Computational Experiment Planning and the Future of Big Data

Christopher Lee

Departments of Computer Science, Chemistry and Biochemistry, UCLA

Why Big Data?

Not everyone here will consider themselves to be working on “Big Data”, but it seems useful for BICOB now because

- *it’s where the discoveries are*: new kinds of high-throughput data are enabling new kinds of discovery. The datasets are huge and require computational analysis.
- *it’s where the field is going*: the same issues are arising again and again as different areas of biology / bioinformatics undergo the same transformation (to Big Data).
- *it’s teaching us*: principles emerge from Big Data analyses that unify disparate areas of methods and give new insights, new capabilities

Big Data: Automate Discovery

- **computational scalability**: algorithms that find a gradient in a lower dimensional space
- **statistical scalability**: as datasets grow huge, IF-THEN rules fail to cut because distributions may overlap, evidence may be weak, even “tiny” error rates may add up to huge FDR.
- **model scalability**: computations can find interesting things even when (initial) models are wrong.

Topics: Empirical Information Metrics for...

1. *model selection*
2. data mining *patterns* and *interactions*
3. data mining *causality*
4. *computational experiment planning*

1. data mining methods: Model Selection

- choose the model that maximizes a *scoring function*
- seems so generic as to cover all the possibilities by definition
- address *computational scalability* algorithmically, by "choosing a space" in which there is a low(er) dimensional gradient pointing in the direction of better (and better) models.

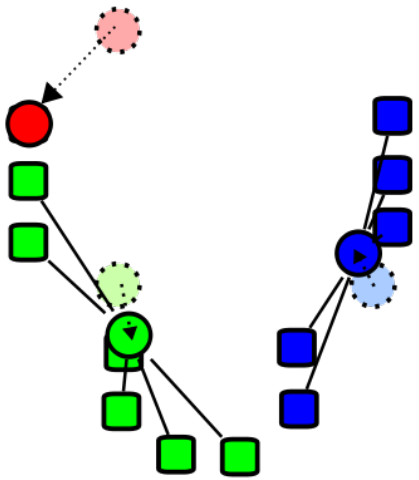
Examples:

- energy-based structure prediction
- maximum likelihood parameter estimation
- "hill-climbing" methods like gradient descent, Expectation-Maximization

data mining methods: Domain-specific Scoring Functions

- potential energy
- k-means (Gaussian clustering): can think of this as k centroids μ_i attached by "springs" to their respective data points x_j , and positioned to minimize the potential energy.

$$E = \sum_{i=1}^{k} \sum_{\vec{x}_j \in S_i} ||\vec{x}_j - \vec{\mu}_i||^2$$



- or any scoring function you can think up...

General Scoring Functions: Why Bother?

Since we can always make up domain-specific scoring functions, this might seem to cover all our possible needs. But historically, people have hit three basic reasons for seeking *general* scoring functions:

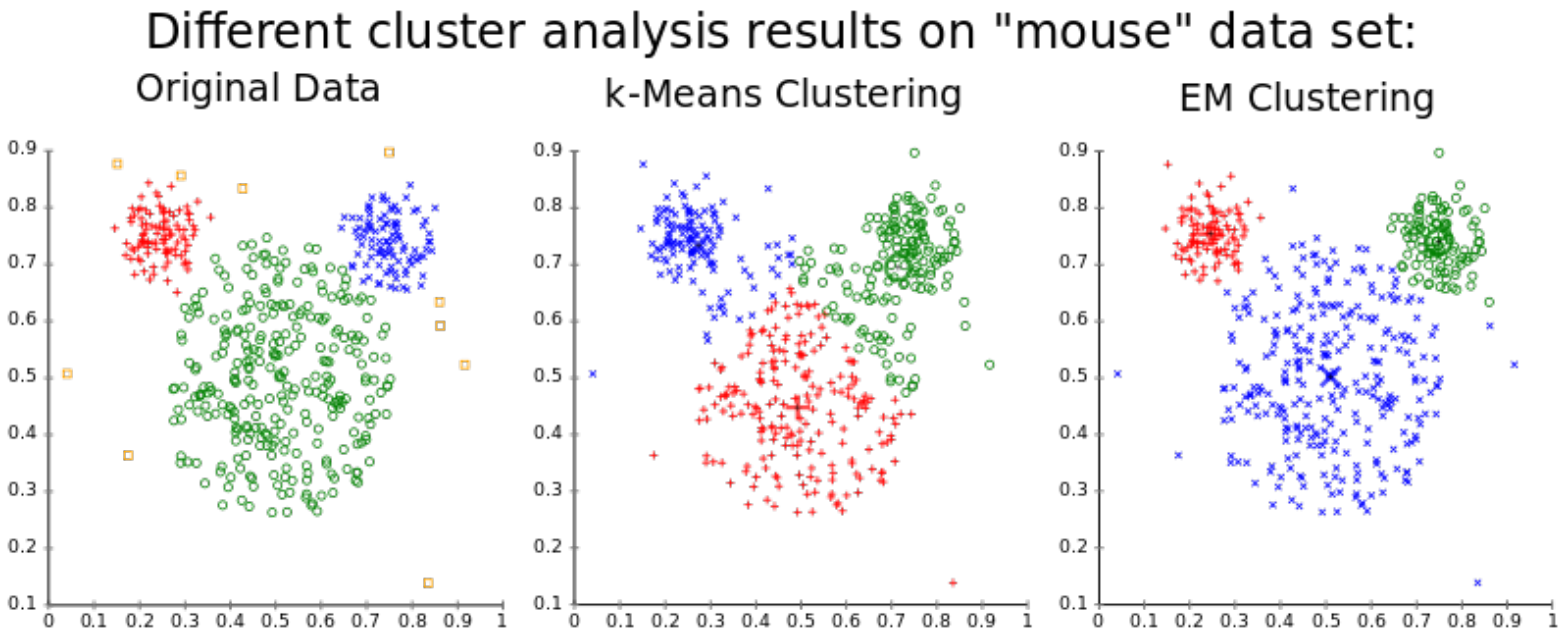
- a domain-specific scoring function only works within narrow range of its (implicit) assumptions
- generalization both *simplifies*, *unifies* and *expands* our understanding (the same idea always works).
- generalization enables automation.

This addresses the need for *model scalability*

Example: k-means

- misclusters even simple data (assumes equal variance)

$$E = \sum_{i=1}^{k} \sum_{\vec{x}_j \in S_i} ||\vec{x}_j - \vec{\mu}_i||^2$$



- overfitting: “optimal” k-means is always *k=n* (*E=0*). Yikes!

What's Wrong? No Cheating Allowed!

We could explicitly take the variance for each cluster into account:

$$E = \sum_{i=1}^{k} \sum_{\vec{x}_j \in S_i} \frac{||\vec{x}_j - \vec{\mu}_i||^2}{\sigma_i^2}$$

But now it always tell us "optimal" is $\sigma \to \infty$. Yikes!
Solution: convert this to a real probability model (Normal distr.):

$$\log p(x_1, x_2, \ldots x_n | \mu_1, \ldots \mu_k, \sigma_1, \ldots \sigma_k) = \sum_{i=1}^{k} \sum_{\vec{x}_j \in S_i} \log \frac{1}{\sigma_i \sqrt{2\pi}} e^{-\frac{||\vec{x}_j - \vec{\mu}_i||^2}{2\sigma_i^2}}$$

$$= \sum_{i=1}^{k} \sum_{\vec{x}_j \in S_i} \left(-\log \sigma_i \sqrt{2\pi} - \frac{||\vec{x}_j - \vec{\mu}_i||^2}{2\sigma_i^2} \right) = n\bar{L}$$

Prediction power "pays" the right price for increasing σ. No cheating!

Generalization: Probabilistic Scoring Functions

Various general scoring functions have been developed based on log-likelihood with corrections to protect against certain types of overfitting, e.g.

- Akaike Information Criterion (minimize)

$$AIC = 2k - 2\log p(x_1, x_2, \ldots x_n|\Psi) = 2k - 2n\bar{L}$$

- Bayesian Information Criterion (minimize)

$$BIC = k\log n - 2n\bar{L}$$

- Bayes' Factor (maximize):

$$BF = \log p(\psi) + n\bar{L}$$

2. Data Mining Patterns and Interactions

Prediction Power, Entropy and Information

The long-term prediction power *E(L)* for observable X with probability distribution *p(X)* is just

$$E(L) = \sum_X p(X) \log p(X) = -H(X)$$

where *H(X)* is defined as the entropy of random variable X.
In 1948 Shannon used this to define information as a *reduction in uncertainty* (increase in prediction power). Specifically, the average amount of information about X that we gain from knowing some other variable Y (averaged over all possible values of X and Y) is defined as

$$I(X;Y) = H(X) - H(X|Y) = E(L(X|Y)) - E(L(X))$$

which is called the *mutual information*.

Example: Sequence Logos (Schneider, 1990)

The vertical height of each column is

$$I(X;obs) = H(X) - H(X|obs)$$

- where $H(X)$ is 2 bits for DNA, and *obs* are the observed letters in that column of a multiple sequence alignment.
- illustrates importance of setting metric to the proper *zero point*.
- should not be fooled by weak evidence (*obs*)

Example: Detecting detailed protein-DNA interactions

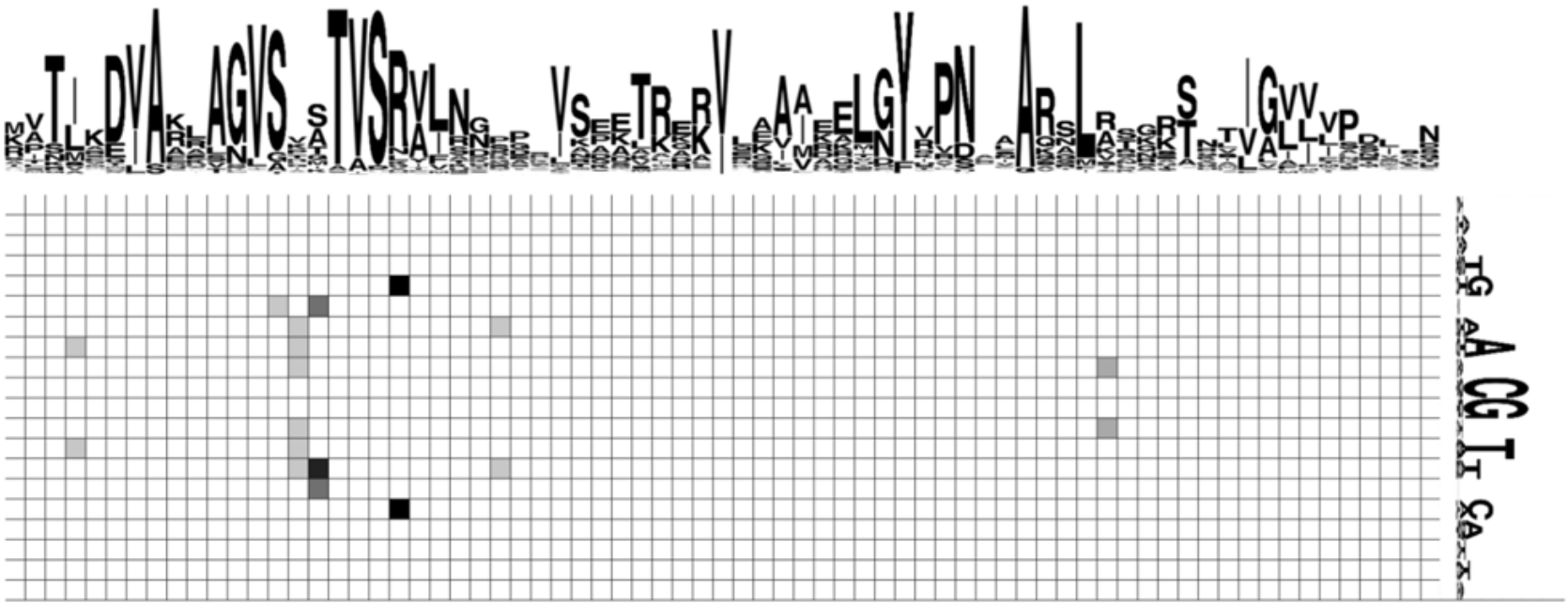
- Say we had a large alignment of one transcription factor protein sequence from many species, and a large alignment of the DNA sequences it binds (from the same set of species).
- In principle *co-variation* between an amino acid site vs. a nucleotide site could reveal specific interactions within the protein-DNA complex.
- mutual information detects precisely this co-variance (or departure from independence):

$$I(X;Y) = E\left(\log \frac{p(X,Y)}{p(X)p(Y)}\right) = D(p(X,Y)||p(X)p(Y))$$

where $D(\cdot||\cdot)$ is defined as the *relative entropy*.

LacI-DNA Binding Mutual Information Mapping

LacI protein sequence (x-axis) vs. DNA binding site (y-axis)



I(X;Y) computed from 1372 LacI sequences vs. 4484 DNA binding sites (Fedonin et al., *Mol. Biol.* 2011).

Note: strong information (interaction) is often seen between *high entropy* sites, rather than highly conserved sites.

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.

The Mutual Information Sampling Problem

Consider the following "mutual information sampling problem":

- draw a specific inference problem (hidden distribution $\Omega(X)$) from some class of real-world problems (e.g. for weight distributions of different animal species, this step would mean randomly choosing one particular animal species);
- draw training data $\vec{X}^t$ and test data X from $\Omega(X)$;
- find a way to estimate the mutual information $I(\vec{X}^t; X)$ on the basis of this single case (single instance of Ω).

$I(\vec{X}^t; X)$ is only defined as an average over total joint distribution of $\vec{X}^t, X$ (over all possible Ω). In fact, if we sample many pairs of $\vec{X}^t, X$ from one value of Ω, we will get *I=0* (because $\vec{X}^t, X$ are conditionally independent given Ω)!

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)

Empirical Information Sampling

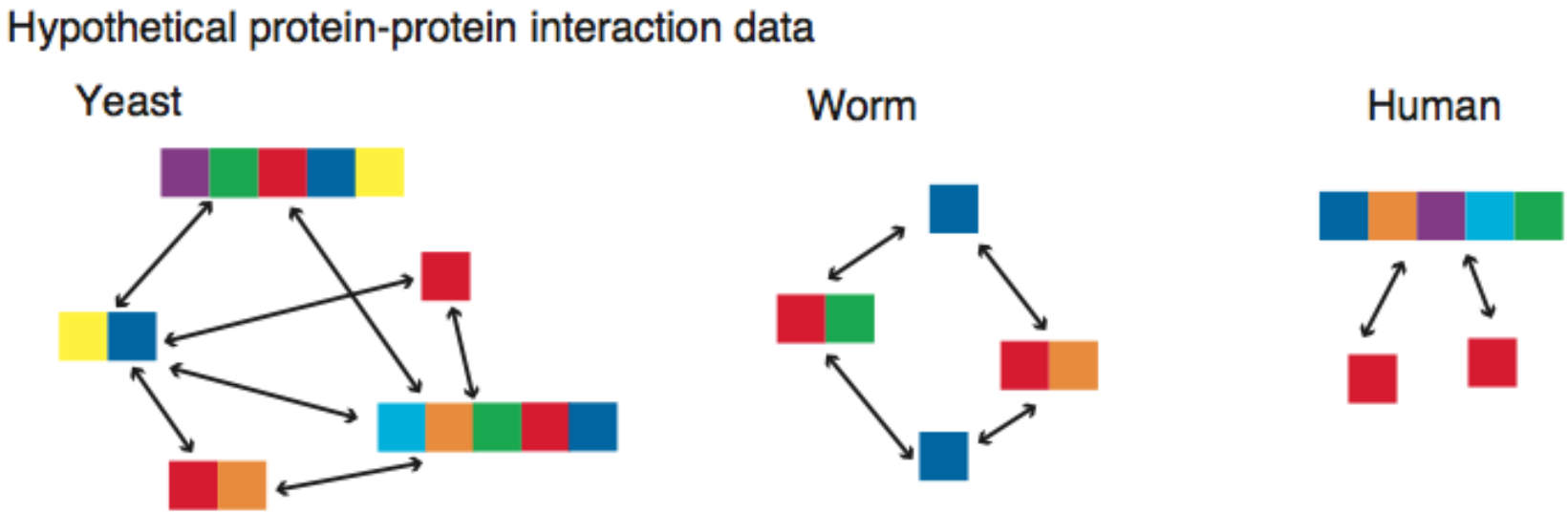
Say we train a model Ψ on training data $\vec{X}^t, X$ from some specific Ω, and measure its prediction power via I_e, and repeat this for many unknowns Ω. What will the average of these empirical information values tell us?

$$E(I_e(\Psi)) = E(L_e(\Psi)) - E(L_e(p)) = E(L_e(\Psi)) + H(X)$$

$$= H(X) - H(X|\vec{X}^t) - E_{\vec{X}^t}(D(p(X|\vec{X}^t)||\Psi(X|\vec{X}^t))) \rightarrow I(X;\vec{X}^t)$$

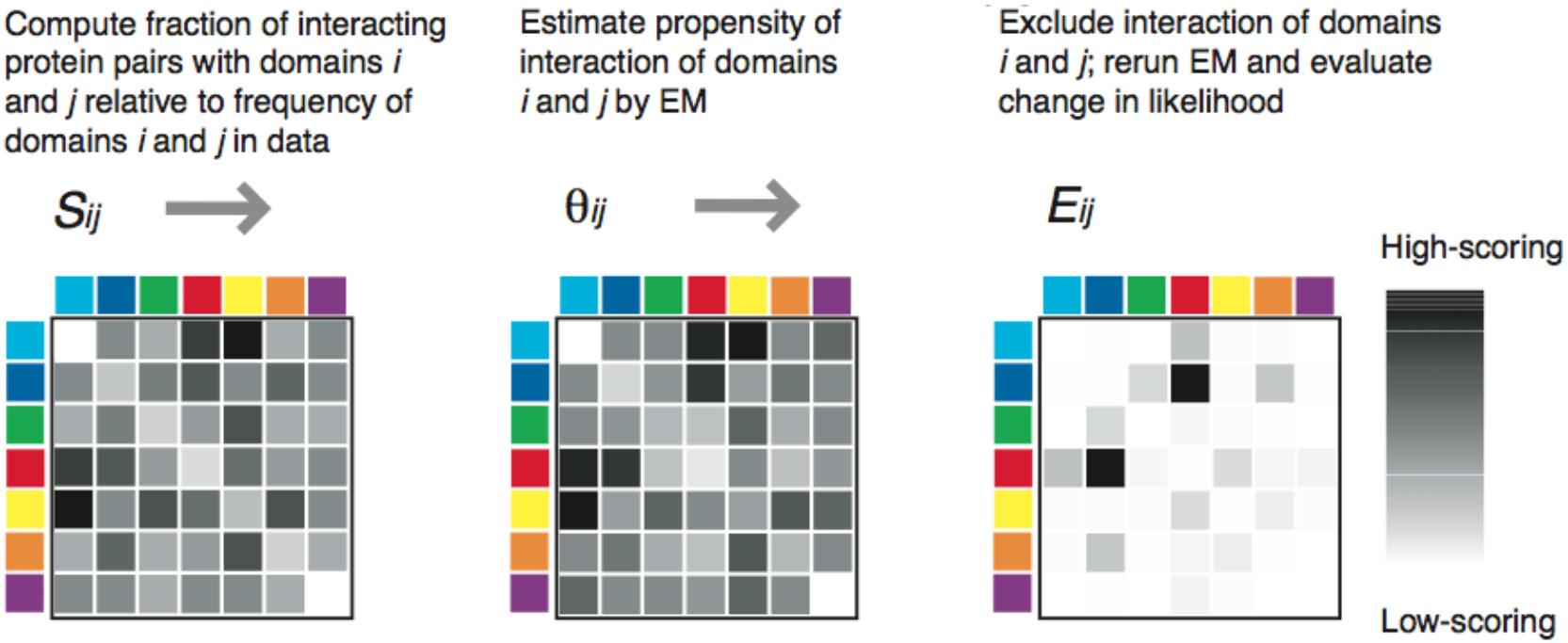
as Ψ becomes increasingly accurate. Hence, I_e solves the mutual information sampling problem. Concretely, we can get an empirical estimator of the mutual information "value" that some factor X yields about some other variable of interest Y, by simply measuring how much X increases our empirical information about Y (even from a single case!).

Example: Domain Interactions from Multi-domain Proteins



- Eukaryotes contain complex multi-domain protein architectures.
- Given a database of protein-protein interaction pairs across many genomes, and the domain composition of each protein, can we deconvolute *which* individual domain-domain pairs mediate these interactions?

Domain Interaction (Riley et al., Genome Biol. 2005)

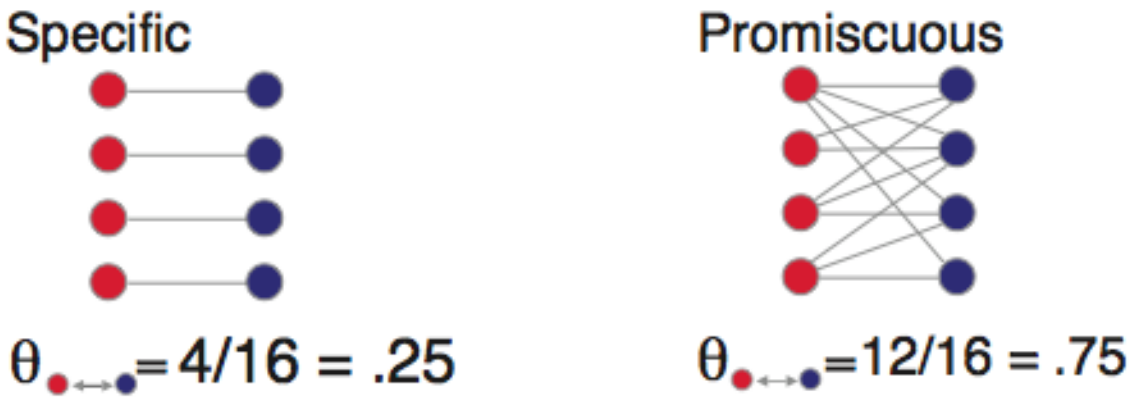


- S_{ij}: fraction of domain *i,j* pairs that are in interacting protein-pairs
- θ_{ij}: fraction of domain *i,j* pairs that directly interact (bind)
- E_{ij}: total strength of evidence that *i,j* directly interact. Concretely, if Ψ^0_{ij} is the model constrained to $\theta_{ij} = 0$, then

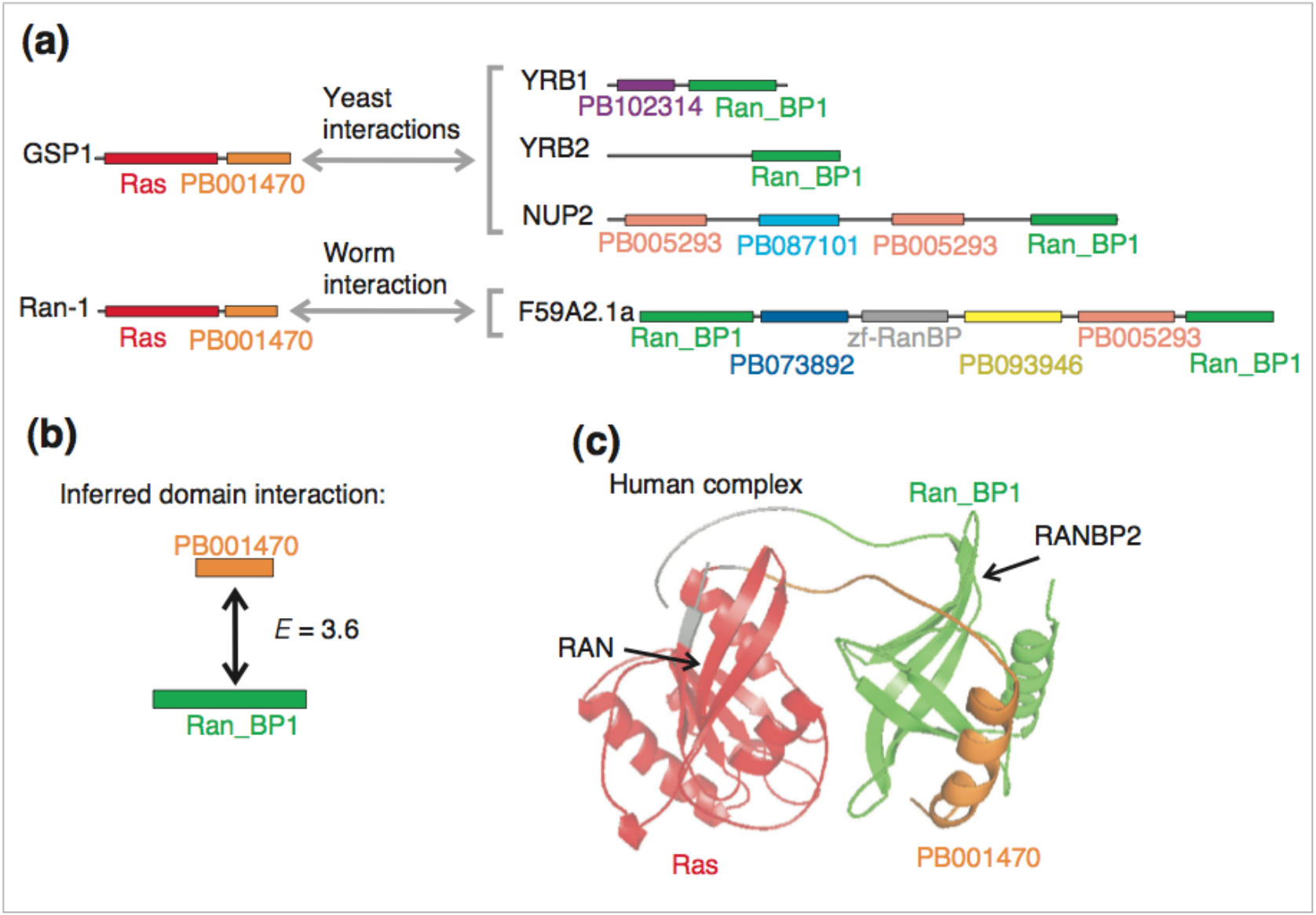
$$E_{ij} = n(\overline{L}_e(\Psi) - \overline{L}_e(\Psi^0_{ij})) = n(\overline{l}_e(\Psi) - \overline{l}_e(\Psi^0_{ij}))$$

Domain Interaction Data Mining

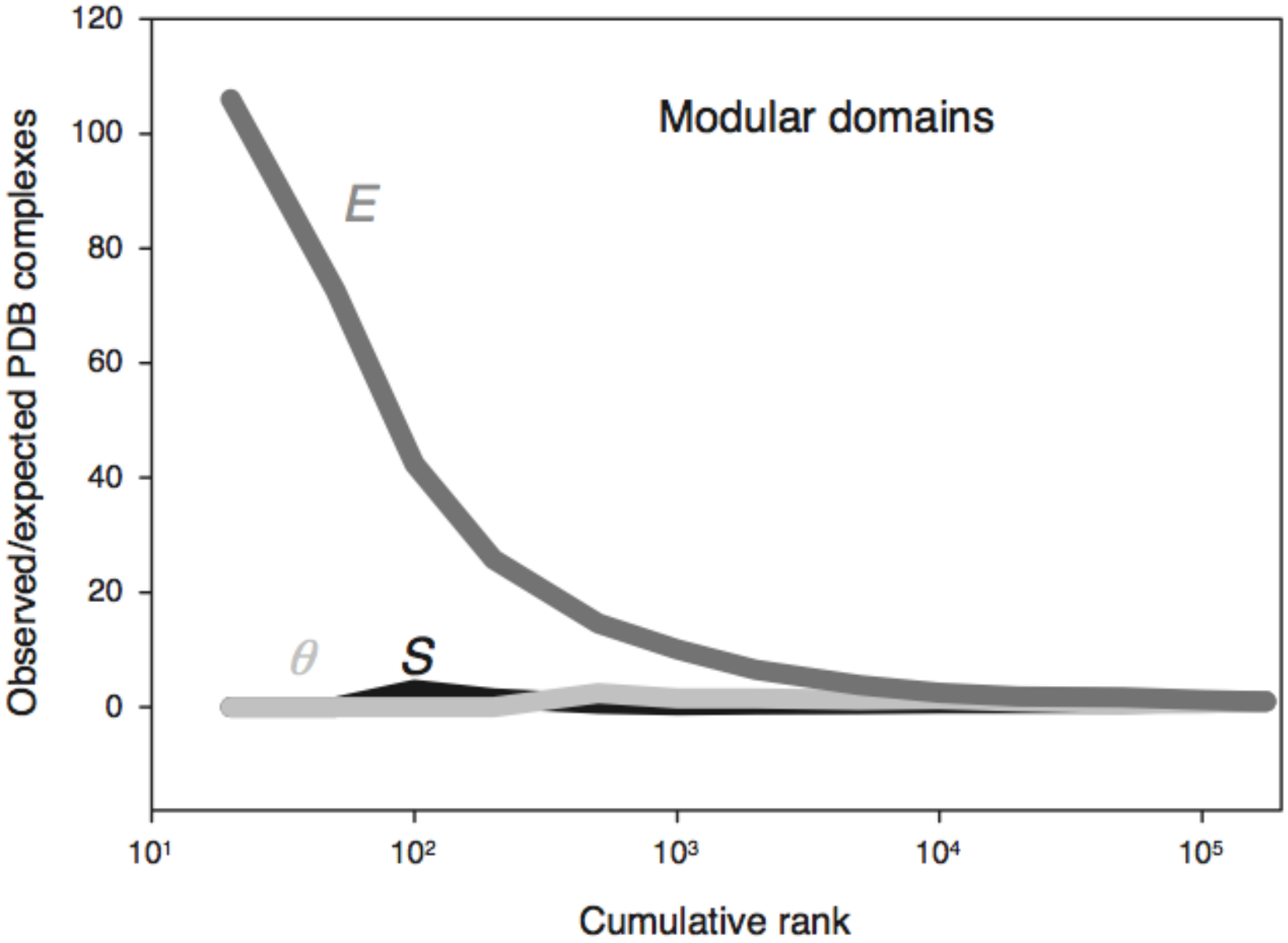
- Database of Interacting Proteins (DIP): 26,032 interaction pairs among 11,403 proteins from 68 organisms
- These proteins contain 12,455 distinct Pfam domain types
- A total of 177,233 possible interacting domain pairs based on co-occurrence in interacting proteins.
- Predicted 3005 domain pairs with $E_{ij} > 3.0$ (p<0.001)
- "promiscuous": high θ_{ij}, high E_{ij}
- "specific": low θ_{ij}, high E_{ij}



Novel Domain Interaction Validated by 3D Structure



Validation by 3D Structure Database (PDB/iPfam)



Domain Interaction Data Mining Conclusions

- E_{ij} used as total evidence measure for the empirical information ΔI_e associated with allowing $\theta_{ij} > 0$, and hence for the mutual information $I(d_{ij}; \beta)$, where d_{ij} represents presence or absence (1 vs. 0) of domains *i,j* in a given protein pair, and β whether that pair binds or not (1 vs. 0).
- greatly out-performs correlation measures in prediction accuracy
- indeed, biologically, high correlation (large θ_{ij}) is not even necessarily what we want to detect (promiscuous interactions). Specificity is a good thing!

3. Data Mining Causality

Chain Rules & Independence

We can always expand a joint probability in any order, e.g.

$$p(X, Y, Z...) = p(X)p(Y|X)p(Z|X, Y)...$$

Or equivalently:

$$H(X, Y, Z...) = H(X) + H(Y|X) + H(Z|X, Y) + ...$$

Of course, this may simplify if some variables are *independent*

$$p(Y|X) = p(Y) \implies H(Y|X) = H(Y) \implies I(X; Y) = 0$$

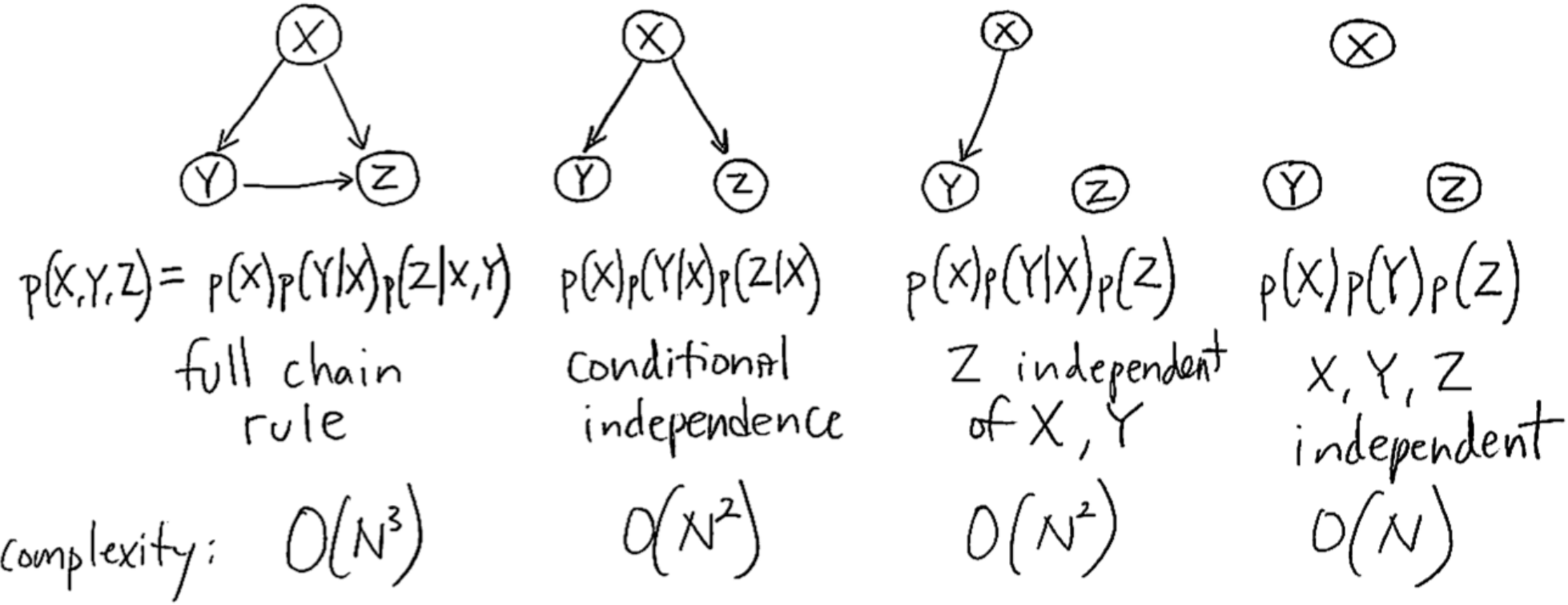
or *conditionally independent*

$$p(Z|X, Y) = p(Z|Y) \implies H(Z|X, Y) = H(Z|Y) \implies I(X; Z|Y) = 0$$

Graphical Models: “Information Graphs”

- gives a picture of a chain rule factoring of a joint probability distribution.
- nodes are the random variables in that joint distribution.
- edges are the conditional probability relations that appear in your chosen chain rule factoring.
- edges represent non-zero information links, i.e. where X is directly informative about Y i.e. $p(Y|X,\cdot) \neq p(Y|\cdot)$.
- They point from *condition* $\rightarrow$ *subject*.
- if the joint probability factoring can be simplified (due to independence) relative to the general chain rule, that should be reflected in the information graph as *missing edges* (some nodes are not directly connected).

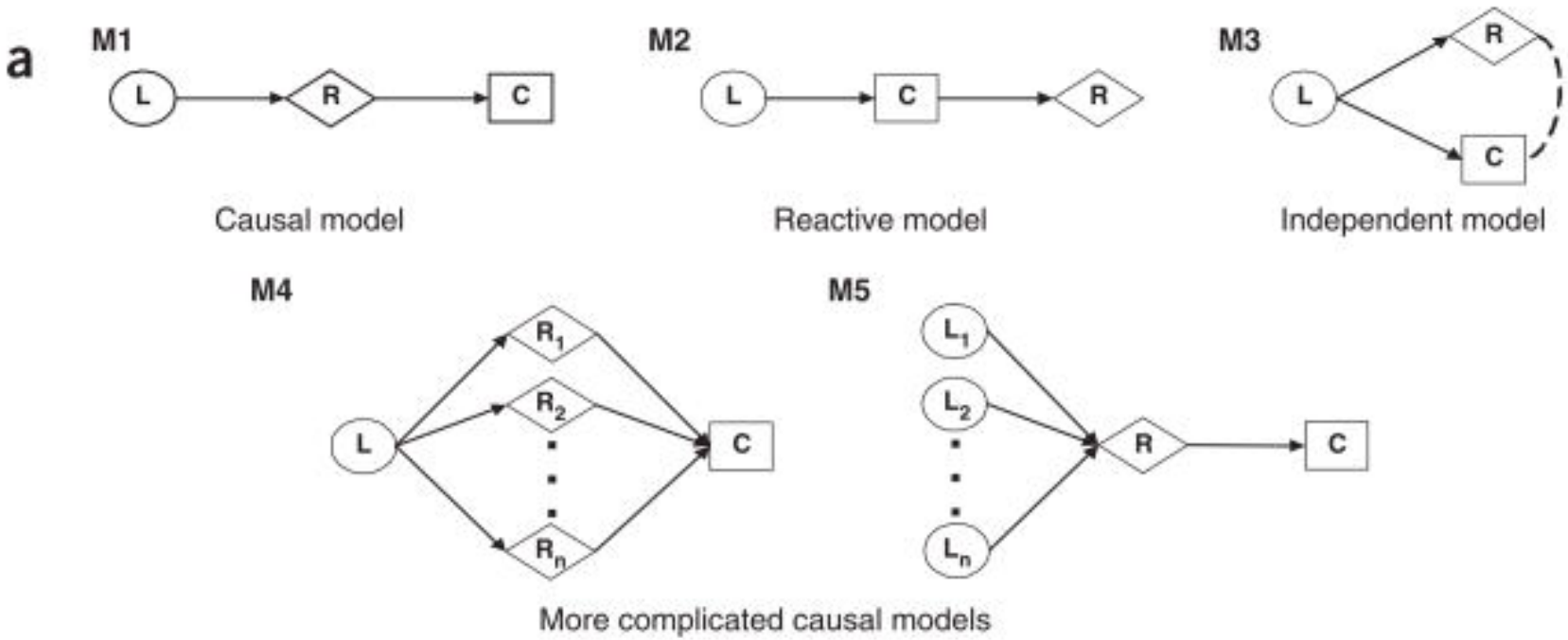
Information Graphs for Three Variables



Missing edges correspond to *zero mutual information* (given the other dependencies).

Example: Causality Analysis (Schadt et al. Nat Genet. 2005)

- consider three interacting factors: SNPs (L), gene expression levels (R), and clinical traits (C).
- generate population variation, e.g. by crossing mouse breeds with big variations in C, R, and looking at F_2 with recombined L.
- SNPs "anchor" the causality analysis: SNPs can cause R and C, but not vice versa.
- Test for non-zero edges via $I(L;C|R)$, $I(L;R|C)$, $I(R;C|L)$.



Empirical Information Tests of Causality

- is SNP L causal for gene expression level R?

$$\Delta I_e = \overline{L}_e(R|L,\ldots) - \overline{L}_e(R|\ldots)$$

- is SNP L causal for clinical trait C?

$$\Delta I_e = \overline{L}_e(C|L,\ldots) - \overline{L}_e(C|\ldots)$$

- is SNP L causal for clinical trait C when R also used in training? (Yes in *independent model*; No in *causal model*).

$$\Delta I_e = \overline{L}_e(C|L,R,\ldots) - \overline{L}_e(C|R,\ldots)$$

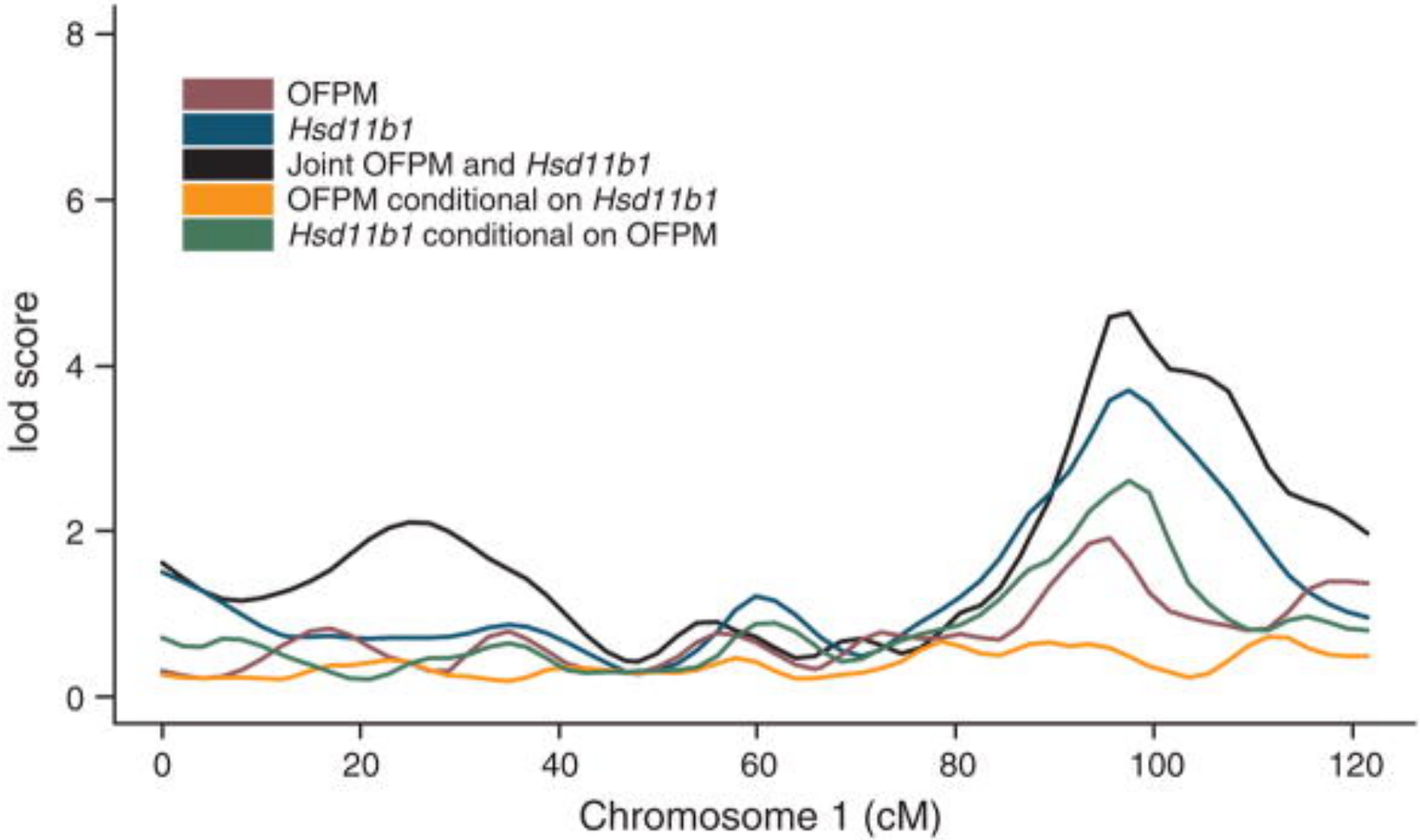
Note: total strength of evidence reported as $n\Delta I_e$.

mouse SNPs vs. expression vs. Obesity Study

Schadt et al. Nat Genet. 2005

- 111 F_2 mice from BXD cross of inbred mouse strains
- L: genome-wide SNP markers genotyped in these mice
- C: obesity clinical trait: omental fat pad mass (OFPM)
- R: genome wide expression dataset (liver)
- 4400 genes showed significant differential expression
- 440 expression traits for which SNPs had predictive value
- 4 major QTL peaks for predicting OFPM

Inferring Causality: SNPs vs. Expression vs. Obesity



$I(L; OFPM|\text{Hsd11b1}) \approx 0$ but $I(L; \text{Hsd11b1}|OFPM) \gg 0$ implies $L \rightarrow \text{Hsd11b1} \rightarrow OFPM$, with *no* direct edge from $L \rightarrow OFPM$.

Four Problems, One Solution

- k-means clustering
- motif discovery
- protein-DNA interaction analysis
- data mining of genetics + expression + clinical traits data to discovery causal pathways

Four rather different problems, but all solved by exactly the same machinery -- because information metric is totally general, to *any* problem.

Unifies a wide variety of problems with a common solution, often much simpler to understand and use. For example a whole field of *causal inference* exists (nicely formulated by J. Pearl's book *Causality: Models, Reasoning, and Inference*, 2000), but one can understand this as just another subcase of *information graphs*.

4. Computational Experiment Planning

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

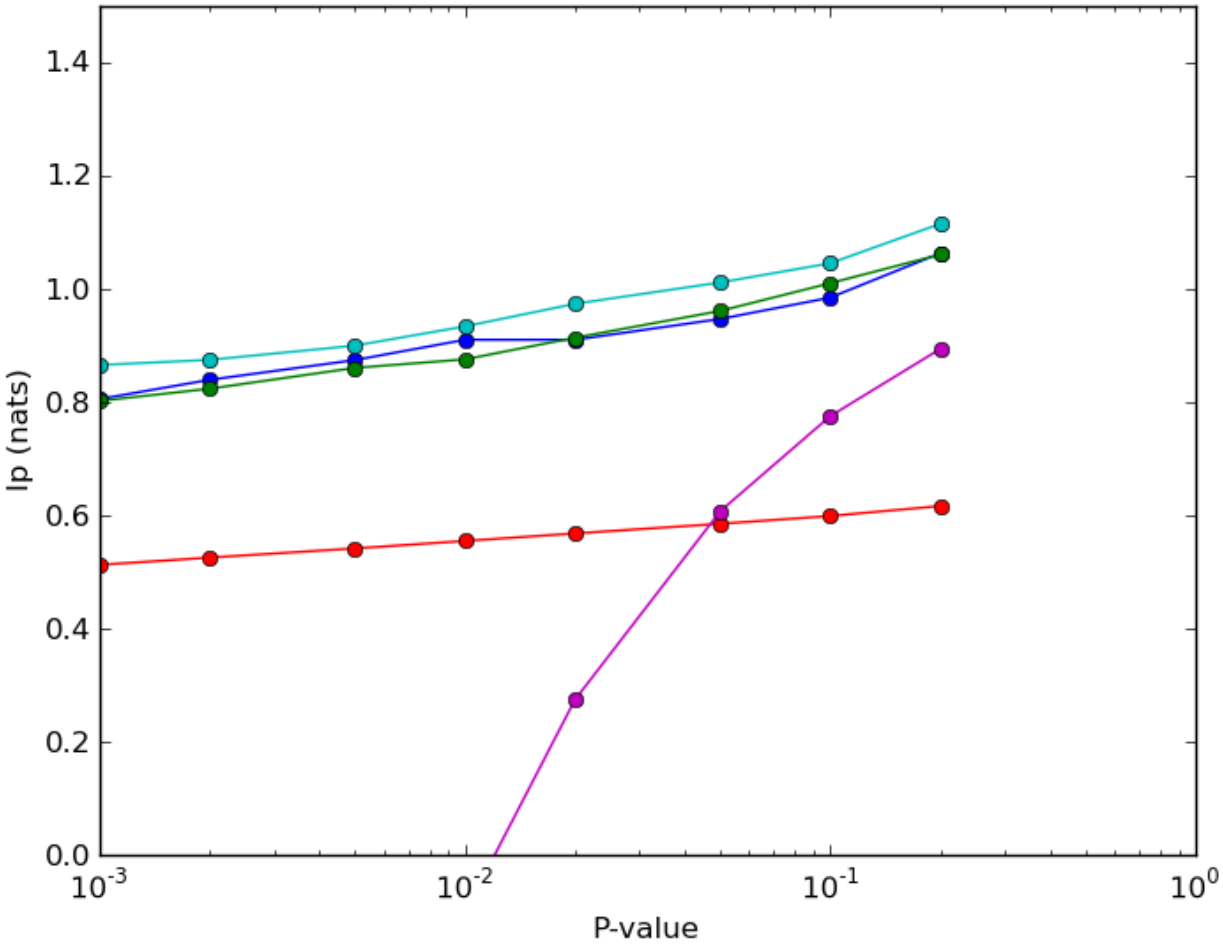
- The Law of Large Numbers guarantees convergence as $n \to \infty$, $\overline{I_p}(\Psi) \to D(\Omega||\Psi)$, the *relative entropy*, a standard information theory measure. Specifically, it guarantees a probabilistic lower bound on D with confidence ϵ:

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

- 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.

Resampling Accurately Estimates I_p Lower Bound



(computed for the Poisson Distribution)

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 outcomes 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 outcome α would be.

Expectation Potential Information

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

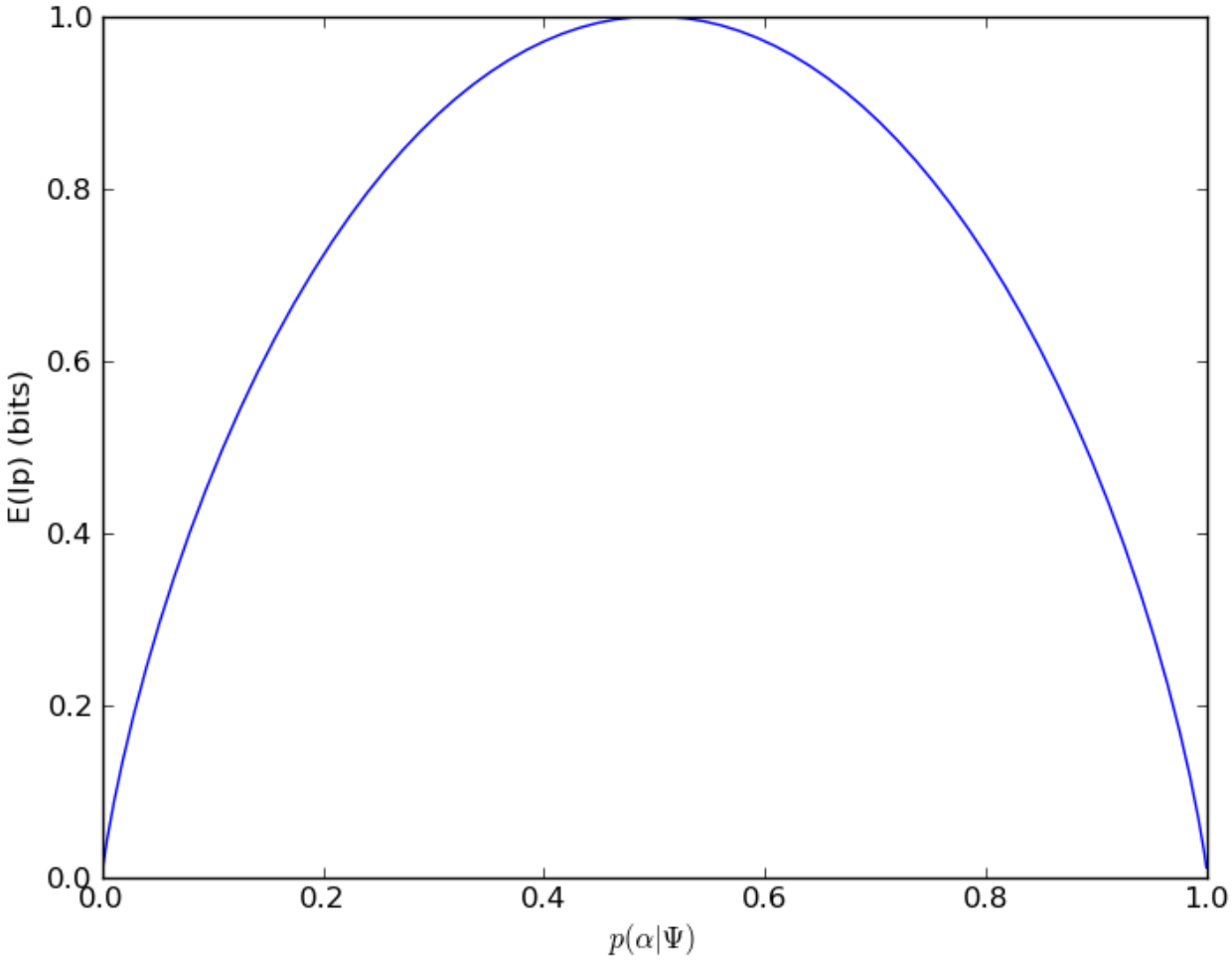
$$E(I_p) = \sum_\alpha p(\alpha|\Psi) D(\alpha||\Psi) = \sum_\alpha p(\alpha|\Psi) D\left(\alpha||\sum_\alpha \alpha p(\alpha|\Psi)\right)$$

- *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?

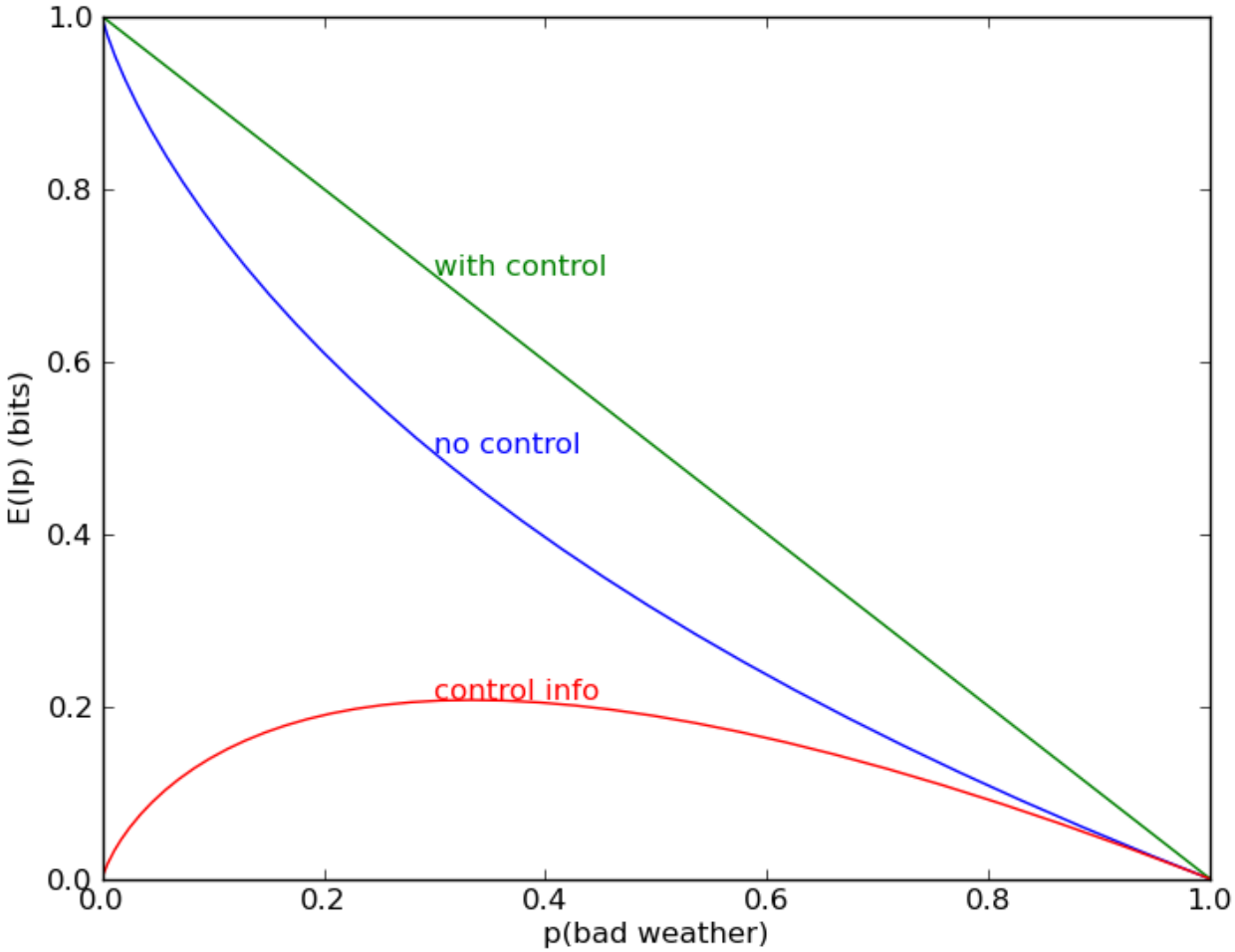
- Experiment: cross two plants $A \times B$, observe whether progeny grow. Assume 50-50 uncertainty = 1 bit of information.

- 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

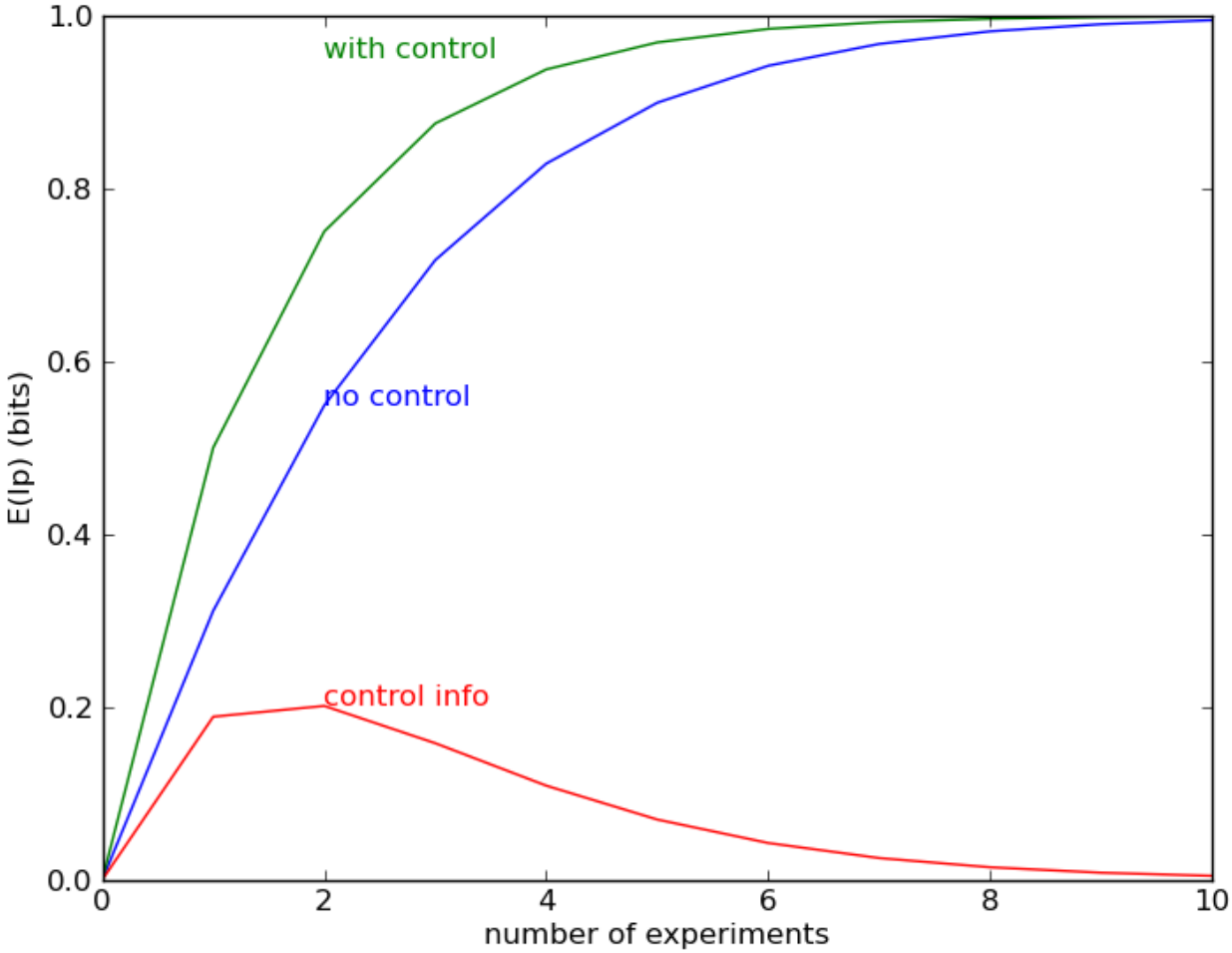
- Factors that vary independently over different repetitions of the experiment affect *the rate* of information production but not the *total* information capacity.

- 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).

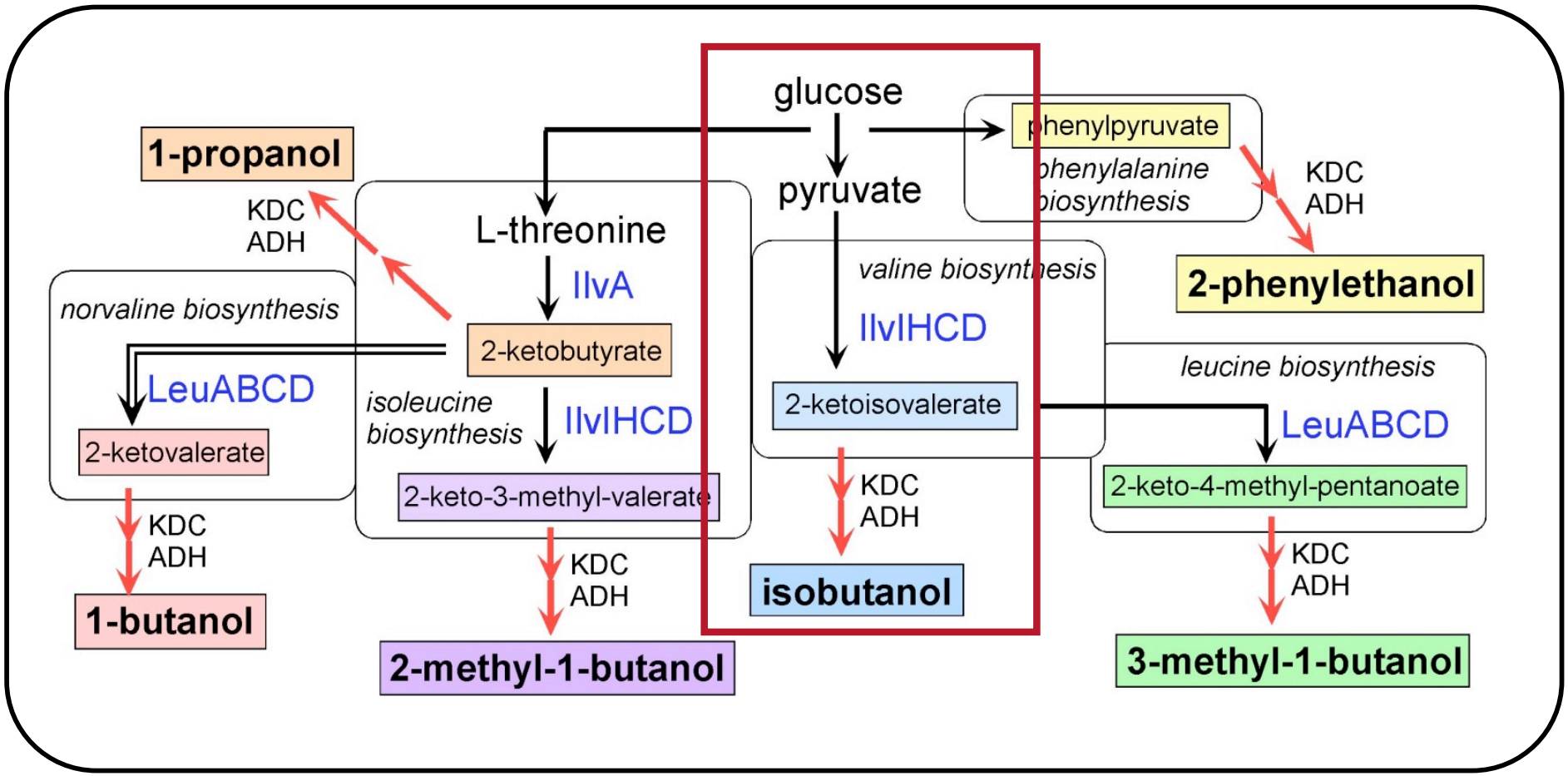
Phenotype Sequencing: identifying the genetic causes of a phenotype directly from sequencing of independent mutants

Chris Lee
UCLA-DOE Institute for Genomics & Proteomics

Phenotypes vs. Causes

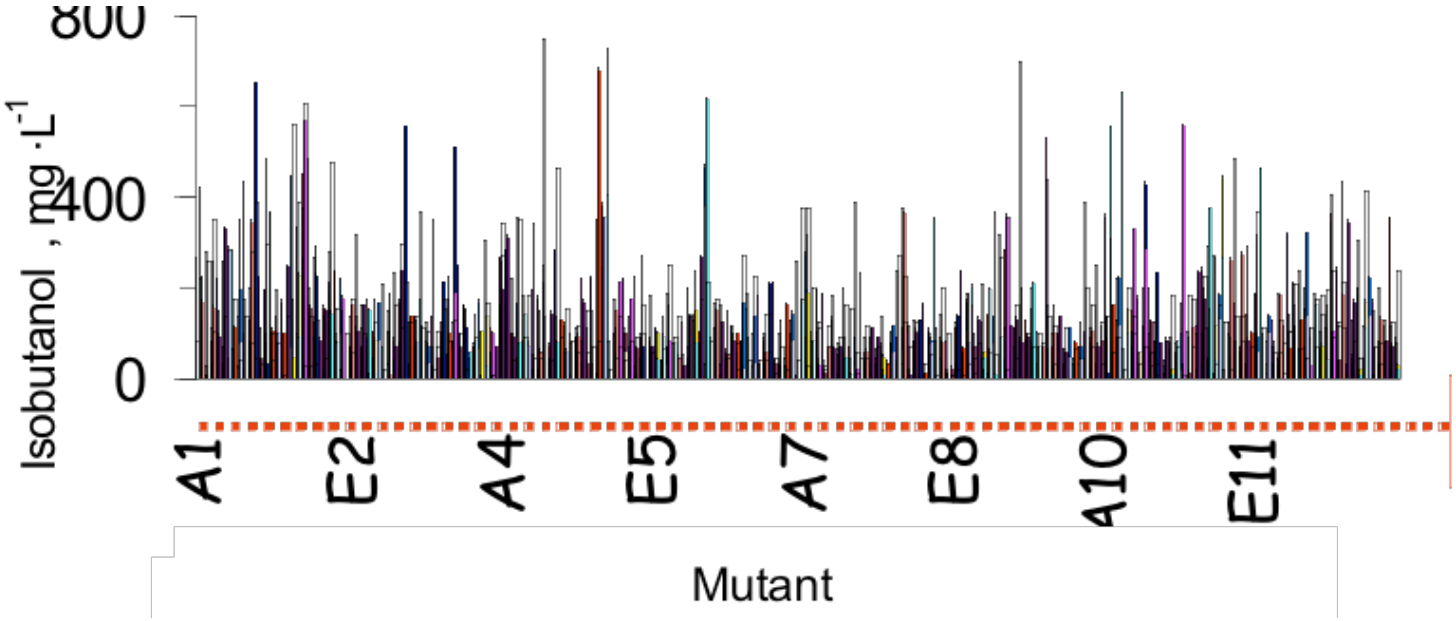
- If a strain with an interesting phenotype contains many mutations, it can be laborious to identify which one is the dominant cause, and which mutations are irrelevant.
- Easier for naturally evolved strains (10-20 mutations), much harder for mutagenized strains (50 - 100 mutations / genome).
- *mutagenesis*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

Liao Lab Pathways for C4, C5 Alcohol Synthesis



Atsumi et al. **Nature 2008**

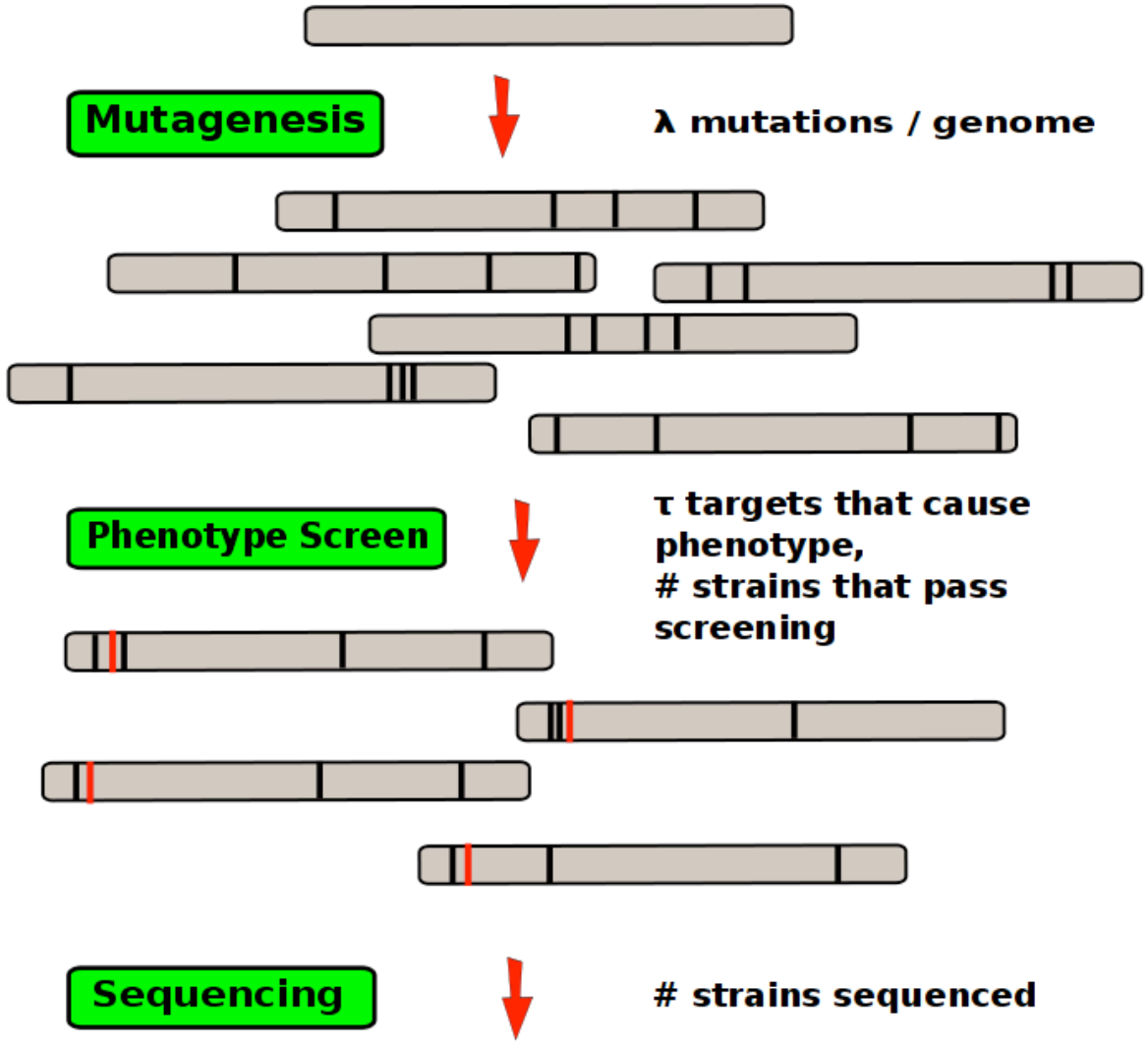
Liao Lab High-Throughput Screen For Increased Isobutanol Production



NTG mutagenesis followed by screening for increased tolerance (reduced toxicity) to isobutanol and increased isobutanol production

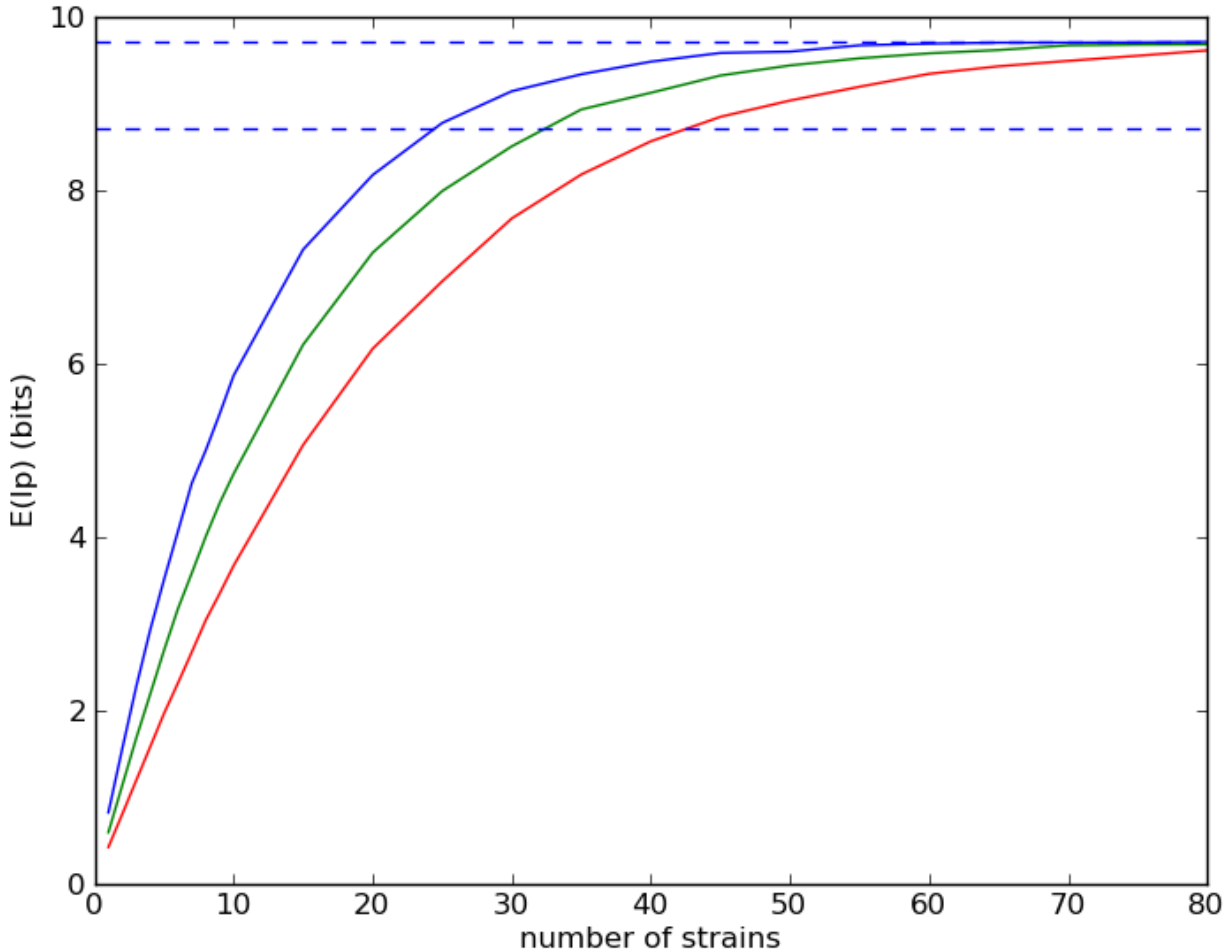
Proposal: Phenotype Sequencing

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.

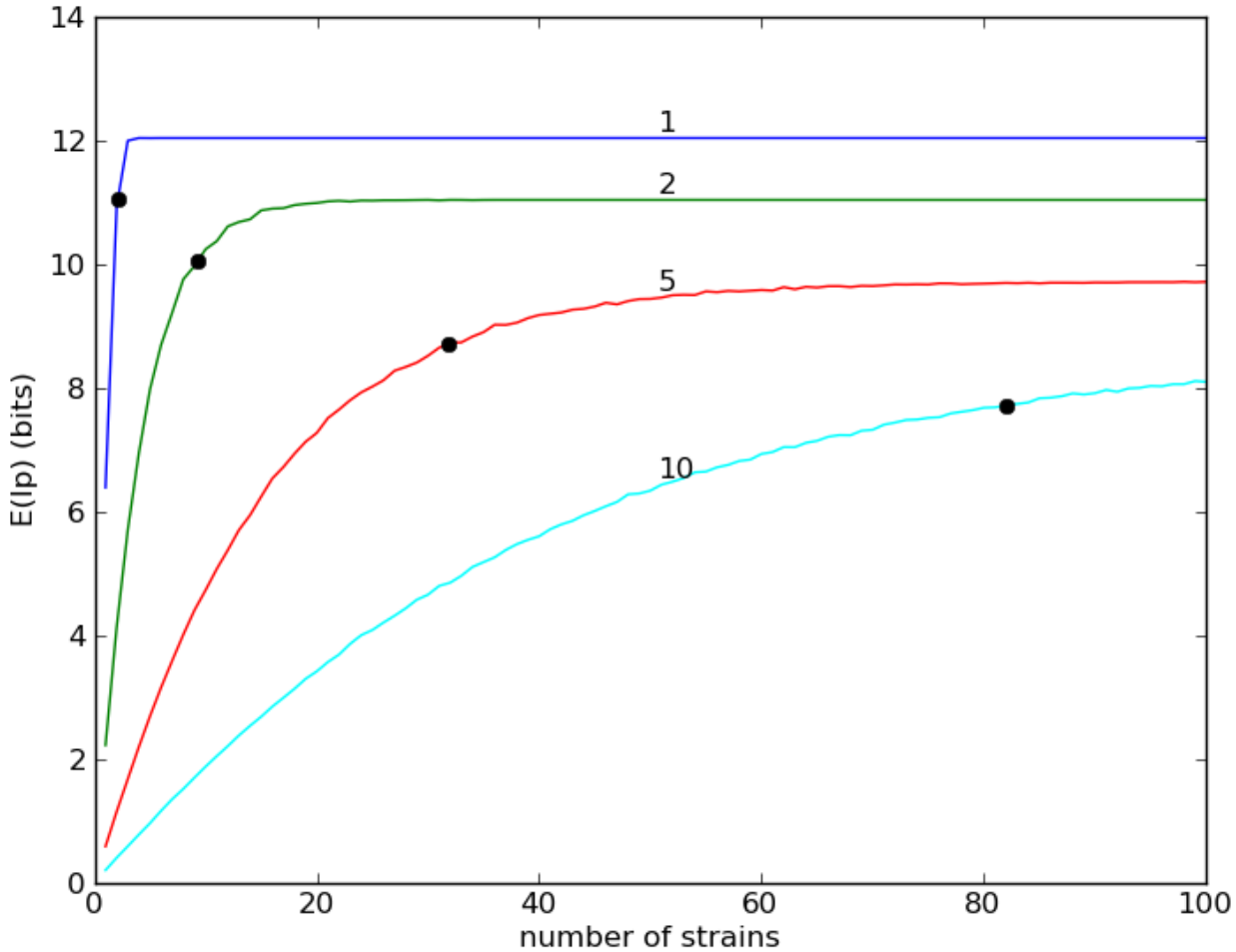


ATTACGAGGGATCCTATGACGC...
ATACCGAGGGATCCTATGACGC...
ATTACGAGCGATCCTATGACGC...
ATAACGAGGGATCCTATGACGC...

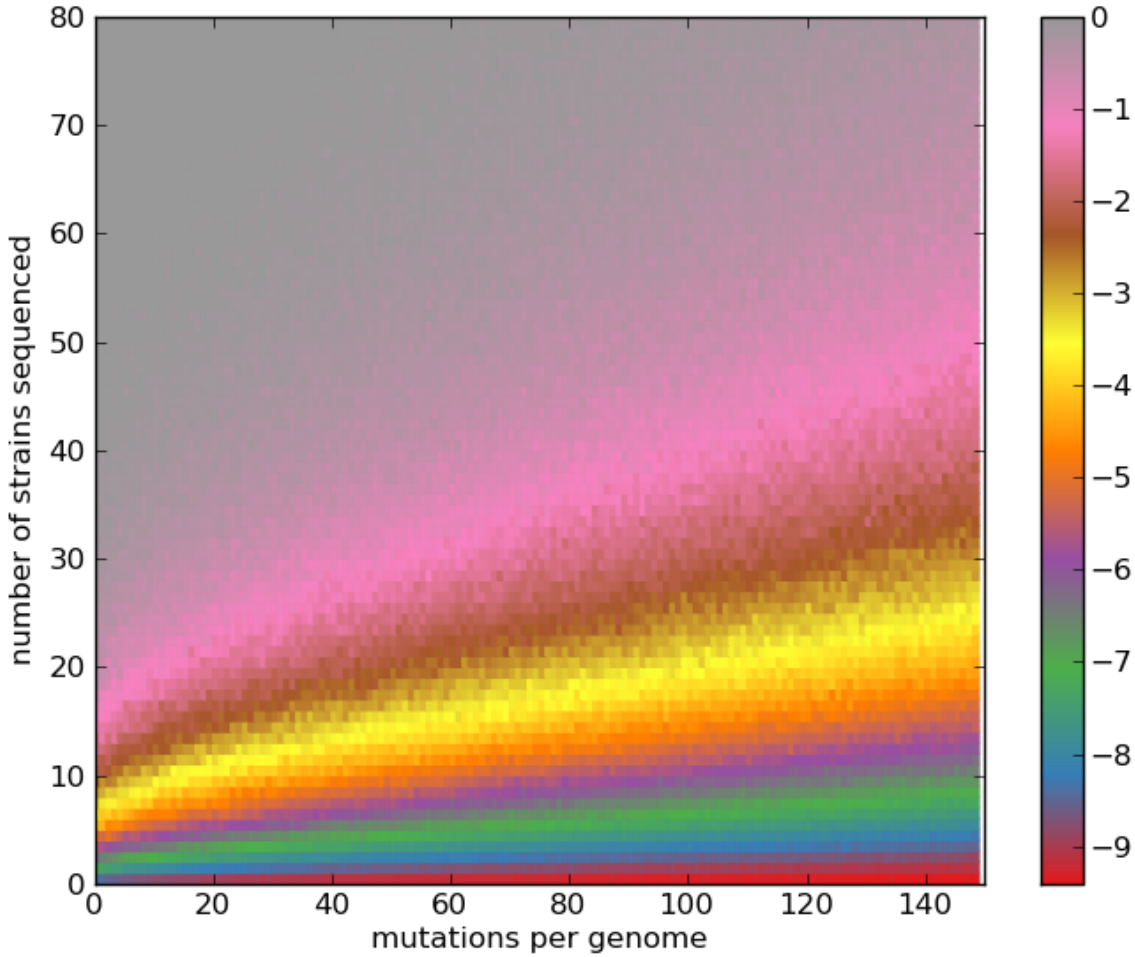
Effect of Mutagenesis Density



Effect of Number of Target Genes



Information Yield of Phenotype Sequencing



“Phenotype Sequencing”

- This approach should work well with the number of isobutanol-tolerant mutants available (80).
- 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.

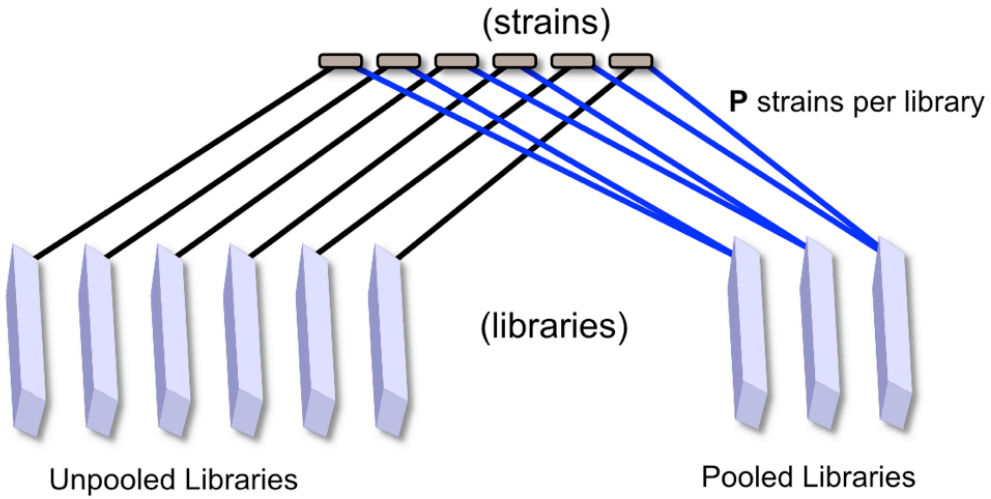
How to Make Phenotype Sequencing Economical

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.

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

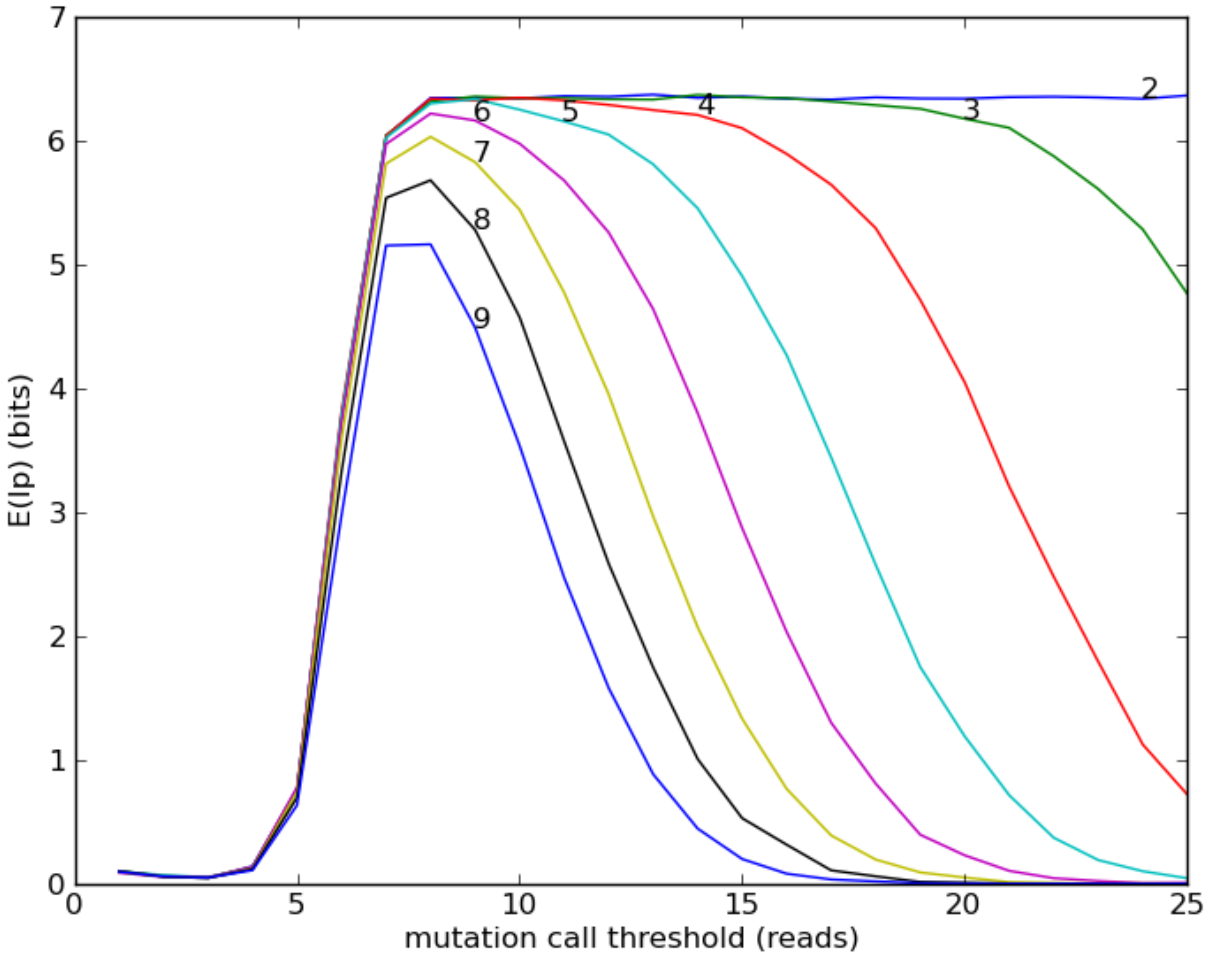
mutation expected
as **1 - ϵ** of reads per library

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).

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 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

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

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.

Phenotype Sequencing Conclusions

- computation allowed us to simulate many aspects of experiment design to understand where the sweet spot is.
- expectation information metric captures many aspects of design (e.g. depth of coverage, number of strains, mutagenesis density, degree of pooling) because it is fully general.
- an example where a new kind of genomics experiment was designed purely computationally.
- experiment worked on the first try.

Example: RoboMendel

- Robot scientist shown same initial observation that Gregor Mendel saw: pea plant with white flowers (instead of the usual purple).
- Selects experiment with highest expected information yield.
- Updates his “genetics model” based on the experimental results.
- Rinse, lather, repeat $\rightarrow$ discover all of classical genetics.
- Simplifying assumption: the only experiments RM can do are *genetic crosses*, so the set of all possible experiments is easily enumerable.

RoboMendel Sees a White Flower...

- Define RM's scope as *heritable variation*, i.e. "genetics".
- Initial model: *species* as separate peaks in observation space
- Like Father Like Son: each child is drawn from same species (peak) as its parents.
- Interspecies crosses not observed to produce any progeny.

initial observations

$I_p = +83$

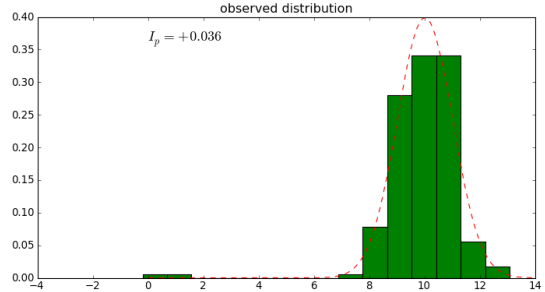
collect more data around discrepancy...

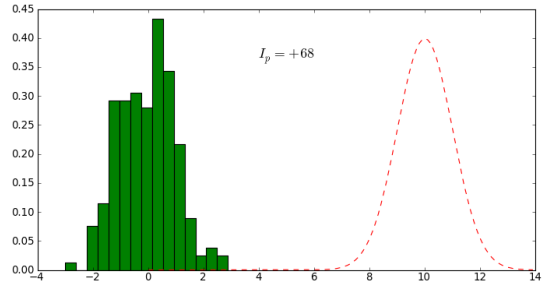
fit better model to observations...

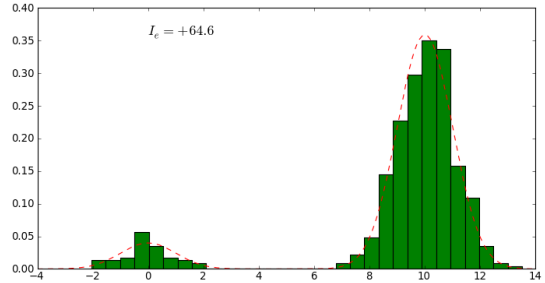
Measure improvement in empirical info on new obs...

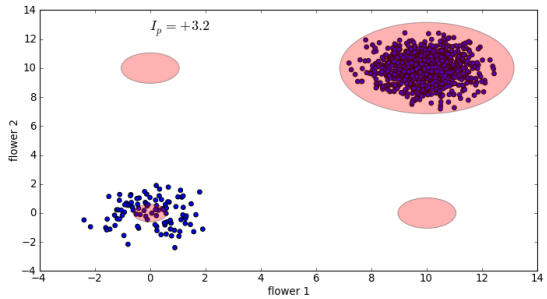
each plant either all Pu or all Wh

observed distribution









RoboMendel Initial Uncertainties

- $p(LFLS) \approx 0.999$: so far, no observed exceptions to Like-Father-Like-Son model, but not impossible...

- $p(Wh - heritable) = 0.5$: Is this even a heritable trait? We don't know. Wh looks different from the Pu species, but this might be *environmental* variation, not genetic.

- $p(same - species) = 0.5$: is Wh a member of the same species as Pu? We don't know.

- We confine ourselves strictly to the question of whether the metric behaves sensibly in its ranking of different experiments, i.e. we don't worry about how to come up with models etc.

Puzzle: What is Wh?

RoboMendel computes the following $E(I_p)$ values for the possible experiments:

Experiment	$E(I_p)$
Wh x Wh	0.5 bits
Wh x Pu	0.09 bits
Mouse x Lion	0.01 bits
Wh x Pu swap	1.2×10^{-6} bits
Pu x Pu swap	0 bits
Pu x Pu self-cross	0 bits

- $Wh \times Wh$ can reliably test $Wh - heritable$, and resolve that uncertainty, so it is picked as the highest information value experiment to perform.

- It conclusively shows $Wh \times Wh \rightarrow Wh$.

Wh is Heritable... ?!?

Experiment	$E(I_p)$
Wh x Pu	0.19 bits
Mouse x Lion	0.01 bits
Pu x Pu swap	0.001 bits
Wh x Wh	0.001 bits
Pu x Pu self-cross	0 bits
Wh x Pu swap	0 bits

- *Wh x Pu* can reliably test the *same-species* model, about which we have strong uncertainty, so it's chosen as the highest information yield.

- It yields progeny, confirming *same-species*, and they are all purple-flowered.

Asymmetric Inheritance?

Experiment	$E(I_p)$
Wh x Pu swap	1.0 bits
Mouse x Lion	0.01 bits
Pu x Pu swap	0.001 bits
Wh x Wh	0.001 bits
Pu x Pu self-cross	0 bits
Wh x Pu	0 bits

- *Wh x Pu swap* can reliably test the *one-parent* model, about which we have strong uncertainty, so it's chosen as the highest information yield.

- Again, the progeny are all purple-flowered, rejecting the *one-parent* model.

Another Try: A "Signal" Model

Experiment	$E(I_p)$
Hy x Wh	1 bits
Hy x Hy	0.98 bits
Mouse x Lion	0.01 bits
Pu x Pu swap	0.001 bits
Wh x Wh	0.001 bits
Pu x Pu self-cross	0 bits
Wh x Pu	0 bits
Wh x Pu swap	0 bits
Hy x Pu	0 bits

- *Hy x Wh* and *Hy x Hy* can reliably test the *transmission* vs. *LFLS* models, about which we have strong uncertainty, so it's chosen as the highest information yield.

- The results reject the *LFLS* model and fit the *transmission* model.

Alternative: “Pu undilutable”

- What if RoboMendel does not come up with the *transmission* model?
- *Pu undilutable*: *Pu* always beats *Wh*. After all, genetic inheritance is the ultimate homeopathy...
- Again, assign a prior $p(Pu - undilutable) = 0.5$ because fit previous obs better than other models, but not yet “tested”.
- Most convincing experimental test: dilute *Pu* in generation after generation of *Wh*, e.g. next step $Wh \times Hy$.
- $Wh \times Hy \rightarrow$ half white, half purple progeny. The results reject *Pu undilutable* and force RoboMendel to the *transmission* model.

Any More Recessive Traits?

Experiment	$E(I_p)$
Pu x Pu self-cross	1.64 bits
Mouse x Lion	0.01 bits
Pu x Pu swap	0.001 bits
Wh x Wh	0.001 bits
Hy x Hy	0.001 bits
Hy x Wh	0.001 bits
Wh x Pu	0 bits
Wh x Pu swap	0 bits
Hy x Pu	0 bits

- The new model predicts that if other recessive traits exist, a self-cross will quickly reveal them.

- Will discover additional recessive traits such as those found by Mendel: *Wrinkled* seeds; *White seed coats*; *Yellow seeds*; *Yellow pods*; *Constricted pods*; *Terminal flowers*; *Short* plants; etc.

RoboMendel Conclusions

- Even with very simplistic model assumptions, the $E(I_p)$ metric guides RoboMendel towards productive experiments that would indeed discover the basic principles of genetics just as Gregor Mendel did.

- Robust: e.g. if RoboMendel doesn't "think" of the *transmission* model but instead comes up with other models such as *Pu undilutable* or *inter-species hybrid*, the $E(I_p)$ metric will still drive him towards decisive experiments for testing these. These experiments in turn reveal the *transmission* model.

- Note: all we tested here was the experiment planning metric. We did not automate any aspect of the process of proposing new models, which would be required if you actually wanted an autonomous robot scientist!

- All the code for our calculations available at https://github.com/cjlee112/darwin.

- Manuscript with full details available at http://potentialinfo.blogspot.com.

Computational Experiment Planning Conclusions

- every possible next step (including computational data mining) has a *cost*
- therefore, treat every possible step as an *experiment*
- use computational experiment planning to assess information yields (per cost) for the possible next steps, then allocate effort to the best return-on-investment.

Three types of generalization

expand the reach of automated data mining:

- general metrics: work for all problems, and always work -- even when our model assumptions are wrong.
- extensible model structures: e.g. rather than implicitly assuming independence, explicitly model possible information graph structures and add edges as the data demand.
- computational experiment planning: don't just mine a fixed dataset. Answer the other side of the question: what *data* would be most valuable to generate. Close the loop!

An Apology and a Request

Due to an urgent grant deadline I have to jump back on a plane...

- But I would really like to follow up with anyone here who has questions or interest in using these kinds of ideas for their problems. Email me at **LEEC@CHEM.UCLA.EDU** (allow a week for the grant to get submitted so I can answer...)
- Slides will be on **potentialinfo.blogspot.com**
- Papers on this topic are on **https://selectedpapers.net/topics/experimentPlanning**

Bioinformatics Teaching Materials Consortium

- open-source repository for reusing, remixing and sharing teaching materials, especially *active-learning*
- concept tests for students to answer in-class with smartphone / laptop
- “cloud projects”: packaged as Virtual Machine Images
- problems, exercises etc.
- over 2000 questions, explanations, exercises, videos already
- software tools for in-class question system, remixing materials etc.
- not yet launched online

Described on **potentialinfo.blogspot.com**, very rough technology demo online at **teachpub.org**. Contact me if you’re interested in this effort or in trying out any of the materials.

Bioinformatics "Flipped" Course Results