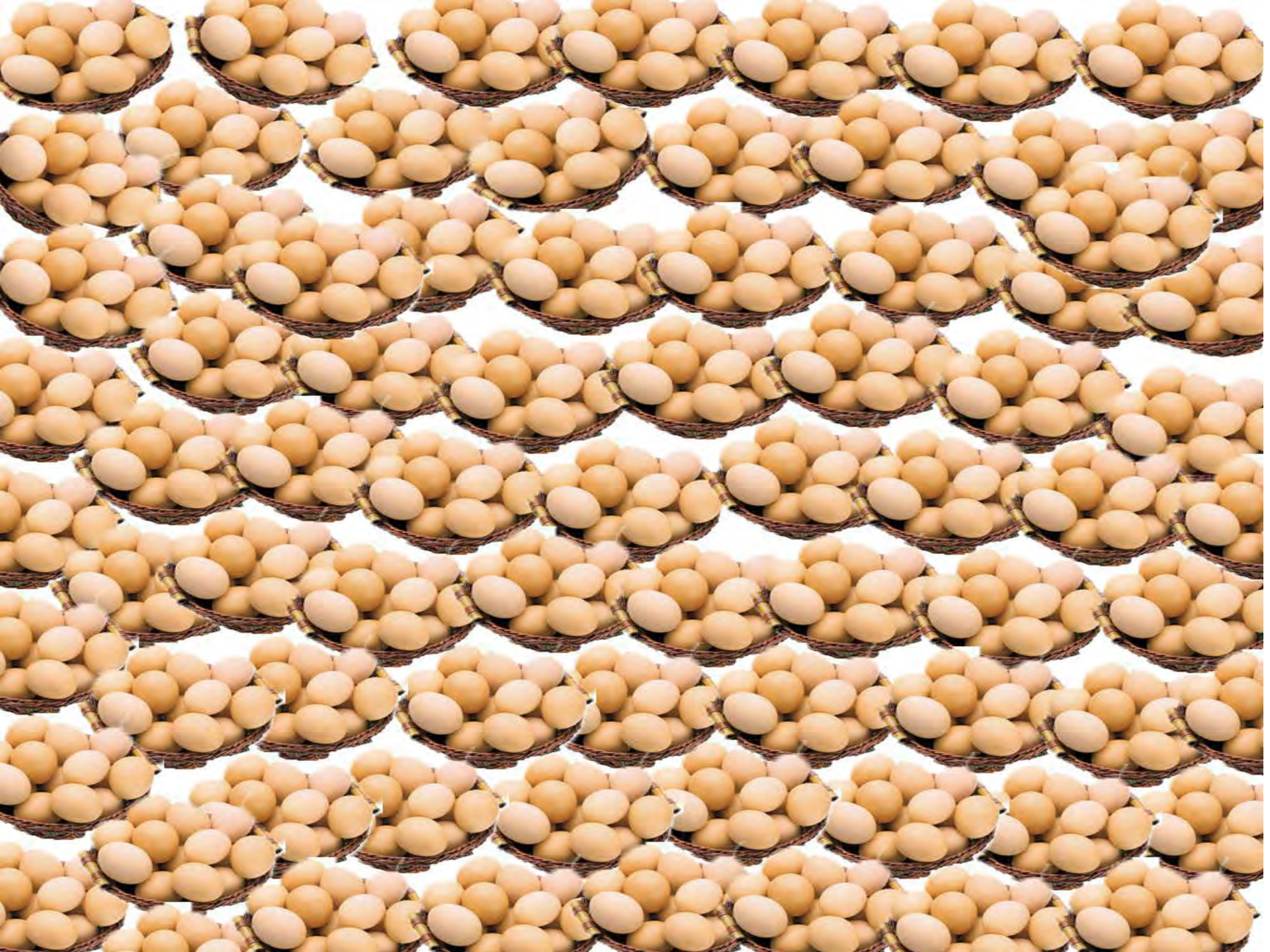
To Review...

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	STL@Chi	6.3	\$7,800	✘
UTIL	Blake Wheeler	Wpg@Tor	4.8	\$7,200	✘

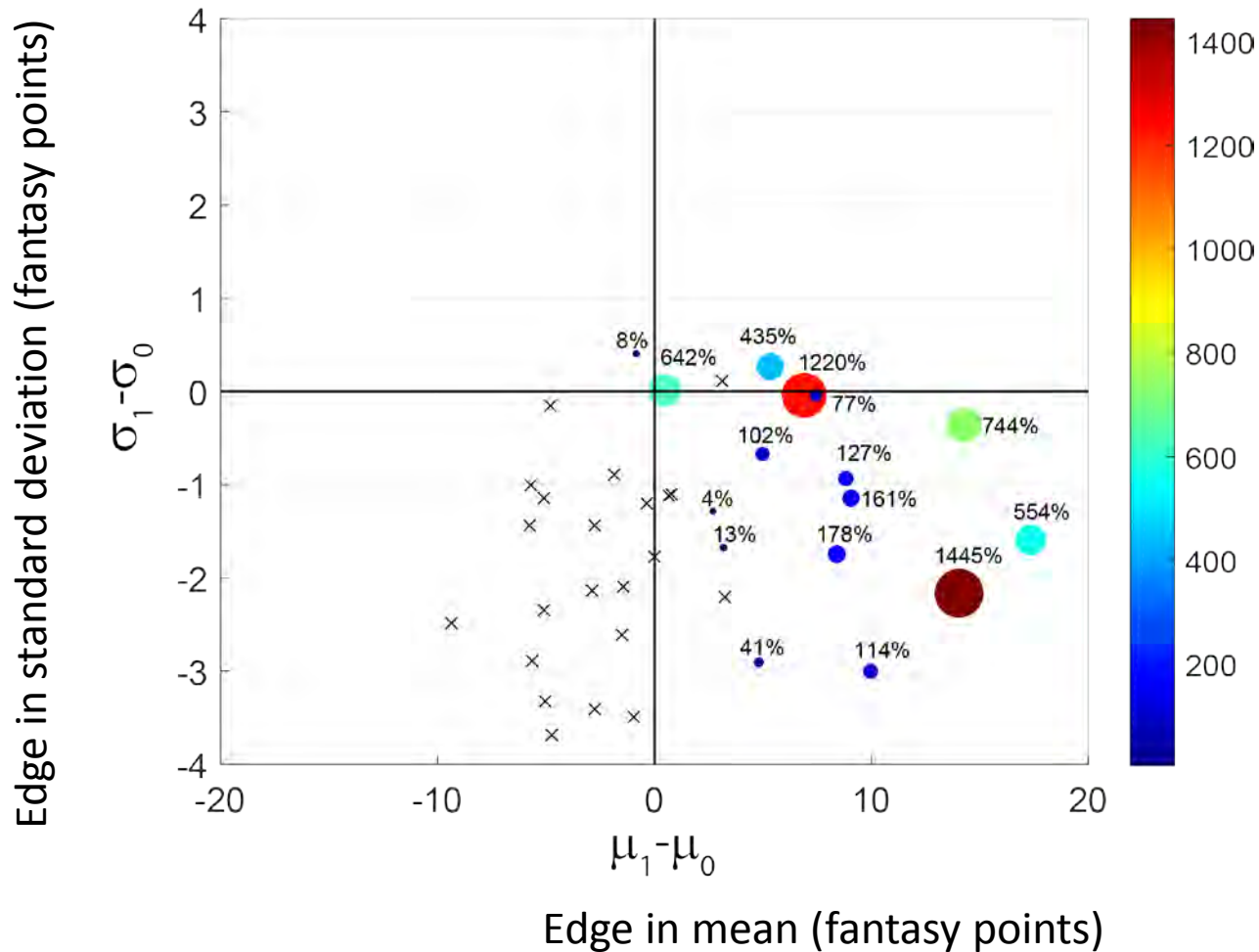




PERFORMANCE ON REAL CONTESTS

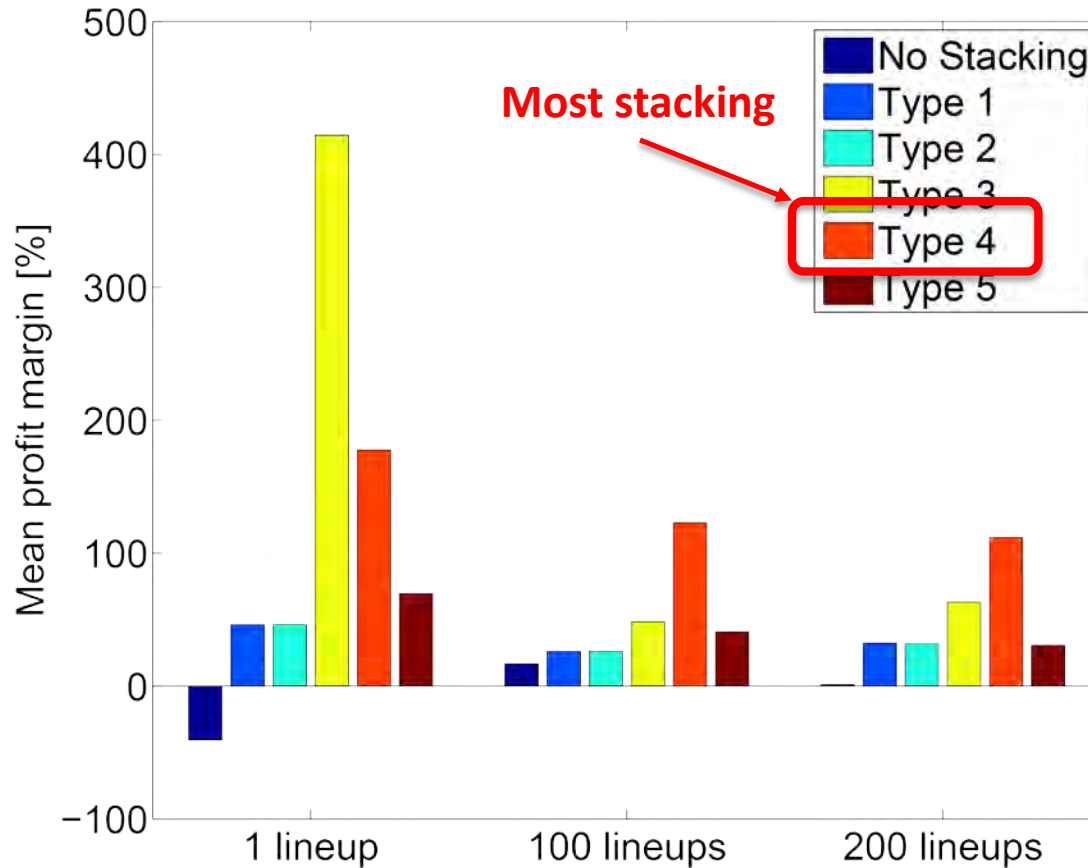
Performance on Real Contests

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



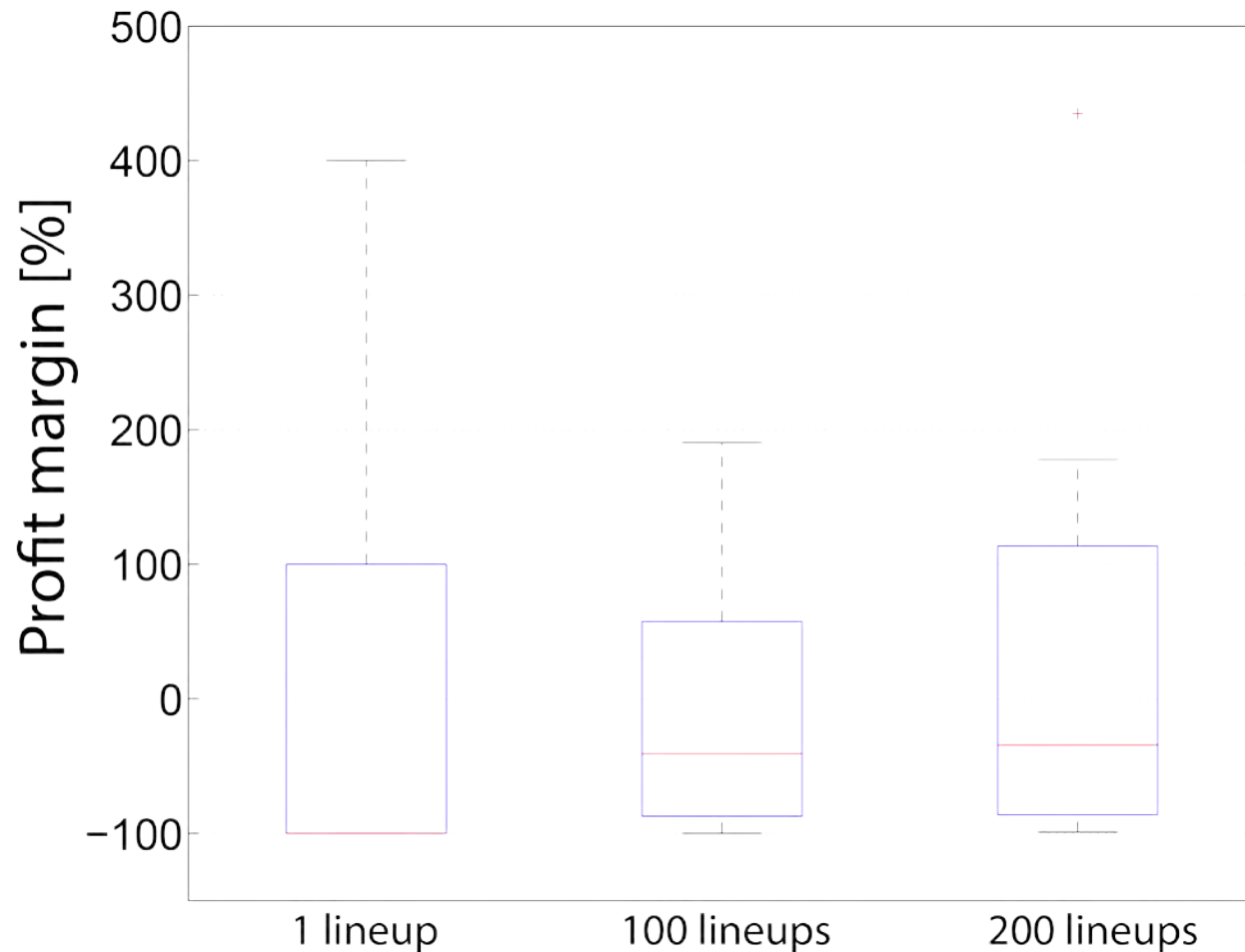
Impact of Stacking

- Used a maximum overlap of 7, and 200 lineups



More stacking is better

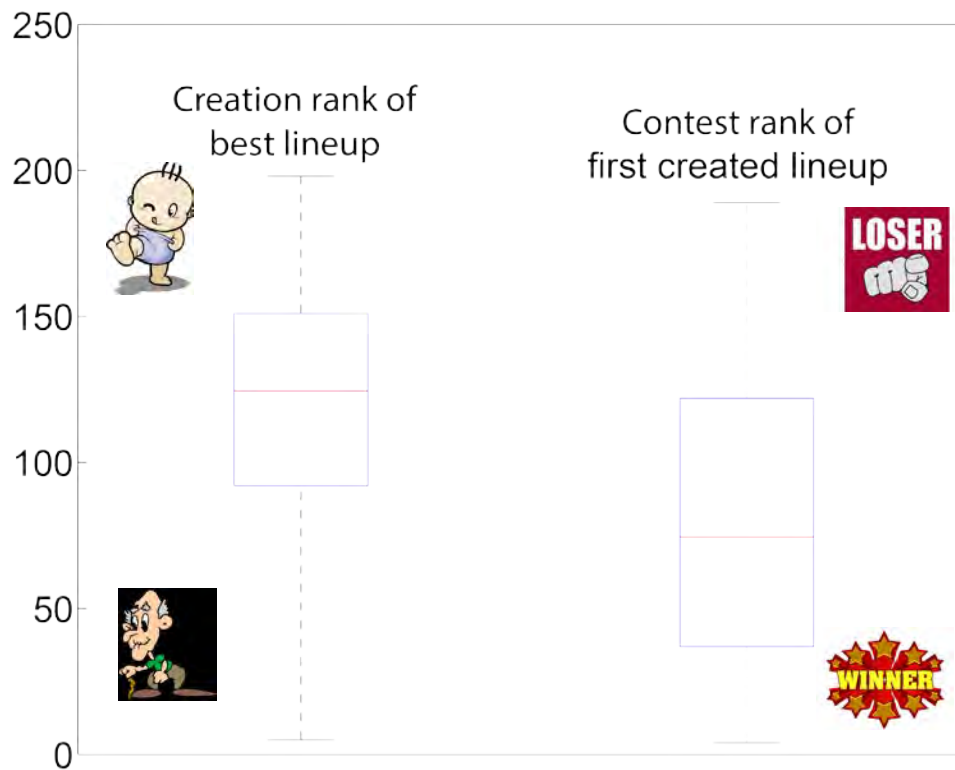
Impact of Number of Lineups



More lineups is better

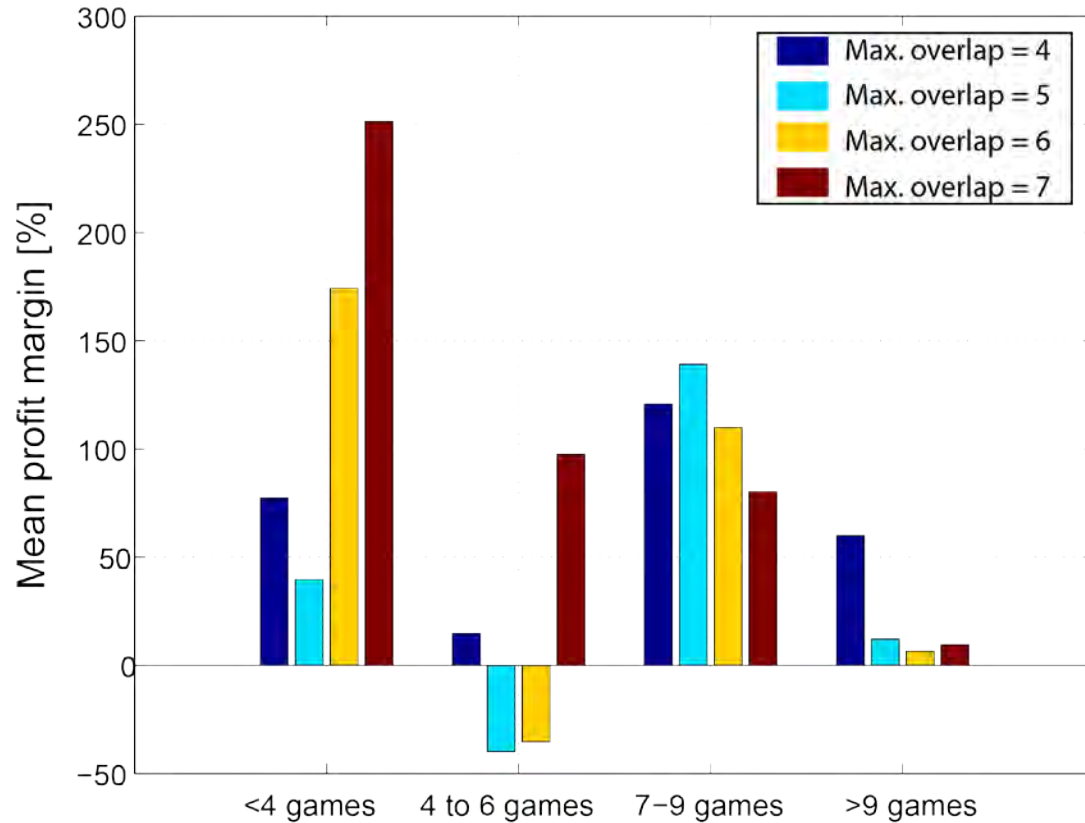
Impact of Lineup Birth Order

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



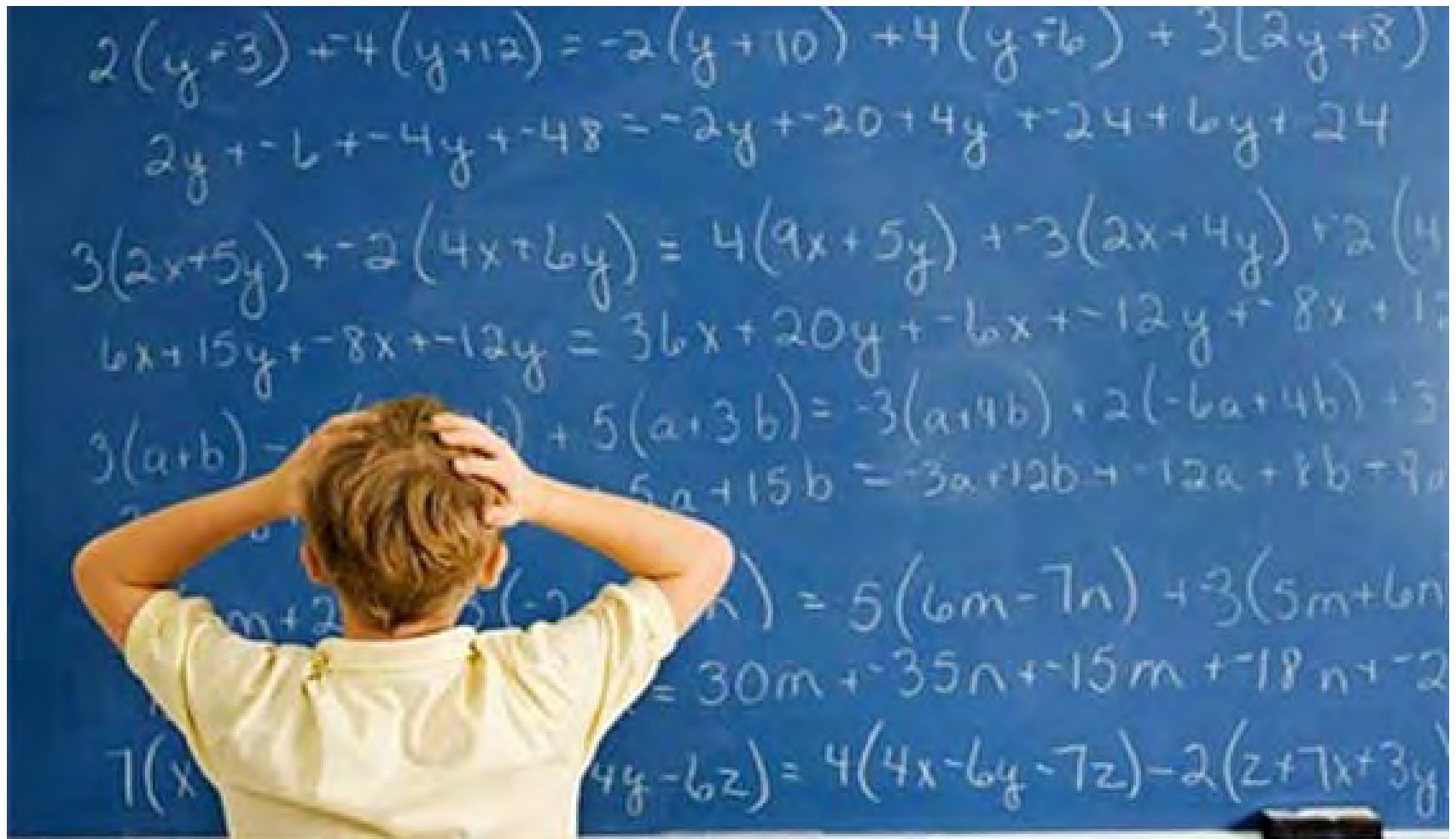
First lineup isn't usually the best lineup

Impact of Diversity



More games -> Use more diversity

How can you do it?



Lineup Construction Procedure

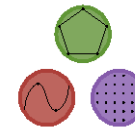
- Get projection data
 - Make sure you wait until the starting goalies are announced
- Solve integer program for each lineup one at a time
 - But add in the new diversity constraints for each new lineup



< 30 Minutes



How can you do it?



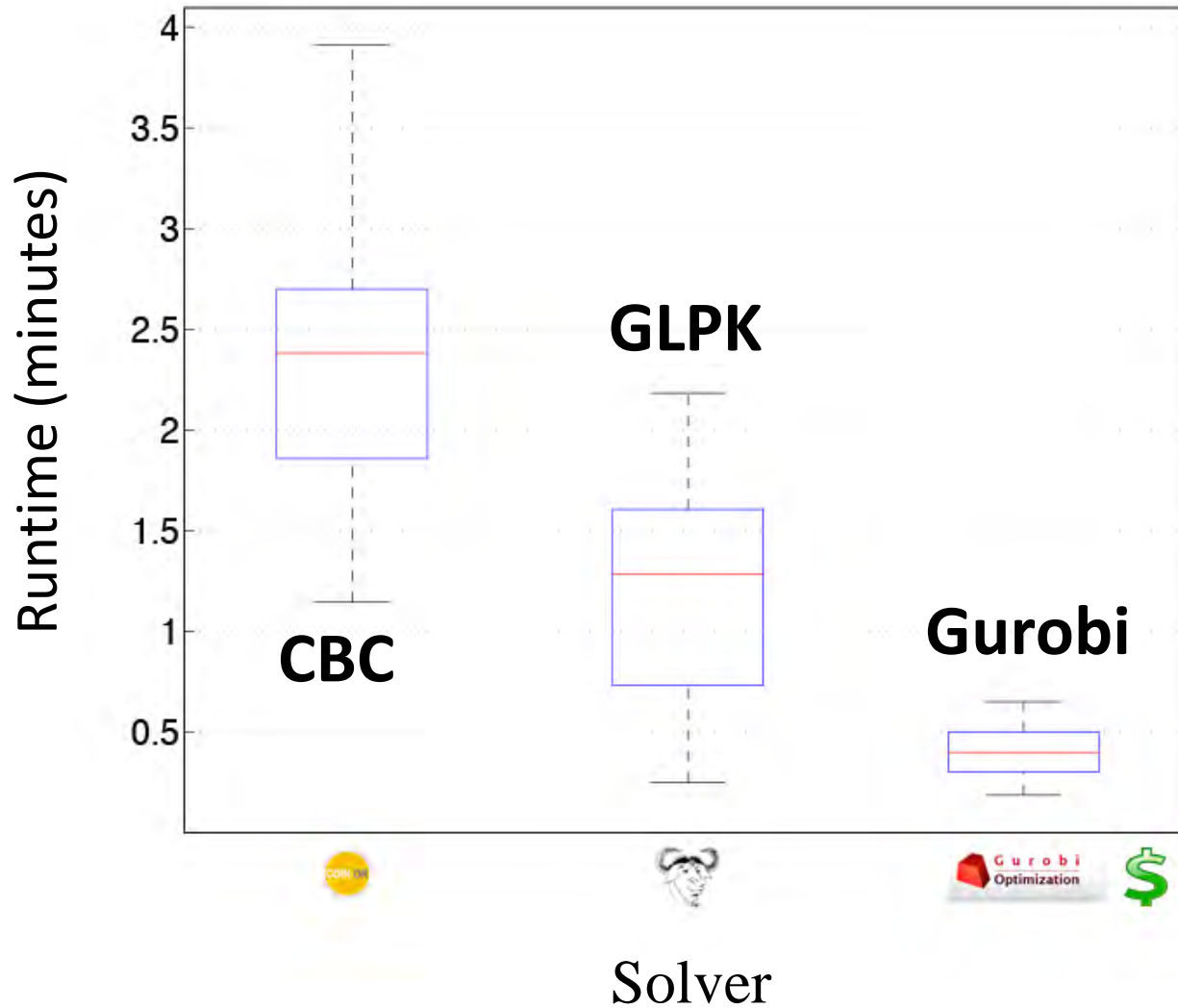
JuMP

Download Code from Github:

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

```
172 function one_lineup_type(skaters, goalies, lineups, num_overlap, num_skaters, num_goalies, centers, wingers, defenders, num_teams, skaters_teams, goalie_opponents, team_lines, num_lines, P1_Info)
173     # = Model{infeasible}()
174
175     # Lineups
176     @defVar(m, skaters_lineup[1:num_skaters], Bin)
177     @defVar(m, goalies_lineup[1:num_goalies], Bin)
178
179     # num goalies
180     @addConstraint(m, sum(goalies_lineup[1, 1:num_goalies]) == 1)
181
182     # num skaters
183     @addConstraint(m, sum(skaters_lineup[1, 1:num_skaters]) == 3)
184
185     # Teams 2 with 4 goalies
186     @addConstraint(m, sum(centers[1] * skaters_lineup[1, 1:num_skaters]) == 2)
187     @addConstraint(m, 2 == sum(centers[1] * skaters_lineup[1, 1:num_skaters]))
188
189     # Teams 2 with 2 wingers
190     @addConstraint(m, sum(wingers[1] * skaters_lineup[1, 1:num_skaters]) == 4)
191     @addConstraint(m, 3 == sum(wingers[1] * skaters_lineup[1, 1:num_skaters]))
192
193     # Teams 2 with 4 defenders
194     @addConstraint(m, 2 == sum(defenders[1] * skaters_lineup[1, 1:num_skaters]))
195     @addConstraint(m, sum(defenders[1] * skaters_lineup[1, 1:num_skaters]) == 7)
196
197     # Goalies 2 with 2 salaries
198     @addConstraint(m, sum(skaters[1, Salary] * skaters_lineup[1, 1:num_skaters]) + sum(goalies[1, Salary] * goalies_lineup[1, 1:num_goalies]) == 60000)
199
200
201     # Restrict # of teams per team for each skaters lineup
202     @defVar(m, num_on_team_pos[1:num_teams])
203     @defVar(m, num_on_team_neg[1:num_teams])
204
205     @addConstraint(m, xyconstr[1:num_teams], num_on_team_pos[1, t] - num_on_team_neg[1, t] == sum(skaters_teams[t, 1] * skaters_lineup[1, t] * num_skaters) * 0.5)
206
207     @defVar(m, used_team[1:num_teams], Bin)
208     @addConstraint(m, constr1[1:num_teams], num_on_team_pos[1, t] == 0 * used_team[1, t])
209     @addConstraint(m, constr2[1:num_teams], num_on_team_neg[1, t] == 1 * used_team[1, t])
210
211
212     # No goalie can play against skaters
213     @addConstraint(m, constr1[1:num_goalies], @goalies_lineup[1, k] == sum(goalie_opponents[k, 1] * skaters_lineup[1, k])
214
215
216     # Restrict # of lines per lineup (user to modify)
217     @defVar(m, pos_num_in_line[1:num_lines])
218     @defVar(m, neg_num_in_line[1:num_lines])
219     @addConstraint(m, constr1[1:num_lines], pos_num_in_line[1, k] == sum(team_lines[k, 1] * skaters_lineup[1, k]) * num_skaters * 0.5)
220
221     @defVar(m, line_stack[1:num_lines], Bin)
222     @addConstraint(m, constr1[1:num_lines], pos_num_in_line[1, k] == line_stack[1, k])
223     @addConstraint(m, constr2[1:num_lines], neg_num_in_line[1, k] == 1 * line_stack[1, k])
224
225     # Restrict # of lines per lineup (user to modify)
226     @defVar(m, pos_num_in_line2[1:num_lines])
227     @defVar(m, neg_num_in_line2[1:num_lines])
228     @addConstraint(m, constr1[1:num_lines], pos_num_in_line2[1, k] == sum(team_lines[k, 2] * skaters_lineup[1, k]) * num_skaters * 1.1)
229     @defVar(m, line_stack2[1:num_lines], Bin)
230     @addConstraint(m, constr1[1:num_lines], pos_num_in_line2[1, k] == 2 * line_stack2[1, k])
231     @addConstraint(m, constr2[1:num_lines], neg_num_in_line2[1, k] == 2 * line_stack2[1, k])
232
233     @addConstraint(m, sum(line_stack2[1, 1:num_lines]) == 2)
234
235
236
237
238     # The constraint must be on P1
239     @addConstraint(m, sum(sum(defenders[1] * P1_Info[1, j] * skaters_lineup[1, j]) * num_teams) == sum(defenders[1] * skaters_lineup[1, 1:num_skaters]))
240
241     # Overlap constraint
242     @addConstraint(m, constr1[1:size(lineups)[1]], sum(lineups[1, 1] * skaters_lineup[1, j]) * num_skaters > sum(lineups[num_skaters + 1, 1] * goalies_lineup[1, j]) * num_goalies == num_overlap)
243
244
245     # Objective
246     @setObjective(m, Max, sum(skaters[1, ROTO_Proj] * skaters_lineup[1, 1:num_skaters]) + sum(goalies[1, ROTO_Proj] * goalies_lineup[1, 1:num_goalies])
247     println("Solving Problem...")
248     @print("mip")
249     status = solve(m)
250
251
252     # status == Optimal
253     skaters_lineup_copy = Array{Int64, 0}
254     for i=1:num_skaters
255         @ getvalue(skaters_lineup[1, i]) == 0.0 @ getvalue(skaters_lineup[1, i]) == 1.0
256         skaters_lineup_copy = vcat(skaters_lineup_copy, fill(i, 1))
257     end
258     skaters_lineup_copy = vcat(skaters_lineup_copy, fill(0, 1))
259
260     end
261
262     # num goalies
263     @ getvalue(goalies_lineup[1, 1]) == 0.0 @ getvalue(goalies_lineup[1, 1]) == 1.0
264     skaters_lineup_copy = vcat(skaters_lineup_copy, fill(i, 1))
265     skaters_lineup_copy = vcat(skaters_lineup_copy, fill(0, 1))
266
267     end
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000
```

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>

