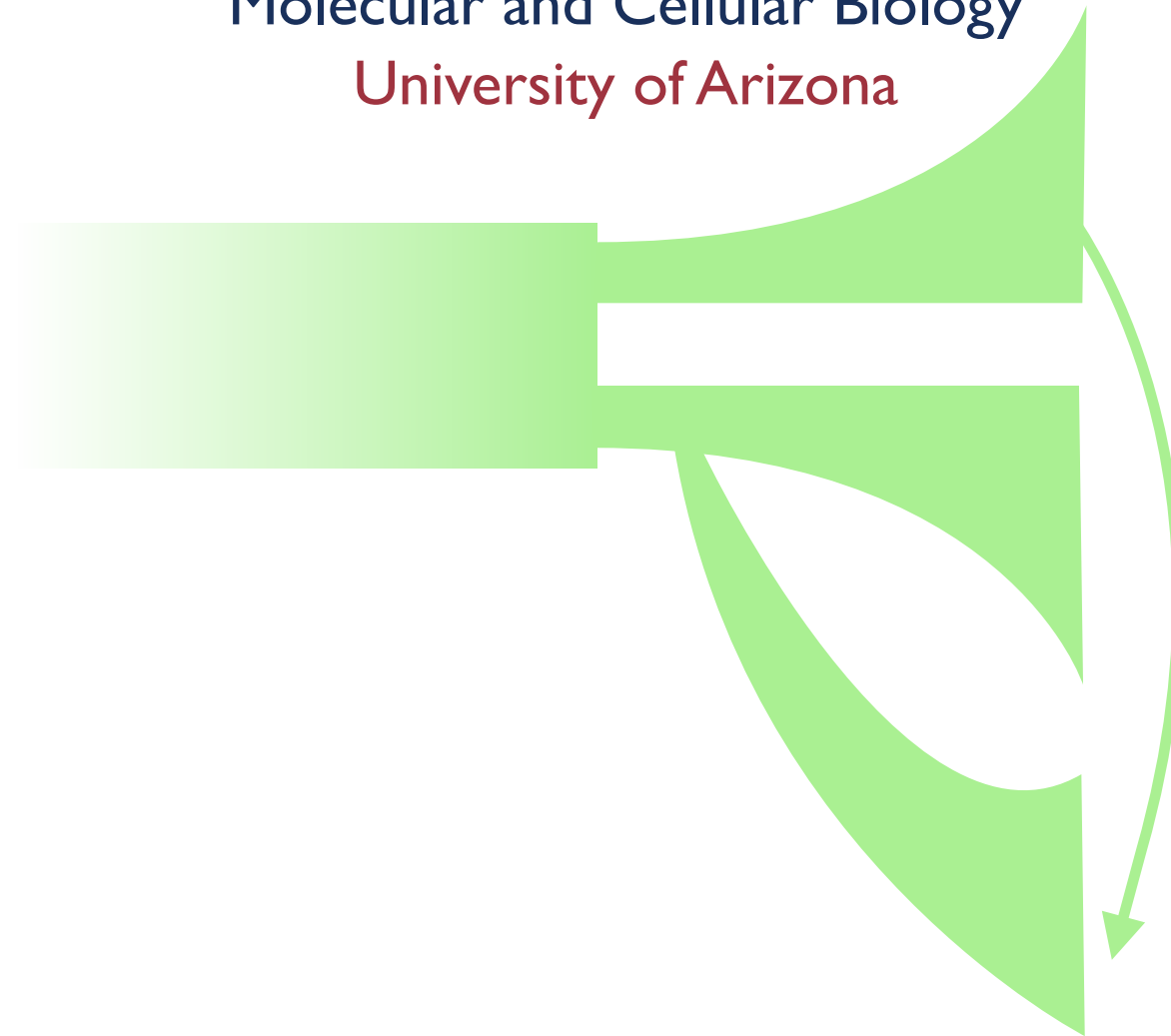


dadi: Diffusion Approximations for Demographic Inference

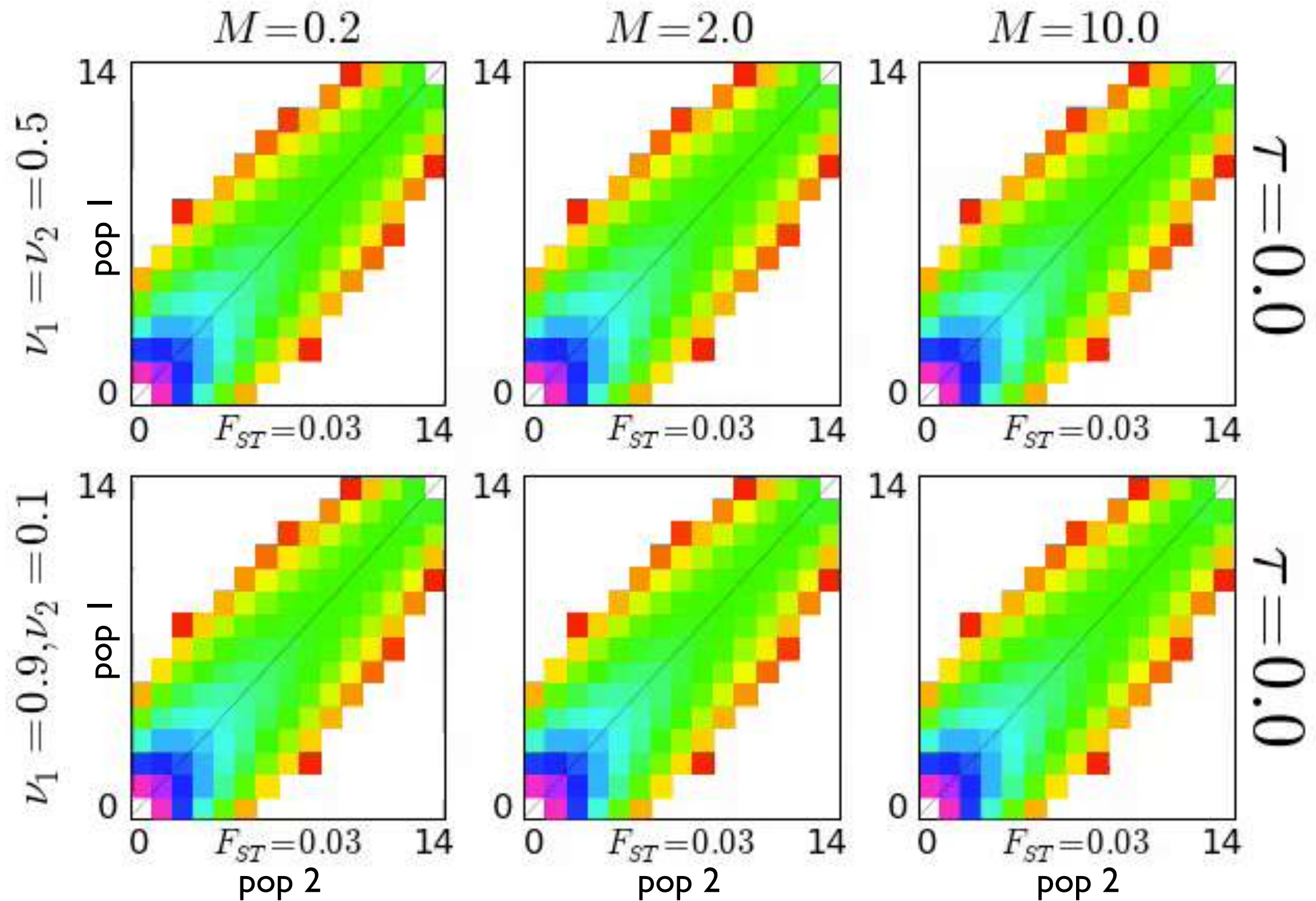
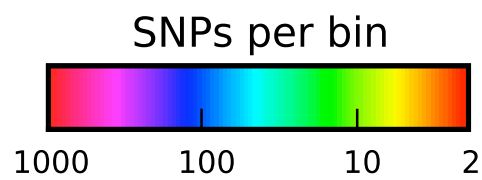
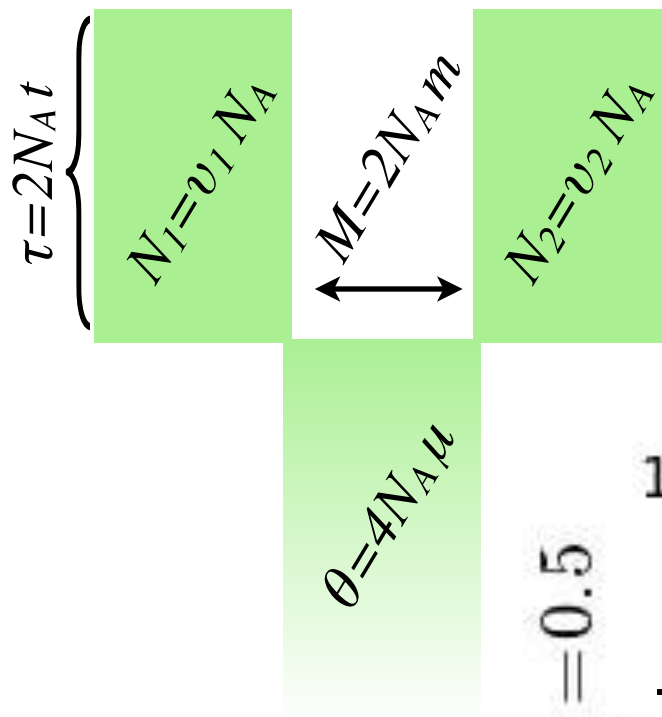
Ryan Gutenkunst

Molecular and Cellular Biology
University of Arizona



<http://bitbucket.org/gutenkunstlab/dadi/>
<http://groups.google.com/group/dadi-user>

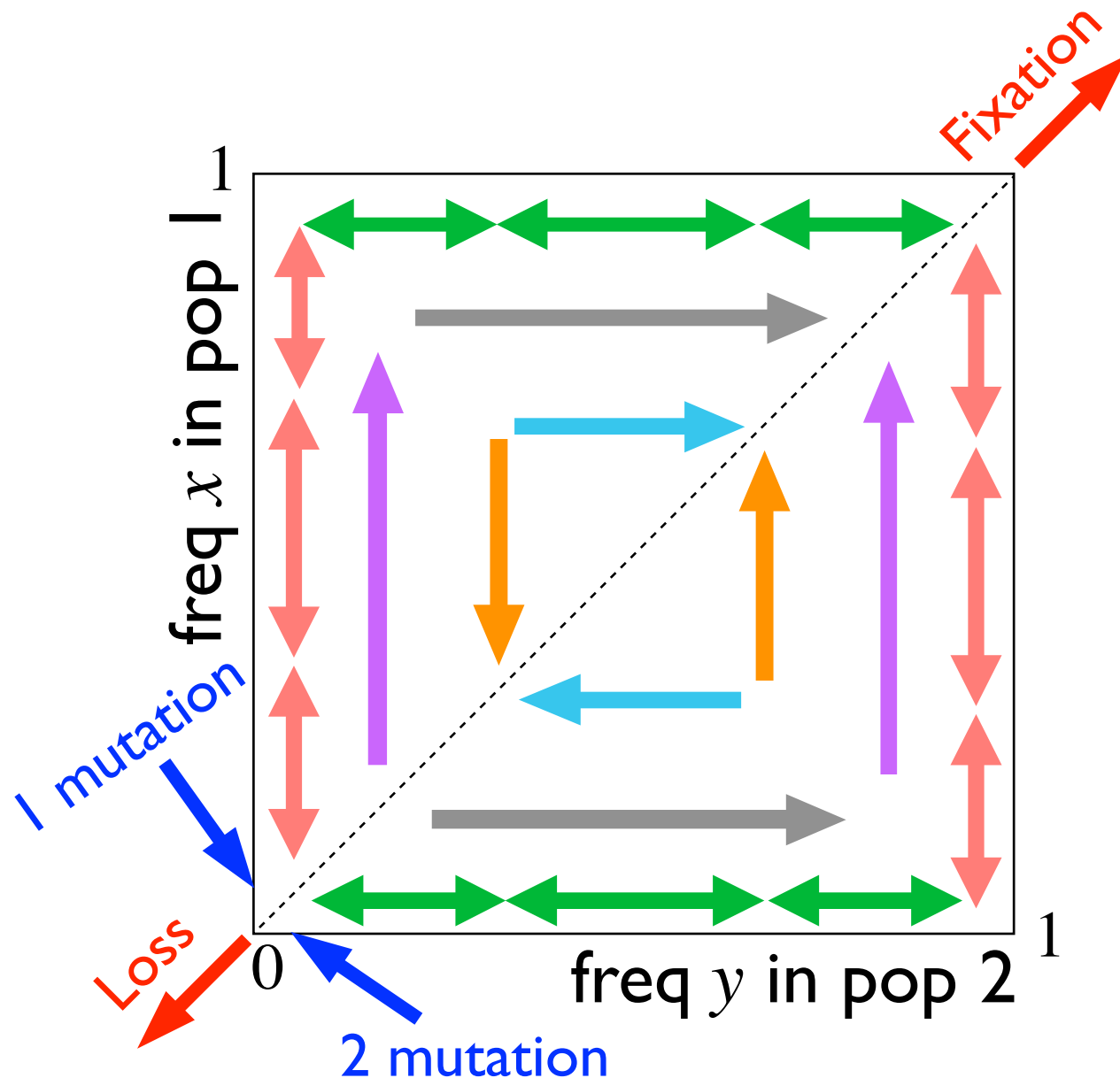
Frequency spectrum gallery



Diffusion simulation of joint AFS

$\phi(x,y,t)$: density of SNPs at freq x in pop 1 and y in pop 2.

$$\begin{aligned} \frac{\partial \phi}{\partial \tau} = & \frac{1}{2} \frac{\partial^2}{\partial^2 x} \left[\frac{x(1-x)}{v_1} \phi \right] - \frac{\partial}{\partial x} \left[\left(M_{1 \leftarrow 2}(y-x) + \gamma_1 x(1-x) \right) \phi \right] \\ & + \frac{1}{2} \frac{\partial^2}{\partial^2 y} \left[\frac{y(1-y)}{v_2} \phi \right] - \frac{\partial}{\partial y} \left[\left(M_{2 \leftarrow 1}(x-y) + \gamma_2 y(1-y) \right) \phi \right] \end{aligned}$$



Splittings

Pop 2 diverges from pop 1:

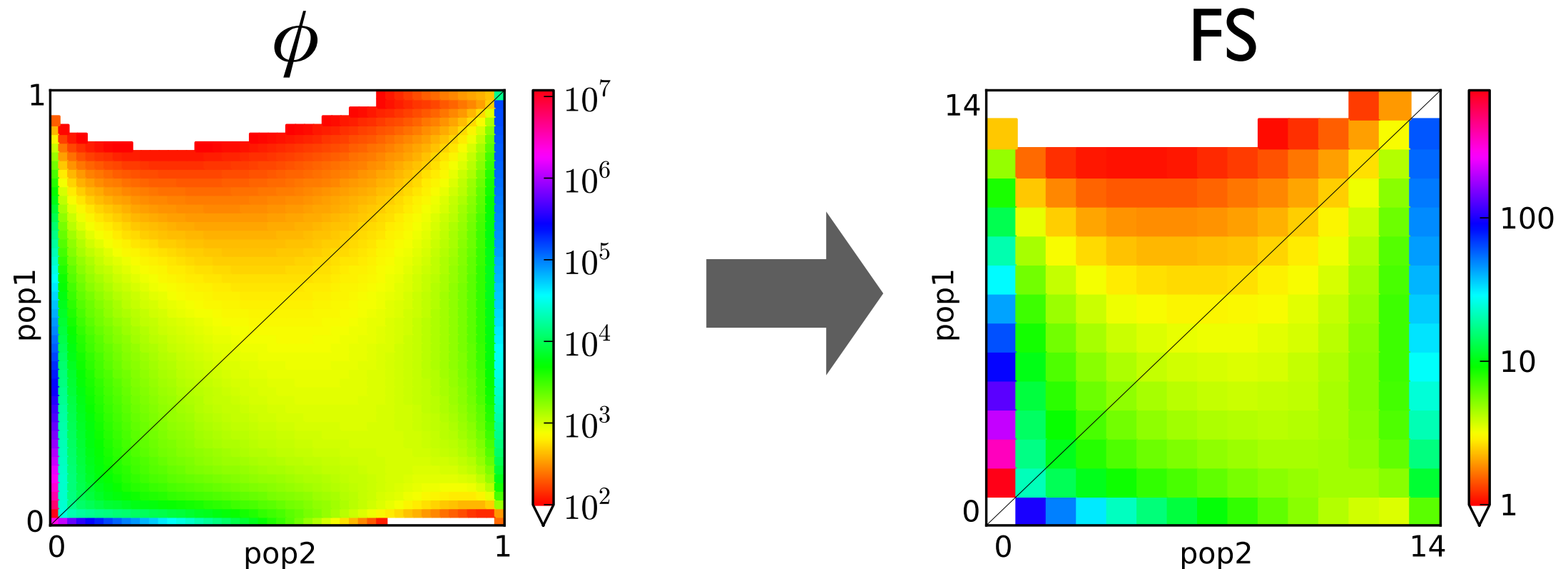
$$\phi(x,y) = \phi(x) \delta(y-x)$$

Numerical solution via
alternating direction implicit
finite-difference method

ϕ to spectrum to likelihood

$$FS[i, j] = \int_0^1 dx \int_0^1 dy \binom{n_1}{i} x^i (1-x)^{n_1-i} \binom{n_2}{j} y^j (1-y)^{n_2-j} \phi(x, y)$$

... can also model (some) ascertainment



$$\text{log-likelihood} = \log \left[\prod_i^{n_1} \prod_j^{n_2} \text{Poisson}(\text{drawing Data}[i, j] | FS[i, j]) \right]$$

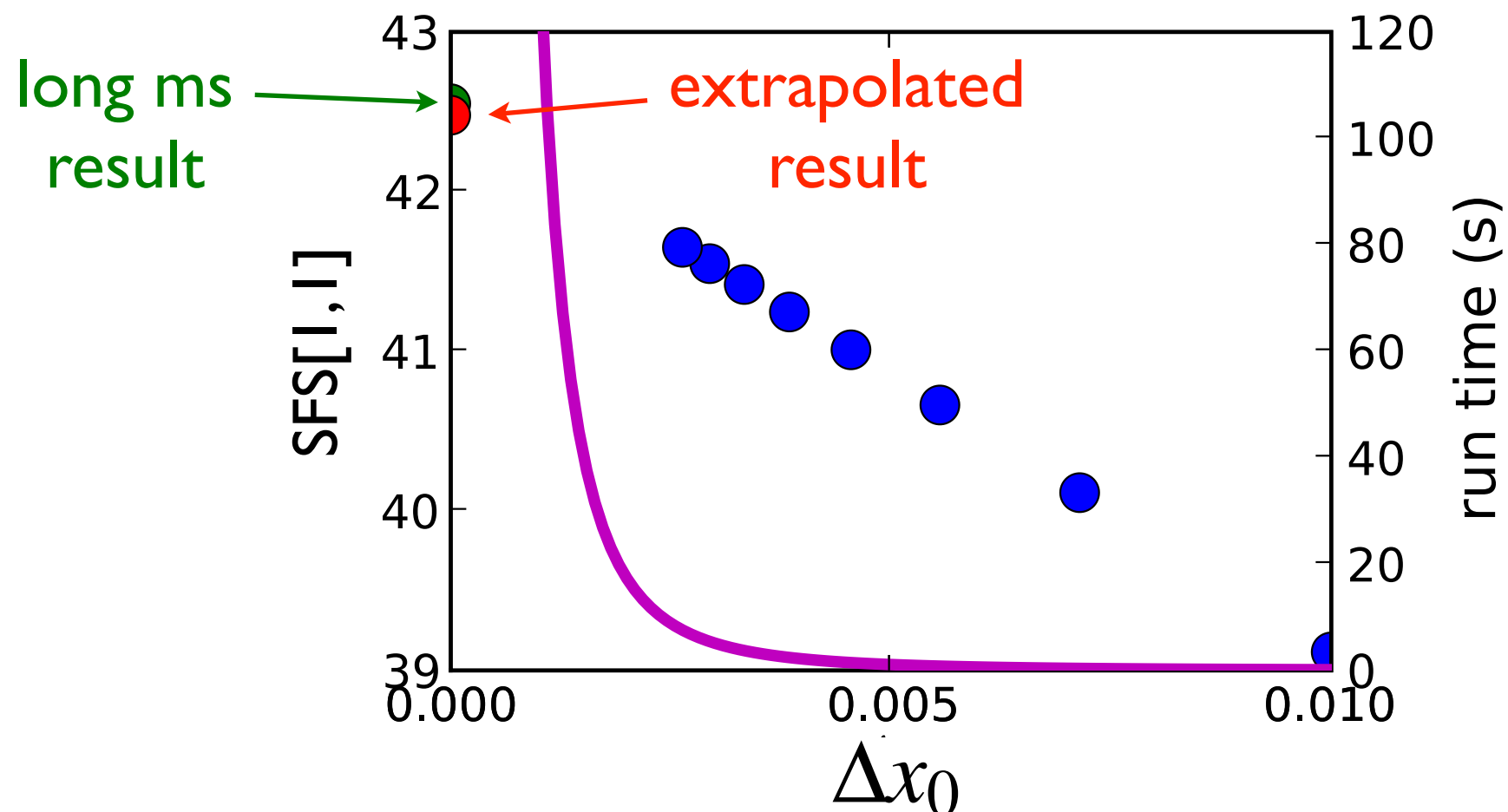
... assuming no linkage

Overcoming finite gridsize

Run time scales as $(\# \text{ grid points})^P$.
(e.g. $100 \times 100 \times 100$ grid = 10^6 points.)

Solution: Richardson extrapolation

$$\log \text{Calc}[i,j] = \log \text{Actual}[i,j] + a\Delta x_0 + b\Delta x_0^2$$



Specifying grid size

- Usually called `pts_1` in scripts.
- Generally want quadratic extrapolation, so `pts_1` should be a list of 3 elements.
- The smallest value should be larger than the largest dimension of your AFS.
- For example, if you have sample sizes of [14, 20, 50] individuals, your AFS will have size [29, 41, 101]. A good setting for `pts_1` might be [120, 130, 140].
- If your model involves small population sizes, high migration rates, or strong selection, you get warnings that extrapolation has failed. In that case, you should try increasing `pts_1`.

Parameter optimization



Parameter optimization

- Parameter optimization is an art, not a science. No algorithm can be guaranteed to converge to the true maximum likelihood in general.
- Hence I always recommend multiple optimization runs from different starting points. (The `perturb_params` method helps with this.) You can be confident if you see the same maximum likelihood repeated several times.
- For example, we often run until the best 3 likelihoods found are all within 1% or 0.1% of each other.
- `∂a∂i` includes a few optimization algorithms.

Optimization algorithms

- `optimize_log`: Based on BFGS algorithm, which uses derivative information. Fast if your starting point is close to the maximum likelihood.
- `optimize_log_fmin`: Based on Nelder-Mead simplex algorithm, which doesn't use derivatives. Slower, but more robust.
- `optimize_grid`: Basic grid search. Very robust, but very inefficient.

Optimization bounds

- Certain parameter settings cause AFS evaluation to be extremely slow, so you should set bounds to avoid those ranges.
- Avoid small population sizes, so maybe set lower bound $\sim 1e-3$.
- Avoid long divergence times, so maybe set upper bound ~ 5 .
- Avoid high migration rates, so maybe set upper bound ~ 10 .

Implicit θ

- The overall genetic diversity of the populations is set by $\theta = 4N_a\mu L$. Here N_a is the ancestral population size, μ is the per-base mutation rate, and L is the length of sequence.
- It turns out that the optimal θ for any demographic model is easy to compute once the other parameters are set, so by default it isn't explicitly included in $\partial a \partial i$ models. In this case, you use the `_multinomial` methods.
- In some cases, you may want to hold the parameter θ fixed, which you can do.

Ancestral states

- Your inference will have the greatest power if you have ancestral states, to call derived versus ancestral alleles.
- But even with a good out group (e.g. human vs. chimp), you'll still have some misidentification.
- This can be corrected statistically (Hernandez et al. (2007)), but it's a little touchy.
- You can just fold the spectrum, and only consider minor vs major alleles.
- But now we typically just misidentification as a model parameter.

Missing data

- If your data are incompletely called, not all SNPs may be called for all individuals.
- If only a small portion of SNPs are missing, they can be dropped from the analysis (and L adjusted).
- But if this is a common problem, our current solution is to *project* the SNPs downward to a common sample size. You then discard SNPs with fewer calls than this smaller sample size.
- The projection is essentially averaging over all resamplings of a smaller number of samples from your called samples.

Parameter uncertainties

- The most robust way to estimate parameter uncertainties is via bootstrap.
 - Divide your data into large ~unlinked blocks.
 - Generate many resampled data sets from those blocks.
 - Fit those resampled sets to estimate confidence intervals.
- Bootstrapping this way is very computationally expensive.
- Recently, we've used an approximation based on Godambe information, which is much faster to compute.

Suggested workflow

- Don't jump in by fitting the most complicated model you can conceive!
- Start by fitting very simple models to single populations.
 - This will both give you quick experience running $\partial a \partial i$ and insight into what demographic events happened in the past to your population.
 - For example, if your 1D fits indicate population growth, make sure that's included in your 2D fits.
- Use residual plots and comparison of likelihoods to judge which parameters to add for next model.
- This can be formalized in likelihood ratio tests.

Exercise time!