A Simpler Explanation of Differential Privacy and Its Applications to Machine Learning

Nina Zumel
Win-Vector, LLC
December 2, 2015

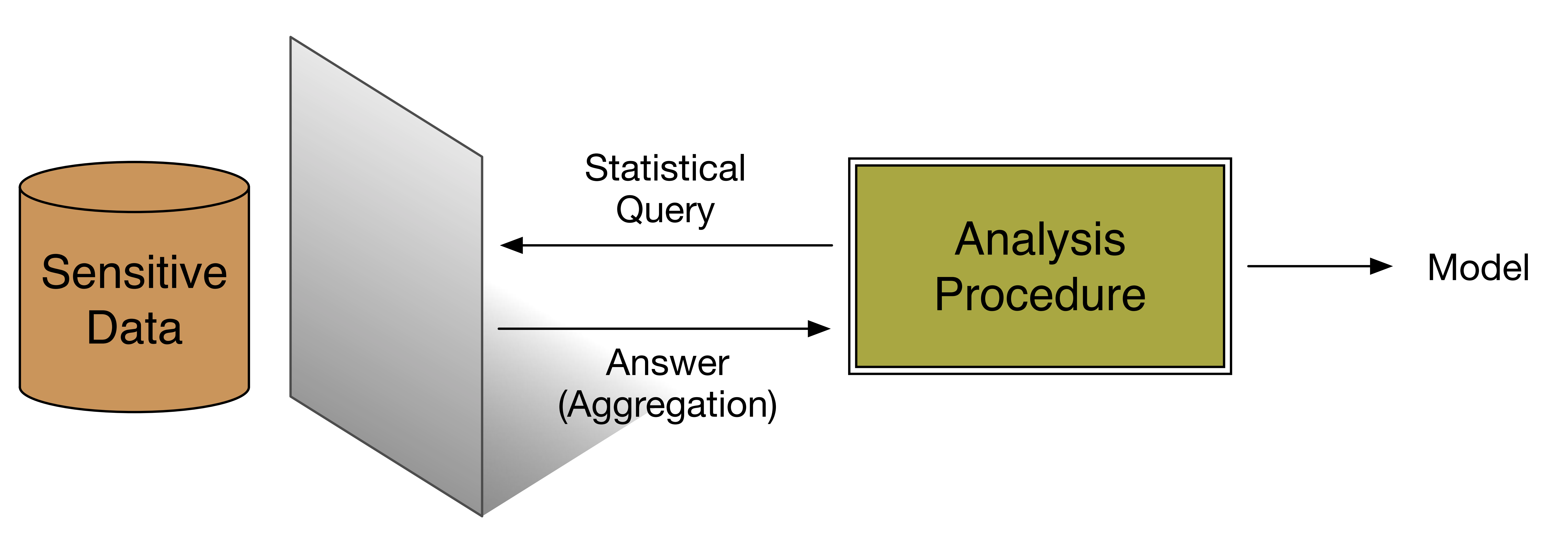
Win-Vector LLC

Differential Privacy

- Secure Analysis over Sensitive Data
  - 2006: AOL Search Data "Anonymized" Release
  - Netflix Data
- Can we analyze data without leaking information?

Thelma Arnold, User #4417749

Useful Aggregations

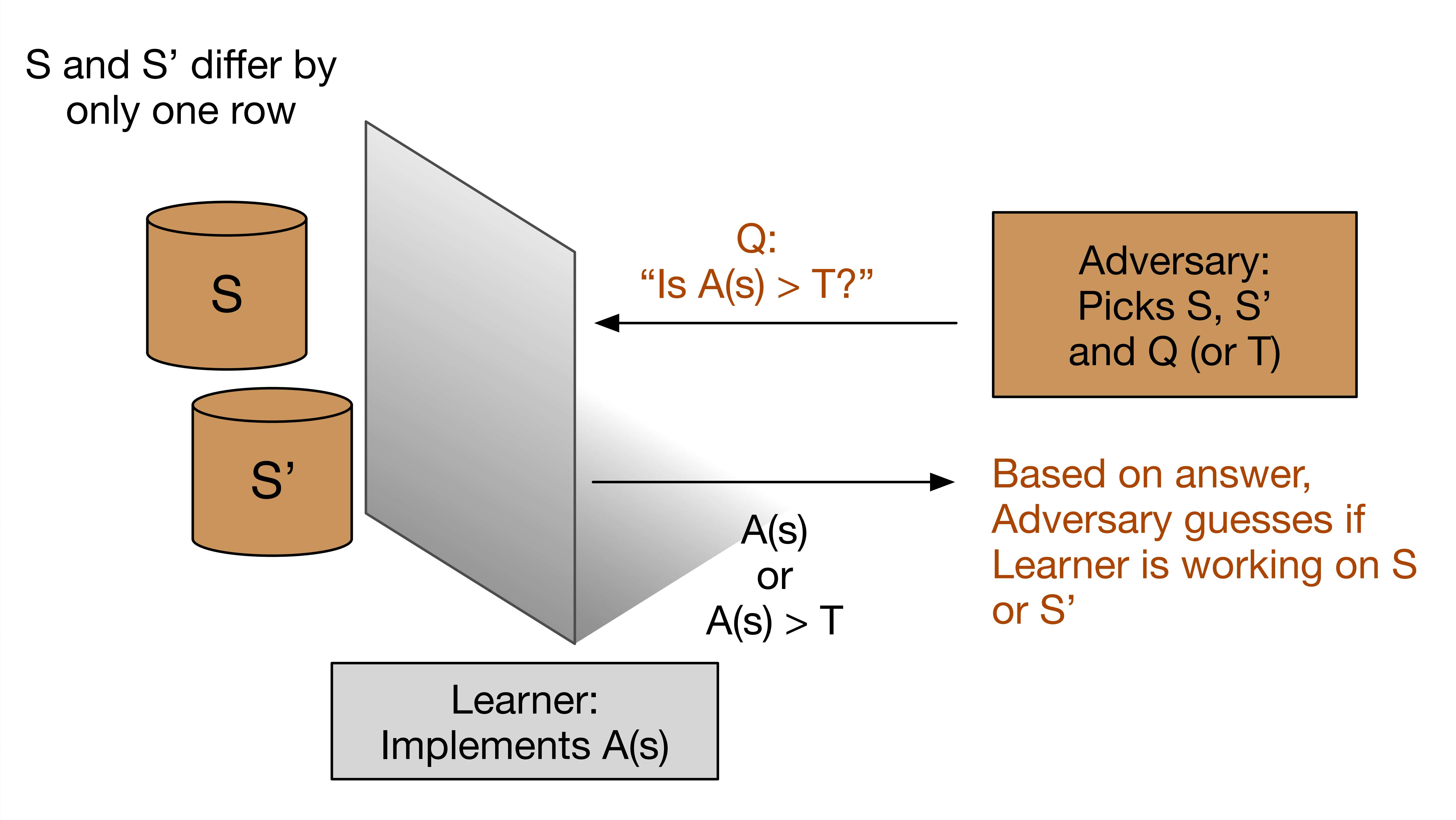


- Mixed Success in Analysis
- Recent Results for Machine Learning
  - Differential Privacy to Reuse Test Data
  - Reduce Upward Bias in Model Evaluation

Outline

- Define Differential Privacy
- Give an example of Recent Results
  - Reusable Hold-out
  - Nested Models

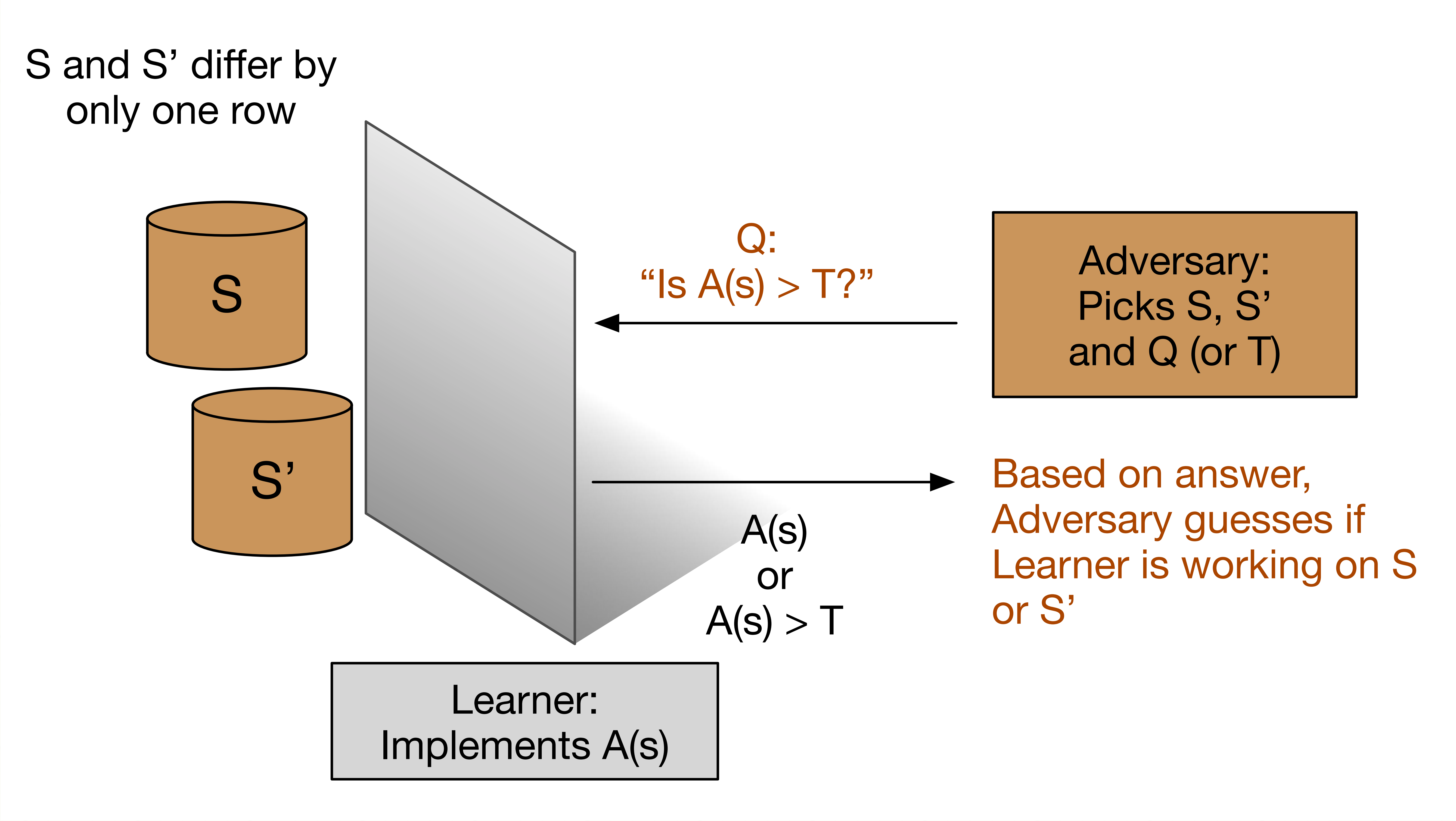
The Differential Privacy Game



Assume A() returns a value in [0,1]

Assume Q is the interval (T 1] (so adversary picks T)

The Differential Privacy Game



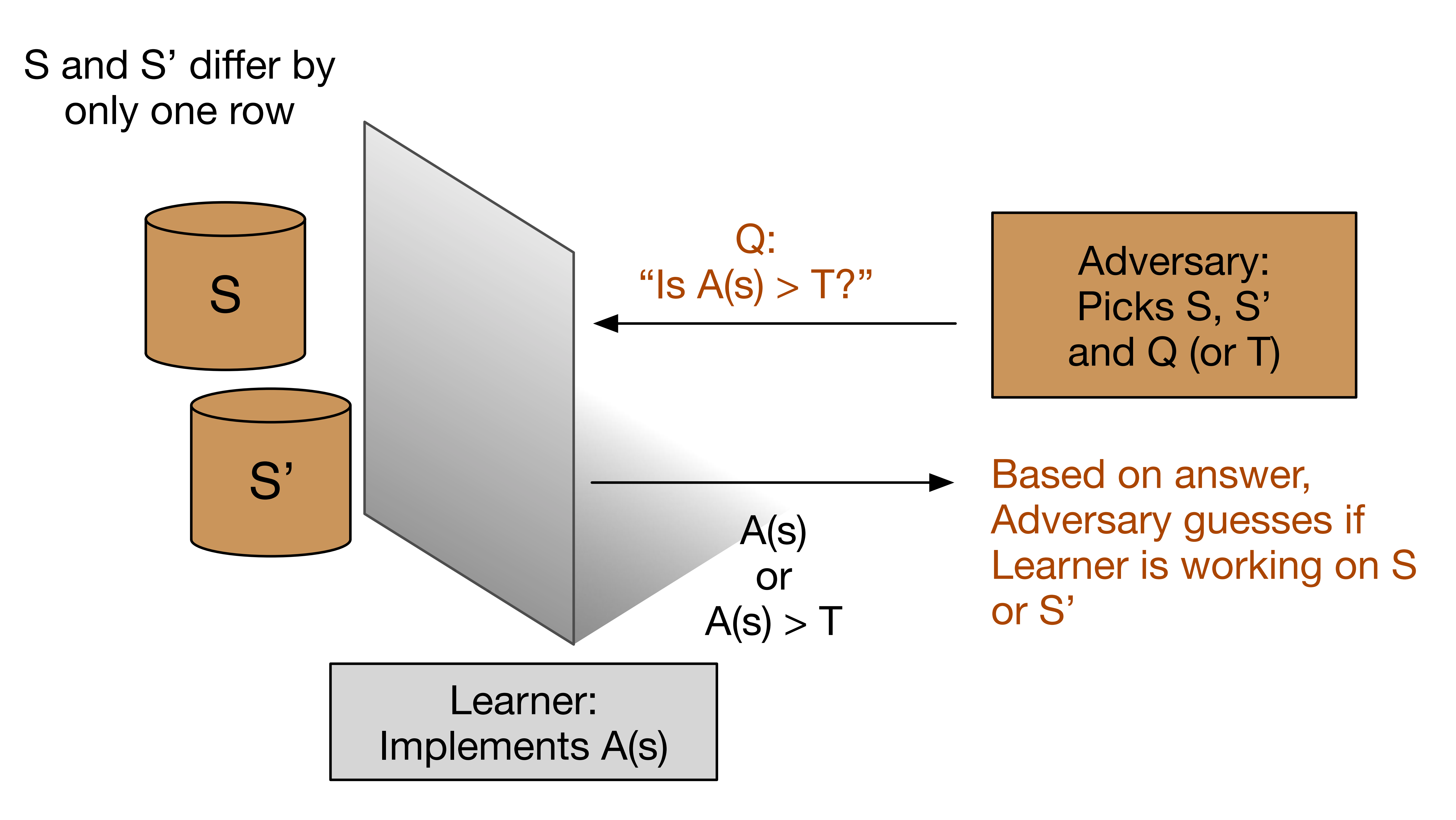
Over many rounds of the game (with the same S, S′):

A(S) > T with probability p
A(S′) > T with probability p′

If p >> p′ (or vice versa), adversary usually wins.

If p/p′ ~1, adversary can't do better than random guesses.

ε-Differential Privacy



A() is ε-differentially Private if

$$\left| \log \left(\frac{\mathrm{Prob}[A(S) \in Q]}{\mathrm{Prob}[A(S') \in Q]} \right) \right| \leq \epsilon$$

for all choices of S, S′, Q

In English: A(S) looks a lot like A(S′)

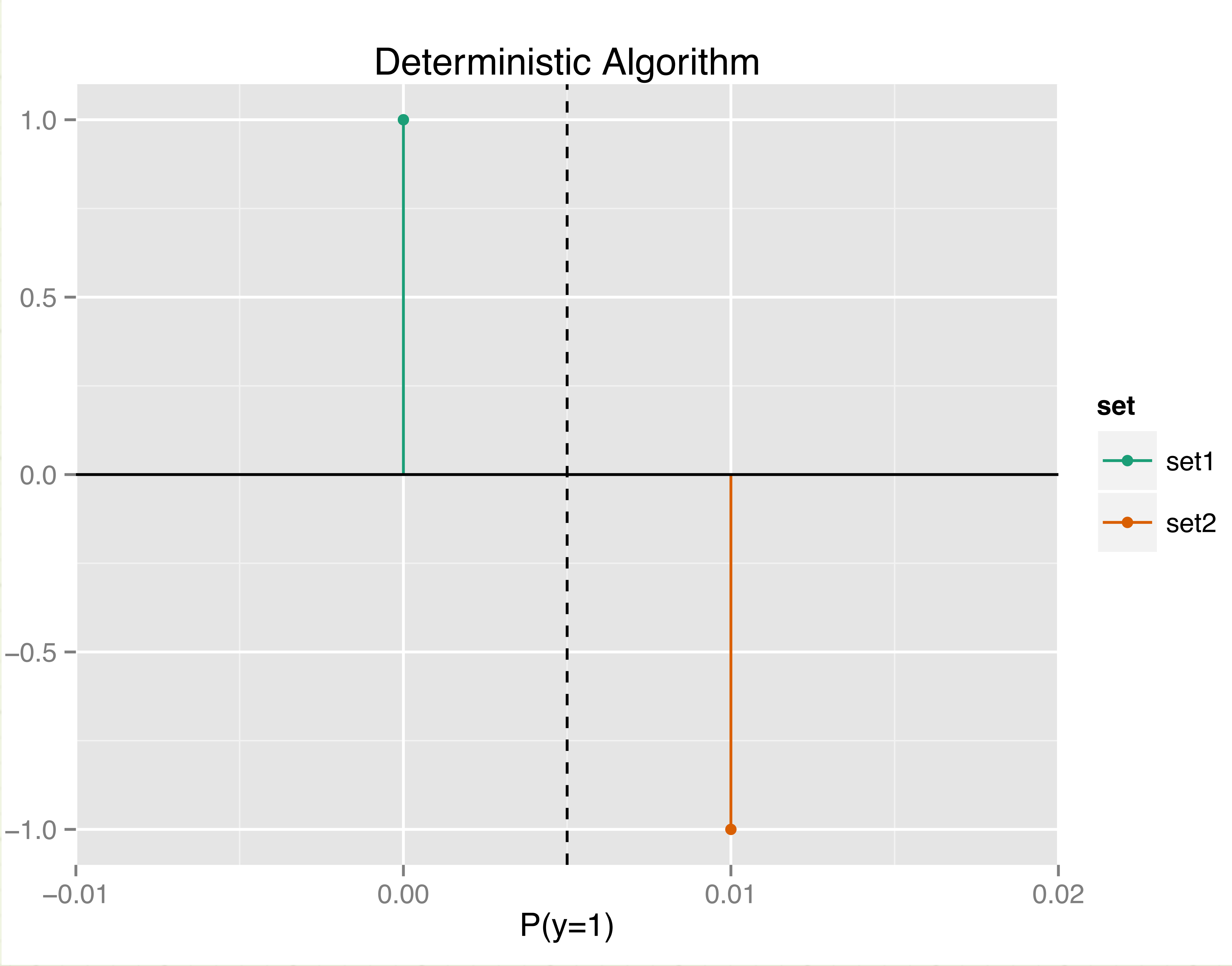
Example

- A(s) : returns the approximate mean value of s
- S: {0,0,…,0} (100 zeros)
- S’: {1,0,…,0} (1 one, 99 zeros)
- Adversary picks T so that if A(s)>T, s is S’
  (with high probability)

Win-Vector LLC

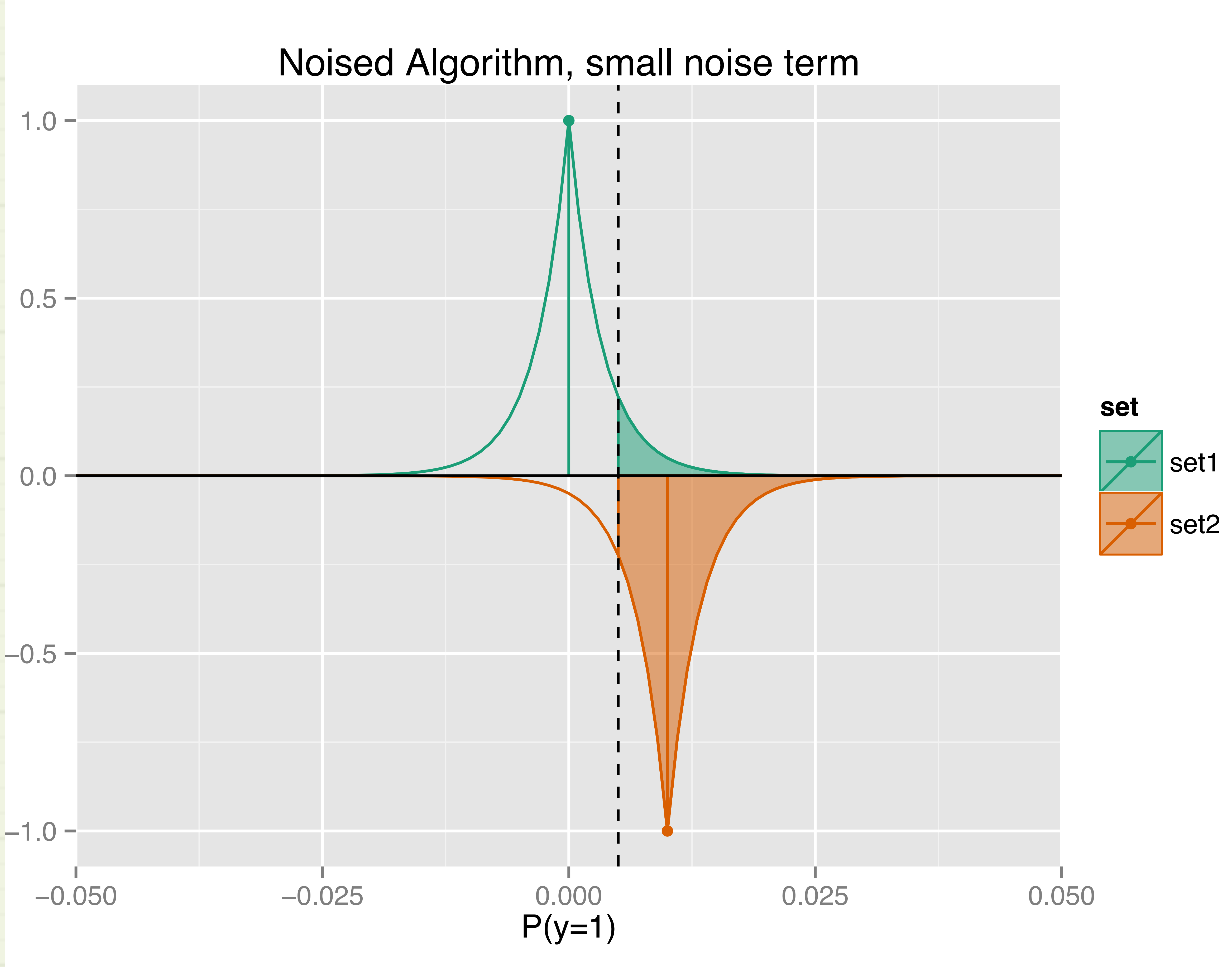
Deterministic Case: A(s) = E(s)

- A(S) = 0, A(S’) = 0.01
- Adversary picks T=0.005
- Not differentially private for any ε.



Add Noise

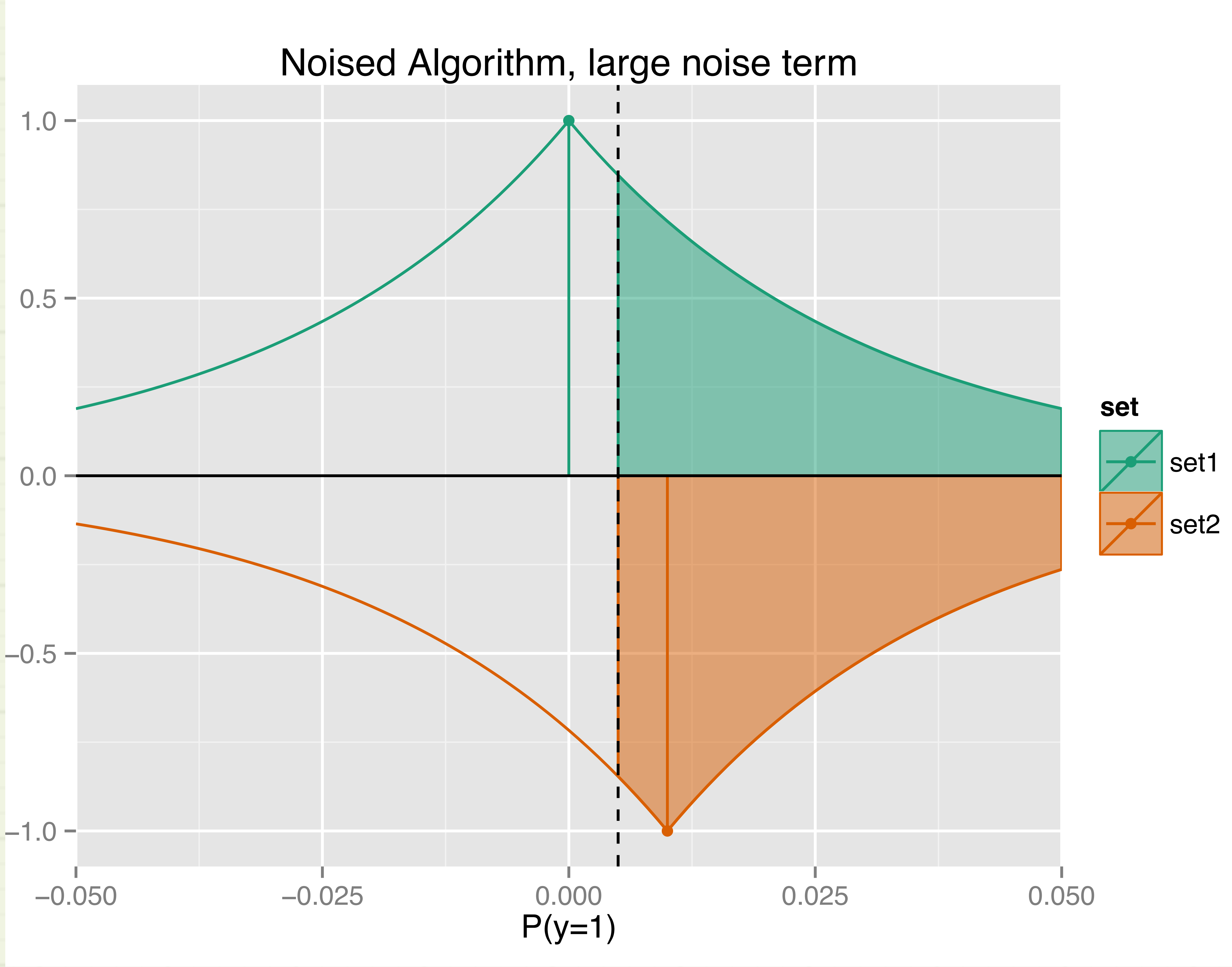
- Laplacian Noise: L(0, σ)
  - σ = 1/3n
- Now sometimes A(S) > T
- Need more noise



Add More Noise

- Need σ > 1/n
- σ = 3/n = 0.03
- Now often A(S) > T
- If R = ratio of green:orange

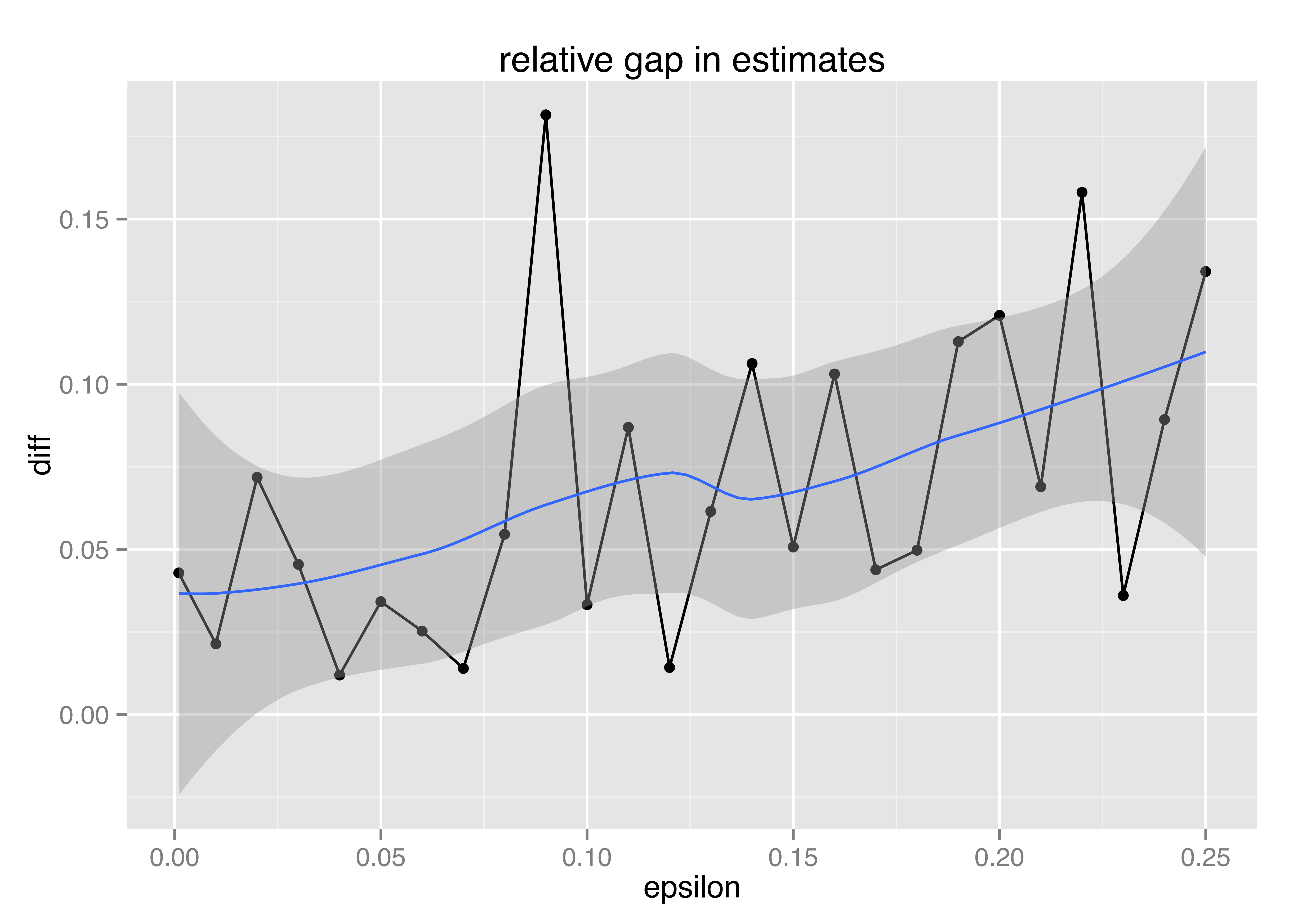
log(abs(R)) = ε



Win-Vector LLC

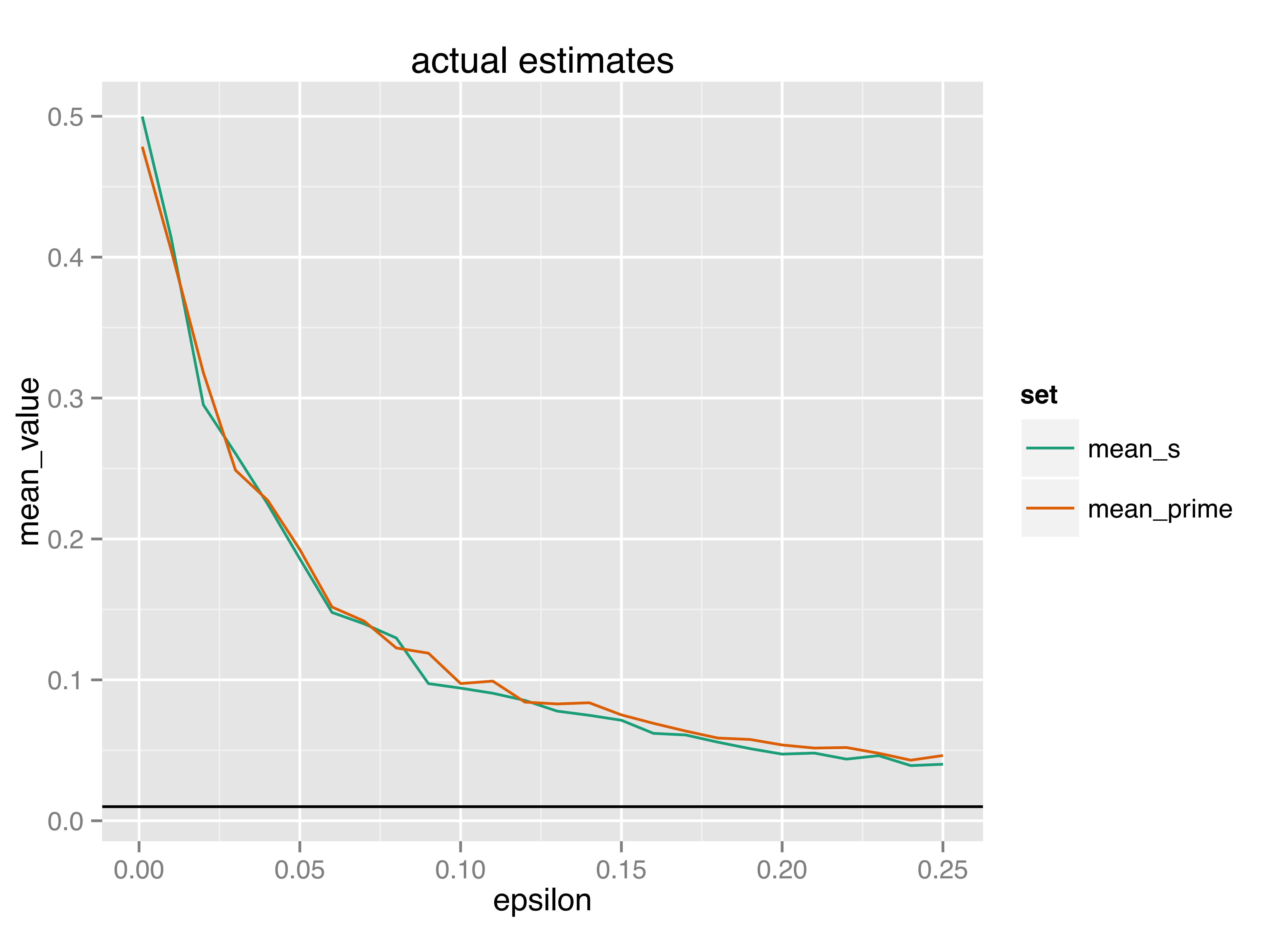
Stricter ε : A(S) → A(S')

- We can simulate the game I described
  - https://github.com/WinVector/Examples/blob/master/DiffPriv/DiffPrivExample.R
- 1000 rounds
- A(S) and A(S') get closer (in % difference)



Stricter ε : Estimates Poorer

- E(S) = 0; E(S') = 0.01
- Hard to balance privacy and good analysis!



Win-Vector LLC

Differential Privacy Applied to Reusable Holdout Data

- Standard ML Practice: Training/Test split
  - or Training/Calibration/Test
- Ideally: Look at Test only once
- In practice: Look at Test, tweak model, look at Test…
- Upward-biased performance estimates on Training

— and Test

Win-Vector LLC

How Many Times Can You Use The Test Set?

- In Theory: exp(N) times, where N is size of Test
- In Practice: N*N times —non-adaptively
  - not true if you tune model after a query
- New results: N*N times **adaptively**
  - Dwork, Feldman, Hardt, Pitassi, Reingold, Roth, 2015

Win-Vector LLC

The Idea

- Use differential privacy to evaluate candidate models on holdout sets “without looking at data.”

- Reduce the bias from test set performance estimates: test set estimates should approximate true out-of-sample performance.

Example: Stepwise Regression

- Use the training set to train a model with k parameters, and the test set to evaluate its accuracy, and pick the best (most improved) k-parameter model.

- Greedy: kth-step uses previous best k-1 parameters

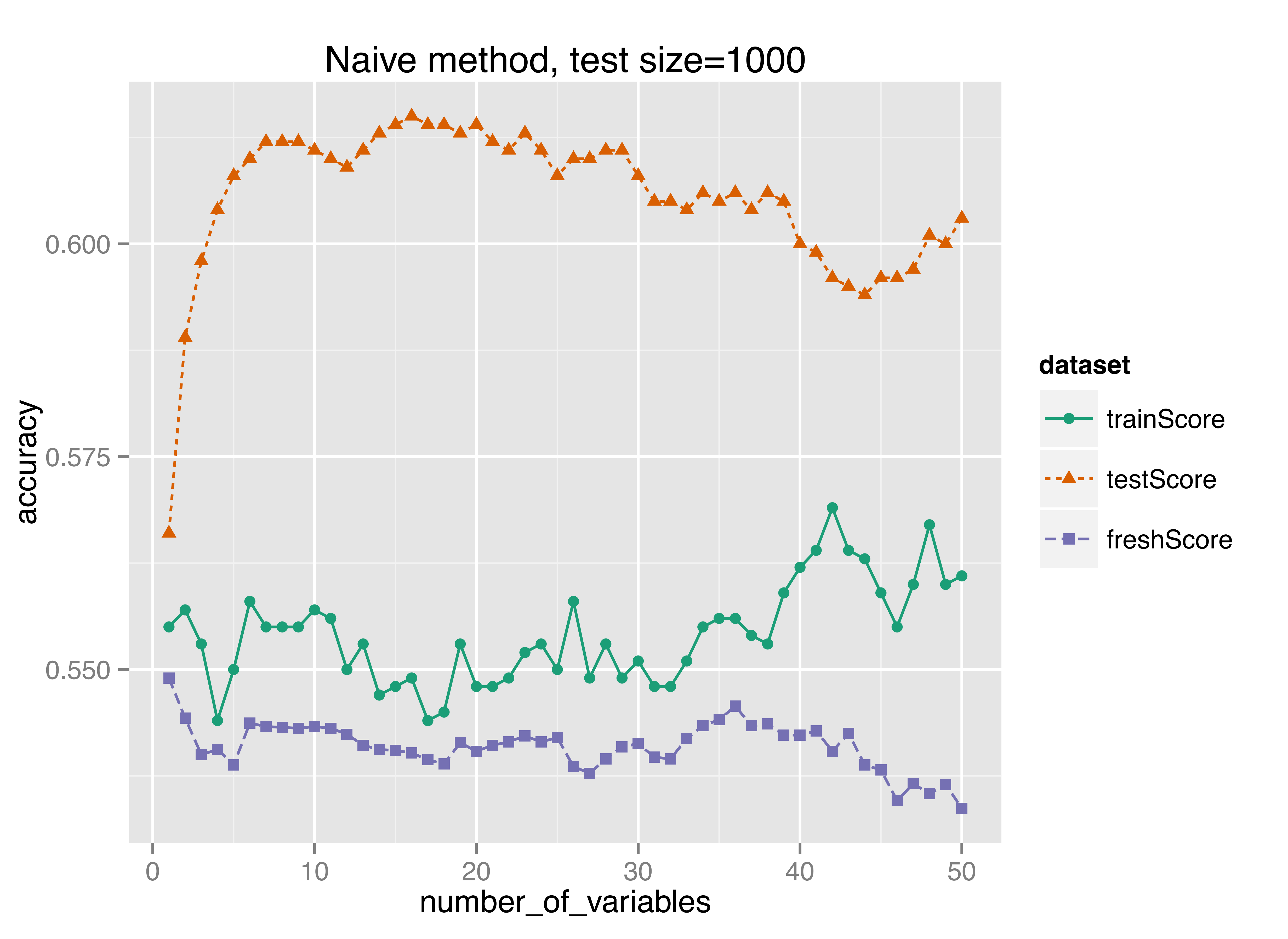
- Run until k=50

Experiment

- Simulated data
- Binary classification (50% positive class)
- 110 candidate variables
    - 10 with signal, 100 with pure noise
- 1000 rows training, 1000 rows test
- Estimate true out-of-sample performance with “fresh” set of 10,000 rows

Naive Method

- Test set more up-biased than training!
- Algorithm only picked 1 signal variable (the first)
- Neither test nor training sets estimate true model performance



Thresholdout

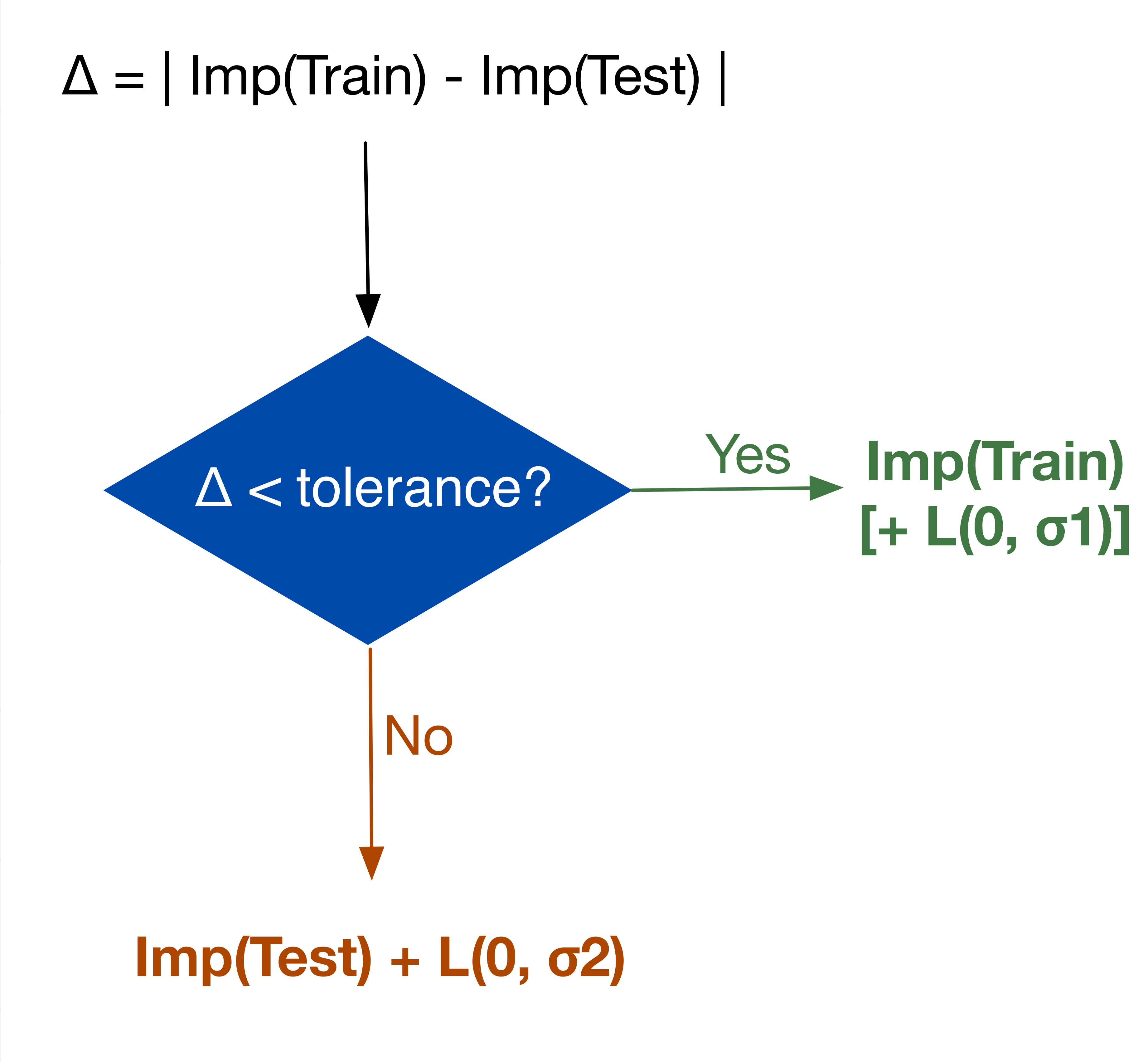
- Goal — Use Test to both:
    - Evaluate models
    - Estimate out-of-sample model performance

- Improvement:
  Accuracy(k) - Accuracy(k-1)

- Tolerance:
  σ/2 + L(0, σ/2)

- Never directly inspect Test, so leak information slower

Dwork, Feldman, Hardt, Pitassi, Reingold, Roth, 2015



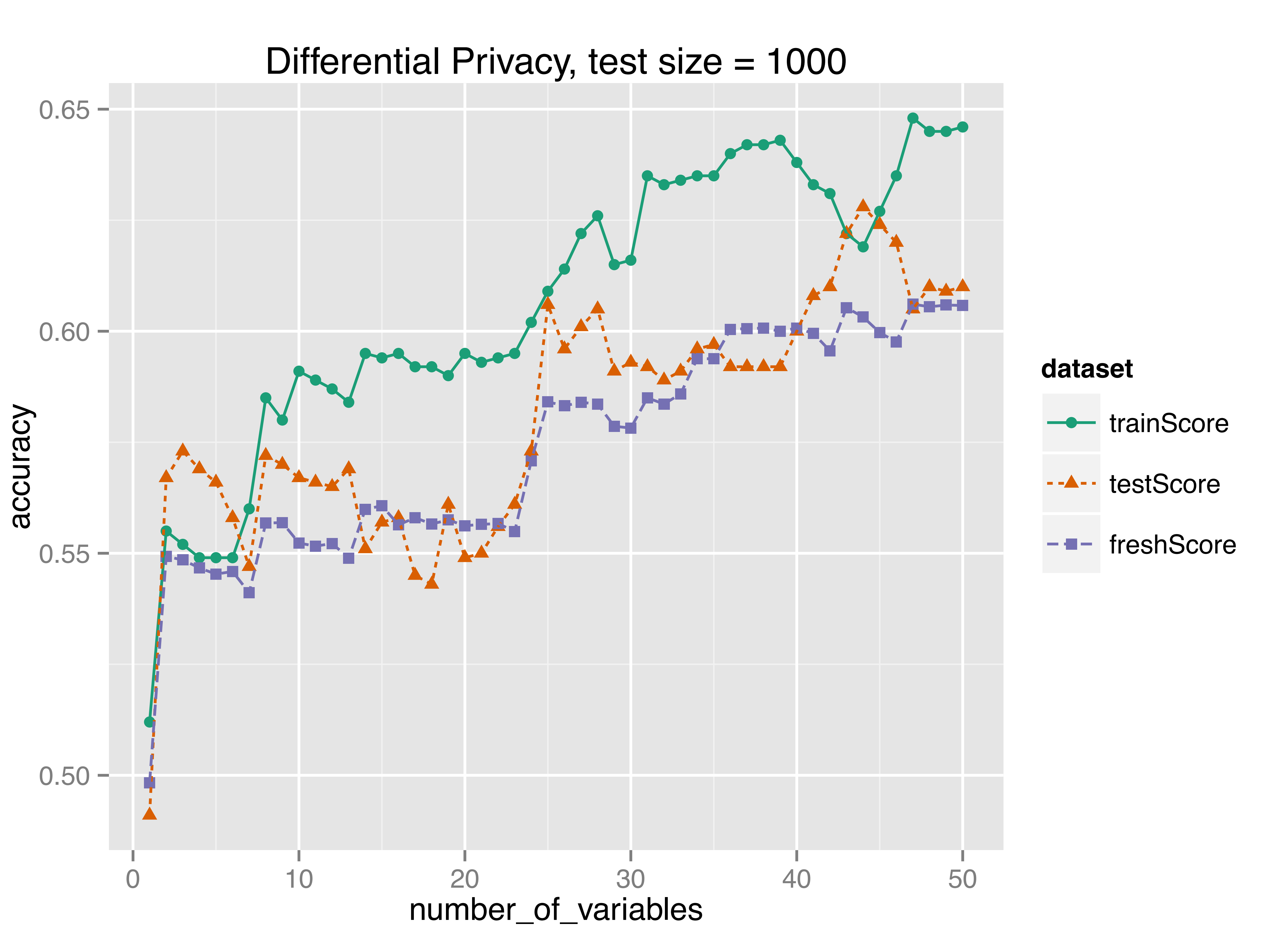
Win-Vector LLC

Result

- Test performance tracks Fresh performance

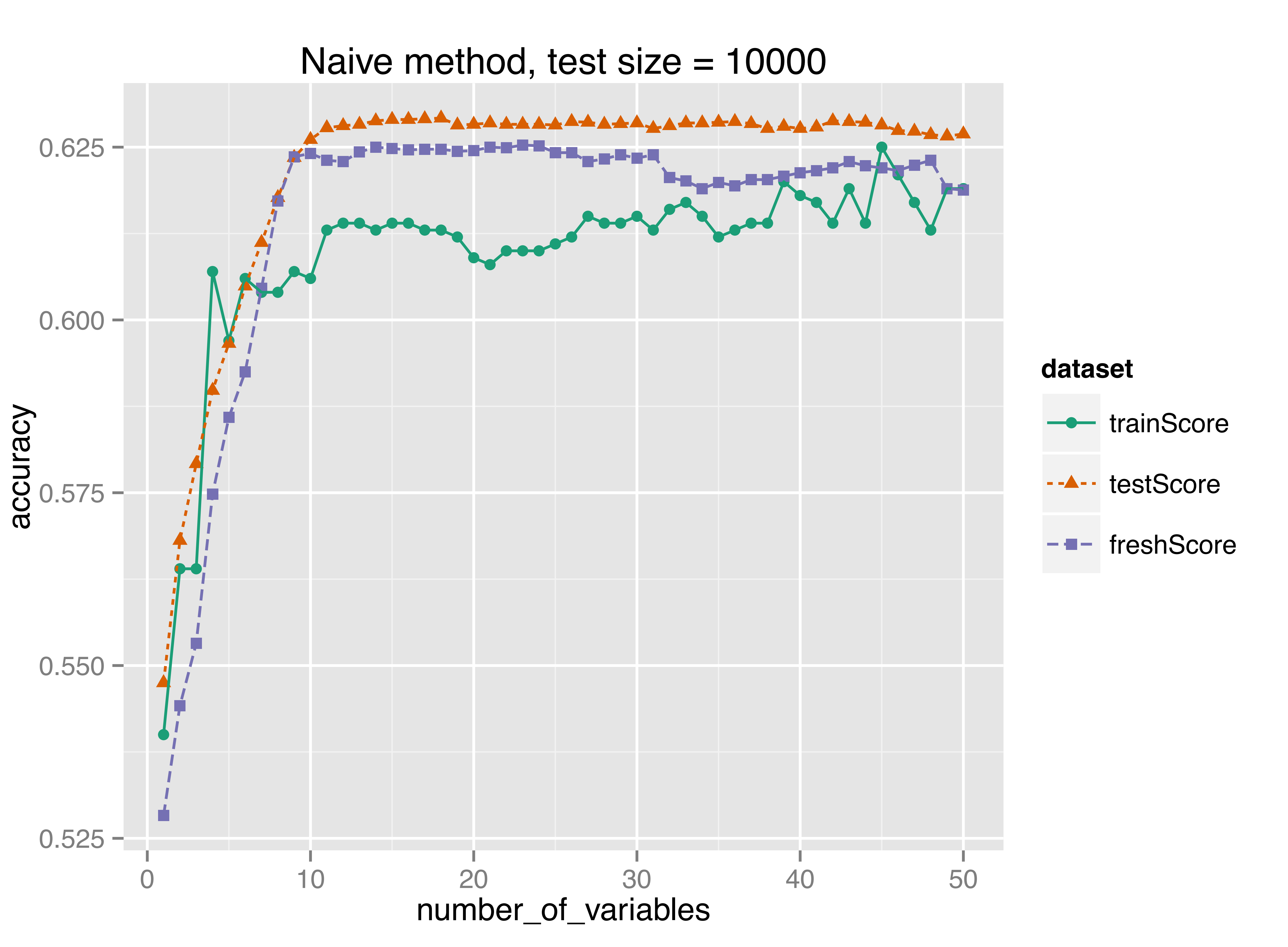
- Found all 10 signal variables
  - But started picking noise early
  - Last signal variable: #36

- Peak accuracy ~61%



For Comparison: LARGE Test Set

- N=10,000, no DP
- Found 9 signal variables immediately
- Accuracy ~62.5% (9 vars)
- Test set only slightly upwardly biased
  - So large, we don't contaminate it much

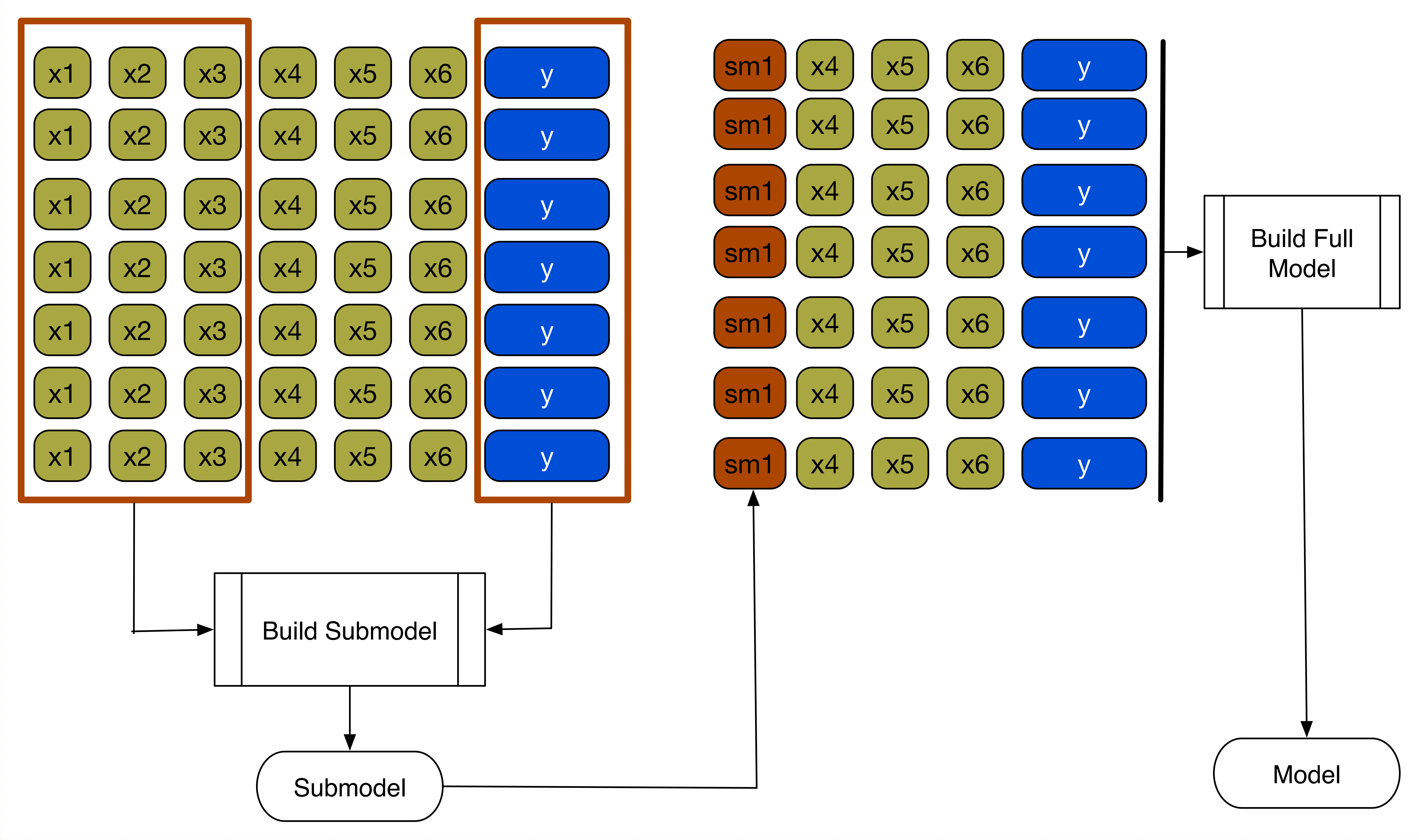


Takeaways

- Can think of Thresholdout as simulating a larger test set.
- DP designed to minimize excess generalization error — not find best possible model
    - The two are related, of course
- Stepwise Regression is dangerous
    - LOTS of queries

Win-Vector LLC

Differential Privacy applied to Nested Models



Win-Vector LLC

Example: Effects Coding

- For categorical variables with many levels.
  - K levels = K-1 indicator vars

- Re-encode the categorical variable as a few numerical variables.

Make_Model	Price	...	SoldInWeek
VW_Golf	$26,000	...	Yes
Mazda_Miata	$24,000	...	No
VW_Golf	$32,000	...	Yes
Toyota_Prius	$21,500	...	No

Bayesian Model or Model by Counts

Make_Model	P(SoldIn Week)	Impact
VW_Golf	0.6	0.2
Mazda_Miata	0.34	-0.06
Chevy_Camaro	0.16	-0.24
Toyota_Prius	0.72	0.32
Lotus_Elise	**1.0**	**0.6**
…	…	…
Overall	0.4	0

Bayesian

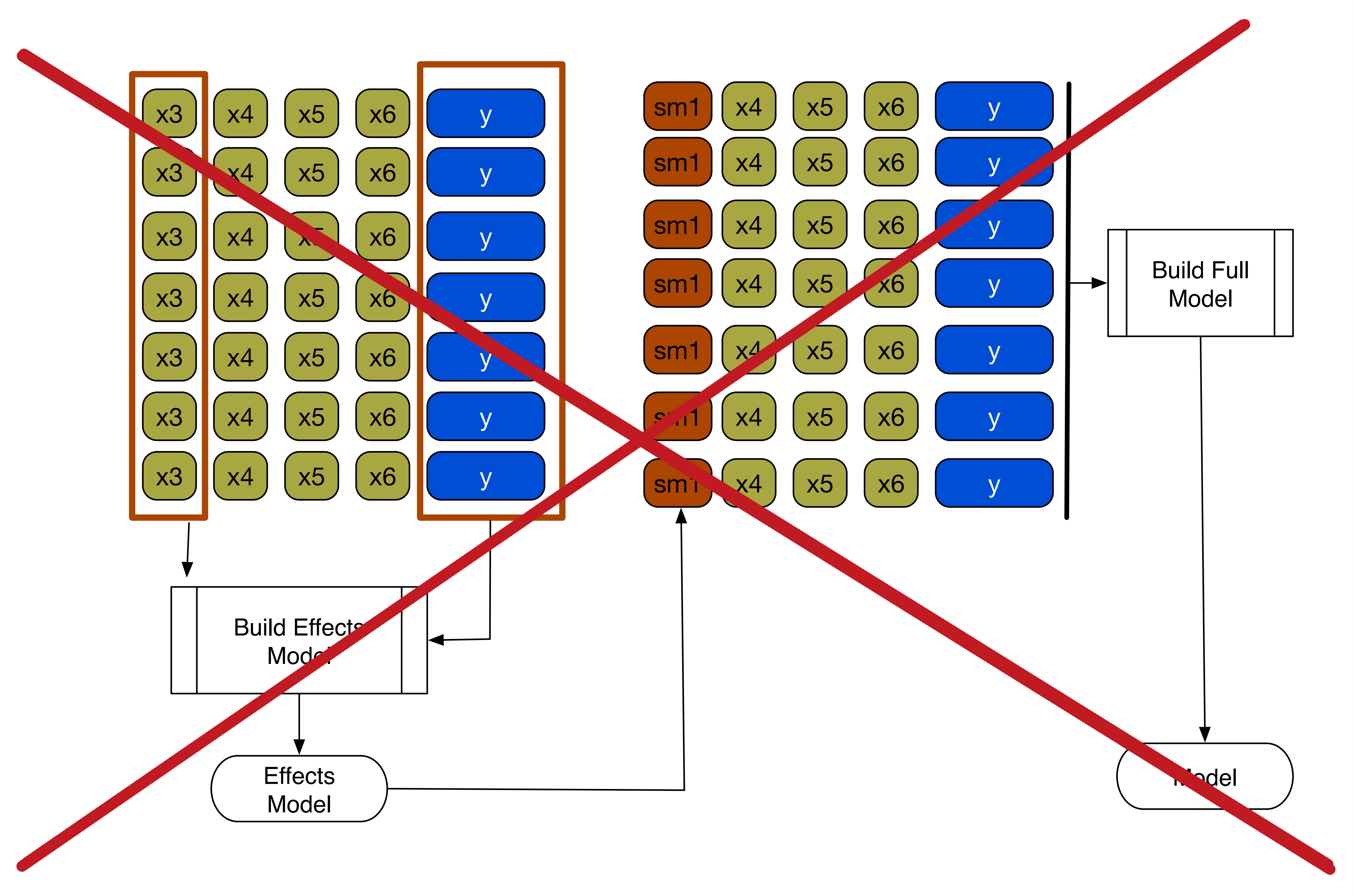
Make_Model	N_SoldIn Week	N_NotSold InWeek	LogDiff	IsRare
VW_Golf	60	40	0.41	No
Mazda_Miata	68	132	-0.66	No
Chevy_Camaro	8	42	-1.6	No
Toyota_Prius	108	42	0.94	No
Lotus_Elise	**1**	**0**	**1E+06**	**Yes**
…	…	…		

Model by Counts

Win-Vector LLC

Can’t use Training Data to Effects Code!

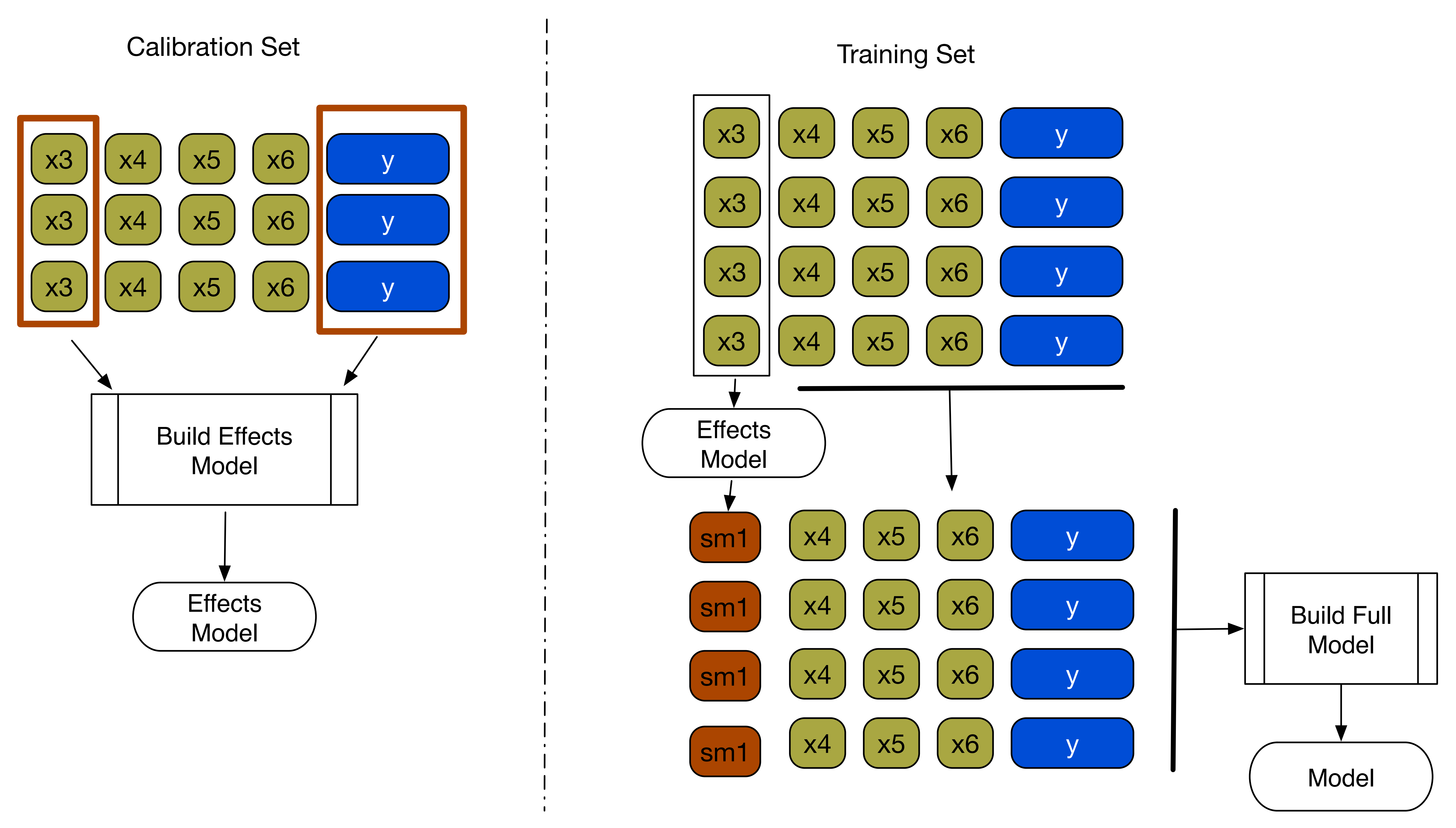
- Effects model can memorize the training data
  - “Lotus Elise always sells in a week”
- Full model may overestimate the value of effects-coded variable
  - Overfit



Win-Vector LLC

Training the Effects Model

Best Solution:
A separate calibration set for effects model



Win-Vector LLC

Alternative Solution: Prune Rare Levels

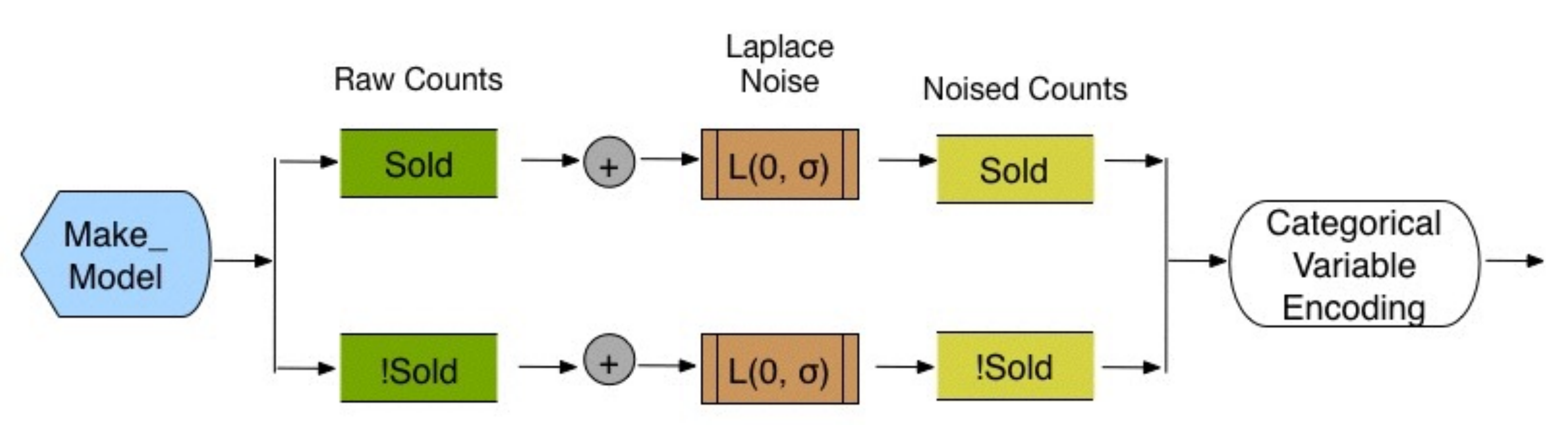
Make_Model	P(SoldIn Week)	Impact	Nobsv
VW_Golf	0.6	0.2	100
Mazda_Miata	0.34	-0.06	200
Chevy_Camaro	0.16	-0.24	50
Toyota_Prius	0.72	0.32	150
~~**Lotus_Elise**~~	~~**1.0**~~	~~**0.6**~~	~~**1**~~
~~**Yugo_GV**~~	~~**0.33**~~	~~**-0.07**~~	~~**3**~~
…	…	…	…
Overall	0.4	0	N

Make_Model	Impact
VW_Golf	0.2
Mazda_Miata	-0.06
Chevy_Camaro	-0.24
Toyota_Prius	0.32
Lotus_Elise	**0**
Yugo_GV	**0**
…	…

Better: use significance of conditional estimate

Win-Vector LLC

Innovative Solution: Differential Privacy



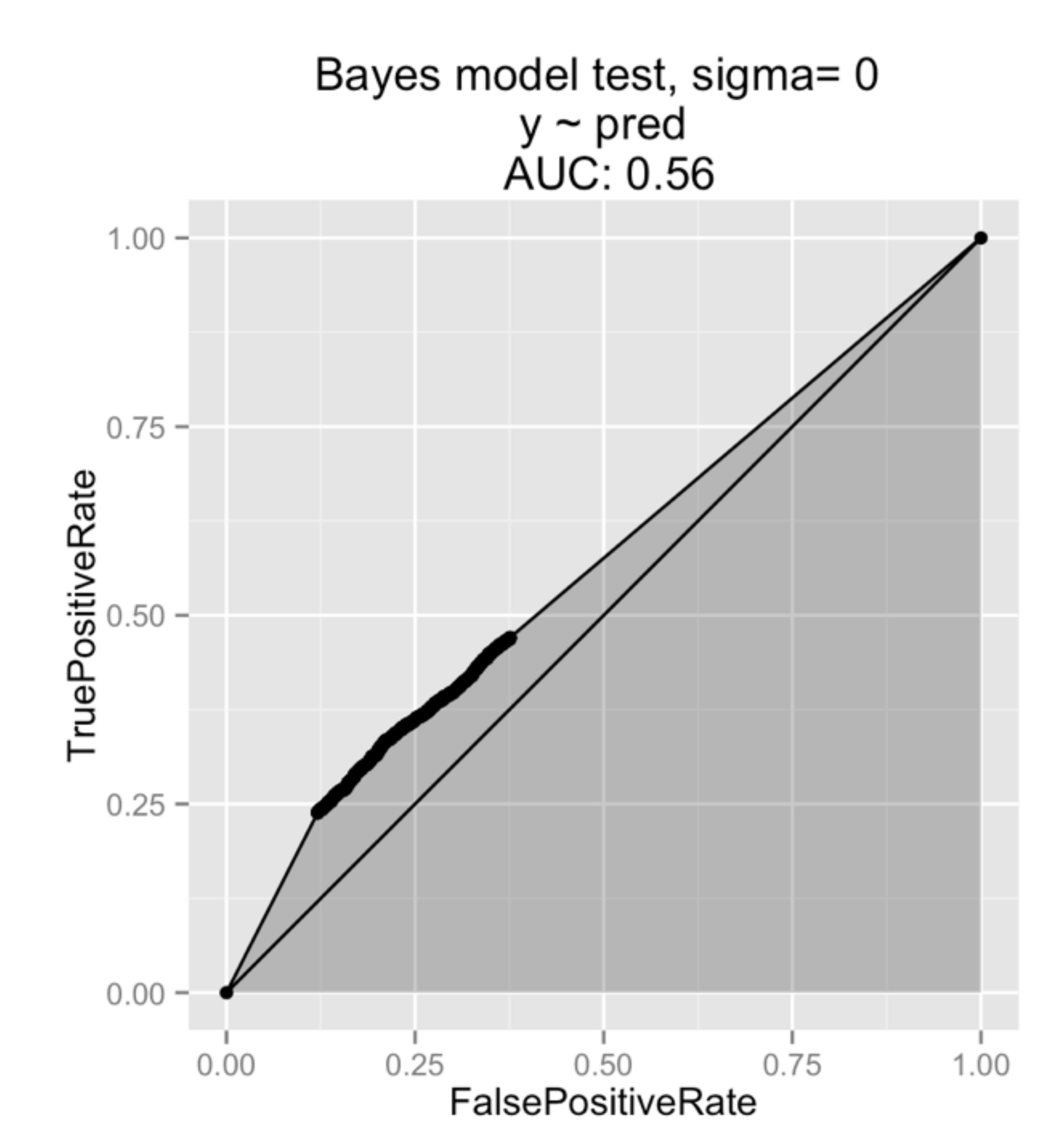
Add noise to training data before passing to effects coding

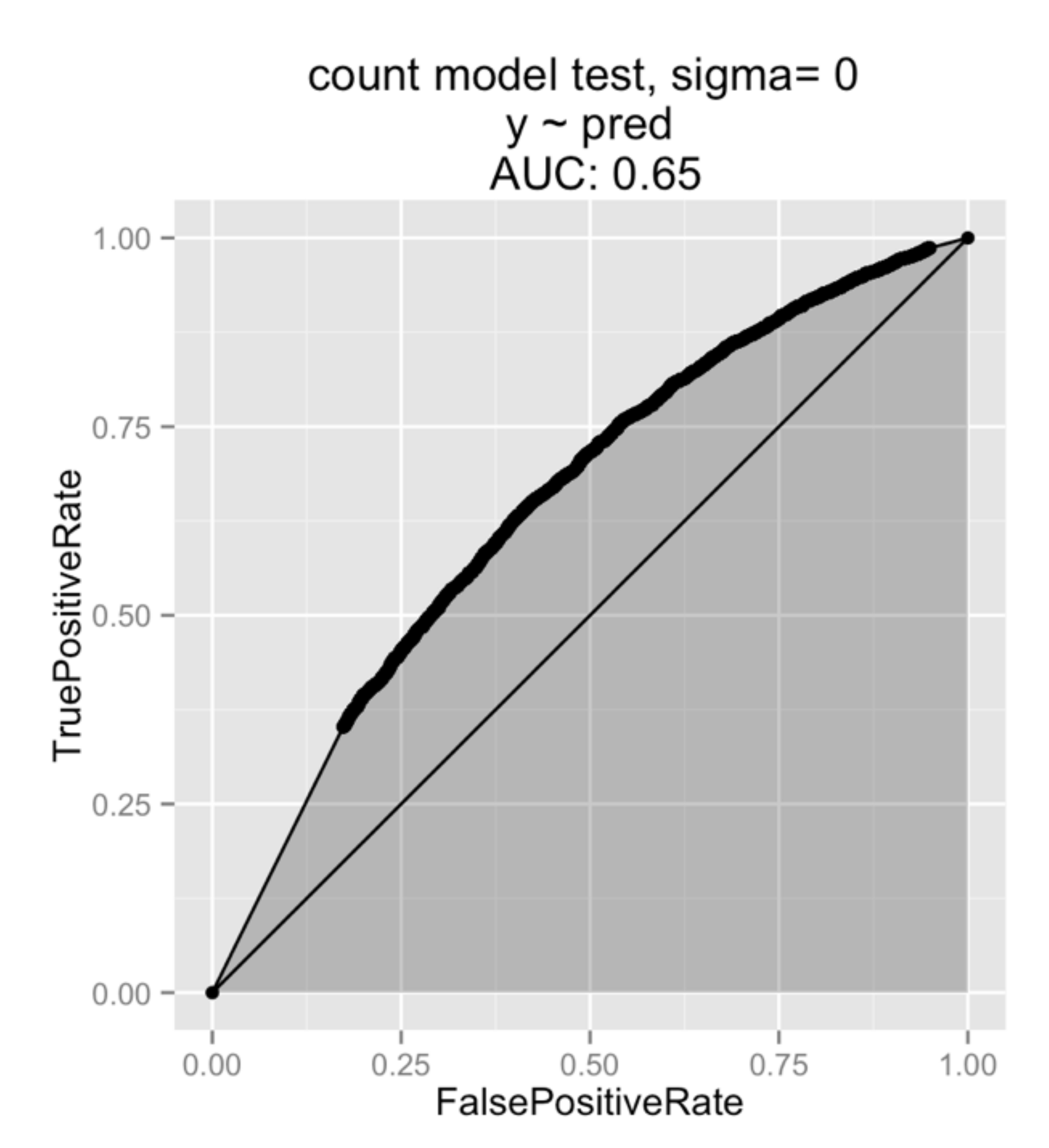
Example

- Synthetic data, 2000 rows training
- 40 categorical variables
  - 10 signal variables with 10 levels each
  - 30 noise variables with 500 levels each
- Classification: Positive class 50% prevalence
- Effects code the variables, then fit a logistic regression model

Naive Modeling

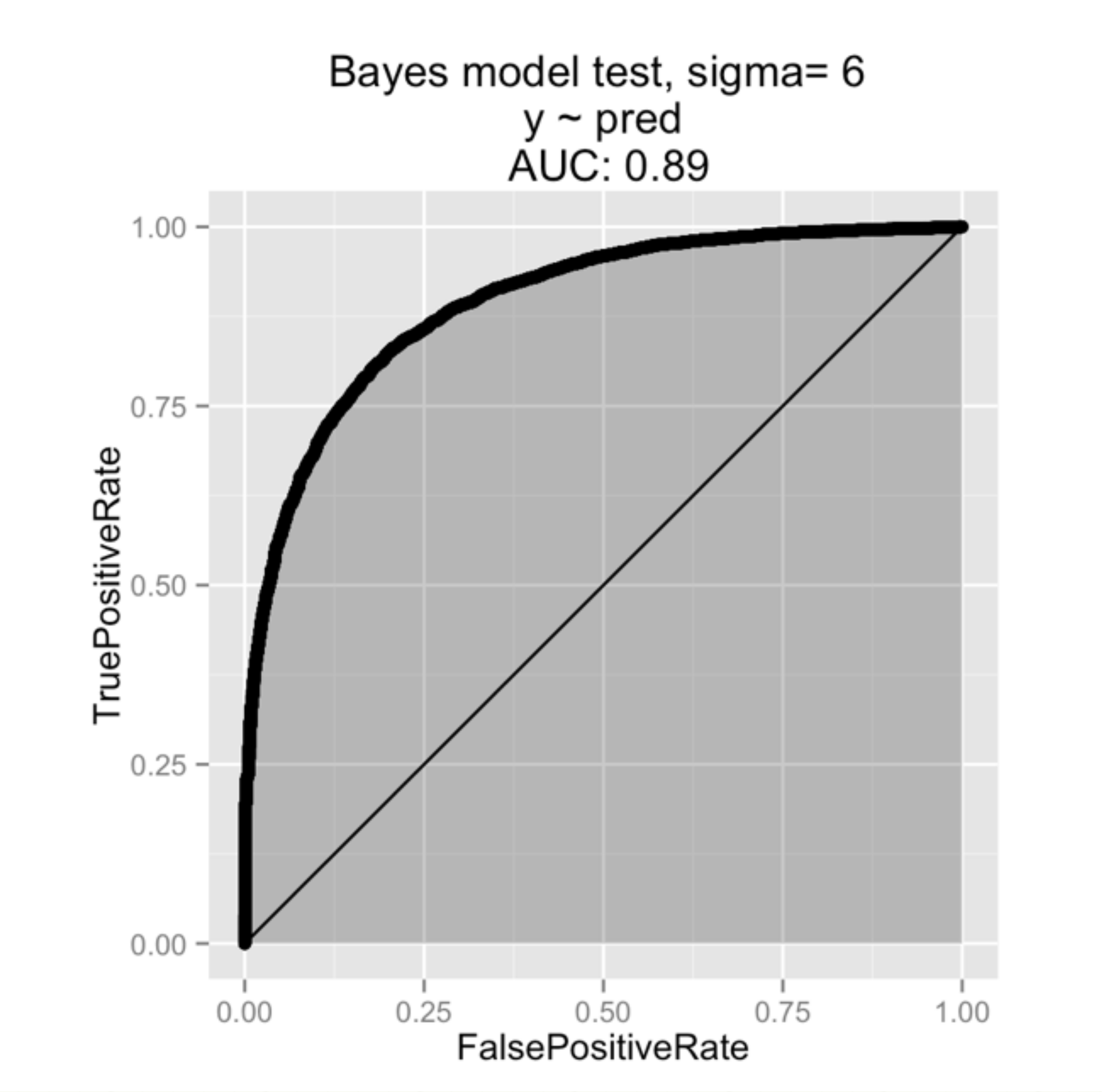
In Training: both models perfect (AUC = 1)



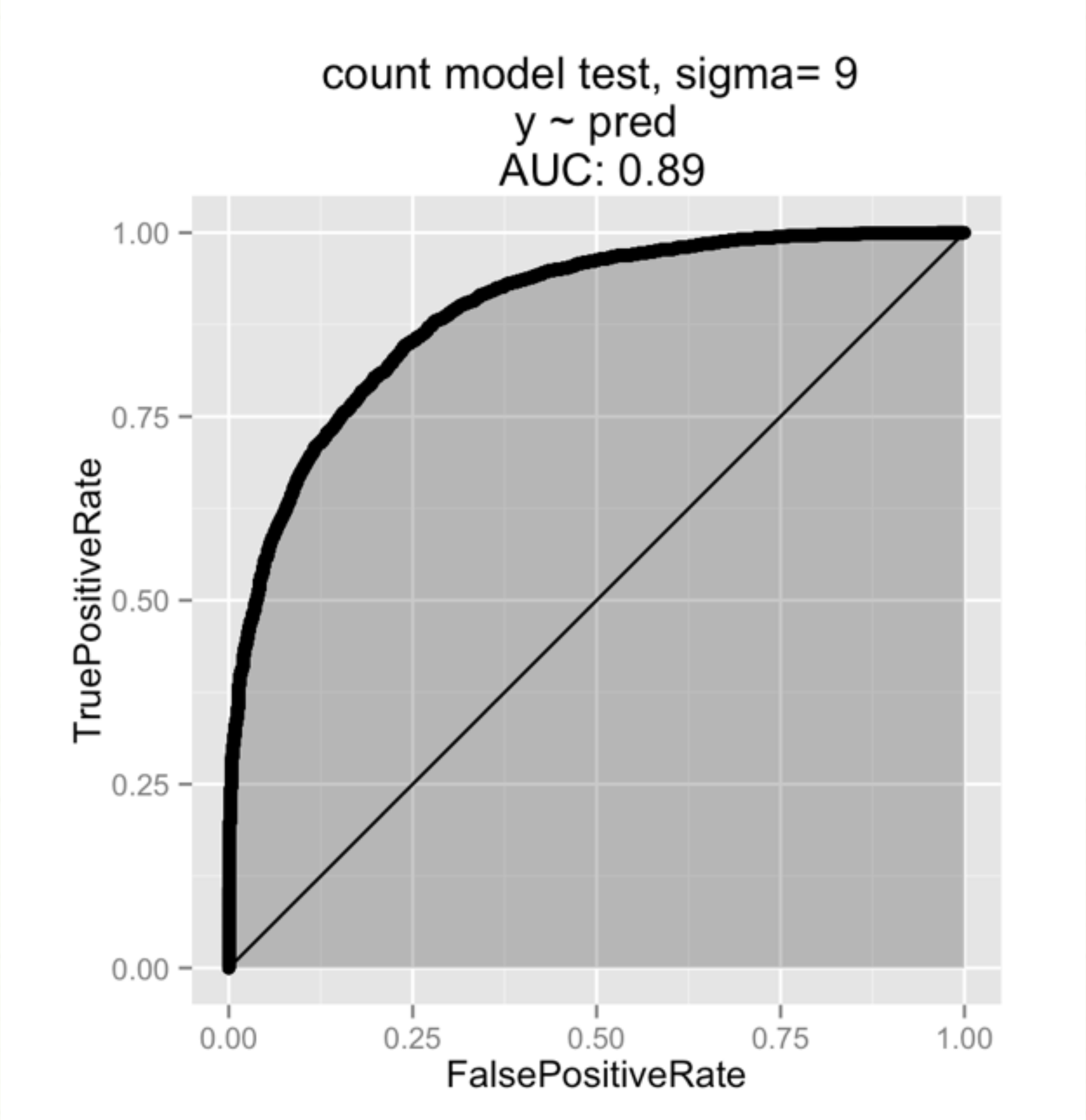


Win-Vector LLC

With Laplace Noise



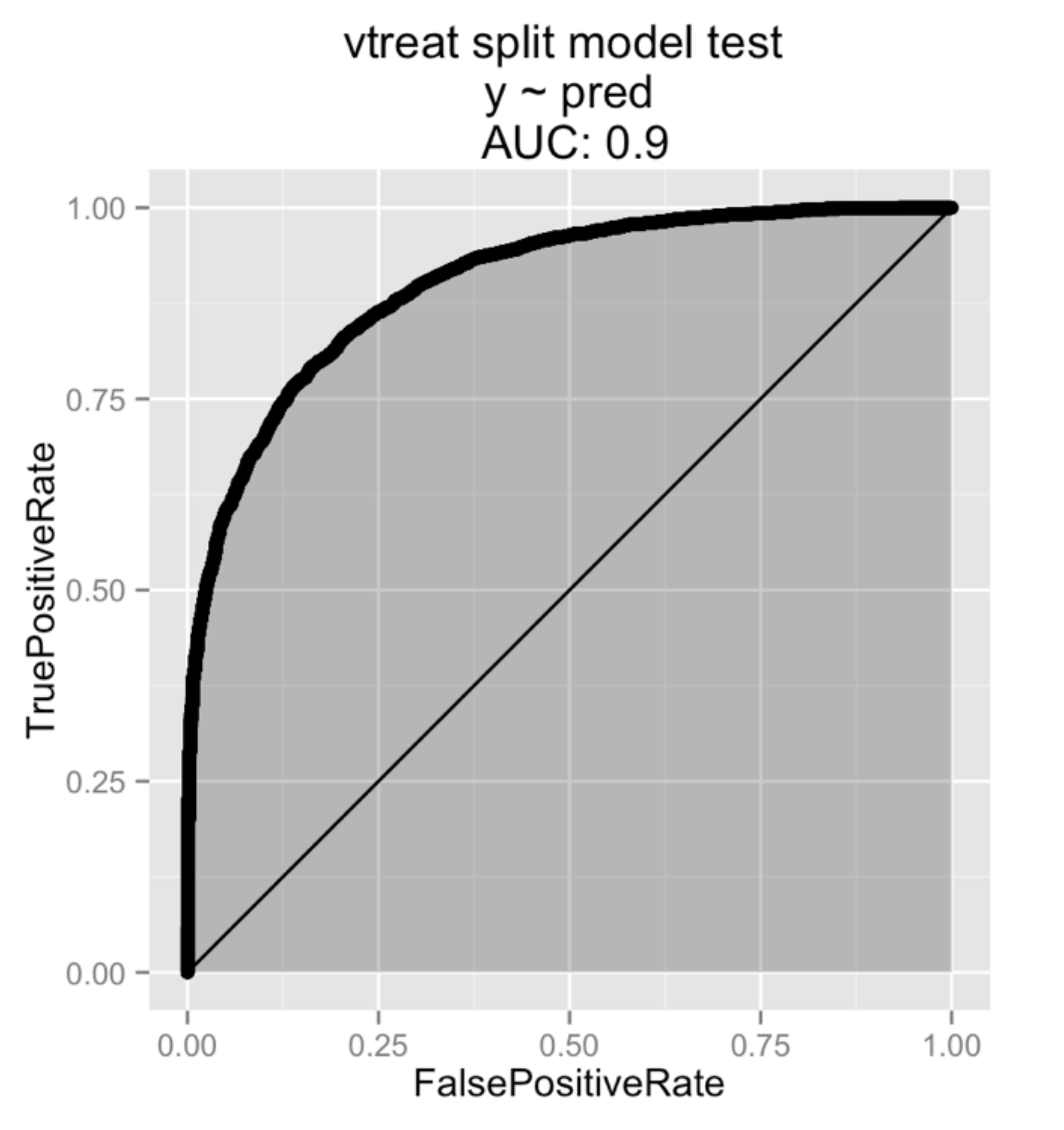
In Training: AUC = 0.95



In Training: AUC = 0.96

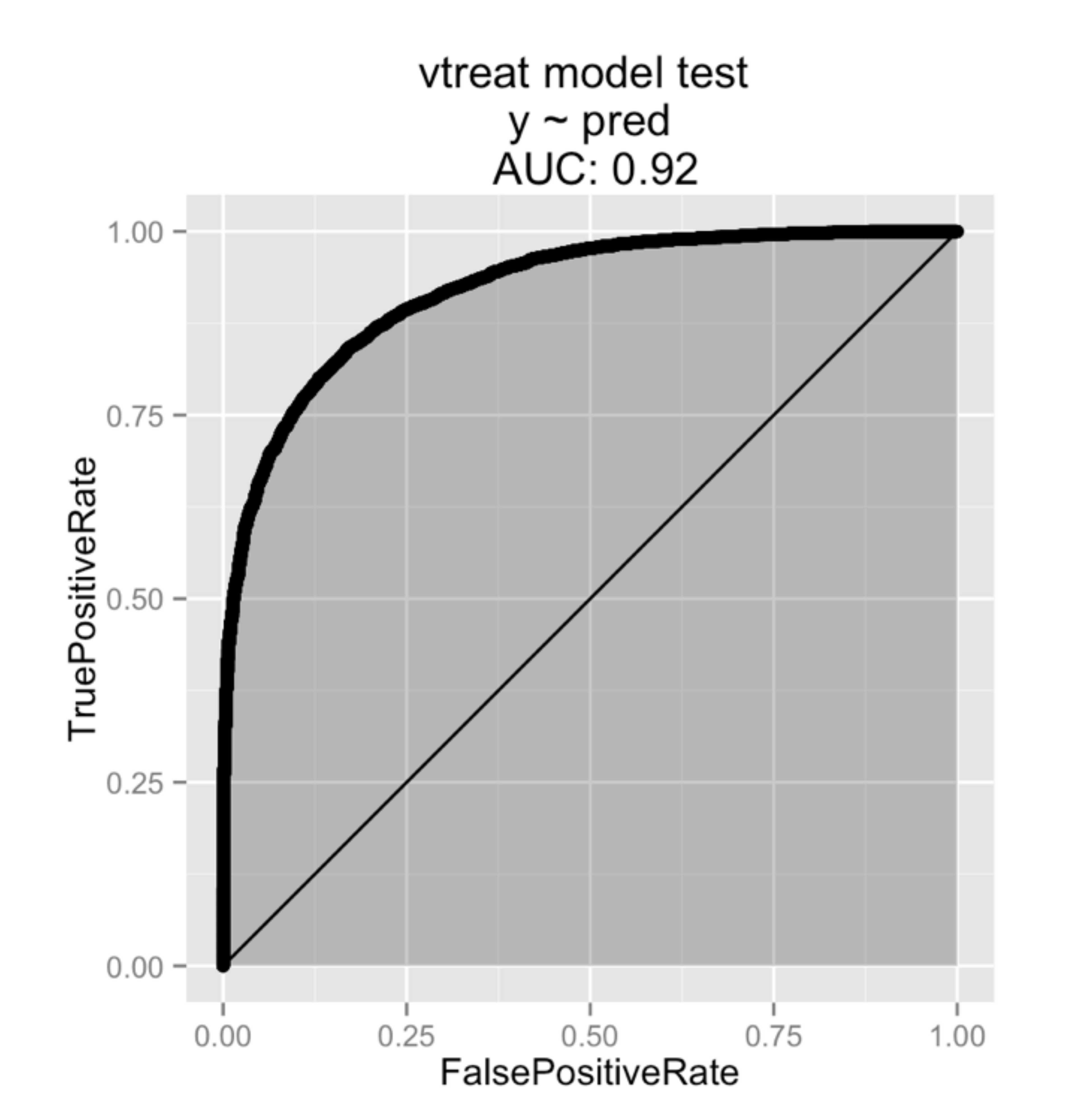
Win-Vector LLC

With Calibration Set



Bayesian model:
In Training: AUC = 0.91

All training data and rare level pruning



Bayesian model:
In Training: AUC = 0.95

Win-Vector LLC

Takeaways

- Differential privacy alleviates the overfit from effects coding (or nested models in general) by masking rare phenomena.
- DP is a useful alternative when there’s not enough data for a calibration set.
  - Or for online situations (with learning by counts)
  - For batch, rare level pruning also works well

References

- Dwork, Cynthia, Vitaly Feldman, Moritz Hardt, Toniann Pitassi, Omer Reingold, Aaron Roth. “Preserving Statistical Validity in Adaptive Data Analysis”, April 2015.
    - http://arxiv.org/abs/1411.2664

- Dwork, Cynthia, *et.al*. “The reusable holdout: Preserving validity in adaptive data analysis”, *Science*, vol 349, no 6248 pp 636-638, August 2015.
    - Abstract: https://www.sciencemag.org/content/349/6248/636.abstract

- Cohen, Jacob and Patricia Cohen. *Applied Multiple Regression/Correlation Analysis for the Behavioral Sciences*, 2nd edition, 1983

- Bilenko, Misha. “Big Learning made Easy — with Counts!” *Machine Learning Blog* http://blogs.technet.com/b/machinelearning/archive/2015/02/17/big-learning-made-easy-with-counts.aspx

References

- Blog posts (Differential privacy mini-series):
  - http://www.win-vector.com/blog/2015/11/our-differential-privacy-mini-series/

- Our code, data and examples:
  - https://github.com/WinVector/Examples/tree/master/DiffPriv/PrivStep
  - https://github.com/WinVector/PreparingDataWorkshop/tree/master/NestedModels

Thank You