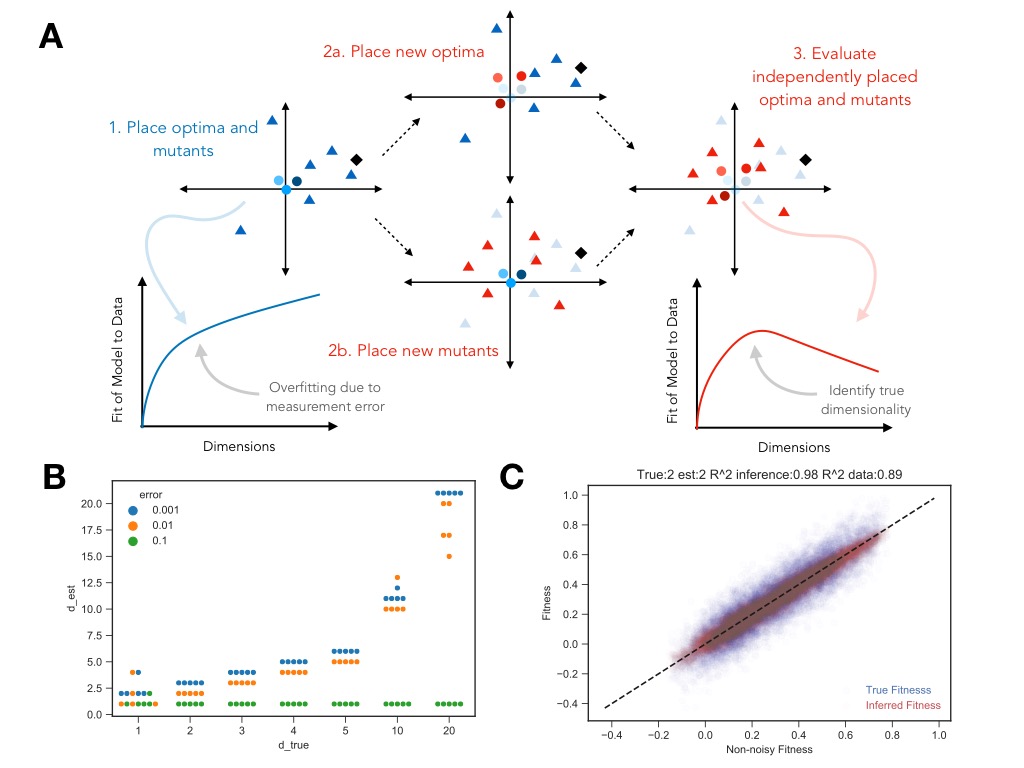
****

**Figure 1. Fisher’s geometric model and approaches to infer dimensionality. A.** In Fisher’s geometric model, phenotypes are represented as orthogonal dimensions and mutations are represented as vectors in phenotype space. The fitness of each mutant is determined by its the distance from the origin, relative to the distance from the origin to the ancestral genotype (black diamond). Thus, everything the same distance away from the origin as the ancestor is neutral (on dashed line), and everything closer is adaptive. **B.** Estimating the number of dimensions using the distribution of fitness effects assumes a fixed mutation distribution and that the ancestor is optimally fit. **C.** Estimating the number of dimensions using epistasis data assumes a mutation distribution and also that the effect of mutations on phenotype is additive, i.e. the vector representing a double mutation is the sum of the vectors of the subsequent single mutations. **D.** Our approach. We use subtle environmental perturbations to triangulate mutants and conditions. Relative fitness measurements give us information about the distance between a mutant and the condition’s optimum, relative to the distance between the ancestor and that optimum.

****

**Figure 2. Cross validation scheme and simulation results. A.** A representation of the cross validation scheme. First, we place a “training” set of optima and mutants in each d-dimensional space. Here, the fit of the model to the data will continuously increase as we add more dimensions. Second, we hold this space constant and place new optima using the information of the mutants in the training set and separately place new mutants using the optima in the training set. Finally, we evaluate the predicted fitness of the “test” set, which consists of the new, independently-placed optima and mutants. The fit of the model with the test set should decrease at some critical number of dimensions, representing where overfitting of the training set occurred, and serving as an estimate for the true underlying dimensionality. **B.** Comparison of the estimated dimensionality and true dimensionality across various simulations with varying levels of error. **C.** Visualization of the fit of the model to true underlying simulated data compared to the “true” fitness with measurement error. Shown in red is the fit of the inferred fitness to the simulated data without noise (). Shown in blue is the fit of the true measured fitness with error to the simulated data without noise ().

Identifying the number of fitness-relevant phenotypes represented by a collection of mutants is equivalent to identifying the appropriate number of components (and its corresponding model) from SVD. One first instinct would be to select the number of components that fully captures the values we observe in the data (the one with the most components possible). However, this approach does not take into consideration that our observations are imperfect due to measurement noise, and there will be some components that are only fitting this noise rather than real biological signal. This phenomenon is known as overfitting in the statistical literature, and various methods have been proposed in order to select the model that balances (1) select a model that explains enough signal and (2) avoids fitting measurement noise.

One common approach to avoid overfitting is to place an explicit penalty on additional parameters, such that only models that provide a sufficiently large gain in explanatory power over the less complex ones are chosen. Several such criteria have been proposed for use in cases similar to SVD [CITE paper from Owen and Perry], though all of these suggestions fail to accurately identify the correct complexity in our simulated data (Figure 2 somewhere).

In order to estimate the phenotypic components of the mutants and the weightings of these components in each environment, we decompose the observed fitness data of mutants into two matrices: one representing the trait values of the mutants and another that represents the weightings of these traits for fitness in each environment using singular-value decomposition (SVD). SVD orders these component traits, such that the first component represents the best possible linear, single-component model for the observed data. Subsequently, the first and second component combine to produce the best two-component model.

Both the SVD and FGM-based approaches face the problem of overfitting, such that when we only consider model fit to the data, the models with the most parameters (maximum number of fitness-relevant phenotypes) appear to be the best (Figure 2 somewhere). This is a common problem for statistical model fitting in general [CITE some stuff], and the statistics literature has put forth many approaches for choosing an appropriate number of parameters, several of which we consider here with our simulated data.

To infer fitness components in the linear model, we rely on singular-value decomposition (SVD). This method dissects the matrix into two component parts: a matrix representing the trait values of the mutants and another matrix that represents the weightings of these traits for fitness in each environment. Moreover, component traits in SVD are ordered, such that the first component represents the best possible linear, one-component model (according to some metric “Frobenius norm”), the first and second component together represent the best possible linear, two-component model, etc.

We take two independent approaches to identify the number of fitness-relevant phenotypes. The first explicitly uses the framework of Fisher’s model and uses optimization techniques to solve for the best fitting phenotype space and dimensionality for the simulated fitness data. The second approach uses singular-value decomposition (SVD) to identify the number of factors represented in the fitness data by decomposing mutants and conditions into linear vectors. [do we need a figure for the two methods?]

This method dissects the matrix into two component parts: a matrix representing the trait values of the mutants and another matrix that represents the weightings of these traits for fitness in each environment

To infer the number of dimensions represented by the adaptive mutants, we use two independent approaches. The first relies on singular-value decomposition (SVD) to represent fitness data of mutants across many conditions in a low-dimensional form. We select the num

The second approach explicitly steps from Fisher’s model and relies on

[A geometric representation of this framework is Fisher’s Geometric Model (Fisher 1930). In this model, organisms are represented as fixed points in a *D*-dimensional phenotype space, with each dimension represented traits that are independent with respect to their effect on fitness. Fitness in a given environment is a function of this phenotype space. A commonly used function is the Gaussian, where there exists an optimal combination of traits in that environment, and fitness decreases with the distance from this optimum. Different environments are represented by different fitness functions, and in the Gaussian case, by different locations of optima and/or different widths (Figure 1). Thus, identifying the number of fitness-relevant traits is equivalent to inferring the number of “dimensions” if Fisher’s representation. Moreover, subtle environmental changes are depicted by small changes in the locations of the optima.]

A separate representation characterizes fitness as a linear function of traits, where each trait has some weighting

In order to break down the complexity of phenotypes, we imagine a group of single mutants for which we have measured their fitness in many subtly different environments. Each mutant’s fitness is defined by the relationship between that mutant’s phenotype, composed of a combination of many trait values, and the environment. This mapping from phenotype to fitness is environmentally dependent, with different combinations of traits, and different weightings of those traits, being more or less important in different environments. Thus, if we subtly change the environment that a particular mutant is in, we can subtly change the relative importance and impact of its trait values, and by extension, the fitness of the mutant. These fitness changes reveal the existence of these differing traits, and if enough subtle environmental perturbations are used, can reveal the number of these fitness-relevant traits.

A third approach to identify the correct model complexity is rooted in SVD and relies on knowledge about the amount of measurement error. In this approach, we directly estimate that

A third approach to identify the model complexity [maybe need a figure to explain this part?]

The ability to detect dimensions directly stems from the relative contribution of the signal represented by that dimension and the measurement noise that could obfuscate that signal. Even if measurement error were completely uniform and random, the noise would have particular substructure that would be detected by SVD. Thus, a third approach relies on identifying how strong the largest component of such structure would be, which represents the limit of detection for the true signal. If an estimate for the measurement error is known, we can simulate draws from this distribution, use SVD to identify the structure within this noise, and calculate the contribution of the largest component of noise. Repeating this for many draws, we can compute the expected size of this largest component and compare it to the size of the components from the measured (here simulated) data. This allows us to identify how many phenotypic components are primarily composed of real signal (those before this level of noise) and those that are hidden by the noise (those with contributions below the largest noise component). For simulated data with the same measurement error for all observations, we can also compare this expected level to that anticipated by theory [CITE]. (Also refer to figure 2 in here). With sufficiently low noise, this method also accurately identifies the correct model (figure 2 somewhere),

…. These two approaches allow us to accurately estimate the number of dimensions on our simulated data when noise is sufficiently low and uniform, but how do they perform when noise varies for particular mutants or conditions or if noise is too high? Do they allow us to be able to identify when we should be confident in the number of components we observe? What is the ratio of signal to noise that matters? [something introducing the idea of “dispersal” ]

As conditions and mutants are more different from each other, the signal

For a set level of measurement noise, detectability is determined by several

[To test our method for estimating the phenotypic space from fitness data, we perform a simulation study. First, we simulate data that fits our phenotypic model and then feed the corresponding data into our method to infer the phenotype space and number of fitness-relevant phenotypes. For simplicity and consistency with previous literature, we start with a Gaussian function of distance. Perhaps use the following to justify why you perform the simulations in the way you do, rather than saying, ‘for simplicity’: Because we are interested in detecting the relevant phenotypic differences that lead to differences in fitness, a relevant fitness function must have the property that traits with no differences between organisms cannot factor into the relative fitness differences between these organisms. An exponential function of squared distance (of which the Gaussian is) is only class of functions with this property (see SI).]

methods

We consider an explicit model of phenotypic evolution analogous to Fisher’s Geometric Model (Fisher 1930). Phenotypes are depicted as orthogonal axes in a *D*-dimensional space, with the number of dimensions, *D,* representing the number of traits possibly relevant in this space. Organisms are depicted as points in this *D*-dimensional space, with their position determined by the combination of phenotypes represented by that particular organism. An organism’s absolute fitness in a particular environment is determined by a function of its distance from an optimal phenotype.

Our implementation of the model makes some inherent assumptions about the fitness function that determines fitness in each environment. First, it assumes that each trait contributes independently to fitness in any given environment – this can be done by transforming the space if considering a single optimum (Martin & Lenormand 2006), but is not generally true if the interactions between traits differ between conditions. Second, each trait is rescaled such that they have equal effect on fitness in a given condition. We assume this scaling holds for all conditions. Furthermore, we assume that absolute fitness in a given environment is a Gaussian function of the distance between an organism’s location in phenotype space and the location of the optimum for that particular environment:

where is the height, is the variance, represents theth coordinate of theth mutant, represents the th coordinate of the th optimum. This function is typically assumed for many investigations of Fisher’s model [cite a bunch] due to analytical tractability. Because we are interested in detecting the relevant phenotypic differences that lead to differences in fitness, a relevant fitness function must have the property that traits with no differences between organisms cannot factor into the relative fitness differences between these organisms. An exponential function of squared distance (including the Gaussian function) is only class of functions with this property (see SI).

Our model makes several assumptions about phenotype space and fitness in phenotype space. The main key assumption is that of subtle environmental perturbations – more specifically, this assumption assumes that mutants have a fixed location in phenotype space (we ignore any environmental contribution to phenotype) and is constant across environmental condition and dimension (ignoring any specific, correlated effect of phenotypes on fitness in a given environment).

*Estimation Methods – FGM Approach*

To infer the phenotypic space of our model from a set of mutants, we use an optimization technique on the relative fitness values of the set of mutants across subtle environmental perturbations. For a given number of traits *D*, we aim to find the positions of mutants, optima, and the ancestor that best fits our relative fitness data. We do this by finding the parameters that minimize the function:

where ,, and represent the locations of the mutants, optima, and ancestor, respectively. is the measured relative fitness of mutant in condition with a measure of the measurement uncertainty (see SI).

[section on other dimensionality reduction and why we’re different? (in a conceptual way) – if convincing don’t need to put comparison in the methods? ] The primary difference between our approach and other “dimensionality reduction” techniques like Principle Component Analysis (PCA) and Multidimensional Scaling (MDS) is that these methods find a low-dimensional representation of data based on distances between objects in the same class (in this case the mutants). Instead, we use relative fitness, which is a measure of distance between 2 classes of objects (mutants and optima). [CITE] [why SVD is different – linear (and doesn’t have the nice properties discussed above), not scaled in terms of fitness which makes D=1 mean fitness]

Using environmental perturbations around M3, including measuring this set of mutants in M3 on many different days, as well as a range of other environmental perturbations. We can divide our conditions into 2 distinct sets: the “subtle perturbation set” corresponding of all M3 conditions, as well as those within 2 standard deviations of the average across the M3 conditions, and the “intermediate to far perturbations set”, consisting of the remaining conditions. To learn the number of phenotypes that matter to M3, we can estimate the number of phenotypes that give us predictive power and avoid over fitting in the subtle perturbation set. Furthermore, we divide the mutants into two sets: a training set, picked to have an even representation of all mutation types observed, and a testing set (see Methods for detailed description of the bi-cross validation scheme). To avoid cases where it is substantially difficult to detect …