General Information Metrics for Automated Experiment Planning

Christopher Lee

October 4, 2011

Overview

- Basic theory of empirical information metrics.
- Experiment planning using empirical information metrics.
- Application: “phenotype sequencing”, a new genomics experiment we designed computationally to identify the genetic causes of a phenotype directly from high-throughput sequencing.

Theory vs. Practice

- Information theory assumes that we know the complete joint distribution of all variables $p(X, Y)$.

- In other words, given *complete knowledge* of the relevant system variables and their interactions in all circumstances, this math can compute information metrics.

- By contrast, in science we have the opposite problem: we start with no knowledge of the system, and must infer it from observation. Information metrics would be useful only if they helped us gradually infer this knowledge, one experiment at a time.

Statistical Inference

- Divides the world into *observable* vs. *hidden* variables, and provides a mathematical framework for making inferences about the latter from the former. There are two schools:

- *Maximum likelihood*: find the model θ that emits the observations X with highest likelihood $p(X|\theta)$.

- *Bayesian inference*: considers both likelihoods and *prior probability* of each model, based on Bayes' Law

$$p(\theta|X) = \frac{p(X|\theta)p(\theta)}{p(X)} = \frac{p(X|\theta)p(\theta)}{\sum_\theta p(X|\theta)p(\theta)}$$

Goal: Marry Information Theory and Statistical Inference

- Develop “empirical information metrics” that can be computed from data samples, and which can guide both statistical inference (modeling past observations) and experiment planning (new observations).

- Resolve the seemingly incompatible assumptions of information theory and statistical inference. Bring insights of information theory to statistical inference, and robustness and utility of inference to information theory.

What is Information?

- Should it be measured on *hidden* variables or *observable* variables?

- Statistical inference focuses on infering hidden variables, but I find this leads to fundamental flaws as an information metric (unbounded; easy to fool etc.).

- I define empirical information as *prediction power measured on observable variables*. Bounded by the inherent variation in the observable; no way to make the metric go up except by predicting that variation more accurately.

Empirical Information

- We want to estimate the prediction power of a model Ψ based on a sample of observations $\vec{X}^n = (X_1, X_2, ..., X_n)$ drawn independently from a hidden distribution Ω. We define the *empirical log-likelihood*

$$\overline{L_e}(\Psi) = \frac{1}{n} \sum_{i=1}^{n} \log \Psi(X_i) \rightarrow E(\log \Psi(X)) \text{ in probability}$$

  which by the Law of Large Numbers is guaranteed to converge to the true expectation prediction power as the sample size $n \rightarrow \infty$.

- We can also define an absolute measure of information from this:

$$\overline{I_e}(\Psi) = \overline{L_e}(\Psi) - \overline{L_e}(p)$$

  where $p(X)$ is the uninformative distribution of X. (Lee, *Information*, 2010)

How do you know when you're done?

- Version 1: The set of all possible models of the universe is infinite, but we only calculate a tiny subset of them. How much of the total *possible* prediction power does this subset capture?

- Version 2: the denominator of Bayes' Law requires summing over this infinite set of models. Is our calculated subset a close approximation or totally wrong?

$$p(\theta|X) = \frac{p(X|\theta)p(\theta)}{\sum_\theta p(X|\theta)p(\theta)}$$

- Version 3: Popper: a scientific theory is only useful if it is *falsifiable* – i.e. show that our best model is *not good enough*. Bayes' Law gives no way to do this.

- Is the absolute value of the likelihood good enough? How good *should* it be?

Potential Information

- Define the total information in the infinite series of all models as I_∞. The empirical information I_e represents the terms we've actually calculated. Define *potential information* I_p as the remainder:

$$I_p = I_\infty - I_e$$

- It turns out we can estimate I_p without actually summing any more terms of the infinite series.

$$I_p = E(L(\Omega) - L(p)) - E(L(\Psi) - L(p))$$

$$I_p = -E(L(\Psi)) + E(L(\Omega)) = -E(L(\Psi)) - H(\Omega(X))$$

We can again estimate this via sampling:

$$\overline{I_p} = -\overline{L_e}(\Psi) - \overline{H_e}$$

where we define $\overline{H_e}$ as the *empirical entropy* computed from the sample (again with a Law of Large Numbers convergence proof). (Lee, *Information*, 2010)

Empirical Entropy Estimation

- A lot of kernel-based density estimation methods in effect apply a *model* (e.g. Gaussian) to the data. But the whole point of H_e is to provide a test that is independent of all models. We need a *model-free* density estimation method for calculating empirical entropy.

- Lots of methods possible, e.g. we've used *k-nearest neighbors*

$$\overline{H_e} = -\frac{1}{n}\sum_{j=1}^{n} \log \frac{k-1}{(n-1)(|X_{j:k} - X_j| + |X_{j:k-1} - X_j|)}$$

  where $X_{j:k}$ is the coordinate of point X_j's k-th nearest neighbor.

Potential Information Convergence

- The Law of Large Numbers guarantees convergence as $n \to \infty$, $\overline{I_p}(\Psi) \to D(\Omega||\Psi)$, the *relative entropy*, a standard information theory measure. Specifically, it guarantees a probabilistic lower bound on D with confidence ϵ:

$$p\left(D(\Omega||\Psi) \geq \overline{I_p}(\Psi) - \sqrt{\frac{Var(\log P_e - L_e)}{n\epsilon}}\right) \geq 1 - \epsilon$$

- This is the ultimate hypothesis test, because $D(\Omega||\Psi) \to 0$ iff $\Psi(X) = \Omega(X)$ everywhere.

- LLN is basic and universal, but insensitive, i.e. we can get a better lower-bound on I_p, e.g. via re-sampling.

Experiment Planning

- Empirical information is improved prediction power. If an experiment does not lead to a change in our predictions (i.e. our model Ψ), clearly there is no improvement in prediction power = *no information value*.

- An experimental observation's total capacity to improve our predictions is simply given by its *potential information* vs. our current model.

- Before we do an experiment, we are uncertain about its outcome. But we may be able to list possible scenarios α, and our model may give some probability estimates for these alternatives. On this basis we can directly calculate what the I_p yield for each scenario α would be.

Expectation Potential Information

- The expected information value of an experiment is just the expectation value of these potential information yields:

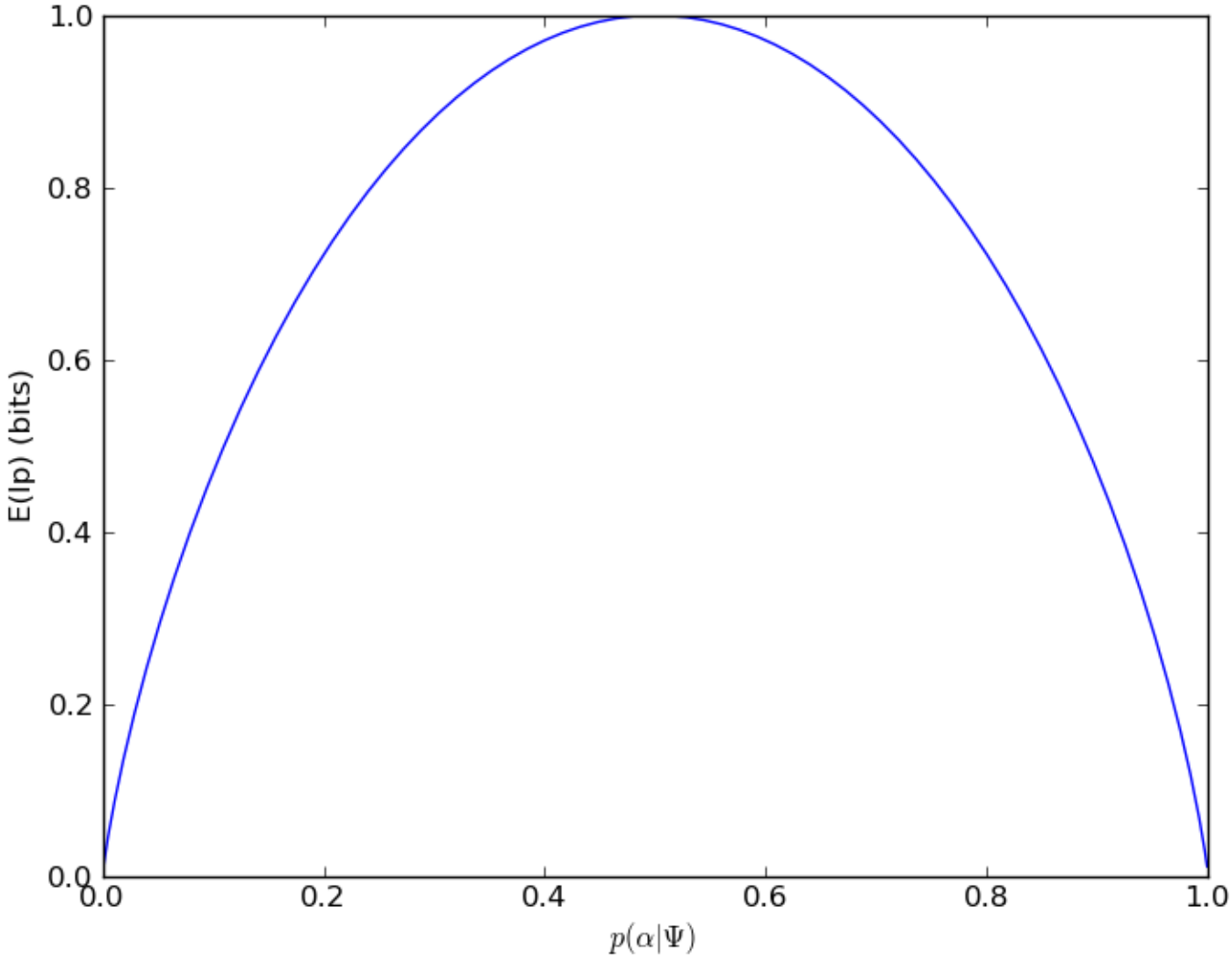
$$E(I_p) = \sum_\alpha p(\alpha|\Psi) D(\alpha||\Psi) = \sum_\alpha p(\alpha|\Psi) D\left(\alpha||\sum_\alpha \alpha p(\alpha|\Psi)\right)$$

- *Disambiguation*: As the estimated outcome probabilities become accurate,

$$E(I_p) \to I(X;\alpha) = H(\alpha) - H(\alpha|X)$$

  i.e. the mutual information measuring how informative the experimental observation X is about the hidden state α. For a "perfect" detector, $H(\alpha|X) = 0$, so $E(I_p) \to H(\alpha)$, our initial uncertainty about the hidden state. Others have proposed using mutual info for experiment planning (Paninski, *Neural Computat.* 2005).

Information Value of Disambiguation



Simple Example: What is the Value of a Control?

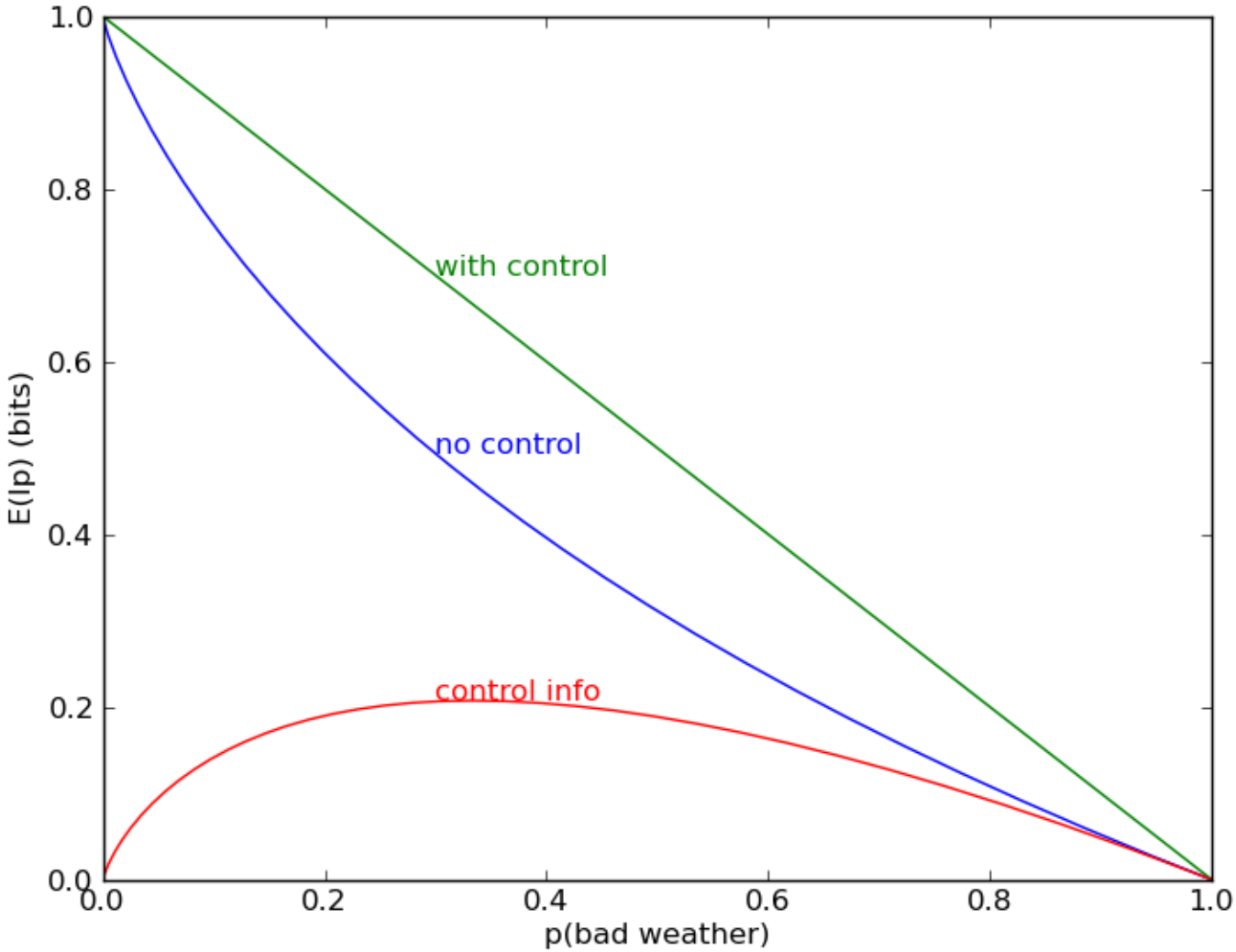
- Experiment: cross two plants $A \times B$, observe whether progeny grow. Assume 50-50 uncertainty = 1 bit of information.

- If *bad weather* occurs, nothing can grow. The experiment becomes uninformative.

- If bad weather occurs with some probability p, we won't know how to interpret a *no-progeny* observation (could be real; could just be bad weather).

- We can include a control cross that we know should grow e.g. $A \times A$.

Computing the Information Value of a Control



Analyzing an Experiment's Information Rate vs. Total Capacity

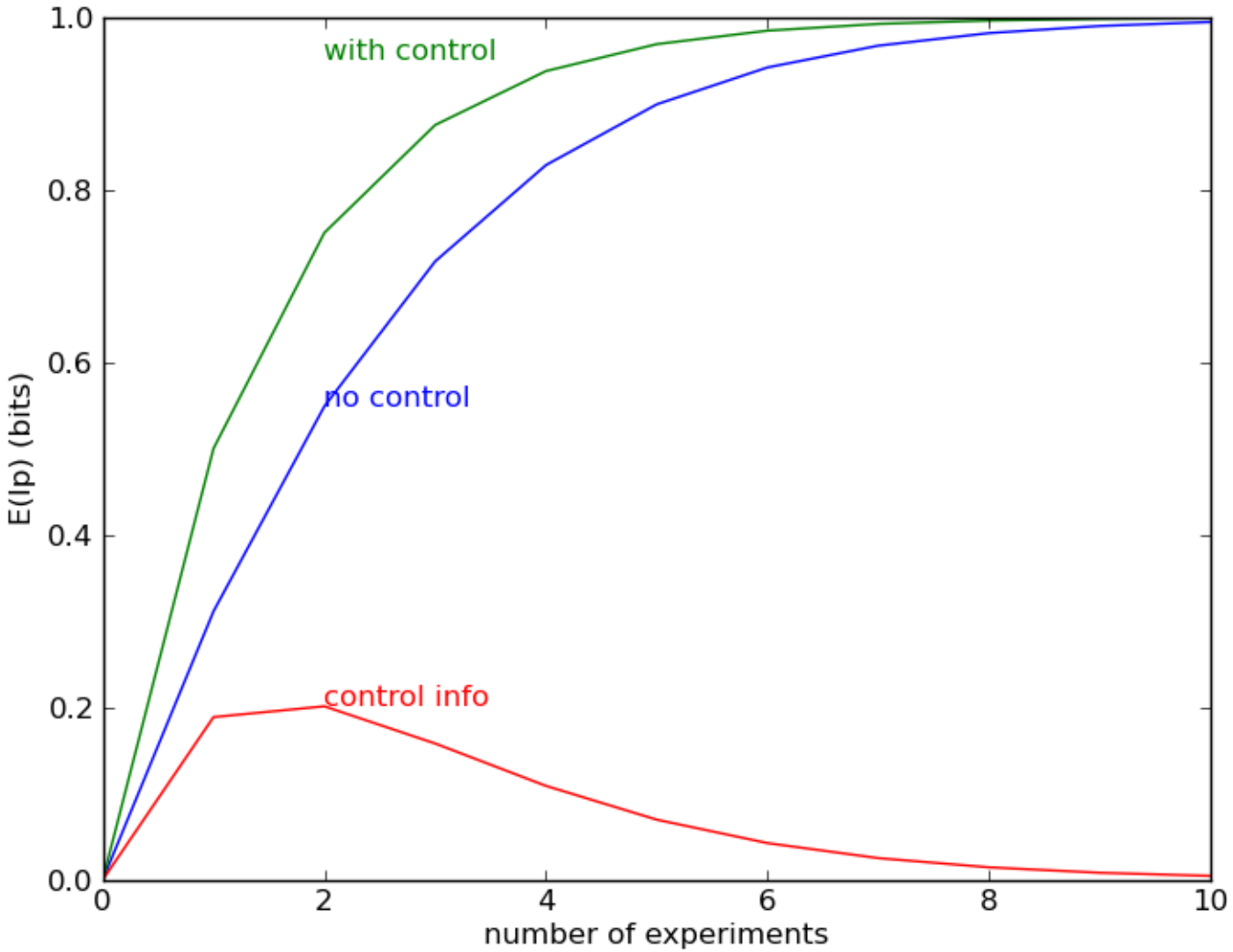
- Factors that vary independently over different repetitions of the experiment affect *the rate* of information production but not the *total* information capacity.

- These rate calculations tell us the efficiency of an experiment design, i.e. its *cost per total information yield*.

- Example: If each repetition of our experiment has a known probability of bad weather (e.g. 50%), we can get a confident result even without a control. E.g. if we get no progeny in 10 experiments, the chance of this being due to bad weather is less than 0.1%.

- Of course, the control still improves the rate of information production – which lowers the cost.

Effect of Control on Information Rate



Factors that Degrade Total Information Yield

- Factors that remain fixed over different repetitions of an experiment (e.g. the experiment design) affect the *total yield* that the experiment can produce (no matter how many times we repeat it).

- "detector failure": in a lot of fields (e.g. molecular biology), there are many factors that can cause an experiment to fail (give a negative result) even if the hypothesis is correct.

- For $E(I_p)$, the high probability of the negative outcome means it produces very little information. A positive outcome could produce a lot of information, but its low probability makes its $E(I_p)$ contribution small.

The Information Evolution Cycle

- When $I_p > 0$, we must extend the model, to "convert" this potential information to empirical information.

- When $I_p \to 0$ for a given set of obs, the model is "good enough", i.e. observationally indistinguishable. More modeling cannot improve it.

- In this case, the only way to get more information, is to seek *new observations* that can resolve uncertainties in the current "model mix" (PL).

- We choose the experiment that maximizes the information yield per cost. (Lather, rinse, repeat).

Phenotype Sequencing: identifying the genetic causes of a phenotype directly from sequencing of independent mutants

Chris Lee
UCLA-DOE Institute for Genomics & Proteomics

Phenotypes vs. Causes

- If a strain with an interesting phenotype contains many mutations, it can be laborious to identify which one is the dominant cause, and which mutations are irrelevant.
- Easier for naturally evolved strains (10-20 mutations), much harder for mutagenized strains (50 - 100 mutations / genome).
- *mutagenesis* + *screen* → *multiple independent mutants* can dissect this powerfully.

Microbial Mutant Sequencing Studies

Study	Strains	SNPs
Srivatsan, *PLoS Genet* 2009	3	115
Le Crom, *PNAS* 2009	2	223
Conrad, *Genome Biol* 2010	11	35
Klockgether, *J Bacteriol* 2010	2	39
Serizawa, *Ant A Chemo* 2010	2	
Lee, *App Env Micro* 2010	1	6
Chen, *PLoS ONE* 2010	8	93

JGI Sequencing of High Production Mutants

	Mutant				
Gene	**lib1-1**	**lib3-15**	**A-37**	**C-5**	**NV3**
thiC		X			X
fdoG		X			X
hemN	X				X
yidE		X			X
malT		X	X		
sdaA			X		X

Each mutant strain had 50-70 mutations throughout the genome, but 7 genes were independently hit in two separate mutant strains.

Basic Statistical Analysis

For uniform mutation probability, mutation counts per gene should follow the Poisson distribution.

For 5 strains each with 50 mutations, 4000 genes:
$p(i \geq 2 \text{ hits} \mid \lambda = 50/4000, t = 5) = 0.0019$

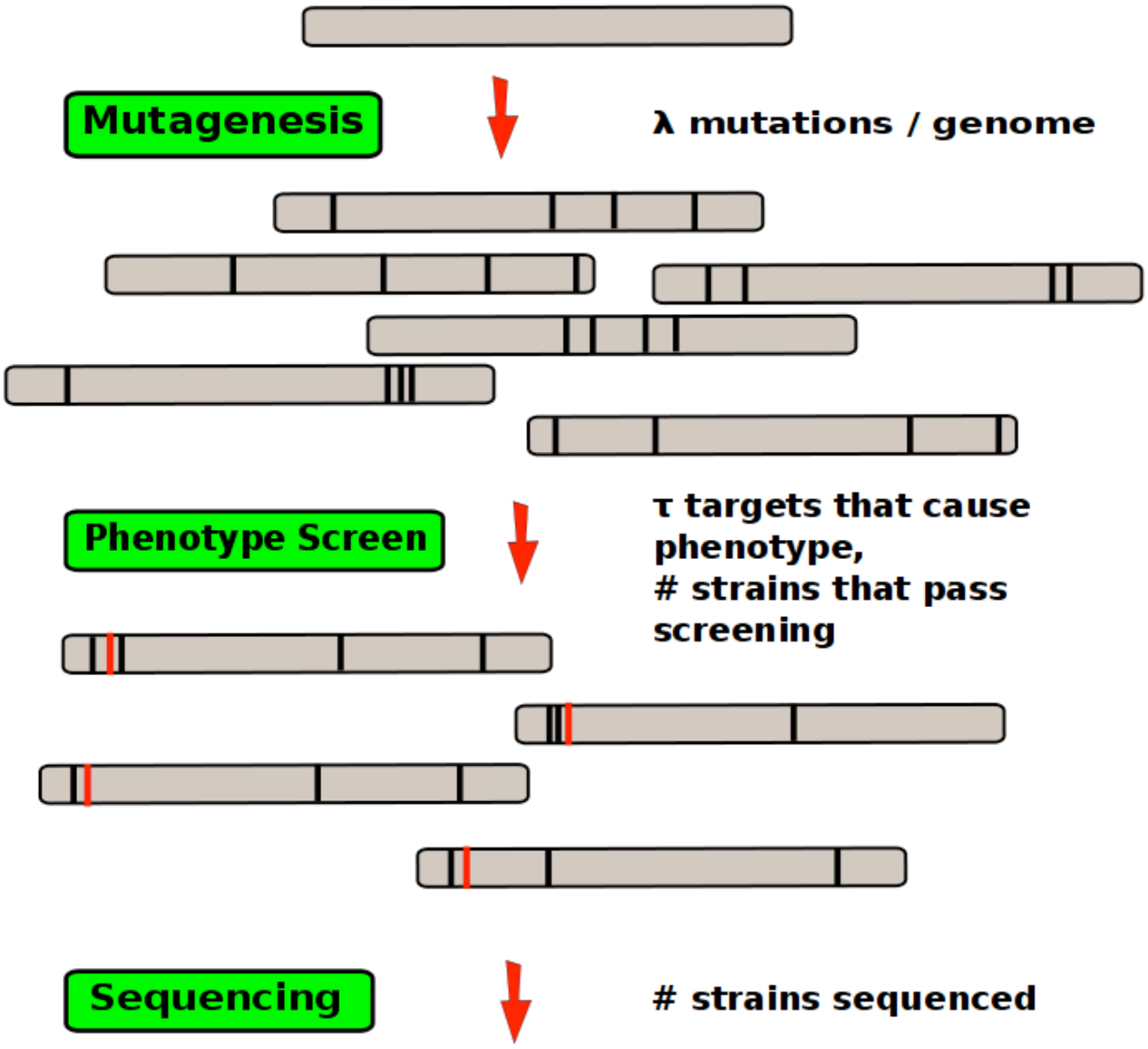
Expected #genes with ≥ 2 hits = 7.5

Not statistically significant.

(for non-uniform models, it's even worse).

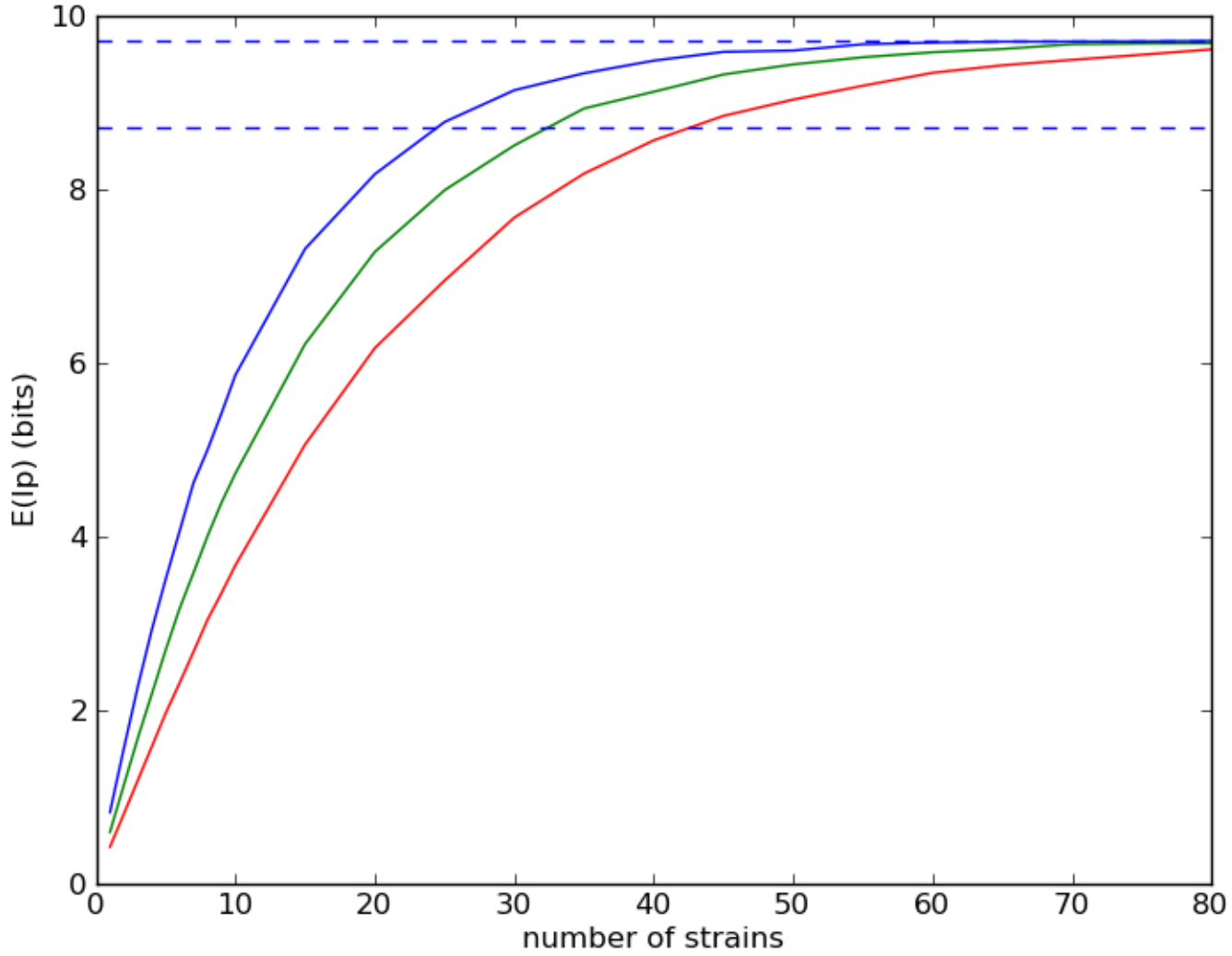
Proposal:
Phenotype Sequencing

Use the statistics of independent selection events to quickly reveal the genes that cause a phenotype, directly from sequencing of mutant strains with the same phenotype.

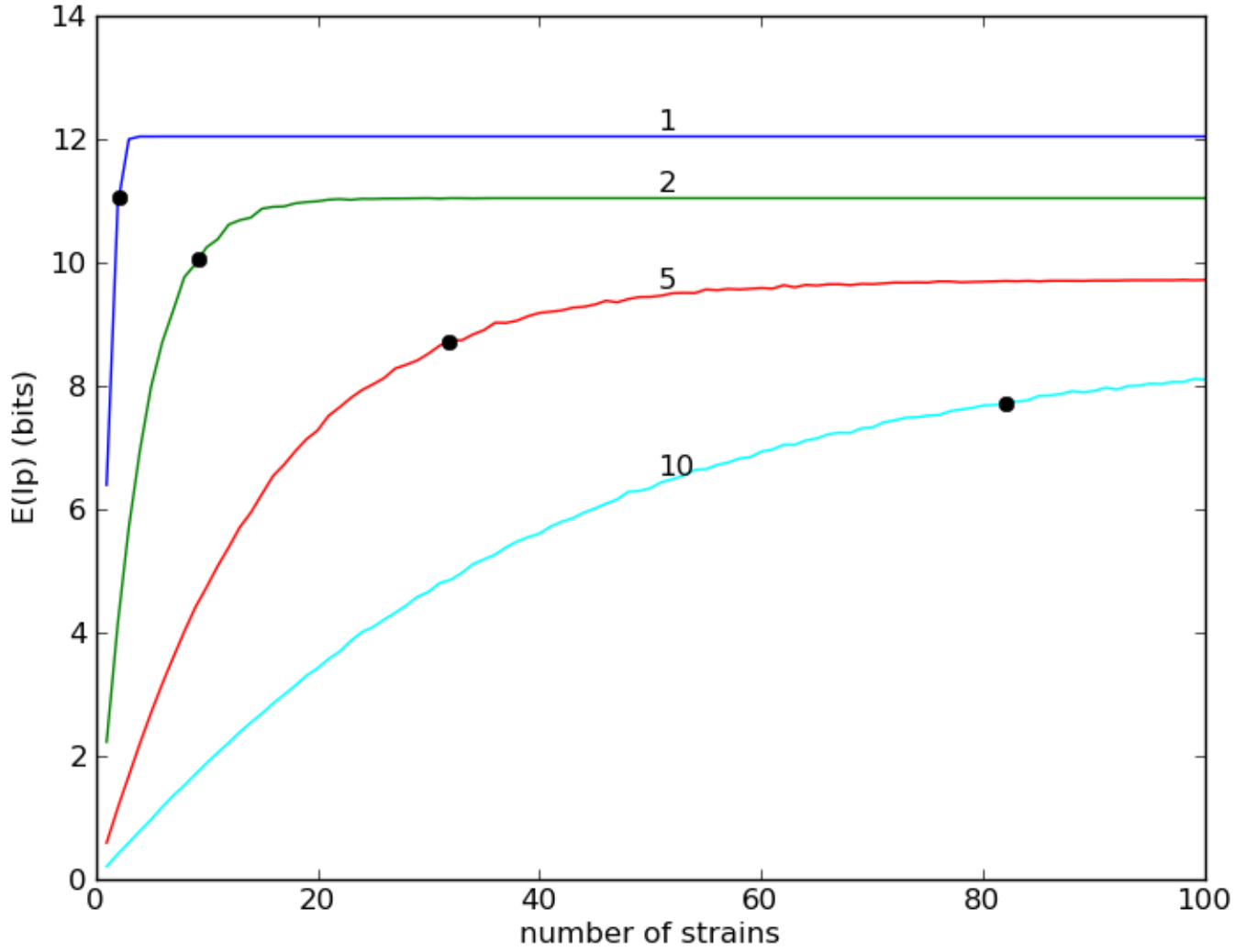


ATTACGAGGGATCCTATGACGC...
ATACCGAGGGATCCTATGACGC...
ATTACGAGCGATCCTATGACGC...
ATAACGAGGGATCCTATGACGC...

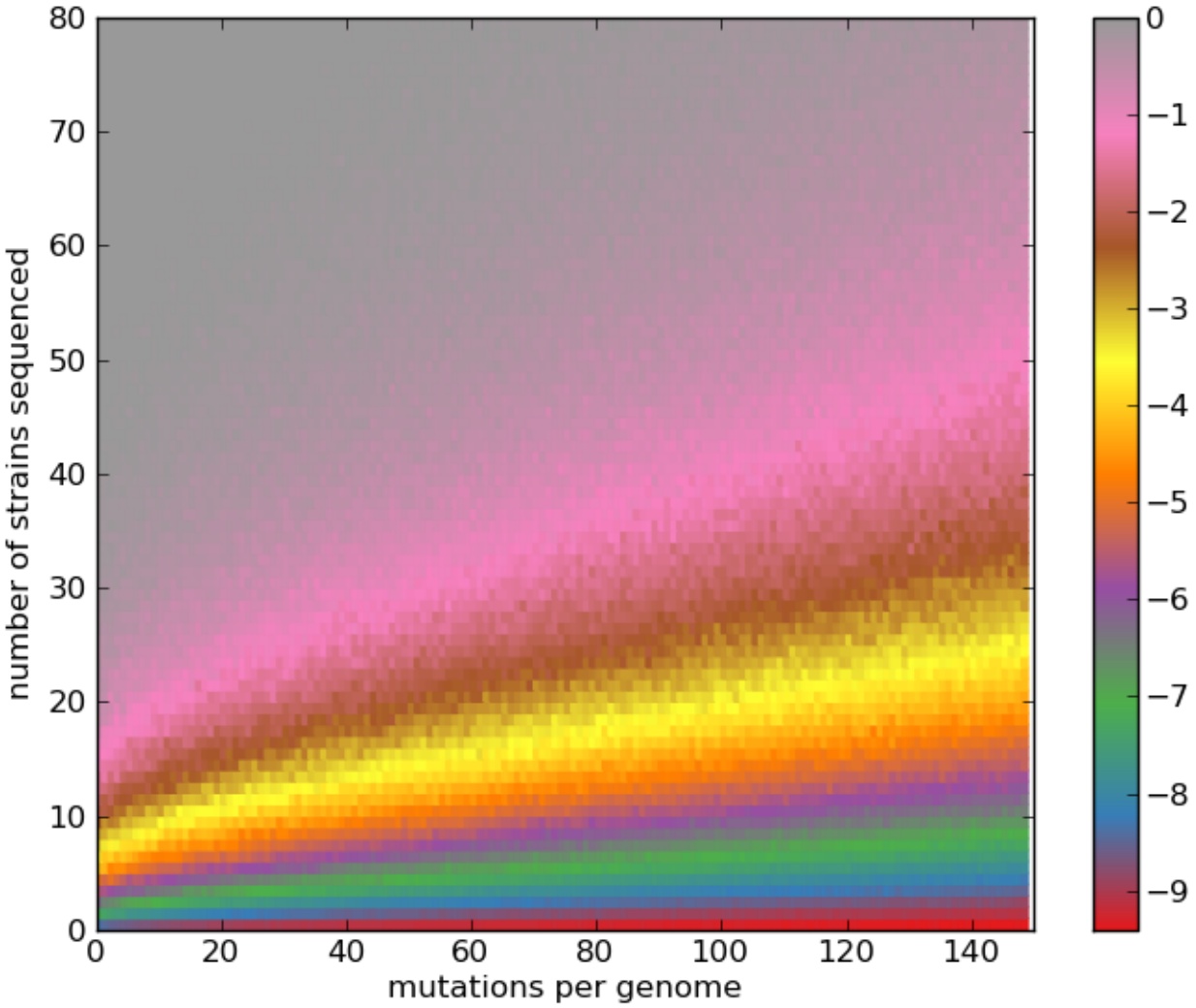
Effect of Mutagenesis Density



Effect of Number of Target Genes



Information Yield of Phenotype Sequencing



(Harper et al., *PLoS ONE* 2011)

“Phenotype Sequencing”

- This approach should work well for phenotypes where mutagenesis can produce many mutants.
- The smaller the number of targets, the easier they are to detect (signal spread over fewer genes).
- Non-uniform target size also makes it easier (concentrates signal into a subset of the targets).
- Lower mutagenesis density is better: requires more screening to find each mutant, but fewer total mutants for successful gene discovery.

Computational Tools for Phenotype Sequencing

- **phenoseq**: created open-source software package for designing and analyzing phenotype sequencing experiments, in Python.
- High-performance: simulates 5 million mutant genomes per second on a laptop.
- Developed statistics for analyzing next-gen sequencing data to score all genes as candidates for causing the phenotype.

  http://github.com/cjlee112/phenoseq

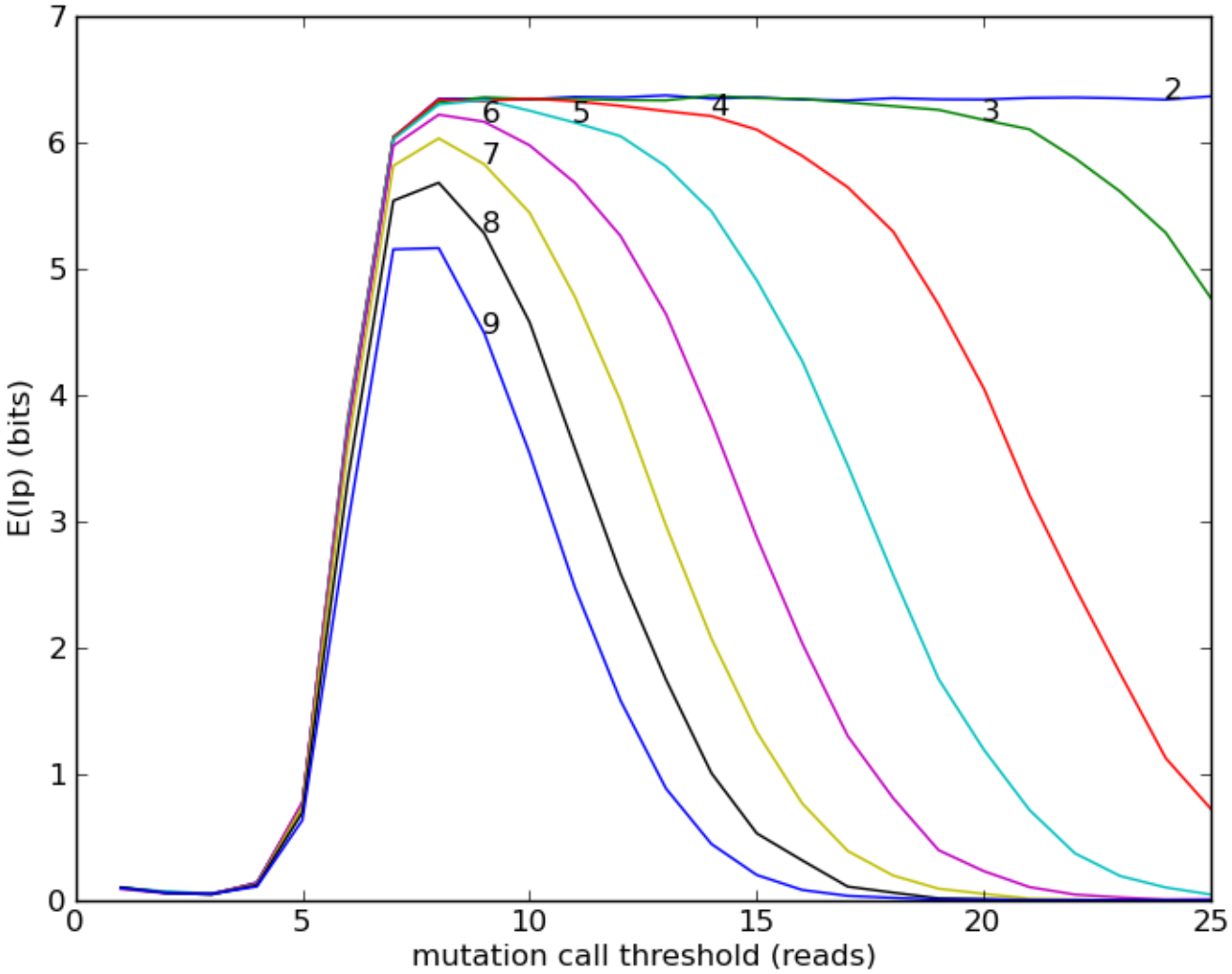
How to Make Phenotype Sequencing Economical

A library-pooling and tag-pooling strategy for greatly reduced experiment costs.

The Sequence is Not the Goal

- What we want is to identify the *genes that cause the phenotype*. The individual mutant sequences are just a means to that end.
- The key piece of data is the *number of times each gene is independently mutated*.
- We can design a sequencing experiment to measure this much more cheaply than individually sequencing each mutant.

Effect of Pooling



Pooling Is a Win-Win

- Increased coverage (reduced pooling) cannot increase the information yield beyond the limit set by the *total number of strains*.
- So moderate pooling loses *no* information.
- But it reduces costs by about five-fold.

Experimental Results

Deciphering the genetic causes of isobutanol biofuel tolerance in *E. coli* mutant strains generated with NTG mutagenesis, from James Liao's lab

Sequencing 32 isobutanol tolerant mutant strains

- Pooled in 10 libraries (3 strains/library)
- Sequenced on three replicate lanes
- 90 million single-end reads from Illumina GA2x
- 4099 SNPs: 3988 average per lane, of which 3702 replicated in all 3 lanes, 265 replicated in 2 lanes, 21 (0.5%) only in one lane. Each unique to one strain (excluded 23 parental mutations)
- 3596 mapped to 1808 genes; 2739 non-synonymous SNPs in 1426 genes.

Top 20 Genes by P-value

p-value	Genes	Description
9.5×10^{-20}	**acrB**	multidrug efflux system protein
1.4×10^{-5}	**marC**	inner membrane protein, UPF0056 family
1.8×10^{-4}	*stfP*	e14 prophage; predicted protein
0.0011	*ykgC*	predicted pyridine nucleotide-disulfide oxidoreductase
0.0035	*aes*	acetyl esterase; GO:0016052 - carbohydrate catabolic process
0.017	*ampH*	penicillin-binding protein yaiH
0.038	*paoC*	PaoABC aldehyde oxidoreductase, Moco-containing subunit
0.039	*nfrA*	bacteriophage N4 receptor, outer membrane subunit
0.044	*ydhB*	putative transcriptional regulator LYSR-type
0.12	*yaiP*	predicted glucosyltransferase
0.17	**acrA**	multidrug efflux system
0.25	*xanQ*	xanthine permease, putative transport; Not classified
0.25	*ykgD*	putative ARAC-type regulatory protein
0.35	*yegQ*	predicted peptidase
0.35	*yfjJ*	CP4-57 prophage; predicted protein
0.37	*yagX*	predicted aromatic compound dioxygenase
0.46	*pstA*	phosphate transporter subunit
0.48	*prpE*	propionate–CoA ligase
0.50	*mltF*	putative periplasmic binding transport protein, membrane-bound lytic transglycosylase F
0.63	*purE*	N5-carboxyaminoimidazole ribonucleotide mutase

Harper et al., *PLoS ONE*, 2011

Independent Validation

- Liao lab independently generated isobutanol tolerant strain SA481 via growth in increasing isobutanol over 45 sequential transfers.
- Sequencing SA481 identified 25 IS10 insertions
- Both repair studies and gene deletion studies showed that several genes contributed to isobutanol tolerance: *acrA*, *gatY*, *tnaA*, *yhbJ*, *marC* (*acrB* also inactivated).

Atsumi et al., *Mol Sys Biol*, Dec. 2010

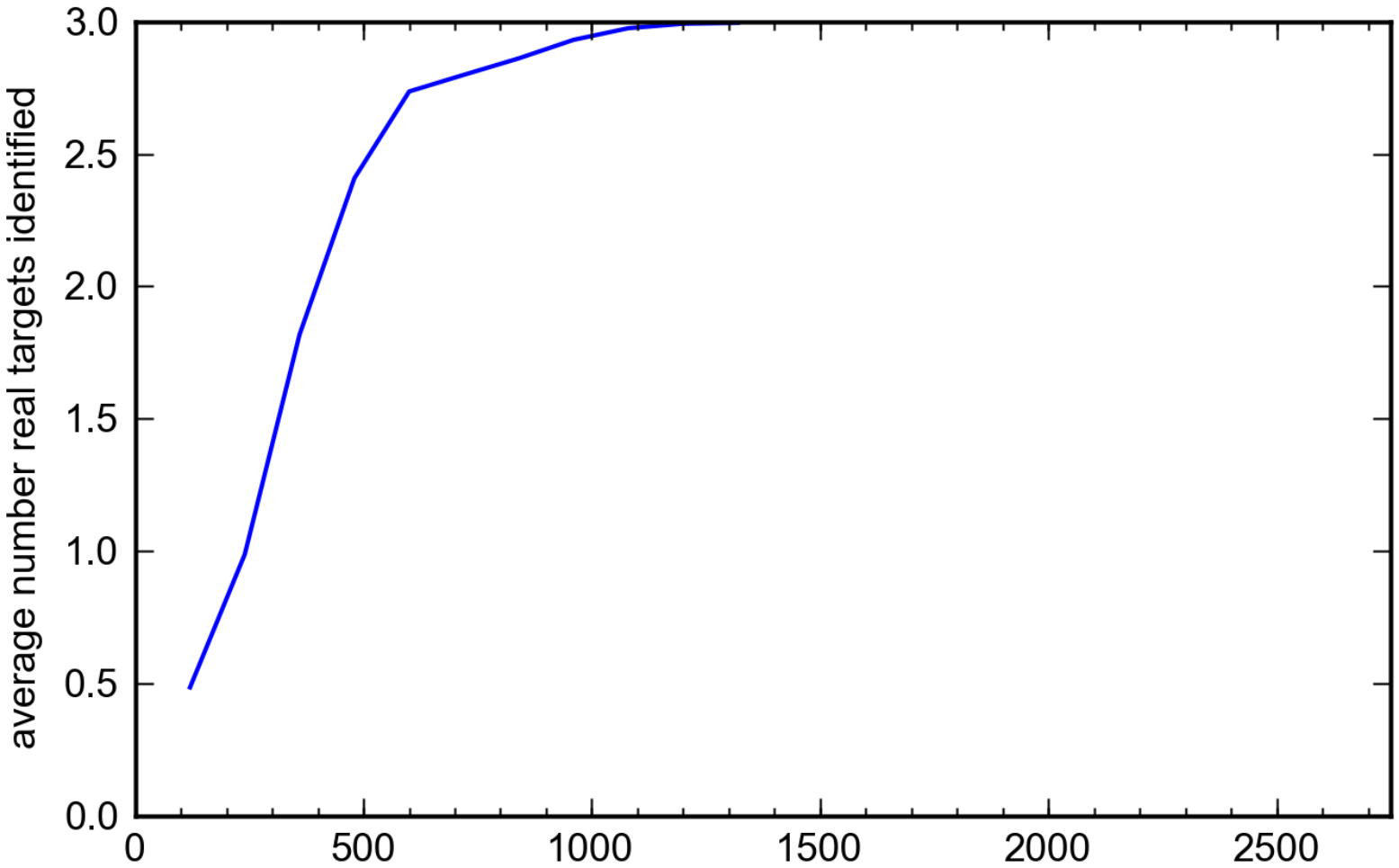
Experimental Results

- Phenotype sequencing identified 9 candidate genes for isobutanol tolerance.
- Three genes validated as causing isobutanol tolerance by repair / deletion experiments.
- Six more candidates need to be studied experimentally.
- Some target genes **easy** to detect: *acrB*

Measuring Phenotype Sequencing Reliability & Cost

Experimental and Bioinformatics analyses

Predicted Yield Curve



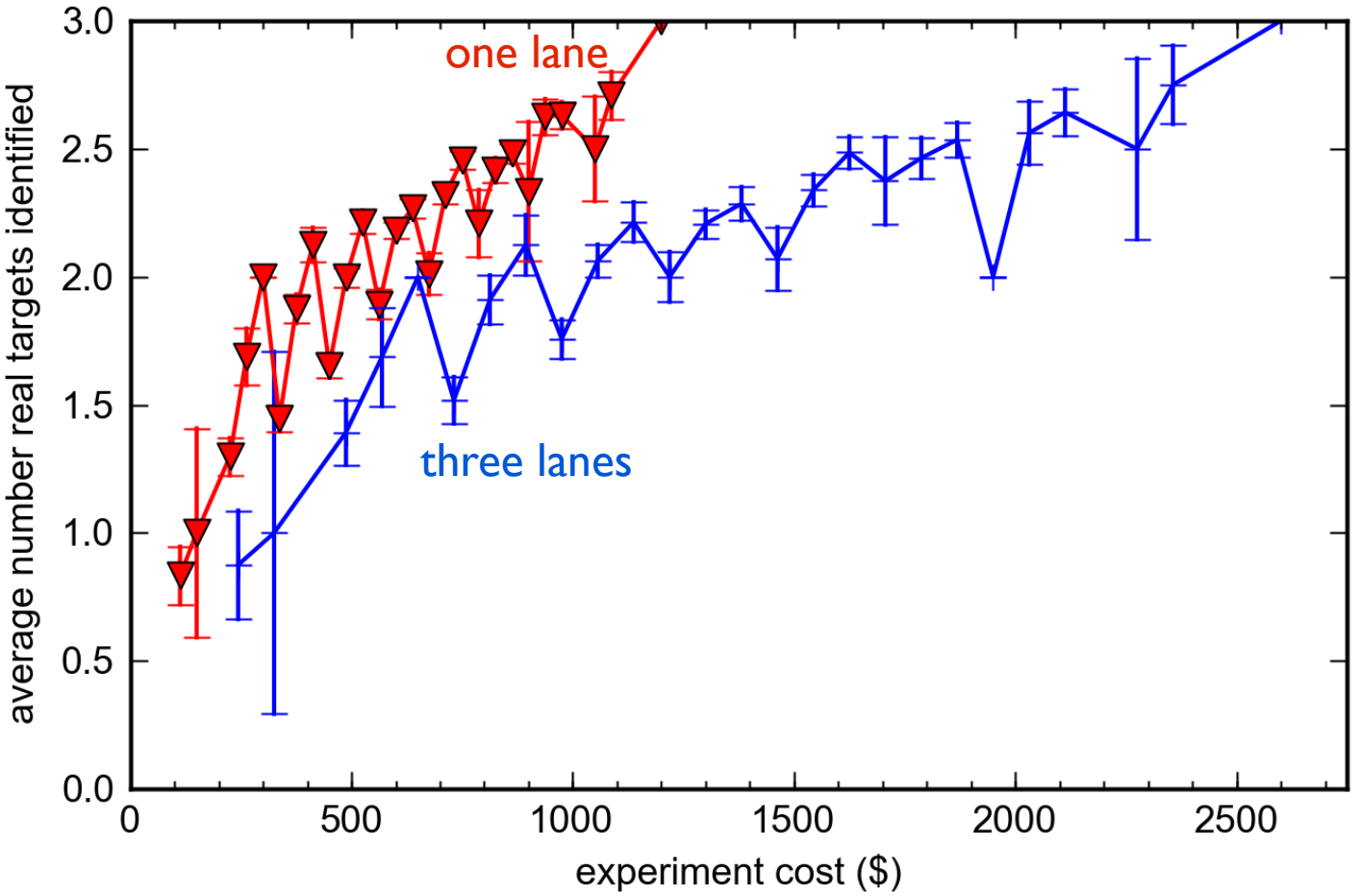
Cost basis:
$700 / lane

$50 / library

(reagents only)

Model for three target genes with equal "hit" probability

Experimental Yield Curve



Cost basis:
$700 / lane

$50 / library

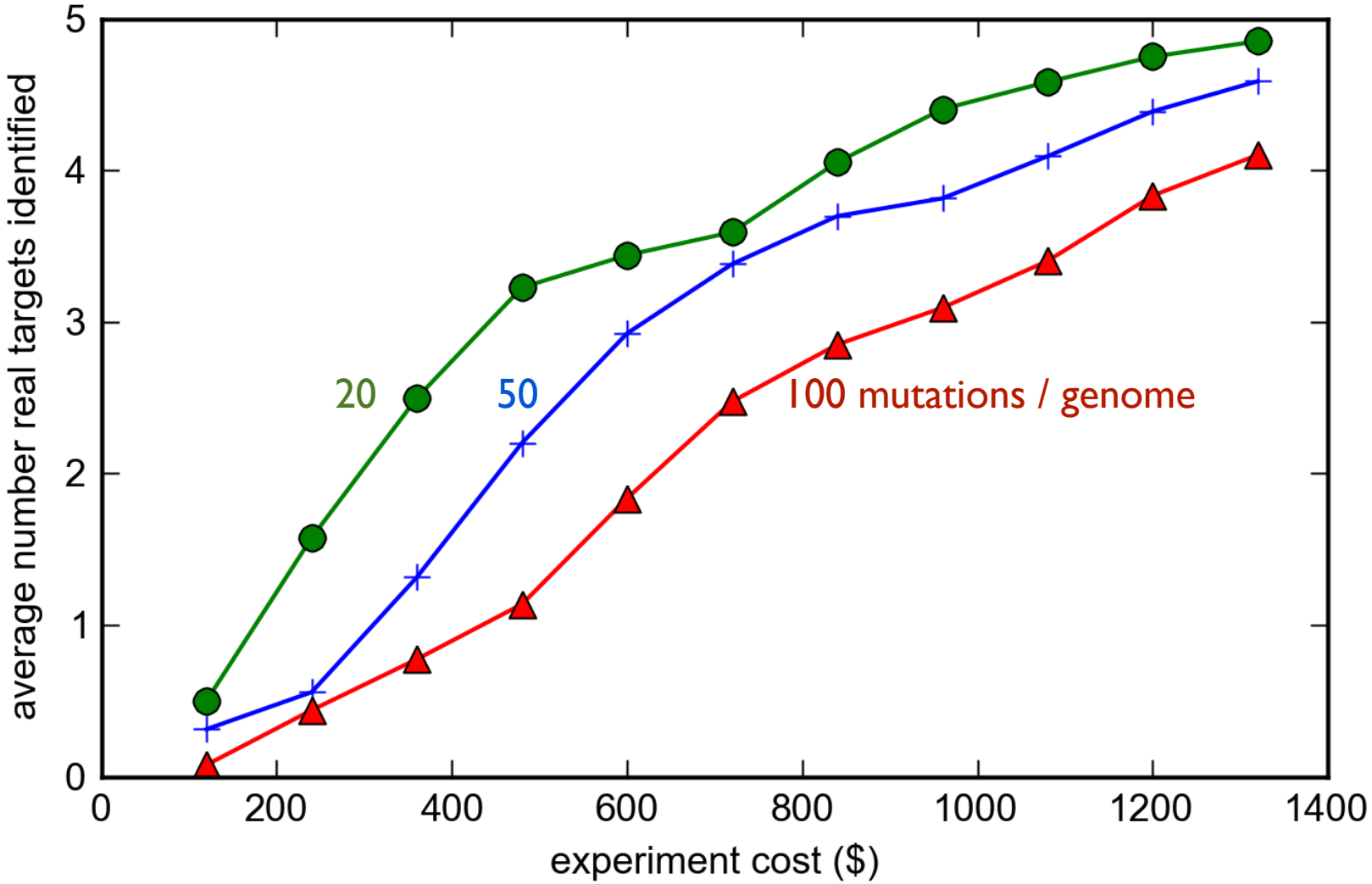
(reagents only)

Test: *acrB*, *marC*, *acrA* among top 20 hits?

Pooling Dramatically Reduced Cost

- Sequencing 3-4 strains ($110-$150) reliably detected *acrB* (detected among top p-values)
- Sequencing 8-14 strains ($340-$525) reliably detected *acrB* and *marC*.
- Detecting all three targets required sequencing the full 32 strains ($1200, vs. $7200 for a conventional genome sequencing design).
- One lane of sequencing gave as good results as three replicate lanes.

Lower mutagenesis reduces cost almost in half



Future Optimization Strategies

- Reducing the sequencing error rate to 0.1% (e.g. multi-base encoding) can reduce cost as much as cutting sequencing lane cost in half.
- Together they reduce phenotype sequencing cost 3-4 fold.
- With reduced sequencing costs, library costs become dominant (i.e. fewer libraries = more efficient phenotype sequencing).

Phenotype Sequencing Applications and Issues

Benefits

- These analyses provide a general method for reliably finding genes that cause a phenotype.
- Applicable to any problem where phenotype screen can produce enough mutants.
- Sequencing cost going down, down over time. May gradually become cost-effective for larger genomes.

Fast & Cheap

- If you have a good screen, mutagenesis can quickly generate a large number of mutants with the phenotype.
- 1-2 weeks for library prep and sequencing
- 1-2 days for data analysis
- Costs start as low as $100 - $1000.
- easy to start with a small number of strains, then expand as needed.

Applications

- So far we have analyzed three phenotype sequencing experiments: two in *E. coli* (4.6 Mb); one for human exome data (30 Mb).
- All three successfully identified genes that cause the phenotype, as validated by independent experimental data.
- May be especially useful for organisms with interesting phenotypes but without good genetic tools for dissecting them in traditional ways, e.g. *Chlamydia*. Just sequence mutants!

Phenotype Sequencing of a Human Disease

- Pilot study: exome sequencing of 6 unrelated patients with a genetic disease (N. Kim, Korea).
- Phenoseq analysis of rare variants identified a single gene. Validated by a separate pedigree study.
- Diploid genomes make phenoseq even more powerful! Method of choice for plant phenotypes?
- Exome sequencing is cheap and scalable. Phenoseq only needs a good fraction of the phenotype to be caused by splicing or protein mutations -- not all.

Collaborations

- Iara Machado (increased isobutanol tolerance), Luisa Gronenberg (metabolic pathway reengineering), James Liao
- Marc Harper: phenoseq analysis
- Zugen Chen, Traci Toy, Stan Nelson (sequencing)
- Namshin Kim (KOBIC; human disease gene discovery)
- Jim Gober (improved cellulosic digestion)

Bio-Information Journal Club

- We're starting a journal club for discussing information metrics in bioinformatics, evolution and biology.

- First meeting: Wed. Oct. 19

- Details will be posted on **http://thinking.bioinformatics.ucla.edu**

Mutant Sources

- *naturally evolved* (under selection): some phenotypes can be hard, slow to generate. Hard to get large numbers of *independent* mutants. Smaller mutation density.
- *mutagenesis*: can generate large numbers of independent mutants. Higher (but controllable) mutation density. Commonly viewed as hard to interpret, but phenotype sequencing solves that!

Challenges of Naturally Evolved Mutants

- Any selection scheme where multiple mutants compete in the same culture will tend to hide all but the "best" target (even a small selective advantage becomes absolutely dominant over time).
- Trivial causes of a phenotype (e.g. lose the plasmid) can obscure interesting causes.
- May not reveal all possible causes.

Shotgun Benefits

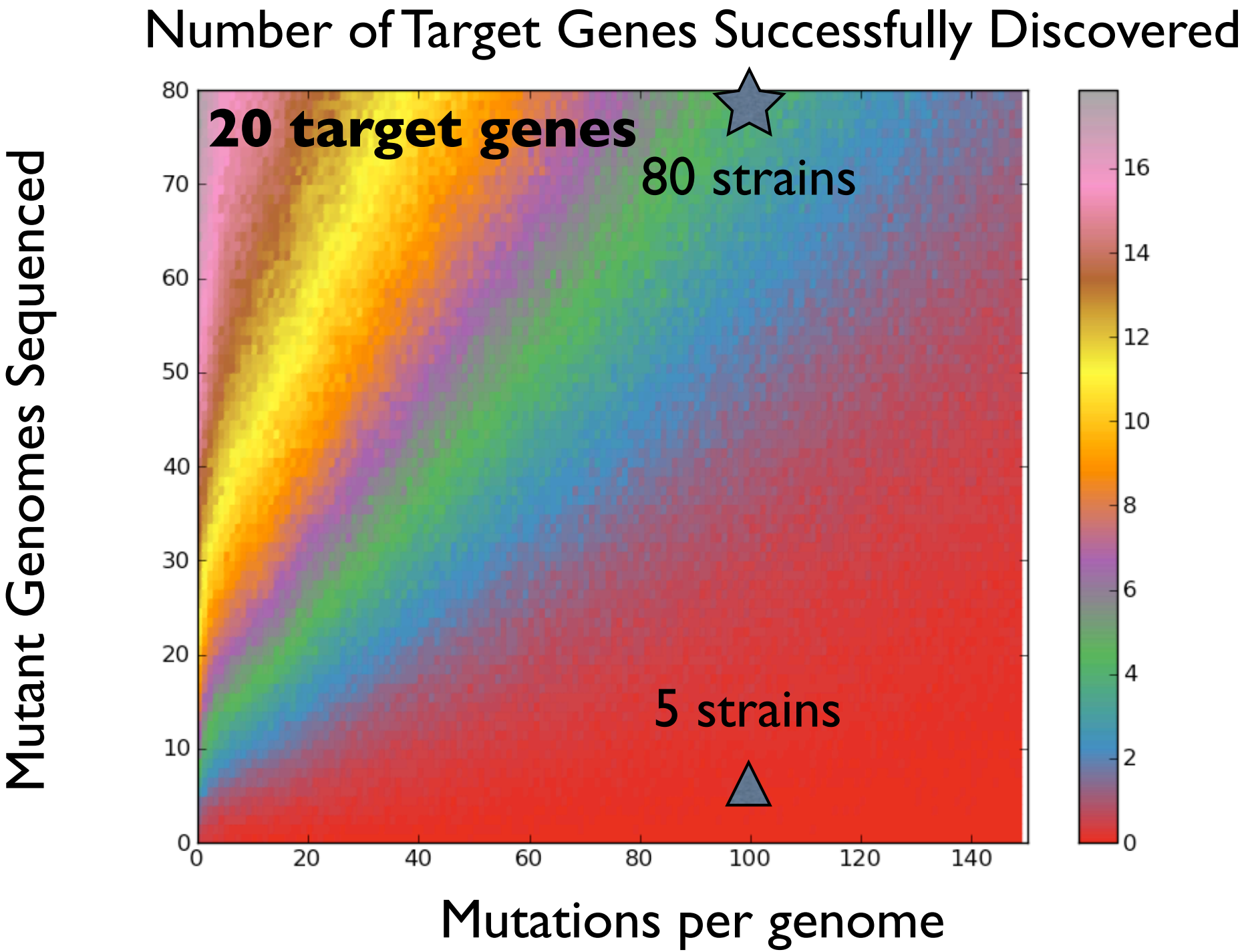
- Mutagenesis hits all genes randomly and comprehensively ("shotgun").
- "Independent mutants" means each culture should contain at most one mutant that passes the screen. This means *they do not compete*.
- A target gene's detection probability depends only on its *size* in the genome.
- Representative picture of *all* causes of a phenotype. They don't hide each other.

Our Conservative Assumptions

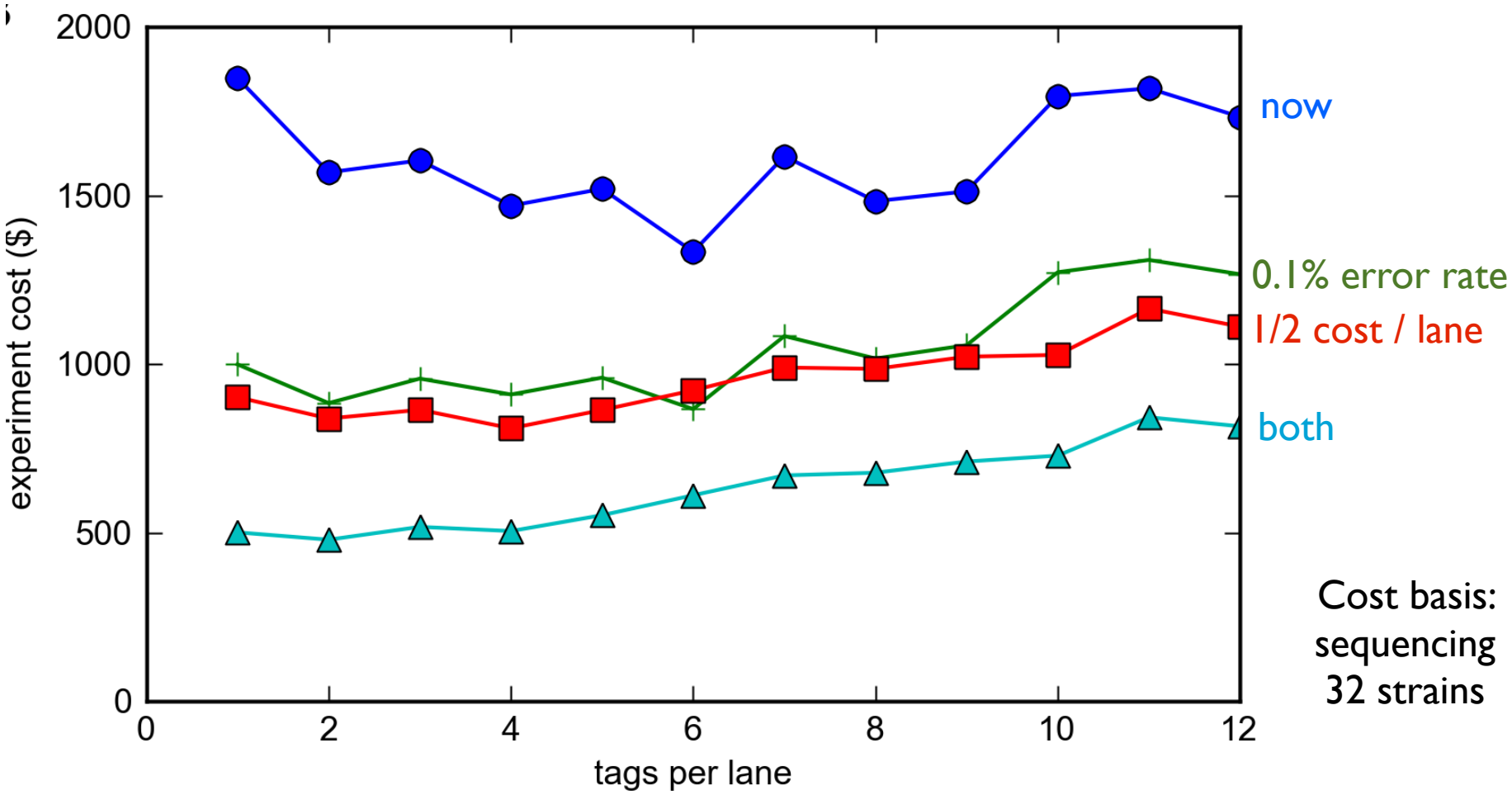
Our analysis assumed hardest cases:

- only one mutation needed for phenotype
- all target genes contribute equally
- high mutagenesis density
- large number of target genes

Real cases likely to be easier than we assumed.


Number of Target Genes Successfully Discovered
20 target genes
80 strains
5 strains
Mutant Genomes Sequenced
Mutations per genome
0
10
20
30
40
50
60
70
80
0
20
40
60
80
100
120
140
0
2
4
6
8
10
12
14
16

Reduced sequencing error vs reduced cost

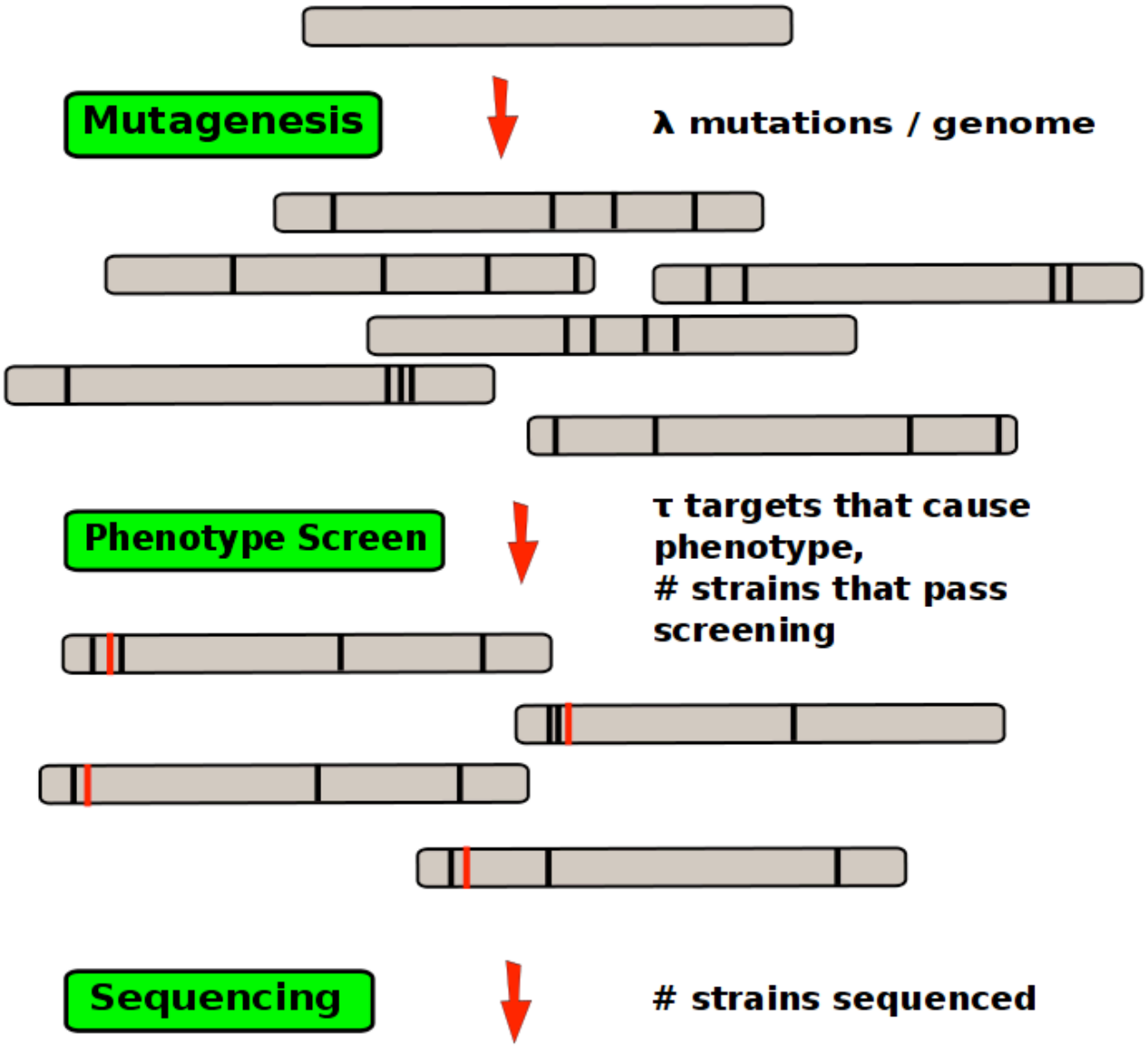


Modeling of Experiment Optimization

- Phenotype sequencing yield is fundamentally limited by the number of strains sequenced.
- The primary goal of experiment optimization is to reduce the cost per strain.
- Modeled the key factors: mutagenesis density; sequencing error rate; sequencing cost.

Requirements

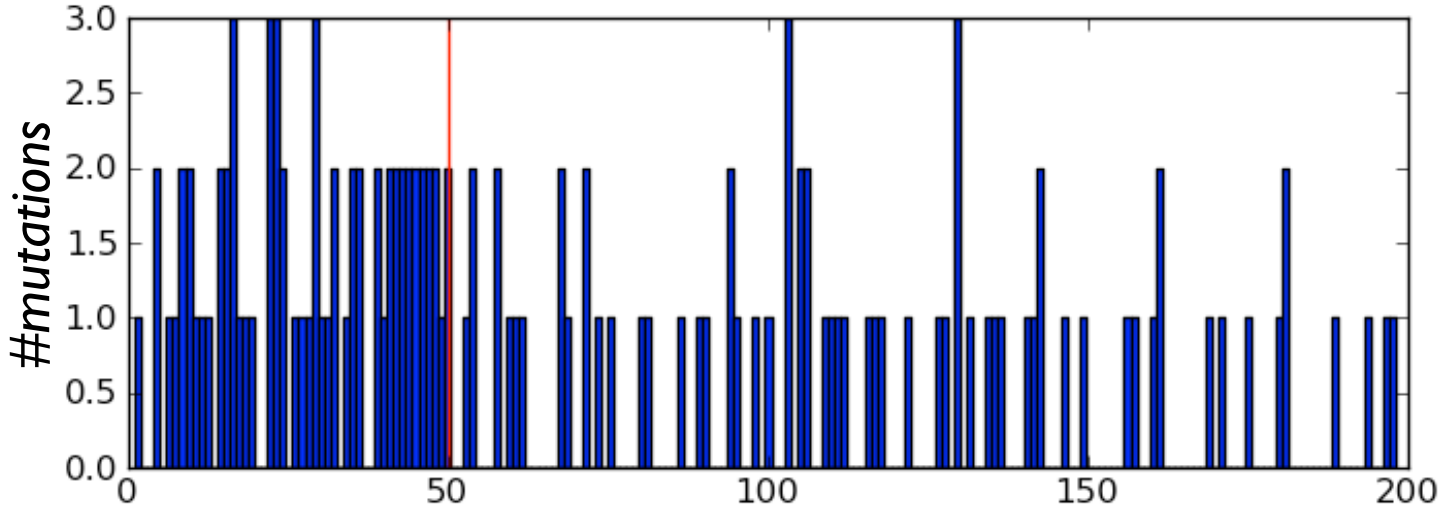
- enough mutant strains, independently generated at a low mutagenesis density.
- a small enough genome to sequence this number of strains at acceptable cost.
- (a reference genome with gene annotations): not required but reduces the work



ATTACGAGGGATCCTATGACGC...
ATACCGAGGGATCCTATGACGC...
ATTACGAGCGATCCTATGACGC...
ATAACGAGGGATCCTATGACGC...

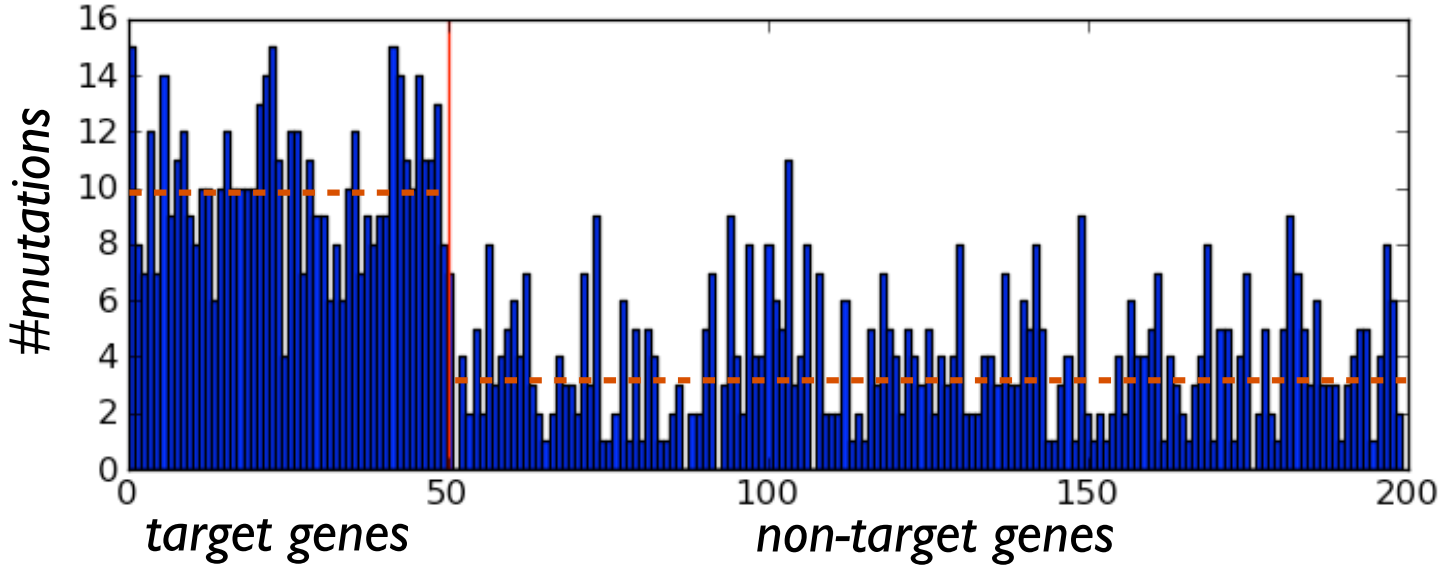
Critical Factor: #Sequenced Strains

few strains



very noisy, hard to distinguish

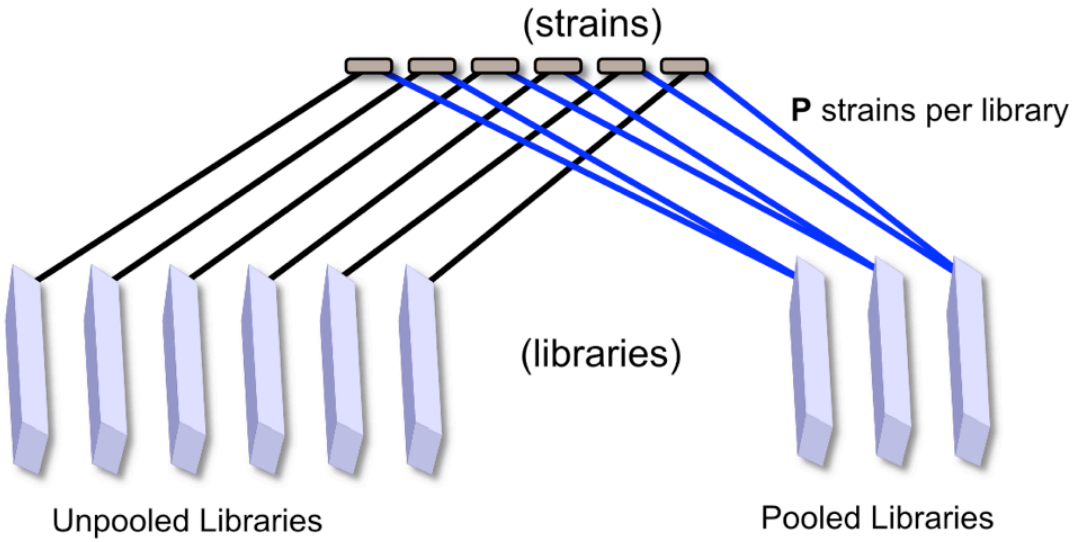
many strains



greater separation, many target genes distinguishable

as more strains sequenced, the #hits/gene converge to means

Standard vs. Pooled Sequencing



Sequencing

ϵ sequencing error rate

A A T G C C	A A T G C C
A A T G C C	A A T G C C
A A T G C C	A A T G C G
T A T G C C	T A T G C C
A A T G C C	A A T G C G
A A T G C C	A A T G C G

mutation expected
as **1 - ϵ** of reads per library

mutation expected
as **(1 - ϵ) / P**
of reads per library

Phenotype Sequencing Via Pooling

- Pooling can count mutations but can't reconstruct each individual sequence.
- Reduces costs by the pooling factor *P*.
- For small *E. coli* genome, we can also sequence many pools (tagged libraries) in a *single* lane.
- How low can we go? We need to keep a real mutation case (c/P reads expected) strongly distinguishable from sequencing error (cε reads expected).

Genome-wide Mutation Calls

too many false positives

ideal zone

too many false negatives

Assuming coverage c=75,

sequencing error ε=0.01,

pooling P=4,

genome size 4 MB

50 true mutations / genome

reliably detecting a small number of true mutations in a pool of multiple genomes requires strong statistical confidence.

Number of Hits per Gene

#SNPs	Genes
32	*acrB*
27	*ydfJ*
12	*cusA, entF*
11	*nfrA, prpE*
10	*febA, rhsD, sbcC*
9	*aesA, bscC, marC, mdlB, paoC, ykgC, yneO*
8	*ampH, kefA, yagX, ybaE, ybaL*

Target Gene Scoring

P-value for k_{obs} hits, for the null hypothesis (not a target gene) follows uniform density (Poisson) model:

$$p(k \geq k_{obs} | \text{non-target}, \lambda) = \sum_{k=k_{obs}}^{\infty} \frac{e^{-\lambda} \lambda^k}{k!}$$

where the gene's mutational cross section λ reflects GC bias of NTG, and the gene's actual GC/AT composition:

$$\lambda = N_{GC} \mu_{GC} + N_{AT} \mu_{AT}$$

Top 20 Genes by P-value

p-value	Genes	Description
9.5×10^{-20}	**acrB**	multidrug efflux system protein
1.4×10^{-5}	**marC**	inner membrane protein, UPF0056 family
1.8×10^{-4}	*stfP*	e14 prophage; predicted protein
0.0011	*ykgC*	predicted pyridine nucleotide-disulfide oxidoreductase
0.0035	*aes*	acetyl esterase; GO:0016052 - carbohydrate catabolic process
0.017	*ampH*	penicillin-binding protein yaiH
0.038	*paoC*	PaoABC aldehyde oxidoreductase, Moco-containing subunit
0.039	*nfrA*	bacteriophage N4 receptor, outer membrane subunit
0.044	*ydhB*	putative transcriptional regulator LYSR-type
0.12	*yaiP*	predicted glucosyltransferase
0.17	**acrA**	multidrug efflux system
0.25	*xanQ*	xanthine permease, putative transport; Not classified
0.25	*ykgD*	putative ARAC-type regulatory protein
0.35	*yegQ*	predicted peptidase
0.35	*yfjJ*	CP4-57 prophage; predicted protein
0.37	*yagX*	predicted aromatic compound dioxygenase
0.46	*pstA*	phosphate transporter subunit
0.48	*prpE*	propionate–CoA ligase
0.50	*mltF*	putative periplasmic binding transport protein, membrane-bound lytic transglycosylase F
0.63	*purE*	N5-carboxyaminoimidazole ribonucleotide mutase

Harper et al., *PLoS ONE*, 2011

Analyzing all the "sub-experiments" in our data

Results from 10 separate libraries allow us to analyze many possible "subexperiments", i.e. all possible combinations of the ten libraries.

1	2	3	4	5	6	7	8	9	10	#strains
+										3
	+									3
+	+									6
		+								3
+		+								6
										...
+	+	+	+	+	+	+	+	+	+	32

How reliably can we discover the correct target genes with just
3 strains?
6 strains?
9 strains?...