

# Mixed Integer Programming (MIP) for Daily Fantasy Sports, Statistics and Marketing

Juan Pablo Vielma

Massachusetts Institute of Technology

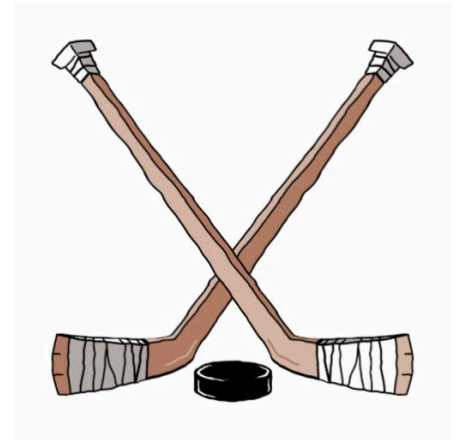
AM/ES 121, SEAS, Harvard.

Boston, MA, November, 2016.

# MIP & Daily Fantasy Sports



# Example Entry



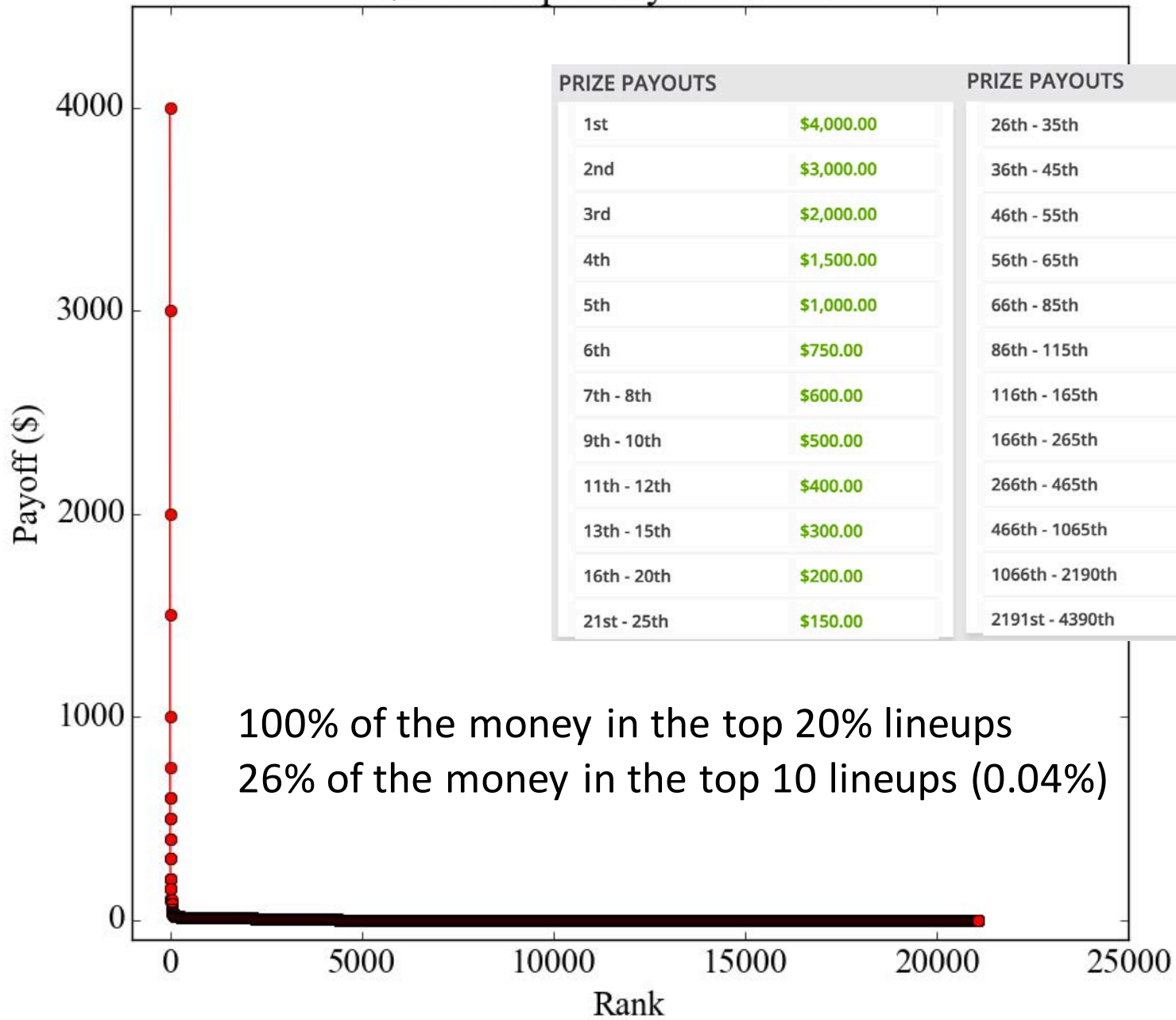
## LINEUP

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

# \$55K Sniper Payoff Structure



# Building a Lineup



# 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

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

# 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

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

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

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

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



# Team Feasibility

- At least 3 different NHL teams

## Team constraints

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

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

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

# Maximize Points

- Forecasted points for player  $p$ :  $f_p$



Score type	Points
Goal	3
Assist	2
Shot on Goal	0.5
Blocked Shot	0.5
Short Handed Point Bonus (Goal/Assist)	1
Shootout Goal	0.2
Hat Trick Bonus	1.5
Win (goalie only)	3
Save (goalie only)	0.2
Goal allowed (goalie only)	-1
Shutout Bonus (goalie only)	2

Table 1 Points system for NHL contests in DraftKings.

## Points Objective Function

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

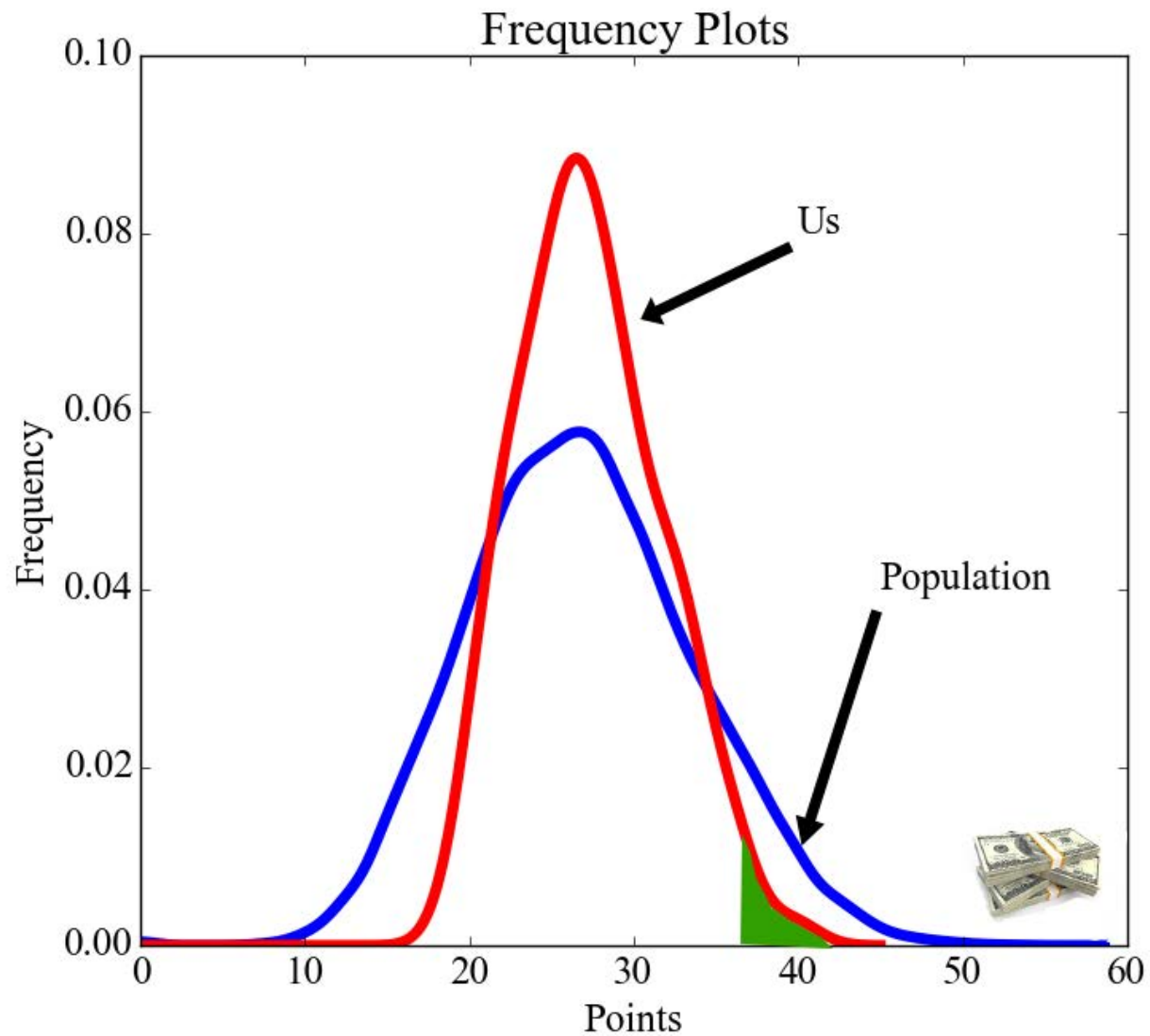
# Lineup

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

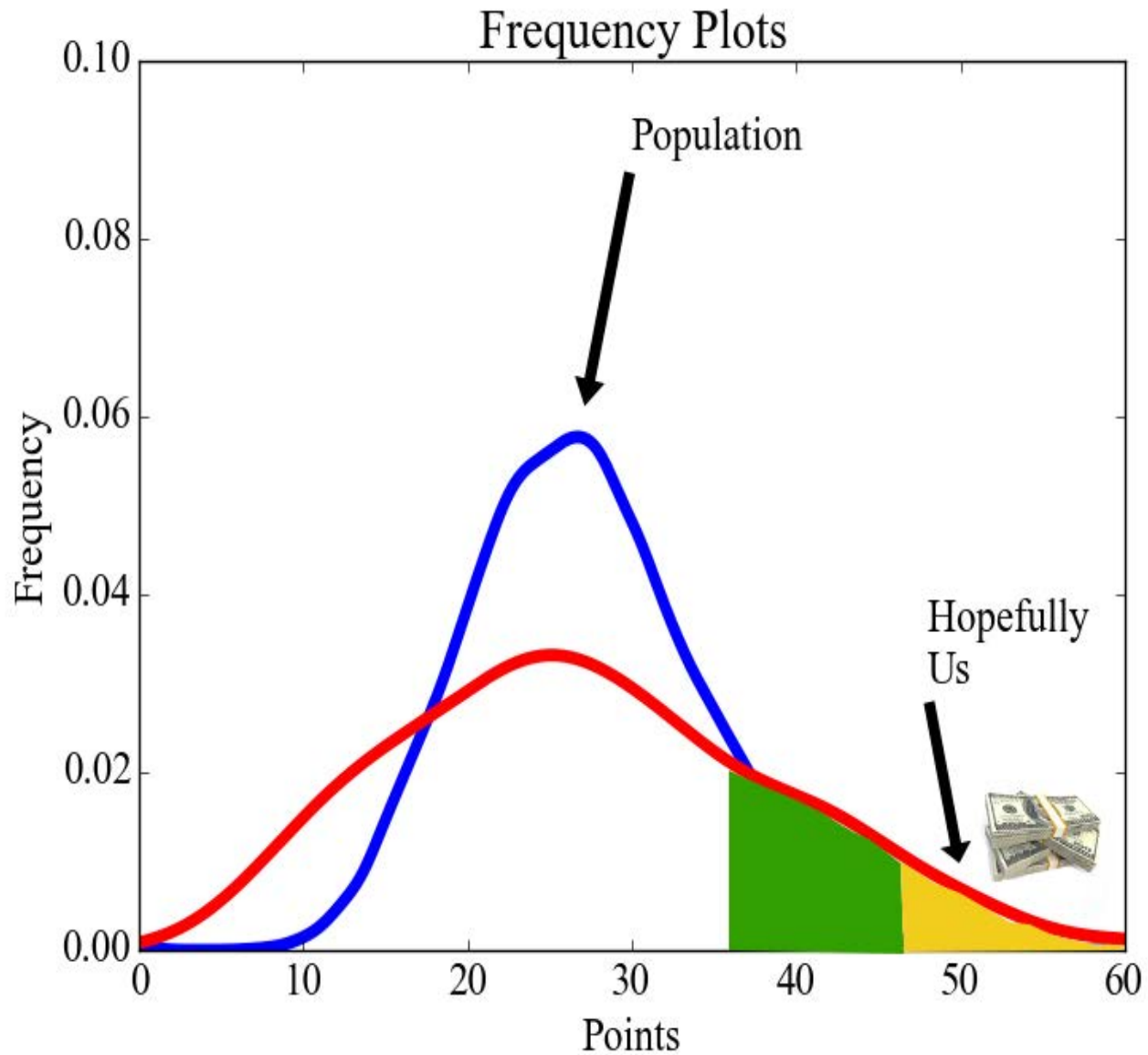


23 points on average

# Need $> 38$ points for a chance to win



# Increase variance to have a chance



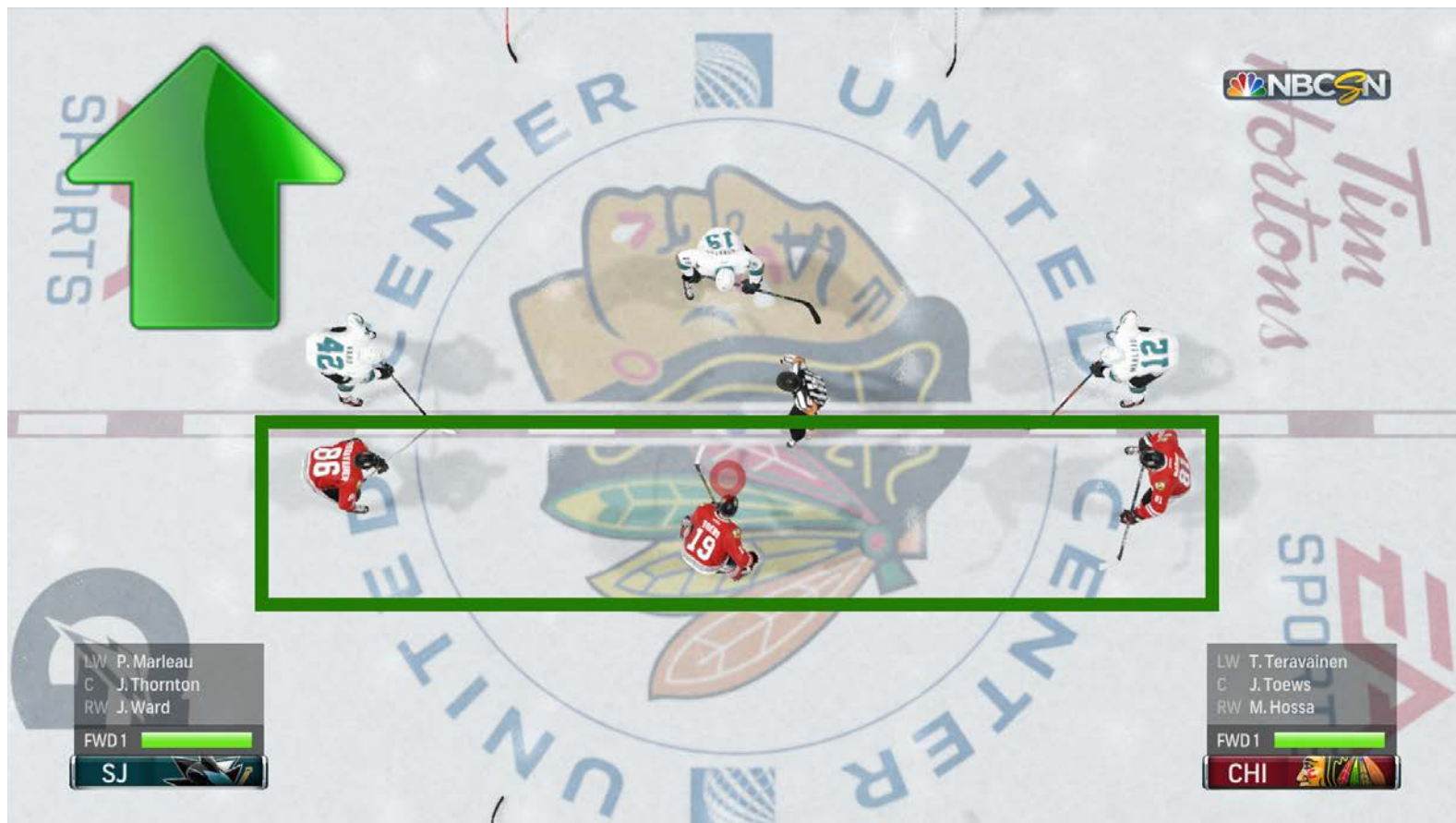


# Structural Correlations - Teams



# Structural Correlations - Lines

- Goal = 3 pt, assist = 2 pt



# Structural Correlations – Lines = Stacking

- 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 Skaters

- No skater against goalie

No skater against goalie constraint

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

# Good, but not great chance

Feasible

Team

Goalie

Line

Line

Not  
Against



# Play many diverse Lineups

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

Diversity constraint

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



# Were we able to do it?

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	\$18.75
28th	zlisto	45.50	\$15.00
28th	zlisto	45.50	

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	

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	

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	

November 15, 2015

November 16, 2015

November 17, 2015

November 23, 2015

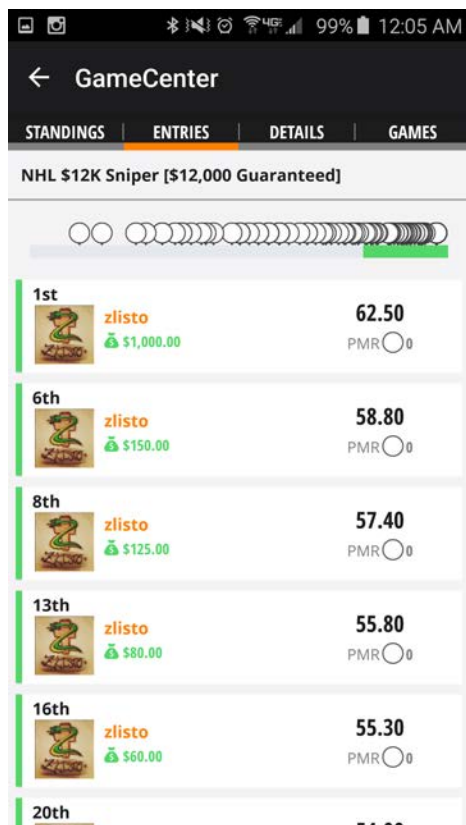
200 lineups

# 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 tabs for STANDINGS, ENTRIES, DETAILS, and GAMES. Below the tabs is a progress bar with 100 circles, the first few of which are filled green. The main content area lists the top performers:

Rank	Player	Score	Prize	PMR
1st	zlisto	62.50	\$1,000.00	0
6th	zlisto	58.80	\$150.00	0
8th	zlisto	57.40	\$125.00	0
13th	zlisto	55.80	\$80.00	0
16th	zlisto	55.30	\$60.00	0
20th		51.00		



> \$15K

December 12, 2015

100 lineups



# How can you do it?



## Download Code from Github:

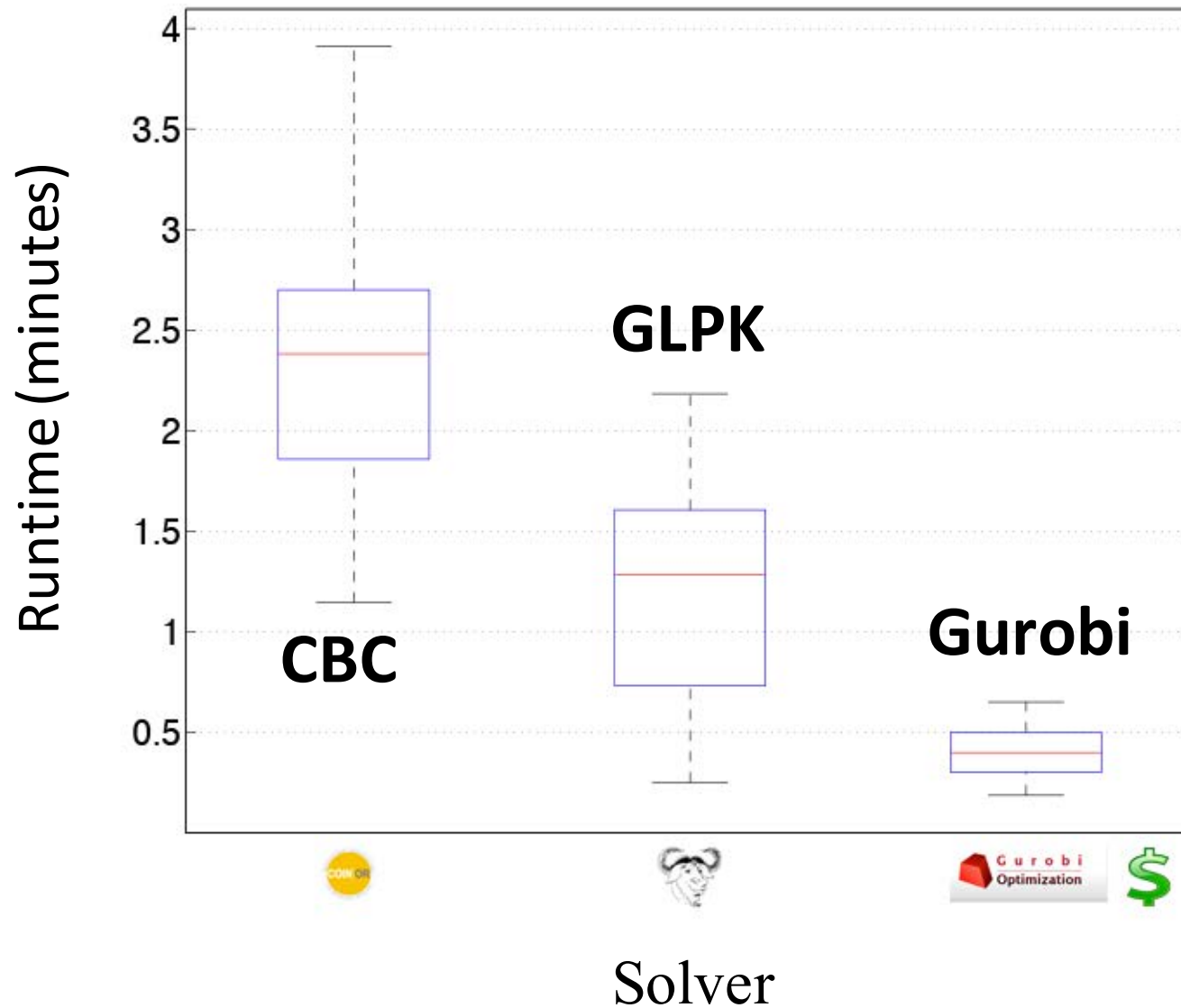
<https://github.com/dscotthunter/Fantasy-Hockey-IP-Code>

```
175 function solve_Lineup_Type_A(skaters, goalies, lineups, num_overlap, num_skaters, num_goalies, centers, wingers, defenders, num_teams, skaters_teams, goalie_opponents, team_lines, num_lines, P1_info)
176     m = Model{Solver}()
177
178     # Variables
179     @variable(m, skaters_lineup[1:num_skaters], Bin)
180     @variable(m, goalies_lineup[1:num_goalies], Bin)
181     @variable(m)
182
183     # Constraints
184     @MIOConstraint(m, sum(goalies_lineup[1], 1:num_skaters) == 1)
185     @MIOConstraint(m, sum(skaters_lineup[1], 1:num_skaters) == 1)
186
187     # Constraints for centers
188     @MIOConstraint(m, sum(centers[1:num_skaters]) - skaters_lineup[1], 1:num_skaters) == 3)
189     @MIOConstraint(m, 2 == sum(centers[1:num_skaters]) - skaters_lineup[1], 1:num_skaters)
190
191     # Constraints for wingers
192     @MIOConstraint(m, sum(wingers[1:num_skaters]) - skaters_lineup[1], 1:num_skaters) == 4)
193     @MIOConstraint(m, 3 == sum(wingers[1:num_skaters]) - skaters_lineup[1], 1:num_skaters)
194
195     # Constraints for D
196     @MIOConstraint(m, 2 == sum(defenders[1:num_skaters]) - skaters_lineup[1], 1:num_skaters)
197     @MIOConstraint(m, sum(defenders[1:num_skaters]) - skaters_lineup[1], 1:num_skaters) == 3)
198
199     # Financial constraint
200     @MIOConstraint(m, sum(skaters[1,Salary] - skaters_lineup[1], 1:num_skaters) + sum(goalies[1,Salary] - goalies_lineup[1], 1:num_goalies) == 50000)
201
202     # Constraint: 3 different teams for the 3 skaters constraint
203     @variable(m, num_on_team_pos[1:num_teams])
204     @variable(m, num_on_team_neg[1:num_teams])
205     @MIOConstraint(m, sum(num_on_team_pos[1:num_teams]) - num_on_team_neg[1:num_teams] == sum(skaters_teams[i], 1:num_skaters) - 3)
206     @variable(m, used_team[1:num_teams], Bin)
207     @MIOConstraint(m, constr[i:num_teams], num_on_team_pos[i] == num_teams[i])
208     @MIOConstraint(m, constr[i:num_teams], num_on_team_neg[i] == 1 - used_team[i])
209     @MIOConstraint(m, sum(used_team[1:num_teams]) == 3)
210
211     # No goalie gets goalie opponent
212     @MIOConstraint(m, constr[1:num_goalies], 6 - goalies_lineup[i] - skaters_lineup[h], 1:num_skaters) == 0)
213
214     # Maximize # of lines and players line in each lineup
215     @variable(m, pos_num_in_line[1:num_lines])
216     @variable(m, neg_num_in_line[1:num_lines])
217     @MIOConstraint(m, constr[i:num_lines], pos_num_in_line[i] - neg_num_in_line[i] == sum(team_lines[h], 1:num_skaters) - 2)
218     @variable(m, line_stack[1:num_lines], Bin)
219     @MIOConstraint(m, constr[i:num_lines], pos_num_in_line[i] == sum(line_stack[i])
220     @MIOConstraint(m, constr[i:num_lines], neg_num_in_line[i] == 3 - sum(line_stack[i])
221     @MIOConstraint(m, sum(line_stack[1], 1:num_lines) == 3)
222
223     # Maximize # of lines in each lineup
224     @variable(m, pos_num_in_line2[1:num_lines])
225     @variable(m, neg_num_in_line2[1:num_lines])
226     @MIOConstraint(m, constr[i:num_lines], pos_num_in_line2[i] - neg_num_in_line2[i] == sum(team_lines[k], 1:num_skaters) - 1)
227     @variable(m, line_stack2[1:num_lines], Bin)
228     @MIOConstraint(m, constr[i:num_lines], pos_num_in_line2[i] == 2 - sum(line_stack2[i])
229     @MIOConstraint(m, constr[i:num_lines], neg_num_in_line2[i] == 2 - sum(line_stack2[i])
230     @MIOConstraint(m, sum(line_stack2[1], 1:num_lines) == 2)
231
232
233     # The defenders must be on P1
234     @MIOConstraint(m, sum(defenders[1:P1_info[1]] - skaters_lineup[1], 1:num_skaters) - sum(defenders[1:num_skaters]) == sum(defenders[1:num_skaters])
235
236     # Number of lines
237     @MIOConstraint(m, constr[1:size(lineups[1])], sum(lineups[j], 1:num_skaters) - sum(lineups[num_skaters+1:j], 1:num_skaters) == num_overlap)
238
239     # Objective
240     @objective(m, Max, sum(skaters[1,NOTO_Proj] - skaters_lineup[1], 1:num_skaters) + sum(goalies[1,NOTO_Proj] - goalies_lineup[1], 1:num_goalies))
241     print("Solving Problem...")
242     @start(m)
243     status = solve(m)
244
245     if status == :Optimal
246         skaters_lineup_copy = Array{Int64, 1}
247         for i in 1:num_skaters
248             if getValue(skaters_lineup[i]) == 0.0 @ getValue(skaters_lineup[i]) == 1.0
249                 skaters_lineup_copy = vcat(skaters_lineup_copy, fill(1, 1))
250             else
251                 skaters_lineup_copy = vcat(skaters_lineup_copy, fill(0, 1))
252             end
253         end
254         for i in 1:num_goalies
255             if getValue(goalies_lineup[i]) == 0.0 @ getValue(goalies_lineup[i]) == 1.0
256                 skaters_lineup_copy = vcat(skaters_lineup_copy, fill(1, 1))
257             else
258                 skaters_lineup_copy = vcat(skaters_lineup_copy, fill(0, 1))
259             end
260         end
261     end
262 end
```

<http://arxiv.org/pdf/1604.01455v1.pdf>



# Performance Time < 30 Minutes



# MIP and Statistics: Inference for the Chilean Earthquake

# The 2010 Chilean Earthquake

---





# 6<sup>th</sup> Strongest in Recorded History (8.8)

---



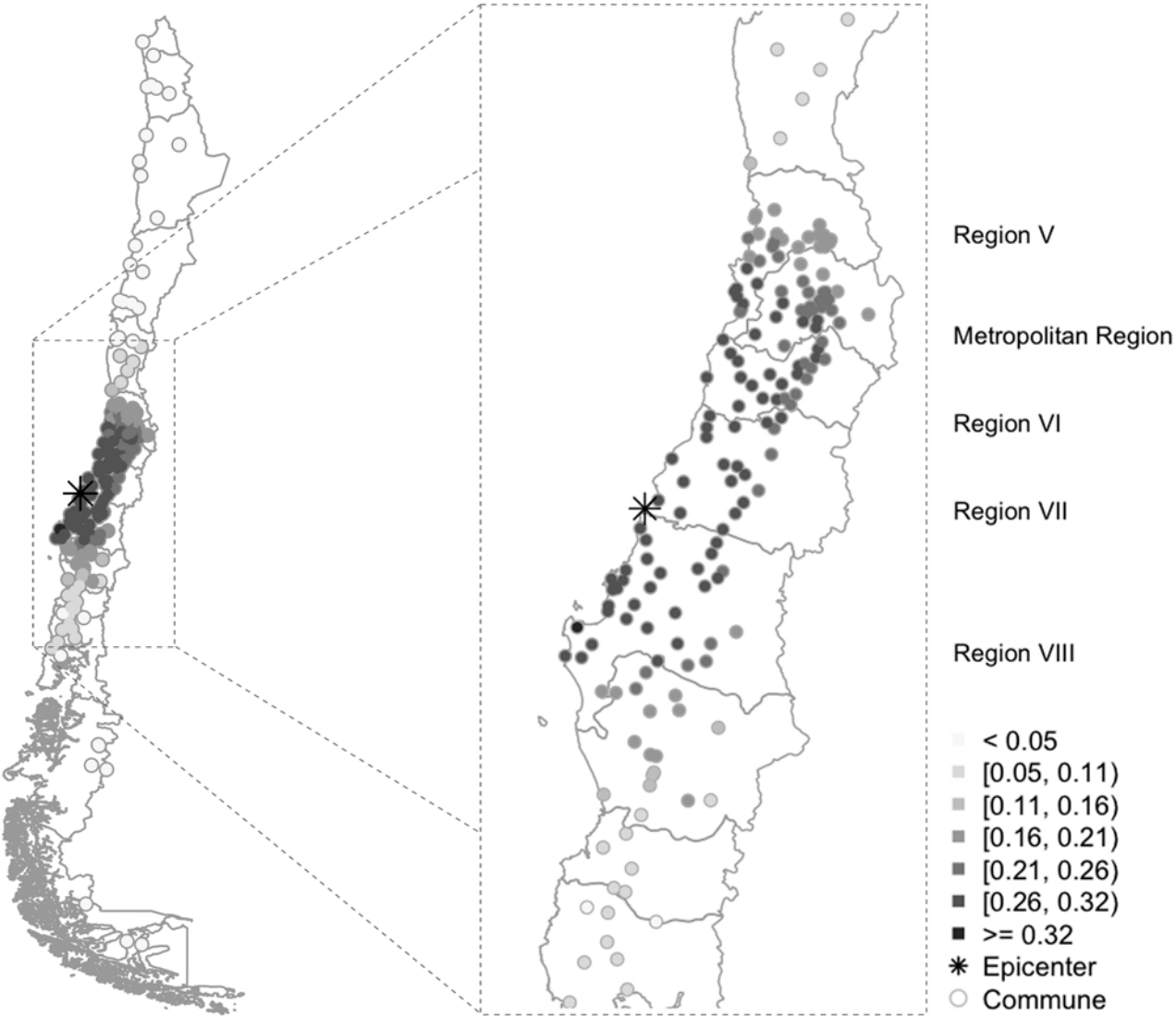


# Impact on Educational Achievement? PSU = SAT

---



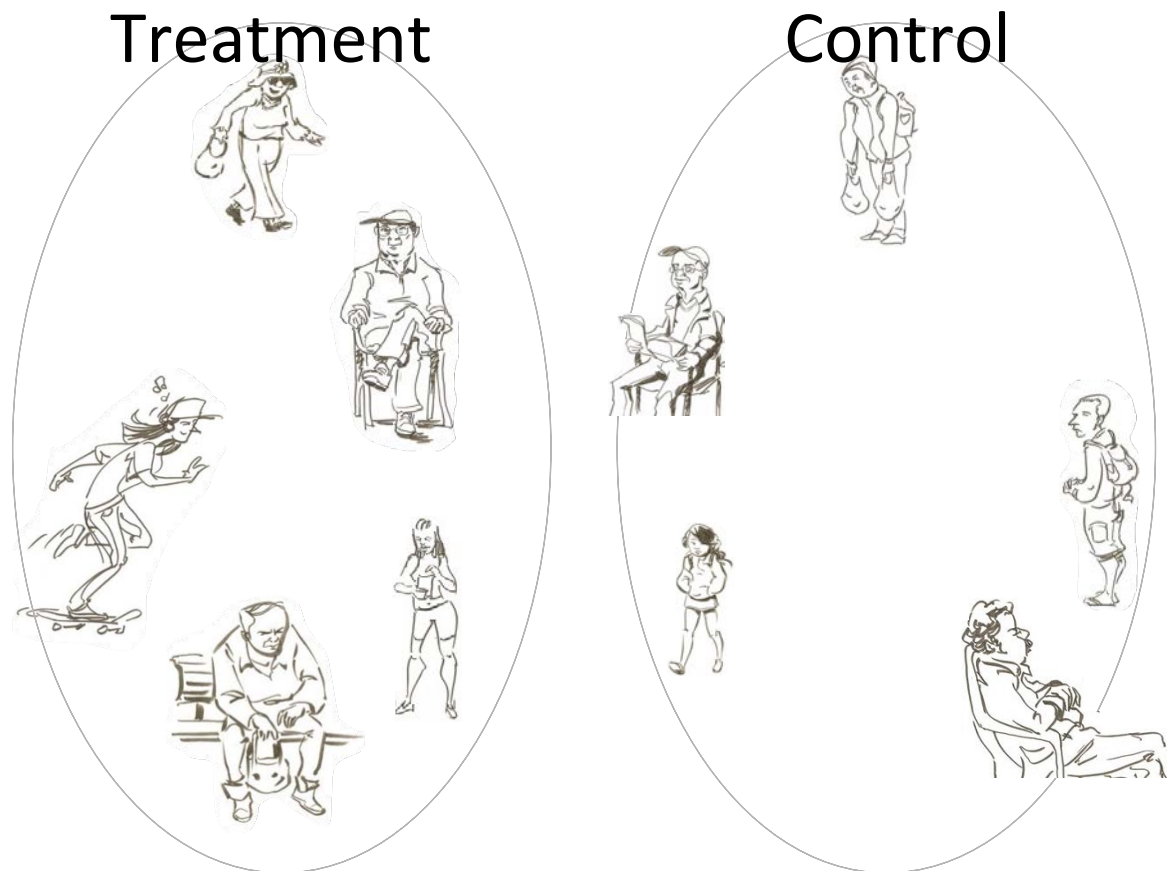
# Earthquake Intensity + Great Demographic Info



# Randomized experiment

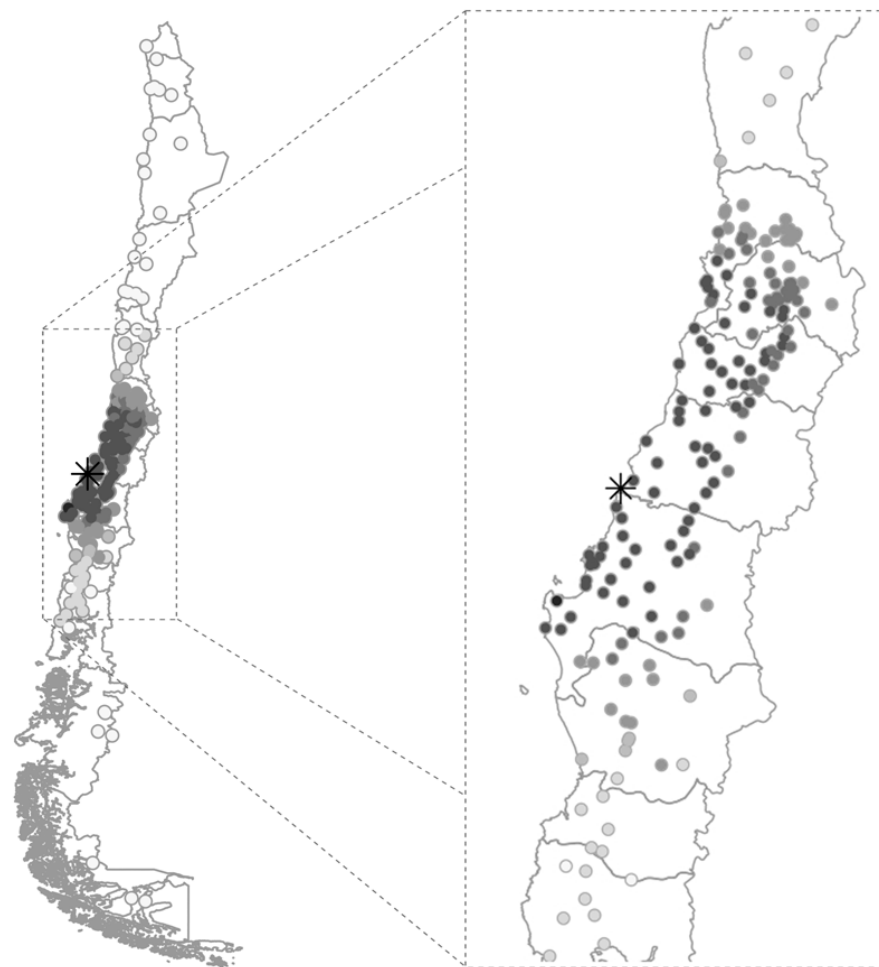
---

- Treatment / control have similar characteristics (covariates).



# Covariate Balance Important for Inference

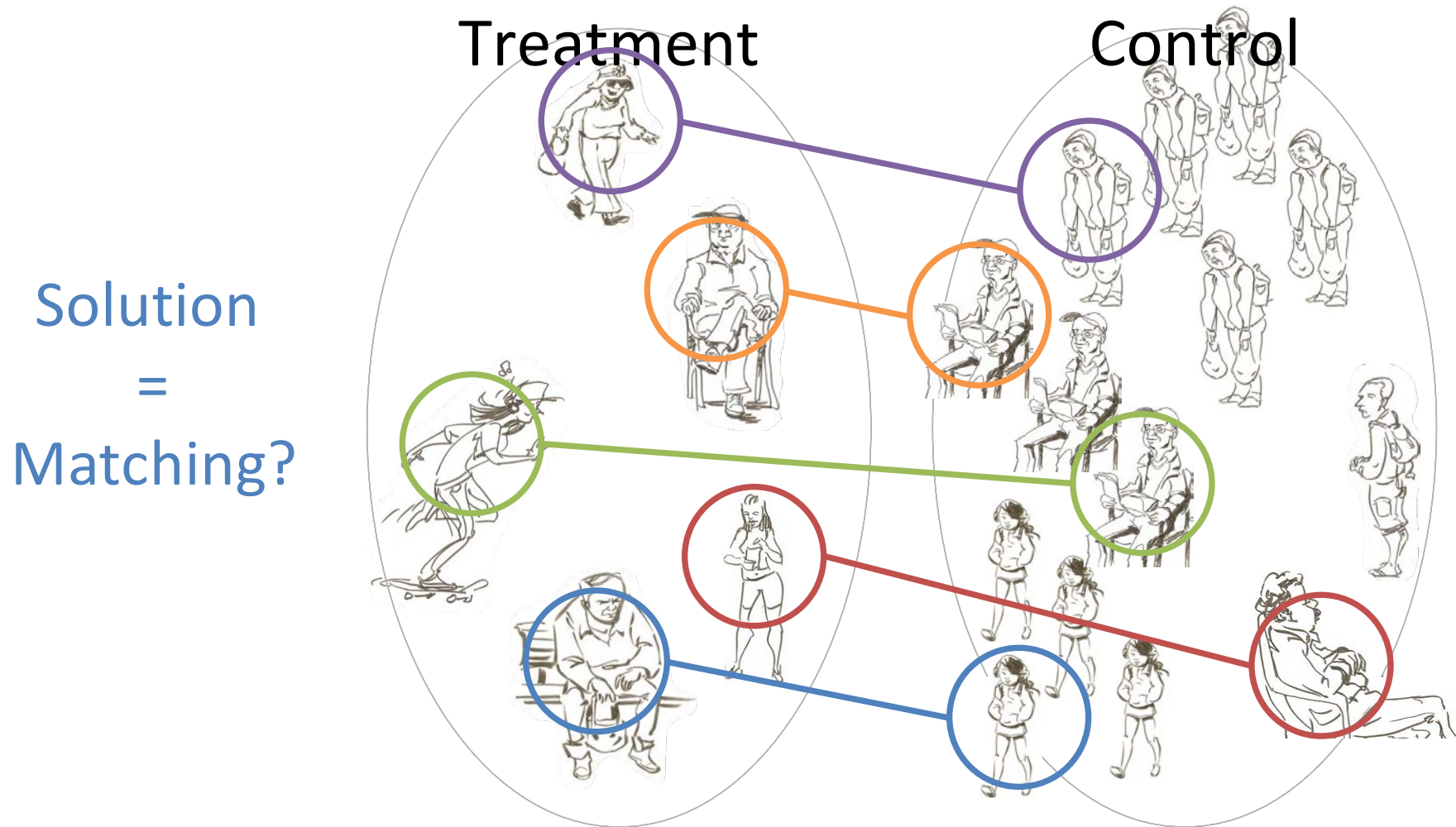
Covariate	Dose	
	1	2
Gender		
Male	462	462
Female	538	538
School SES		
Low	75	75
Mid-low	327	327
Medium	294	294
Mid-high	189	189
High	115	115
Mother's education		
Primary	335	335
Secondary	426	426
Technical	114	114
College	114	114
Missing	11	11
⋮		





# Observational Study: e.g. After Earthquake

- Treatment / control can have different characteristics.



## Matching

---

Treated Units:  $\mathcal{T} = \{t_1, \dots, t_T\}$

Control Units:  $\mathcal{C} = \{c_1, \dots, c_C\}$

Observed Covariates:  $\mathcal{P} = \{p_1, \dots, p_P\}$

Covariate Values:  $\mathbf{x}^t = (x_p^t)_{p \in \mathcal{P}}, \quad t \in \mathcal{T}$

$\mathbf{x}^c = (x_p^c)_{p \in \mathcal{P}}, \quad c \in \mathcal{C}$

# Nearest Neighbor Matching

---

$$\text{minimize}_{\mathbf{m}} \quad \sum_{t \in \mathcal{T}} \sum_{c \in \mathcal{C}} \delta_{t,c} m_{t,c}$$

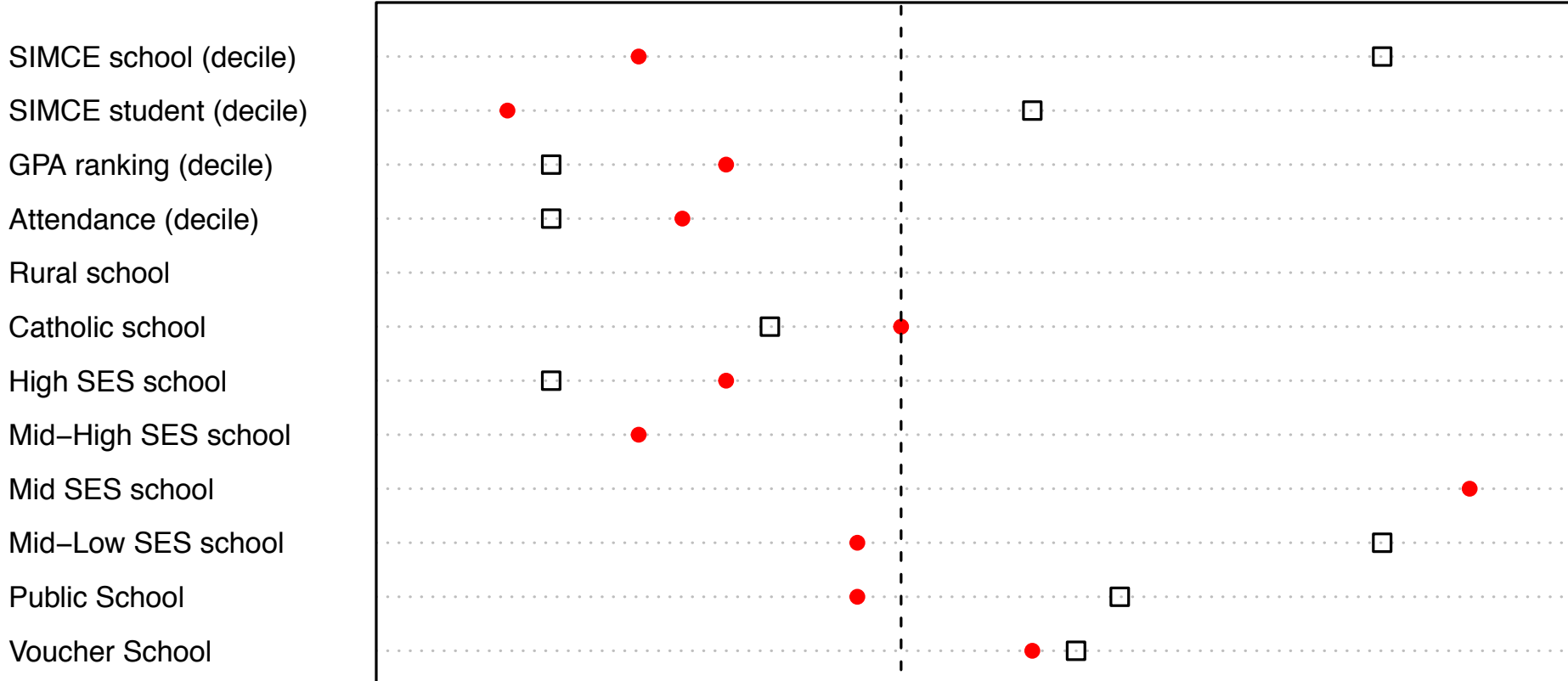
$$\text{subject to} \quad \sum_{c \in \mathcal{C}} m_{t,c} = 1, \quad t \in \mathcal{T}$$

$$\sum_{t \in \mathcal{T}} m_{t,c} \leq 1, \quad c \in \mathcal{C}$$

$$0 \leq m_{t,c} \leq 1 \quad \text{---} m_{t,c} \in \{0, 1\}, \quad t \in \mathcal{T}, c \in \mathcal{C}$$

- e.g.  $\delta_{t,c} = \|\mathbf{x}^t - \mathbf{x}^c\|_2$
- Easy to solve

# Balance Before After Matching



# Maximum Cardinality Matching

---

$$\max \sum_{t \in \mathcal{T}} \sum_{c \in \mathcal{C}} m_{t,c}$$

s.t.

$$\sum_{t \in \mathcal{T}} m_{t,c} \leq 1,$$

$$\sum_{c \in \mathcal{C}} m_{t,c} \leq 1,$$

$$\sum_{t \in \mathcal{T}_{p,k}} \sum_{c \notin \mathcal{C}_{p,k}} m_{t,c} = \sum_{t \notin \mathcal{T}_{p,k}} \sum_{c \in \mathcal{C}_{p,k}} m_{t,c} \quad \forall p \in \mathcal{P}, k \in \mathcal{K}(p)$$

$$m_{t,c} \in \{0, 1\}$$

$$\mathcal{K}(p) = \{\mathbf{x}_p^c\}_{c \in \mathcal{P}} \cup \{\mathbf{x}_p^t\}_{t \in \mathcal{T}}$$

$$\mathcal{C}_{p,k} = \{c \in \mathcal{C} : \mathbf{x}_p^c = k\}$$

$$\mathcal{T}_{p,k} = \{t \in \mathcal{T} : \mathbf{x}_p^t = k\}$$

$$\forall c \in \mathcal{C}$$

$$\forall t \in \mathcal{T}$$

$$\forall p \in \mathcal{P}, k \in \mathcal{K}(p)$$

$$\forall t \in \mathcal{T}, c \in \mathcal{C}.$$

- Very hard to solve ( and very hard to understand! )

# Advanced Maximum Cardinality Matching

---

$$\max \sum_{t \in \mathcal{T}} x_t$$

*s.t.*

$$\sum_{t \in \mathcal{T}} x_t = \sum_{c \in \mathcal{C}} y_c,$$

$$\sum_{t \in \mathcal{T}_{p,k}} x_t = \sum_{c \in \mathcal{C}_{p,k}} y_c, \quad \forall p \in \mathcal{P}, \quad k \in \mathcal{K}(p)$$

$$x_t \in \{0, 1\} \quad \forall t \in \mathcal{T}$$

$$y_c \in \{0, 1\} \quad \forall c \in \mathcal{C}.$$

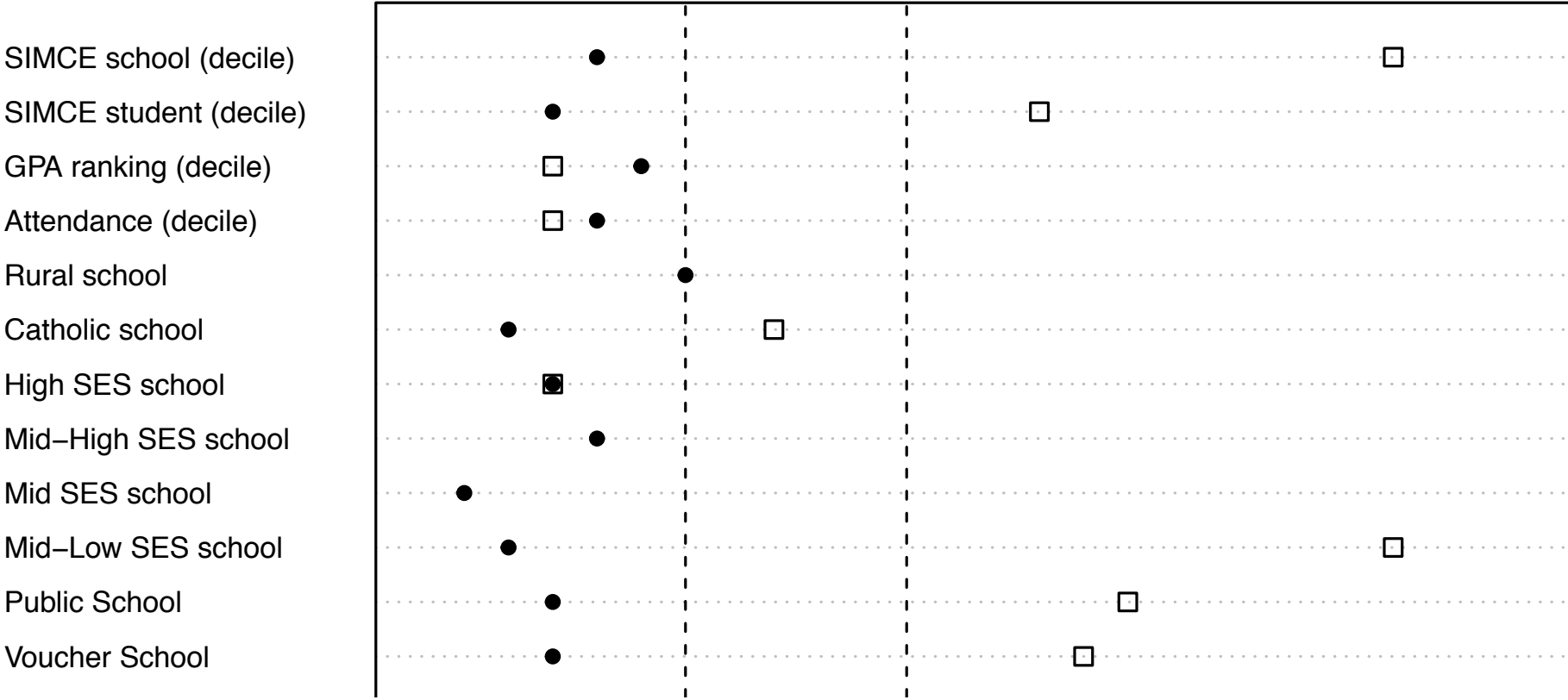
$$\mathcal{K}(p) = \{\mathbf{x}_p^c\}_{c \in \mathcal{P}} \cup \{\mathbf{x}_p^t\}_{t \in \mathcal{T}}$$

$$\mathcal{C}_{p,k} = \{c \in \mathcal{C} : \mathbf{x}_p^c = k\}$$

$$\mathcal{T}_{p,k} = \{t \in \mathcal{T} : \mathbf{x}_p^t = k\}$$

- Matching without matching variables
- Easy to solve: Small, but inherits matching properties

# Balance Before After Cardinality Matching



# Can Also do Multiple Doses

---

- Dose
  1. No quake
  2. Medium quake
  3. Bad quake

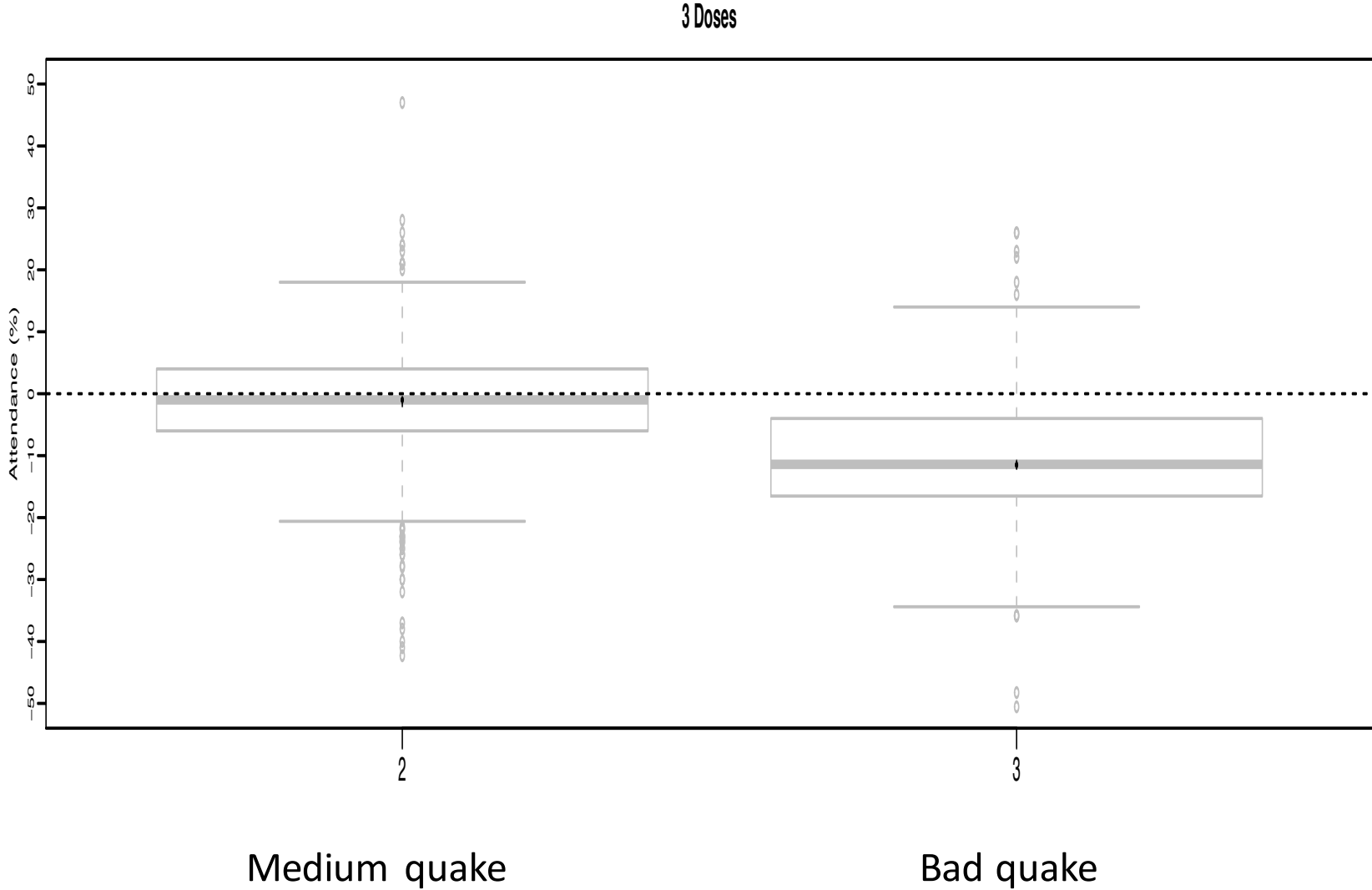
---

Covariate	Dose		
	1	2	3
Gender			
Male	462	462	462
Female	538	538	538
School SES			
Low	75	75	75
Mid-low	327	327	327
Medium	294	294	294
Mid-high	189	189	189
High	115	115	115
Mother's education			
Primary	335	335	335
Secondary	426	426	426
Technical	114	114	114
College	114	114	114
Missing	11	11	11
⋮			

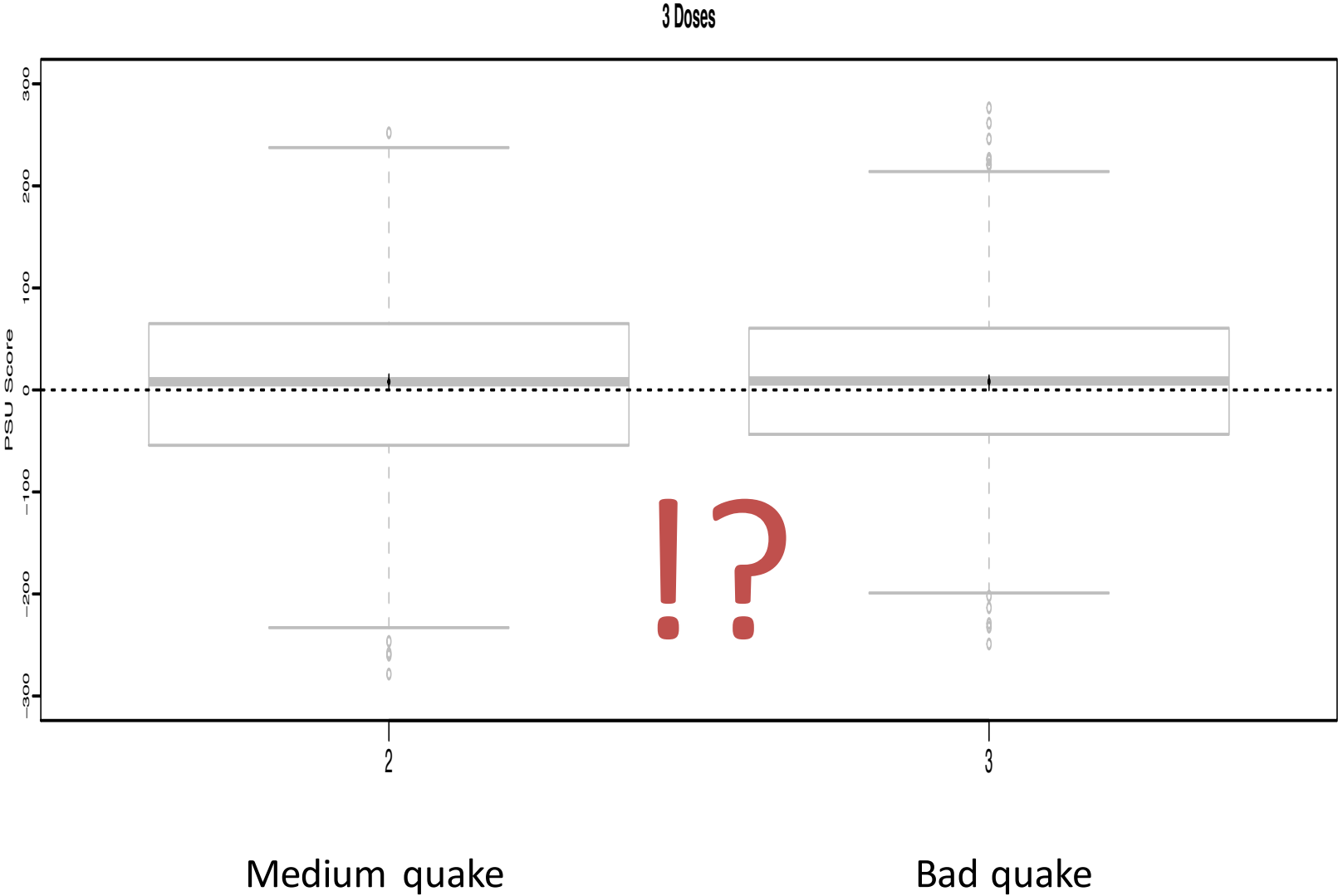
---



# Relative (To no Quake) Attendance Impact



# Relative (To no Quake) PSU Score Impact



MIP and Marketing:  
Chewbacca or BB-8?

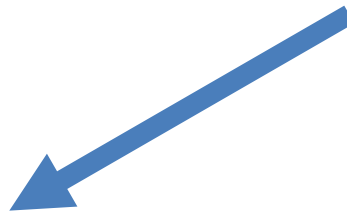
# Adaptive Preference Questionnaires



Feature	SX530	RX100
Zoom	50x	3.6x
Prize	\$249.99	\$399.99
Weight	15.68 ounces	7.5 ounces
Prefer	<input checked="" type="checkbox"/>	<input type="checkbox"/>



Feature	TG-4	G9
Waterproof	Yes	No
Prize	\$249.99	\$399.99
Weight	7.36 lb	7.5 lb
Prefer	<input type="checkbox"/>	<input checked="" type="checkbox"/>



Feature	TG-4	Galaxy 2
Waterproof	Yes	No
Prize	\$249.99	\$399.99
Viewfinder	Electronic	Optical
Prefer	<input checked="" type="checkbox"/>	<input type="checkbox"/>



We recommend:



# Choice-based Conjoint Analysis (CBCA)

---



Feature	Chewbacca	BB-8
Wookiee	Yes	No
Droid	No	Yes
Blaster	Yes	No
I would buy toy	<input checked="" type="checkbox"/>	<input type="checkbox"/>

$$\begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} = x^2$$

Product Profile       $x^1$        $x^2$



# Preference Model and Geometric Interpretation

- Utilities for 2 products, d features, logit model

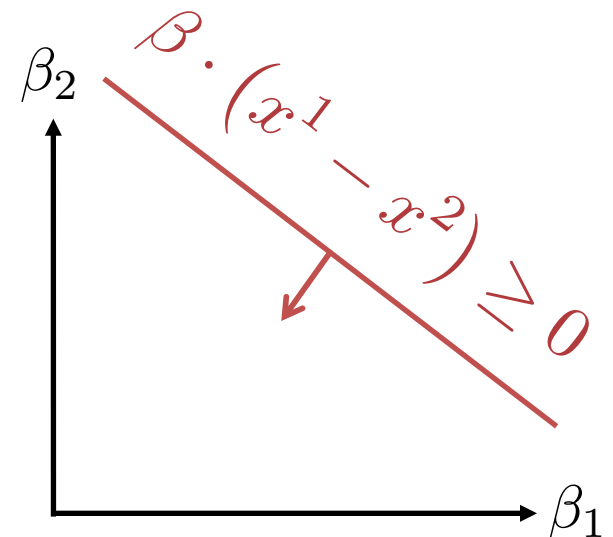
$$U_1 = \beta \cdot x^1 + \epsilon_1 = \sum_{i=1}^d \beta_i x_i^1 + \epsilon_1$$

$$U_2 = \beta \cdot x^2 + \epsilon_2 = \sum_{i=1}^d \beta_i x_i^2 + \epsilon_2$$

part-worths  $\nearrow$   
product profile  $\nearrow$  noise (gumbel)  $\uparrow$

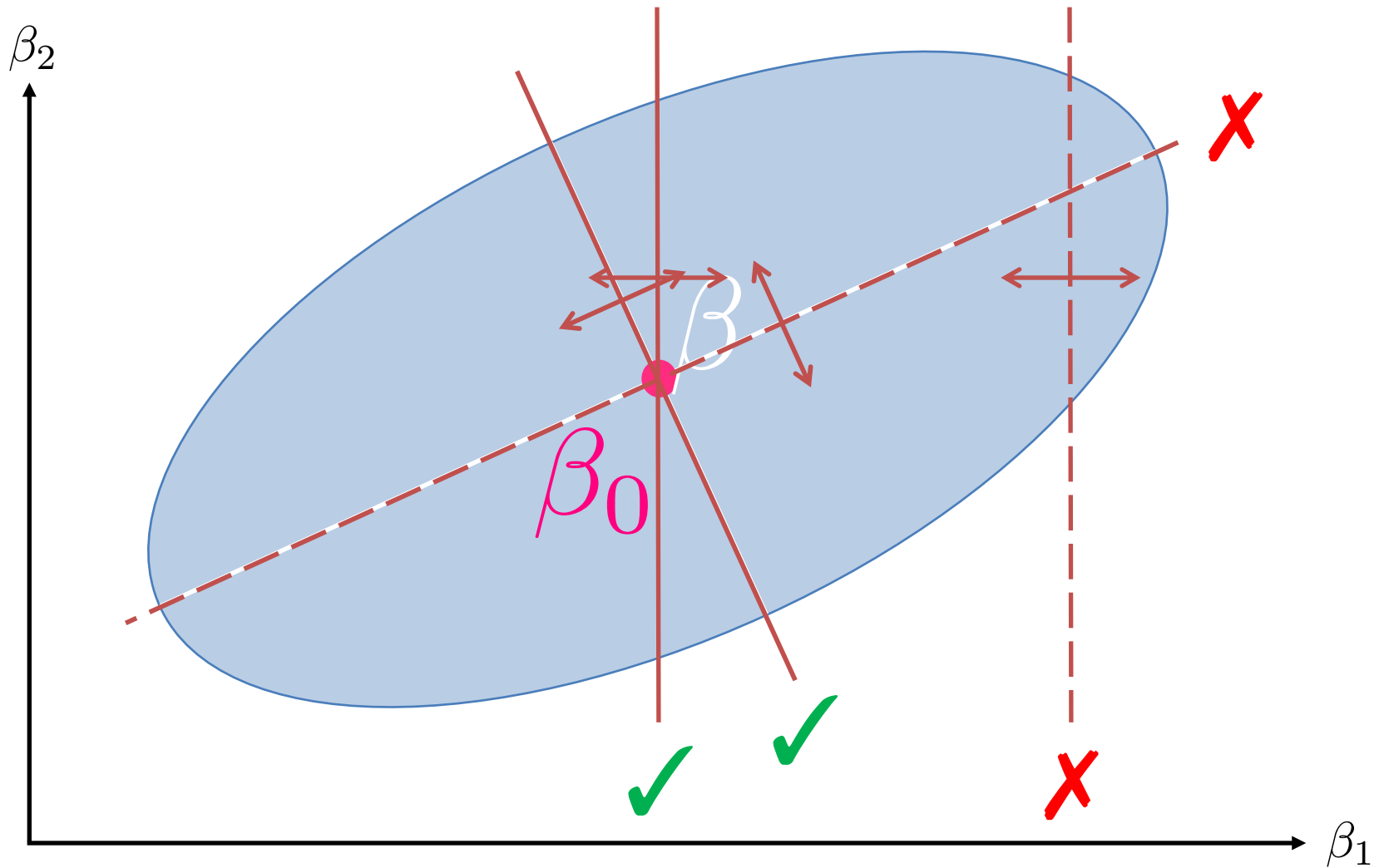
- Utility maximizing customer
  - Geometric interpretation of preference for product 1 **without error**

$$x^1 \succeq x^2 \Leftrightarrow U_1 \geq U_2$$



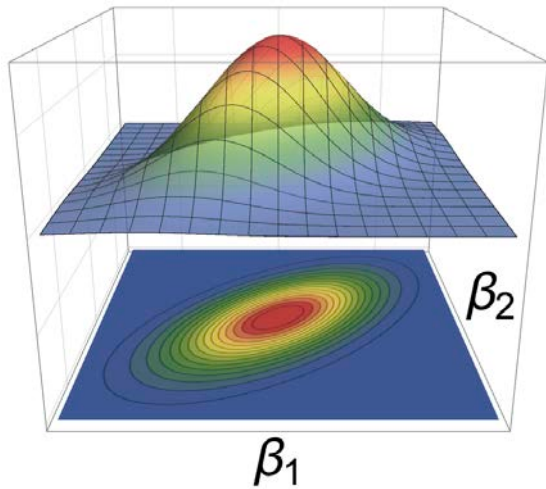
# Next Question = Minimize (Expected) Volume

Good Estimator? for  $\beta$ ? ~~Central tendency~~  $\beta_0$

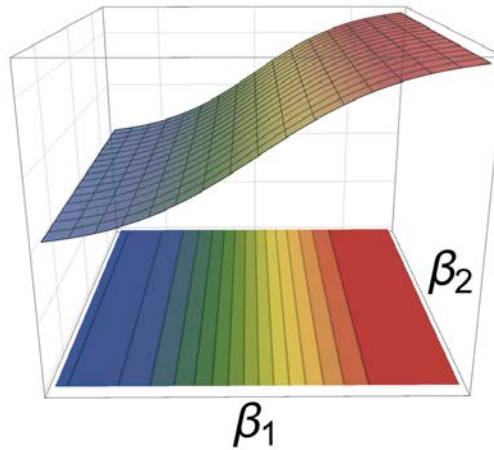


# With Error = Volume of Ellipsoid $f(x^1, x^2)$

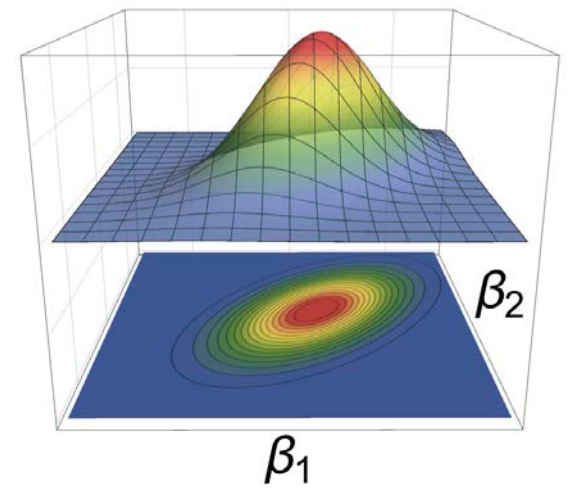
Prior distribution



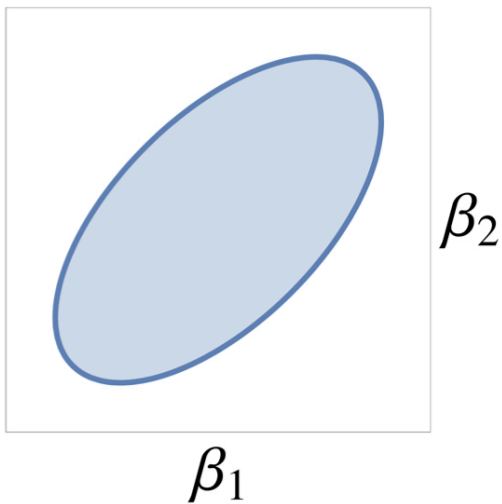
Answer likelihood



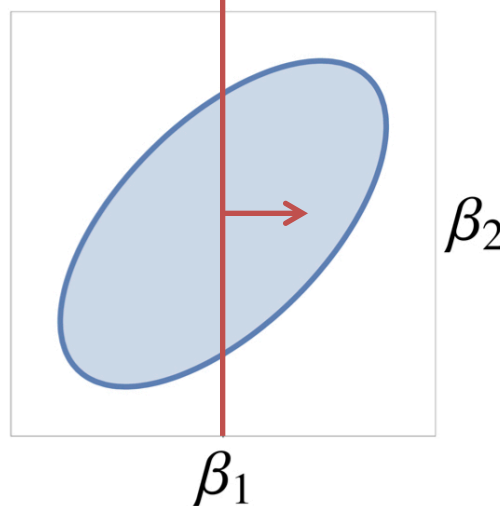
Posterior distribution



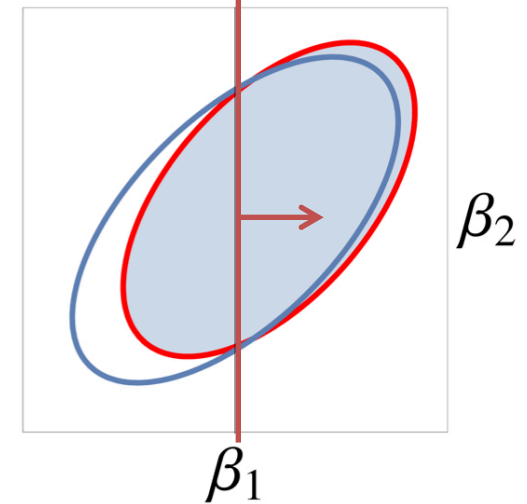
Prior ellipsoid



Question/Answer



Posterior ellipsoid



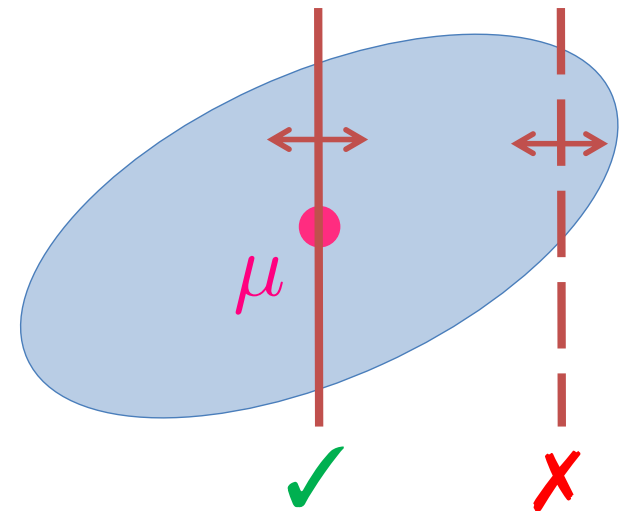
# Rules of Thumb Still Good For Ellipsoid Volume

---

$$(\beta - \mu)' \cdot \Sigma^{-1} \cdot (\beta - \mu) \leq r$$

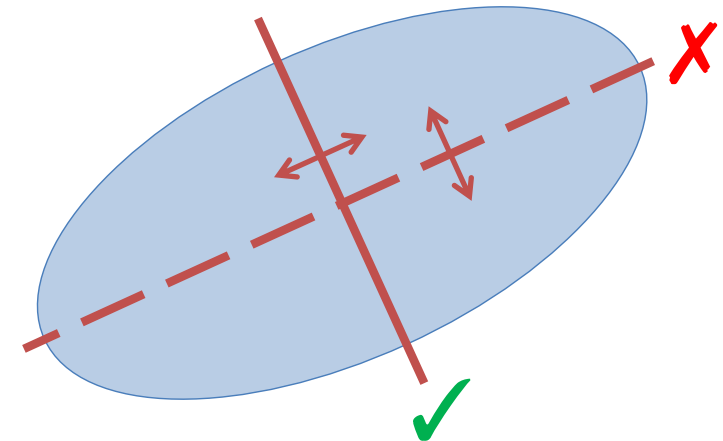
- Choice balance:
  - Minimize **distance** to center

$$\mu \cdot (x^1 - x^2)$$



- Postchoice symmetry:
  - Maximize **variance** of question

$$(x^1 - x^2)' \cdot \Sigma \cdot (x^1 - x^2)$$

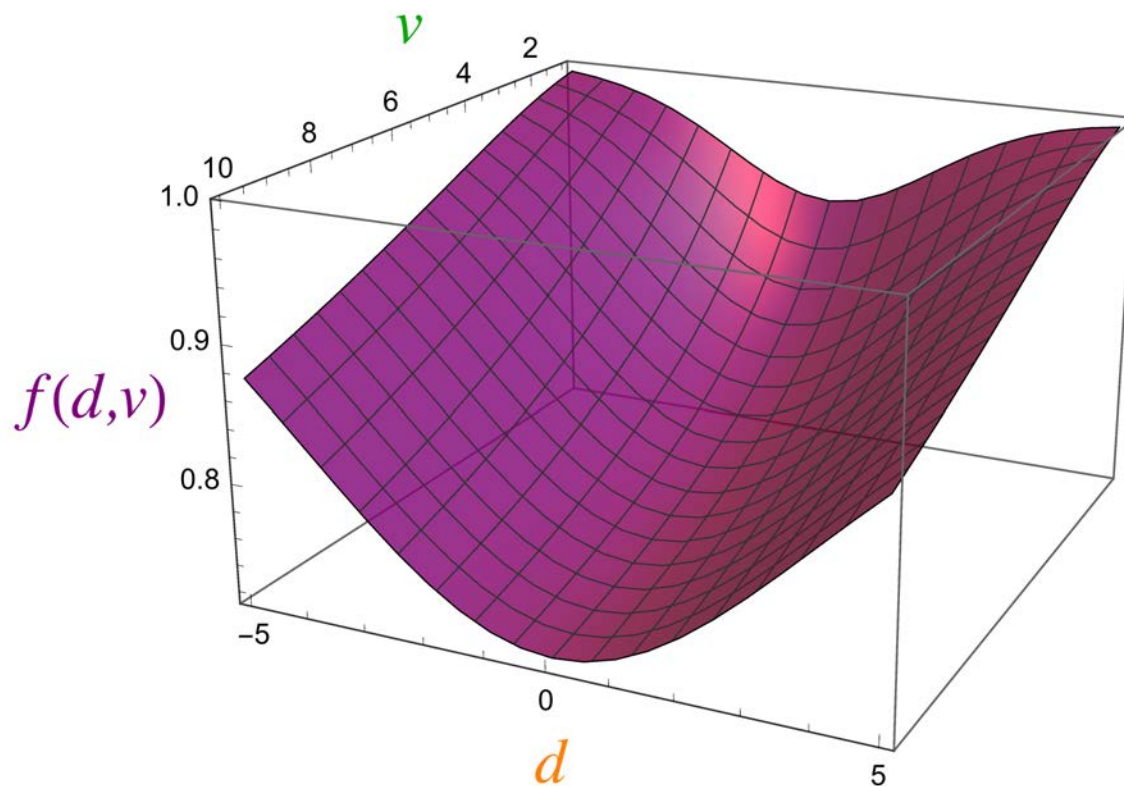


# “Simple” Formula for Expected Volume

- Expected Volume = Non-convex function  $f(d, v)$  of

distance:  $d := \mu \cdot (x^1 - x^2)$

variance:  $v := (x^1 - x^2)' \cdot \Sigma \cdot (x^1 - x^2)$



Can evaluate  $f(d, v)$   
with 1-dim integral 😊



# Optimization Model

---

min

$$f(d, v)$$

~~X~~

s.t.

$$\mu \cdot (x^1 - x^2) = d \quad \checkmark$$

$$(x^1 - x^2)' \cdot \Sigma \cdot (x^1 - x^2) = v \quad \checkmark \times$$

$$A^1 x^1 + A^2 x^2 \leq b \quad \checkmark$$

Formulation trick:

linearize  $x_i^k \cdot x_j^l$

$$x^1 \neq x^2 \quad \checkmark \times$$

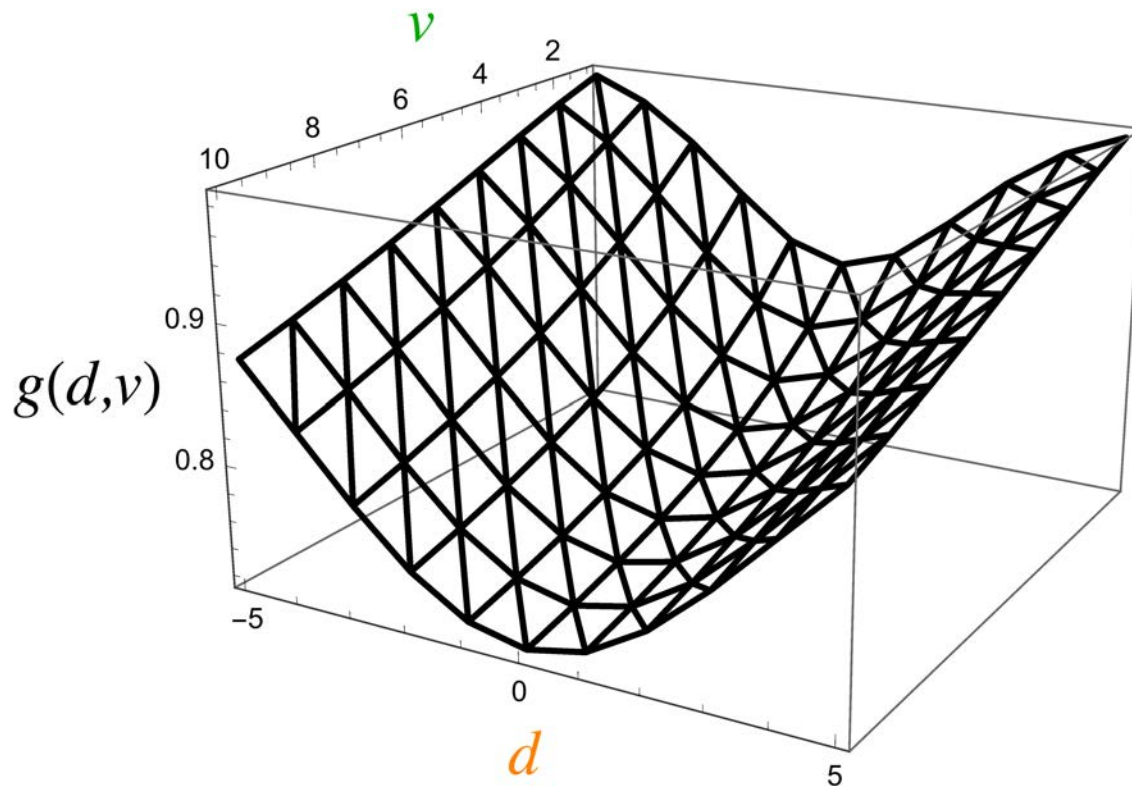
$$x^1, x^2 \in \{0, 1\}^n$$

# Technique 2: Piecewise Linear Functions

- D-efficiency = Non-convex function  $f(d, v)$  of

distance:  $d := \mu \cdot (x^1 - x^2)$

variance:  $v := (x^1 - x^2)' \cdot \Sigma \cdot (x^1 - x^2)$



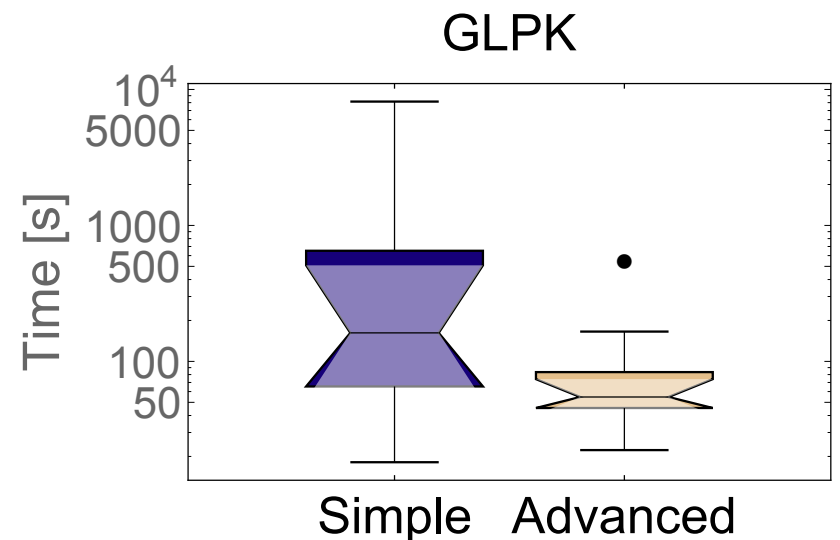
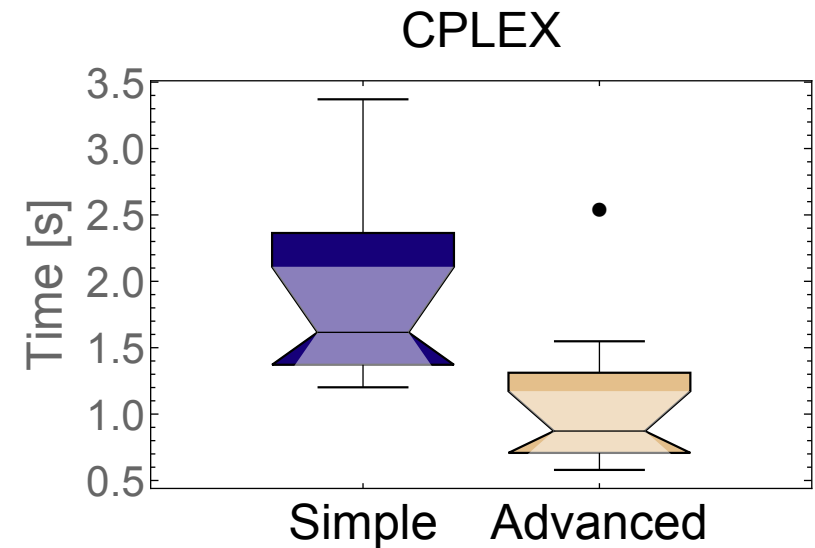
Can evaluate  $f(d, v)$   
with 1-dim integral 😊

Piecewise Linear  
Interpolation

MIP formulation

# Computational Performance

- Advanced formulations provide an computational advantage
- Advantage is significantly more important for free solvers
- State of the art commercial solvers can be significantly better than free solvers
- Still, free is free!



# Summary and Main Messages

---



- Always choose Chewbacca!
- How to YOU use MIP / Optimization / OR / Analytics?
  - Study for the 2<sup>nd</sup> midterm!
  - Use JuMP and Julia Opt.
  - How about grad school down the river?
    - Masters of Business Analytics / OR
    - Ph.D. in Operations Research

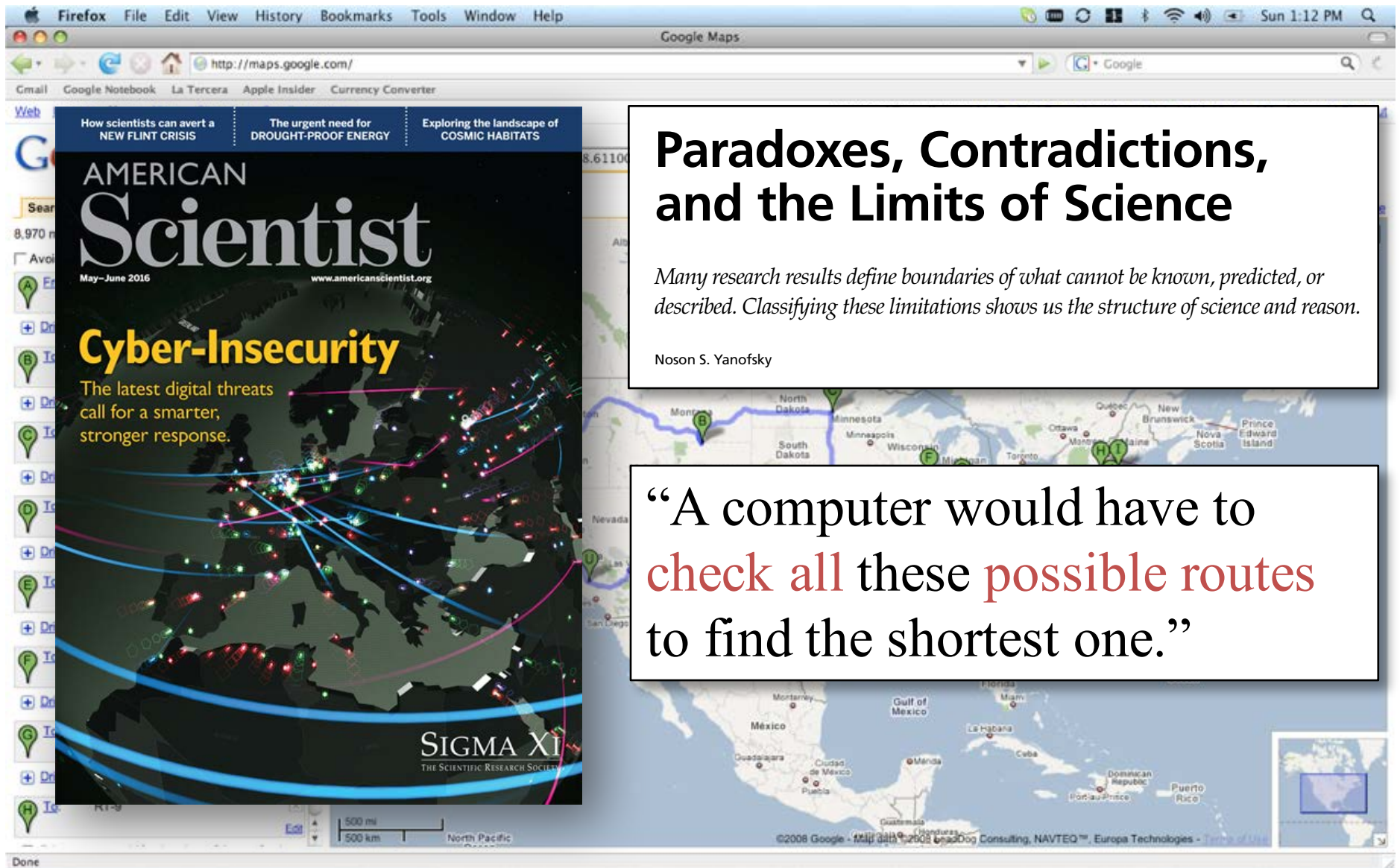


OPERATIONS  
RESEARCH  
CENTER

<https://orc.mit.edu>

How Hard is MIP?

# How hard is MIP: Traveling Salesman Problem ?



The image shows a Firefox browser window with Google Maps open. On the left, the cover of the May-June 2016 issue of 'AMERICAN Scientist' is displayed. The cover features a world map with glowing blue and red lines representing data connections, and the headline 'Cyber-Insecurity: The latest digital threats call for a smarter, stronger response.' Below the headline, it says 'SIGMA XI THE SCIENTIFIC RESEARCH SOCIETY'. The browser's address bar shows 'http://maps.google.com/'.

## Paradoxes, Contradictions, and the Limits of Science

*Many research results define boundaries of what cannot be known, predicted, or described. Classifying these limitations shows us the structure of science and reason.*

Noson S. Yanofsky

“A computer would have to check all these possible routes to find the shortest one.”



# MIP = Avoid Enumeration

---

- Number of tours for 49 cities =  $48!/2 \approx 10^{60}$
- Fastest supercomputer  $\approx 10^{17}$  flops
- Assuming one floating point operation per tour:  
>  $10^{35}$  years  $\approx 10^{25}$  times the age of the universe!
- How long does it take on an iphone?
  - Less than a second!
  - 4 iterations of **cutting plane** method!
  - Dantzig, Fulkerson and Johnson 1954 did it by hand!
  - For more info see tutorial in ConcordeTSP app
  - **Cutting planes** are the key for effectively solving (even NP-hard) MIP problems in practice.

# 50+ Years of MIP = Significant Solver Speedups

---

- Algorithmic Improvements (Machine Independent):
  - **CPLEX** v1.2 (1991) – v11 (2007): 29,000x speedup
  - Gurobi v1 (2009) – v6.5 (2015): 48.7x speedup
  - Commercial, but free for academic use
- (Reasonably) effective free / open source solvers:
  - GLPK, **COIN-OR (CBC)** and SCIP (only for non-commercial)
- Easy to use, fast and versatile modeling languages
  - Julia based JuMP modelling language
  - <http://julialang.org>
  - <http://www.juliaopt.org>

# Technique 1: Binary Quadratic $x^1, x^2 \in \{0, 1\}^n$

---

$$(x^1 - x^2)' \cdot \Sigma \cdot (x^1 - x^2) = v$$

$$X_{i,j}^l = x_i^l \cdot x_j^l \quad (l \in \{1, 2\}, \quad i, j \in \{1, \dots, n\}) :$$

$$X_{i,j}^l \leq x_i^l, \quad X_{i,j}^l \leq x_j^l, \quad X_{i,j}^l \geq x_i^l + x_j^l - 1, \quad X_{i,j}^l \geq 0$$

$$W_{i,j} = x_i^1 \cdot x_j^2 :$$

$$W_{i,j} \leq x_i^1, \quad W_{i,j} \leq x_j^2, \quad W_{i,j} \geq x_i^1 + x_j^2 - 1, \quad W_{i,j} \geq 0$$

$$\sum_{i,j=1}^n (X_{i,j}^1 + X_{i,j}^2 - W_{i,j} - W_{j,i}) \Sigma_{i,j} = v$$

# Technique 1: Binary Quadratic $x^1, x^2 \in \{0, 1\}^n$

---

$$x^1 \neq x^2 \iff \|x^1 - x^2\|_2^2 \geq 1$$

$$X_{i,j}^l = x_i^l \cdot x_j^l \quad (l \in \{1, 2\}, \quad i, j \in \{1, \dots, n\}) :$$

$$X_{i,j}^l \leq x_i^l, \quad X_{i,j}^l \leq x_j^l, \quad X_{i,j}^l \geq x_i^l + x_j^l - 1, \quad X_{i,j}^l \geq 0$$

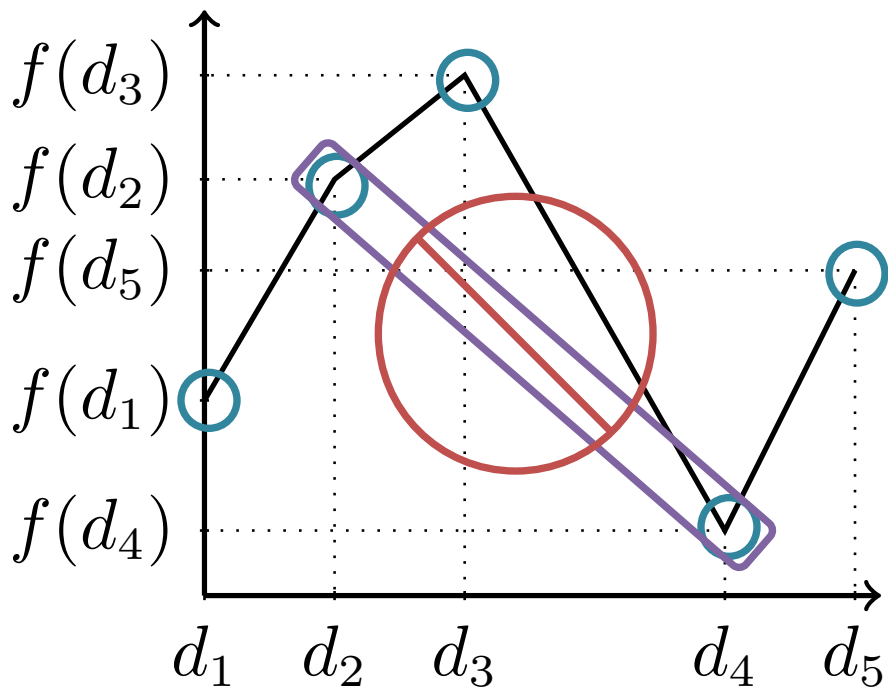
$$W_{i,j} = x_i^1 \cdot x_j^2 :$$

$$W_{i,j} \leq x_i^1, \quad W_{i,j} \leq x_j^2, \quad W_{i,j} \geq x_i^1 + x_j^2 - 1, \quad W_{i,j} \geq 0$$

$$\sum_{i,j=1}^n (X_{i,j}^1 + X_{i,j}^2 - W_{i,j} - W_{j,i}) \geq 1$$

# Simple Formulation for Univariate Functions

$$z = f(x)$$



Size =  $O(\# \text{ of segments})$

Non-Ideal: Fractional Extreme Points

$$\begin{pmatrix} x \\ z \end{pmatrix} = \sum_{j=1}^5 \begin{pmatrix} d_j \\ f(d_j) \end{pmatrix} \lambda_j$$

$$1 = \sum_{j=1}^5 \lambda_j, \quad \lambda_j \geq 0$$

$$y \in \{0, 1\}^4, \quad \sum_{i=1}^4 y_i = 1$$

$$0 \leq \lambda_1 \leq y_1$$

$$0 \leq \lambda_2 \leq y_1 + y_2$$

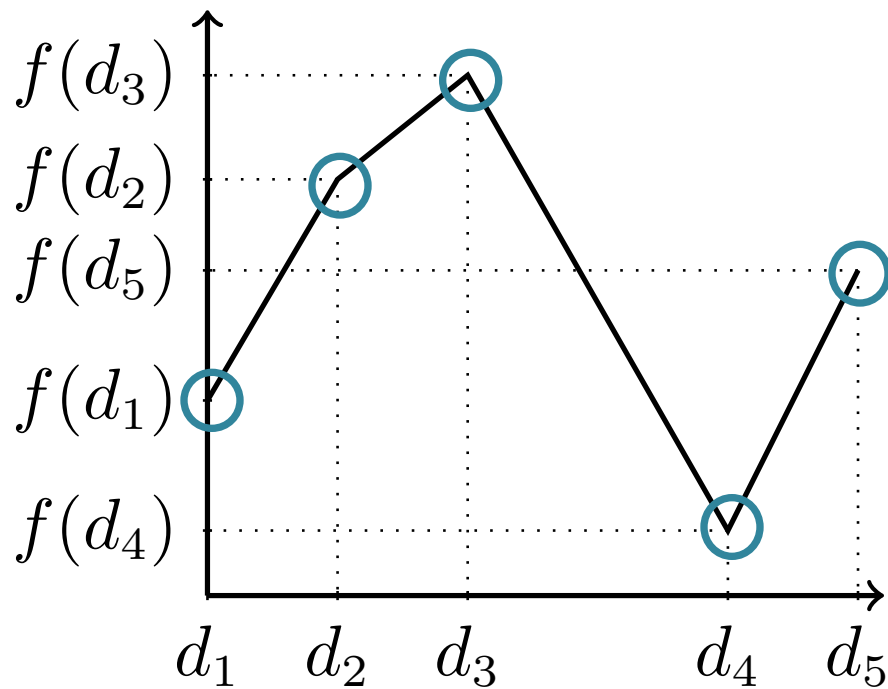
$$0 \leq \lambda_3 \leq y_2 + y_3$$

$$0 \leq \lambda_4 \leq y_3 + y_4$$

$$0 \leq \lambda_5 \leq y_4$$

# Advanced Formulation for Univariate Functions

$$z = f(x)$$



Size =  $O(\log_2 \# \text{ of segments})$

Ideal: Integral Extreme Points

$$\begin{pmatrix} x \\ z \end{pmatrix} = \sum_{j=1}^5 \begin{pmatrix} d_j \\ f(d_j) \end{pmatrix} \lambda_j$$

$$1 = \sum_{j=1}^5 \lambda_j, \quad \lambda_j \geq 0$$

$$y \in \{0, 1\}^2$$

$$0 \leq \lambda_1 + \lambda_5 \leq 1 - y_1$$

$$0 \leq \lambda_3 \leq y_1$$

$$0 \leq \lambda_4 + \lambda_5 \leq 1 - y_2$$

$$0 \leq \lambda_1 + \lambda_2 \leq y_2$$