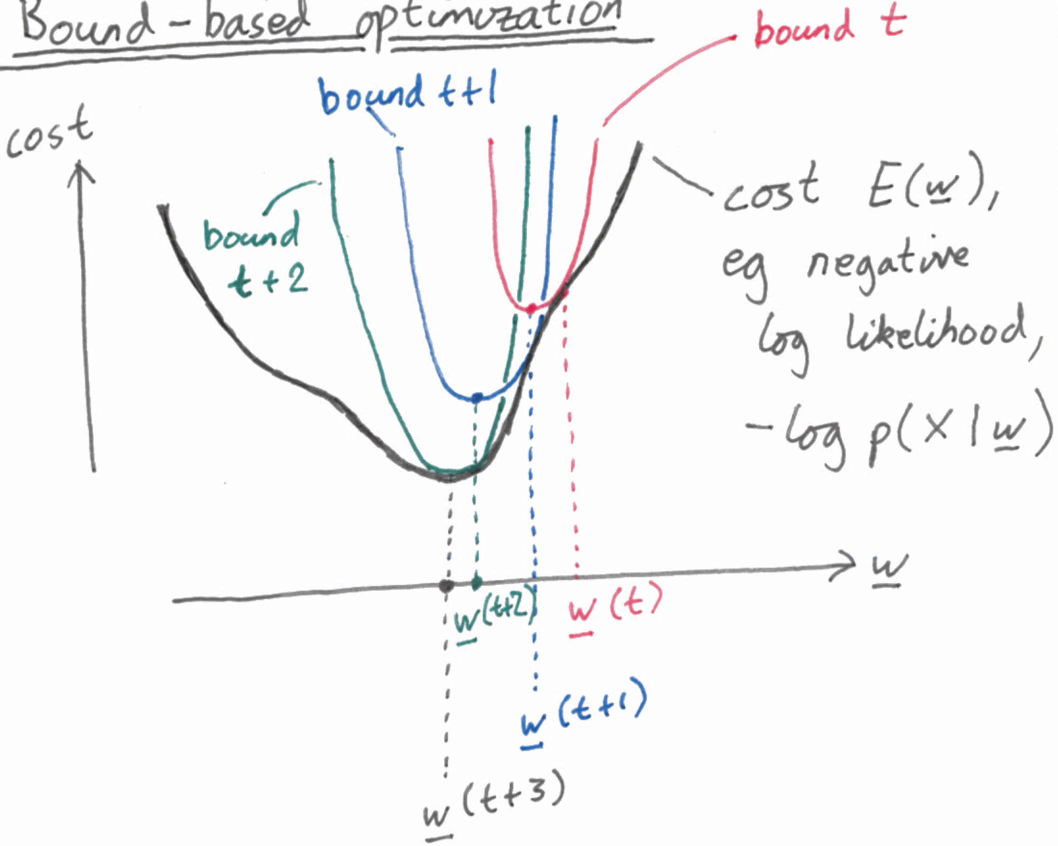
Bound-based optimization



For EM mix Gaussians

Bound minimized in closed form:

1) No step size / learning rate
2) Constraints on Σ, $\underline{\pi}$ satisfied

Revision: Laplace Approximation

"Energy" $E(\underline{w})$

Hessian, $H_{ij} = \dfrac{\partial^2 E}{\partial w_i \partial w_j}$

Approximate quadratic cost:

$$E(\underline{w}) \approx \tfrac{1}{2}(\underline{w} - \underline{w}^*)^T H (\underline{w} - \underline{w}^*) + \text{const.}$$

Newton's Method

eval at $\underline{w}^{(t)}$

$\nabla_{\underline{w}} E(\underline{w})$

- Initialize $\underline{w}^{(0)}$
- $\underline{w}^{(t+1)} = \underline{w}^{(t)} - H^{-1} \underline{g}$

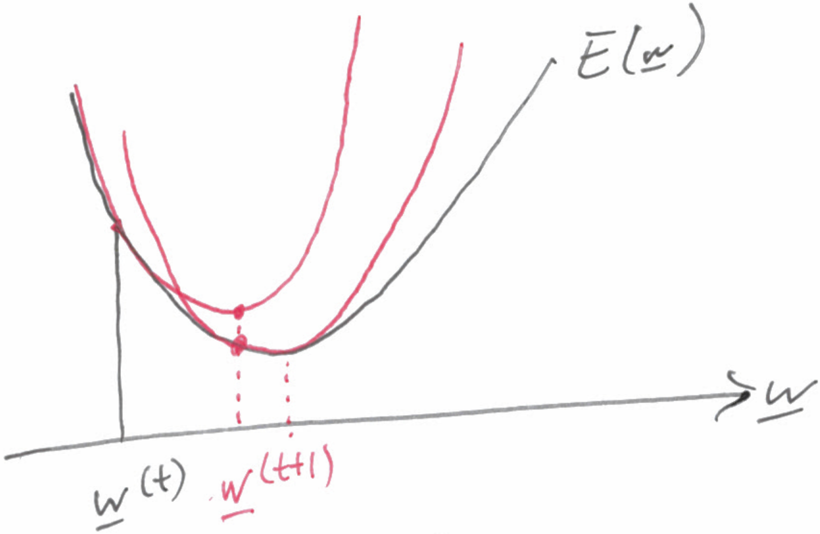
If cost is quadratic:

$\mathbb{I}$ if H invertible

$$\underline{g} = H(\underline{w} - \underline{w}^*)$$

$$\underline{w}^{(t+1)} = \underline{w}^{(t)} - \cancel{H^{-1} H}(\underline{w}^{(t)} - \underline{w}^*)$$

$$= \underline{w}^*$$


E(w)
w
w(t)
w(t+1)

Why use other optimizers?

- Convergence?
- Tuning?
- Constraints?
- SGD can't give "sparse" solutions
  - → some $w_d = 0$

L1 Regularization

$$c(\underline{w}) = \underbrace{E(\underline{w})}_{\text{Training error}} + \lambda \underbrace{\textstyle\sum_d |w_d|}_{\|\underline{w}\|_1}$$

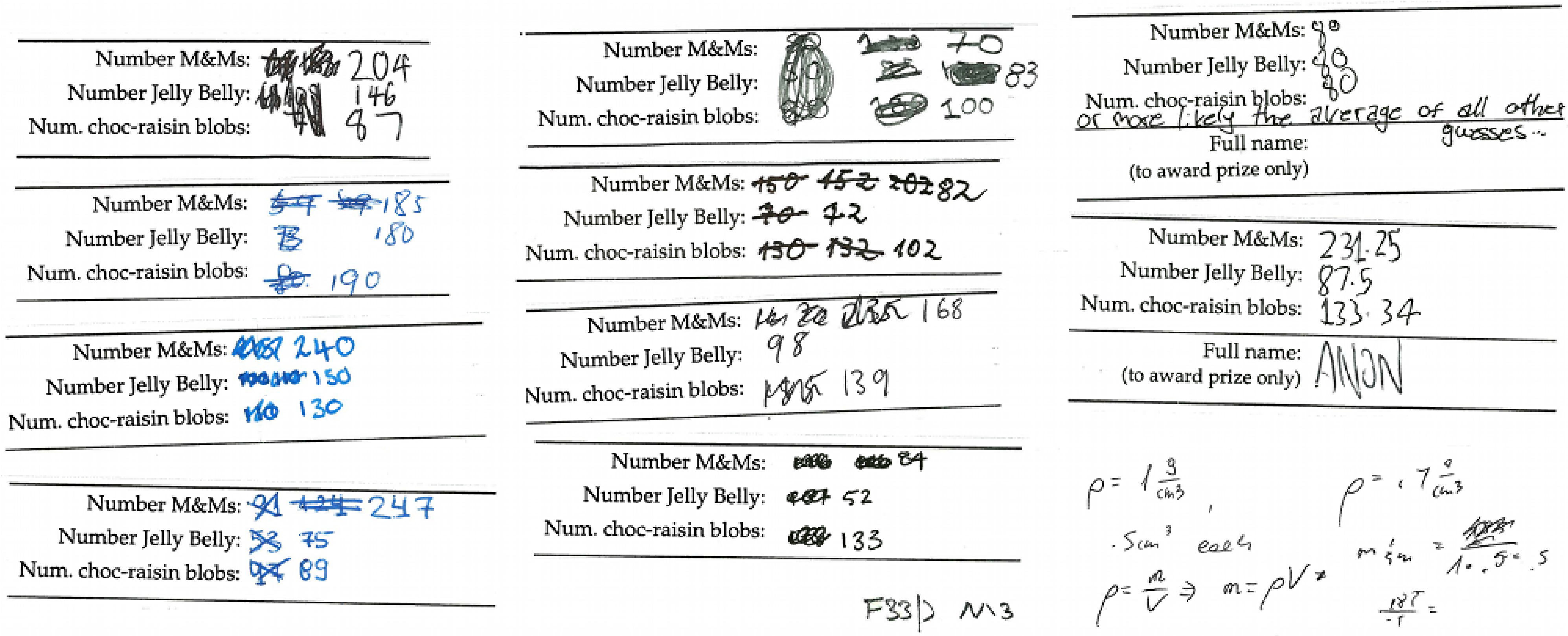
The confection

m&m's
(185g)

Jelly Belly
(100g)

Chocolate Raisins
(200g)

Stuff Inf2b students wrote



Number M&Ms: 204
Number Jelly Belly: 146
Num. choc-raisin blobs: 87

Number M&Ms: 185
Number Jelly Belly: 180
Num. choc-raisin blobs: 190

Number M&Ms: 240
Number Jelly Belly: 150
Num. choc-raisin blobs: 130

Number M&Ms: ~~91~~ ~~124~~ 247
Number Jelly Belly: ~~58~~ 75
Num. choc-raisin blobs: 89

Number M&Ms: 70
Number Jelly Belly: 83
Num. choc-raisin blobs: 100

Number M&Ms: ~~150~~ ~~152~~ ~~202~~ 82
Number Jelly Belly: ~~70~~ 72
Num. choc-raisin blobs: ~~130~~ ~~132~~ 102

Number M&Ms: 168
Number Jelly Belly: 98
Num. choc-raisin blobs: 139

Number M&Ms: 84
Number Jelly Belly: 52
Num. choc-raisin blobs: 133

F33D M13

Number M&Ms: 90
Number Jelly Belly: 80
Num. choc-raisin blobs: 80
or more likely the average of all other guesses...
Full name:
(to award prize only)

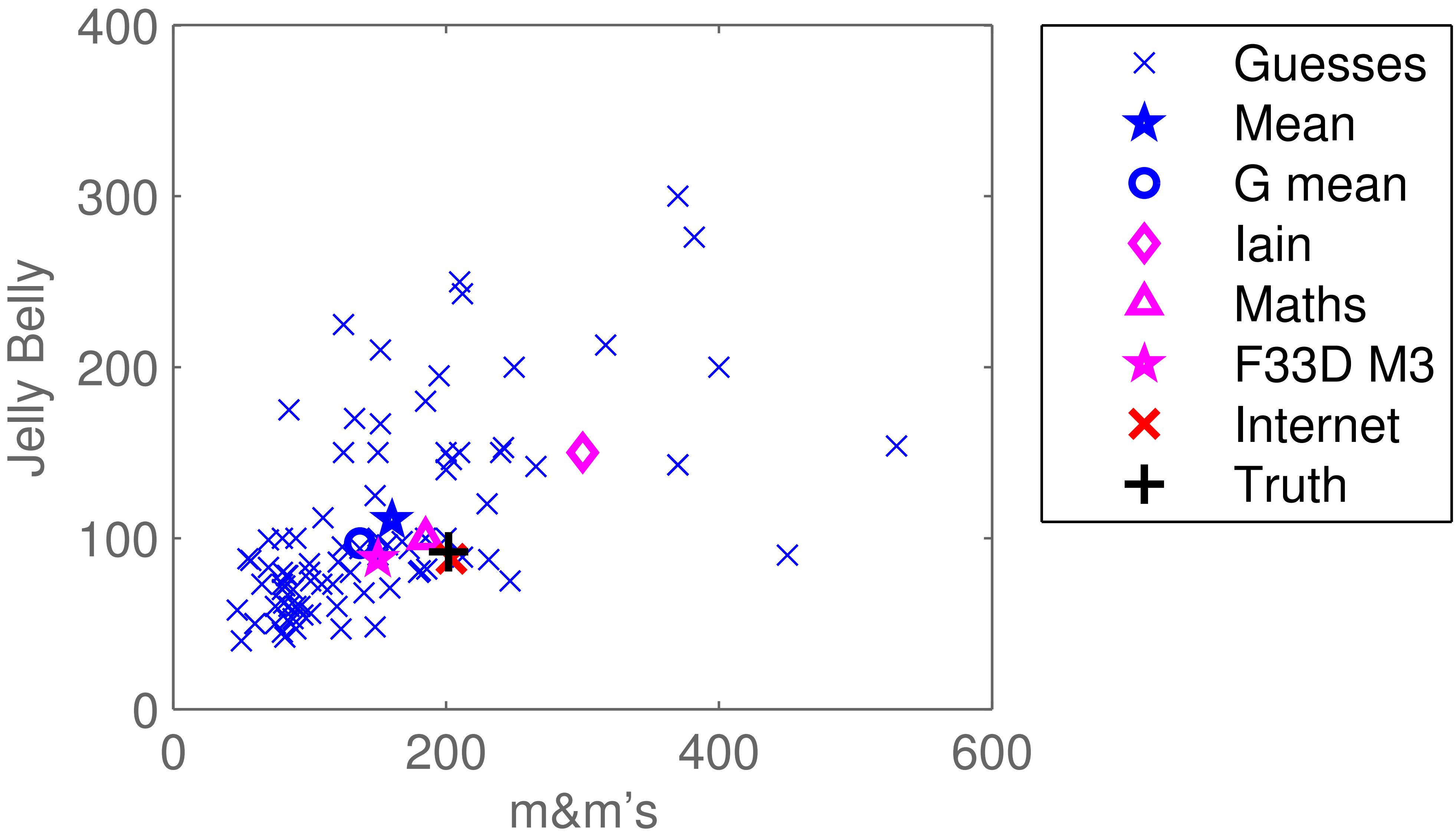
Number M&Ms: 231.25
Number Jelly Belly: 87.5
Num. choc-raisin blobs: 133.34
Full name: ANON
(to award prize only)

$\rho = 1\frac{g}{cm^3}$, $\rho = .7\frac{g}{cm^3}$

$.5cm^3$ each

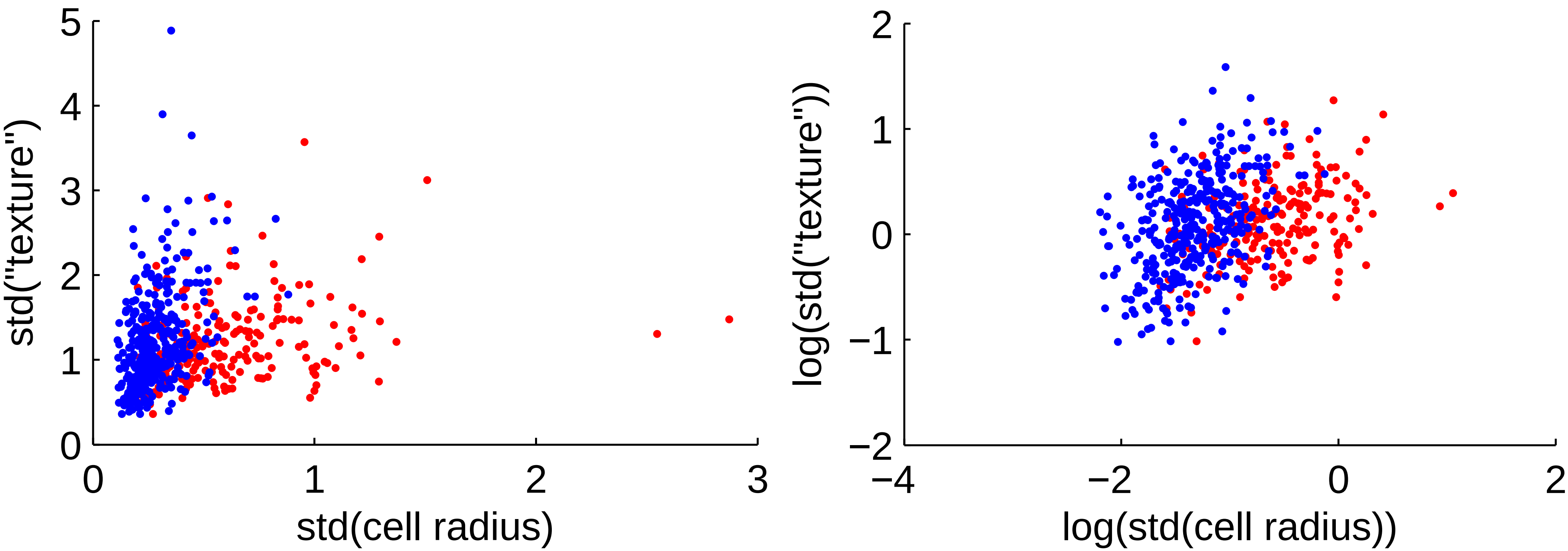
$\rho = \frac{m}{V} \Rightarrow m = \rho V$

A 2D space



For 3D and more, check out the code on the website.

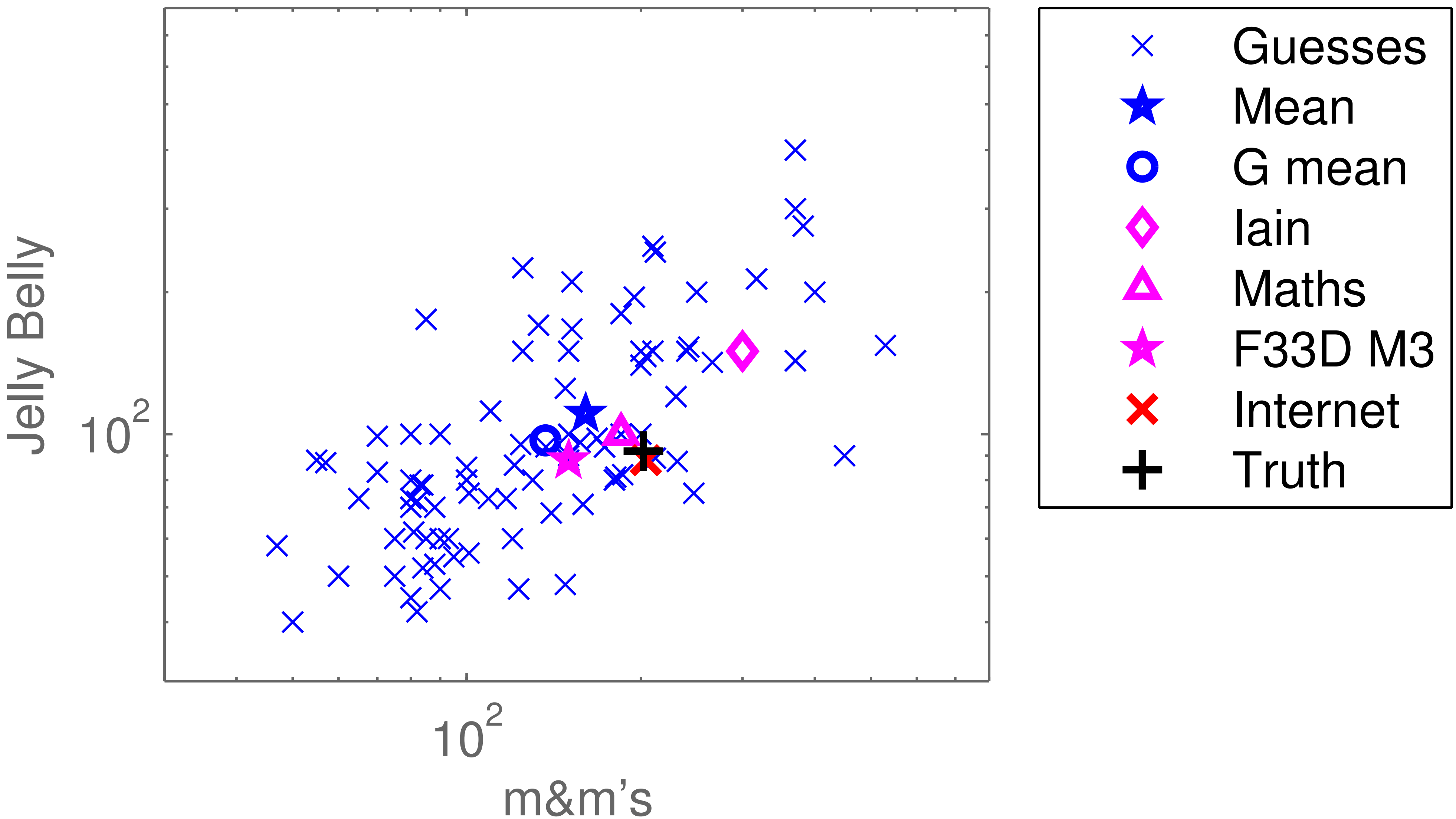
Often log-transform +ve data



Wisconsin breast cancer data

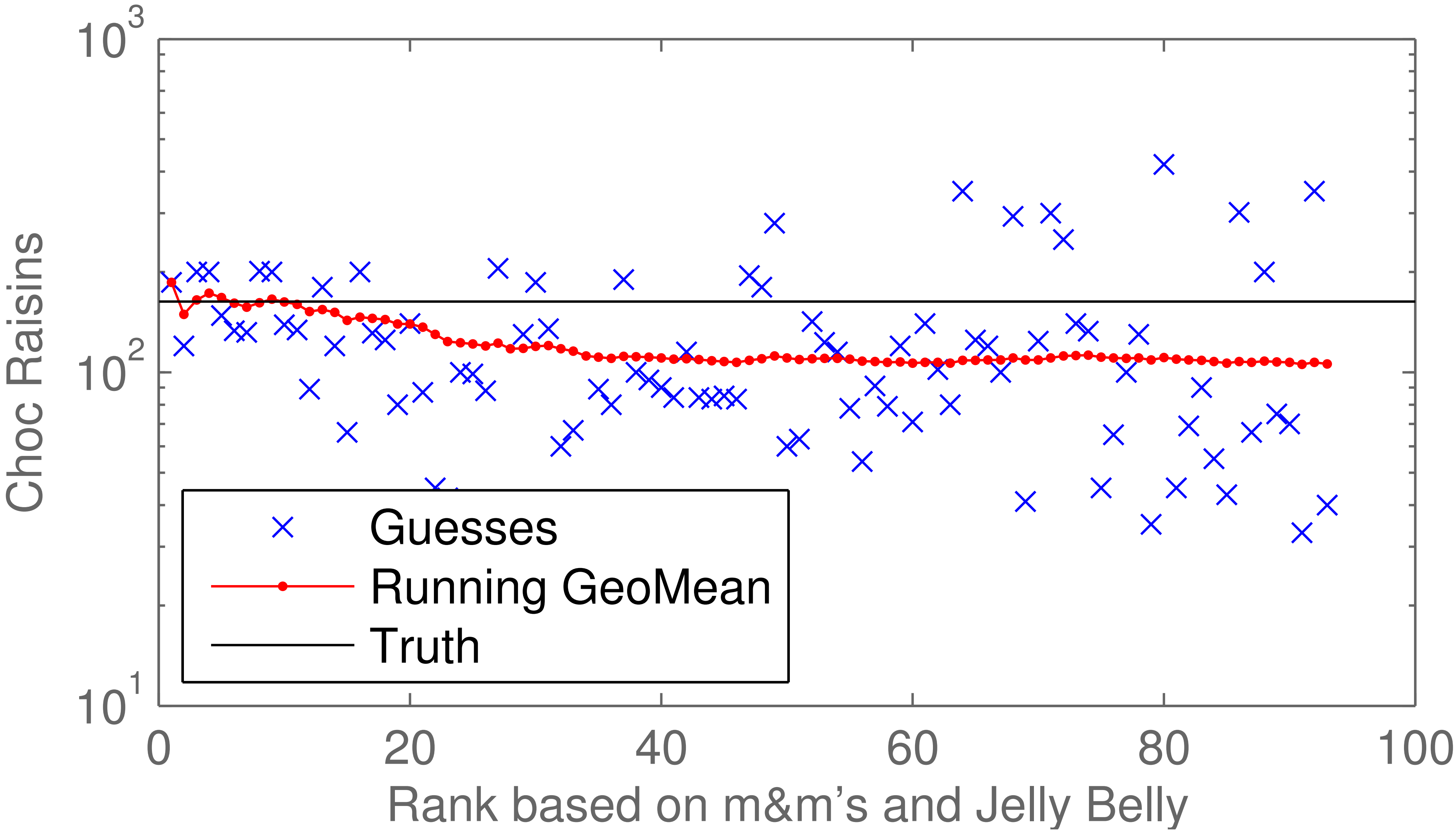
UCI ML repository

Count guesses on log-scale



Were some people just lucky?

Ranking by past performance



Ensemble of Models

Two motivations:

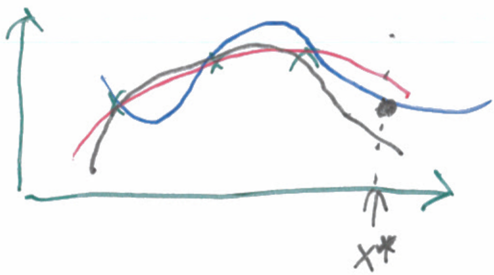
1. Reduce over-fitting
2. Reduce under-fitting

Example 1

Bayesian model averaging:

$$p(y \mid \underline{x}, D) = \int p(y \mid \underline{x}, \underline{w})\, p(\underline{w} \mid D)\, d\underline{w}$$

$$\approx \underbrace{\frac{1}{S} \sum_{s=1}^{S} p(y \mid \underline{x}, \underline{w}^{(s)})}_{\text{Ensemble of } S \text{ predictors}} \qquad \underline{w}^{(s)} \sim p(\underline{w} \mid D)$$



Another similar ensemble Bagging

"Bootstrap aggregation"

N training examples

Training time:

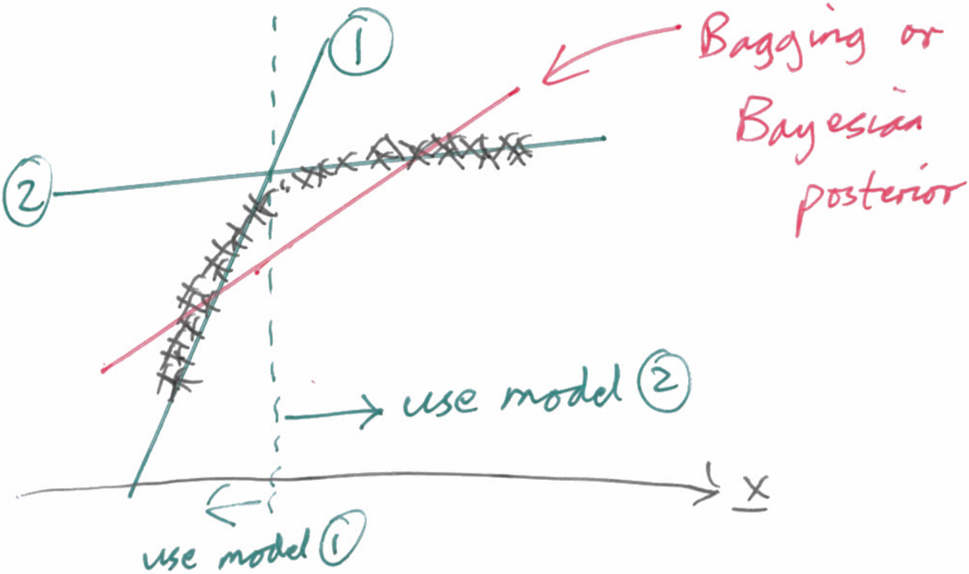
for $s = 1 \ldots S$:

Bootstrap: create a new dataset, sampling N datapoints from training data with replacement

Fit model to dataset → predictor s

Test time:

Average predictions (or majority vote)



2 Model combination

$$p(y|\underline{x},\theta) = \sum_{z} \underbrace{p(y|\underline{x}, z, \theta)}_{\text{Any regression model}} \underbrace{p(z|\underline{x},\theta)}_{\text{"Gating network" Any classifier}}$$

choice expert

"Mixture of experts"

Fit θ, neg log likelihood,
Regularize fit, Bayesian, Bagging, ...

Another way to build complicated model
Boosting.