TITLE IDEAS:

Latent phenotypic complexity of adaptation in a single environment

Local Modularity and Global Pleiotropy of Local Adaptation

Latent Phenotypic Complexity of Local Adaptation

AUTHORS:

Grant Kinsler[1], Dmitri Petrov[1], Kerry Geiler-Samerotte[1,2]

AFFILIATIONS:

[1] Department of Biology, Stanford University, Stanford, CA

[2] Center of Mechanisms of Evolution, School of Life Sciences, Arizona State University, Tempe, AZ

INTRODUCTION

Organisms are very complex and well-integrated machines. This complexity and integration imply that it should be difficult to modify any one part of the organism without changing some or maybe even all other parts as well. This notion of "universal pleiotropy", whereby all mutations affect all traits, with stronger direct effects on some and weaker indirect effects on others, has been proposed by XXX. This notion recently gained additional empirical traction through the analysis of human GWAS data which revealed that practically all polymorphic sites in the human genome can be associated with some effects on most of the studied complex phenotypes (Boyle et al., 2017). In this way, the mapping between standing genetic variation and phenotype is not merely polygenic but rather "omnigenic" and indeed universally pleiotropic.

The notion of universal pleiotropy is difficult to reconcile with abundant evidence of rapid adaptation by single large-effect mutations that have been observed in multiple systems. Such large fitness jumps in the context of universal pleiotropy should be unlikely given how hard it is in principle to optimize a very large number of phenotypes simultaneously. This problem is known as the "cost of complexity" (Orr 2000) and was first recognized by R.A. Fisher in the context of his geometric model of phenotypic evolution. In this model the probability of mutation being adaptive declines as a square root of the number of independent phenotypic dimensions (orthogonal traits) and is one of the reasons that Fisher believed that only very small effect mutations should be able to contribute to adaptation (REFS).

However, as mentioned above, many examples of single mutations of very large fitness are known. For example in natural populations, the non-melanic variant of the peppered moth had a ~30% per generation fitness advantage during the post-industrial period and it is now known that the melanic phenotype is due to a single mutation - an insertion of a transposable element into the gene *cortex* (Cook, 2003; Cook et al., 2012; Hof et al 2016). The armor-reducing variant allele of *Eda* that distinguishes the marine and freshwater stickleback populations has been shown to have a selection coefficient of ~50% per generation in freshwater threespine sticklebacks (Barrett et al., 2008). Cancer driving mutations have been estimated to have fitness effects of ~20% per cell division in mouse models of lung cancer (Rogers et al Nature Genetics 2018) with similar magnitude of selective effects estimated from the cancer sequencing data analysis (Tilk et al., 2019). Individual drug resistance mutations in HIV can have effects of over 100% per generation (Feder et al., 2019). Finally, experimental evolution in multiple systems generically leads to rapid adaptation via single adaptive mutations that often have fitness benefits in the tens of percent per generation/cycle (Levy et al 2015, Venkataram et al 2016, Li et al 2018, others).

It is clear that large effect adaptive mutations are not only possible but are in fact commonplace. How can they be so common in the face of widespread pleiotropy? One possible resolution is to suggest that while most mutations are very pleiotropic and affect many (all) traits, the strongly adaptive mutations are unusual in that they succeed in affecting only some traits strongly and adaptively and other traits not at all. In other words, adaptive mutations are unusually modular (Wagner and Altenberg, 1996). The idea of modularity has been most famously invoked to argue that *cis*-regulatory rather than trans-regulatory or structural mutations should generally drive adaptation as these can tweak gene expression in some tissues but not others (REFS).

While the idea of modularity is very popular it has been hard to definitively argue for or against this notion because it is very hard to assess modularity empirically. To do so not only do we need to map all phenotypic effects of many adaptive mutations but we also need to assess the fitness consequences of specific phenotypic effects. While it is becoming possible to assess the

fitness and phenotypic effects of many adaptive mutations, especially in the experimental evolution context, it is extremely hard to specifically assign fitness values to individual phenotypic changes independently of each other.

Figure 1A illustrates the problem. Consider a mutation that has an effect on the expression of a particular gene (marked by a star in Fig. 1A). This effect subsequently results in changes in multiple protein abundances, which in turn produce other changes in gene expression and other protein abundances. These changes in molecular phenotypes percolate to higher functional levels and affect multiple aspects of cell function such as cell death, cell morphology, cell division rate, or other higher-level phenotypes. This complicated percolation of phenotypic effects, along with its high-level phenotypic effects, may at first seem to suggest that such a mutation is highly pleiotropic. However, it might be that only a subset of these phenotypic effects (say, the level of phosphorylation of a single protein, or the level of a secondary messenger such as cAMP, or the ability to rapidly leave the lag-phase) have substantial effects on the fitness of this mutant in this environment and others (say levels of expression of most individual genes or changes in cell shape) do not. In this respect, an adaptive mutation can be modular even if it affects a very large number of phenotypes to the extent that only a small subset of these phenotypes matter to fitness in the specific environment where the mutation is adaptive. In the end, even though we do now have the ability to measure thousands of molecular and cellular phenotypes at unprecedented levels of throughput, we still do not have an easy way to tell which of these phenotypes change adaptively or maladaptively and thus cannot easily address issues of evolutionary modularity.

Here we suggest a possible way forward. We argue that it might be possible to understand the structure of the fitness-relevant phenotypic space by measuring only the fitness - and not any of the phenotypes themselves - of a set of adaptive mutations in a collection of subtly varying environments (Fig 1B). The idea is that the way each mutant's fitness varies across environments must be related to its causal effects on phenotypes, and thus the way mutants co-vary in fitness across environments tells us whether they affect similar phenotypes. For instance in Fig 1B, mutants 1 and 2 have similar and the mutants 3 and 4 have dissimilar fitness profiles across multiple environments, suggesting that mutants 1 and 2 have similar and mutants 3 and 4 have dissimilar effects on fitness-relevant phenotypes. We then use these fitness profiles to create an abstract multi-dimensional (in Fig. 1B - *k* phenotypic dimensions) phenotypic space (M) with each mutant having a location in this space (e.g. mutant 1 having a location $(m_{11}, m_{12}, m_{13}, \ldots, m_{1k})$). Each environment is represented as a point in multidimensional space (C) where the coordinates of each environment correspond to the importance (weight) of each of the *k* phenotypic values for each mutant. The linear combination of a mutant's phenotype and an environment's weights on the phenotypes determines the fitness of the mutant in the environment and is given by $f_{ij} = m_{i1}c_{1j} + m_{i2}c_{2j} + m_{i3}c_{3j} + \ldots + m_{ik}c_{kj}$.

In order to infer the spaces M and C, we decompose the matrix of fitness profiles for a collection of mutants in a set of environments into mutant phenotypes and environment weights using Singular Value Decomposition (SVD), with an additional procedure to prevent overfitting and select the appropriate number of phenotypes *k.* Because we only measure fitness, we naturally only identify and focus on those underlying phenotypic components that matter to fitness. In the end, if only a small number of phenotypic dimensions is required to accurately predict fitness of a large number of adaptive mutants across many environments we can argue that adaptation is modular and that all adaptive mutations affect only a small number of distinct fitness-relevant phenotypes. To the extent we can use environmental perturbations that do not change the phenotypic effects of mutations too dramatically compared to the evolving condition (subtle perturbations), we can also claim that we have inferred the dimensionality of phenotypic space

in or at least near the evolving condition. Of course, the more subtly we vary the environments the harder it is to measure fitness differences precisely, putting more premium on the precision of fitness measurements.

We apply this approach to a large set of adaptive yeast mutants that drove adaptation in a limited glucose environment (Levy et al 2015) and find that these adaptive mutants affect only a small number of fitness-relevant phenotypes that matter in the evolution condition, confirming the intuition that adaptive mutants of large effect cannot change too many phenotypes that matter at once. However, we also find that these same adaptive mutants have many other pleiotropic effects that are revealed in conditions that are dissimilar from the evolution condition, confirming the intuition that mutants in general are pleiotropic. Integrating these two observations, we argue that strongly adaptive mutants must be locally modular but at the same time can be globally pleiotropic. We term this pattern “latent phenotypic complexity” - and suggest that even modular local adaptation can lead to generation of global phenotypic diversity. Finally we argue that the concept of modularity needs to be rethought to consider the possibility that the extent of modularity itself is context-dependent.

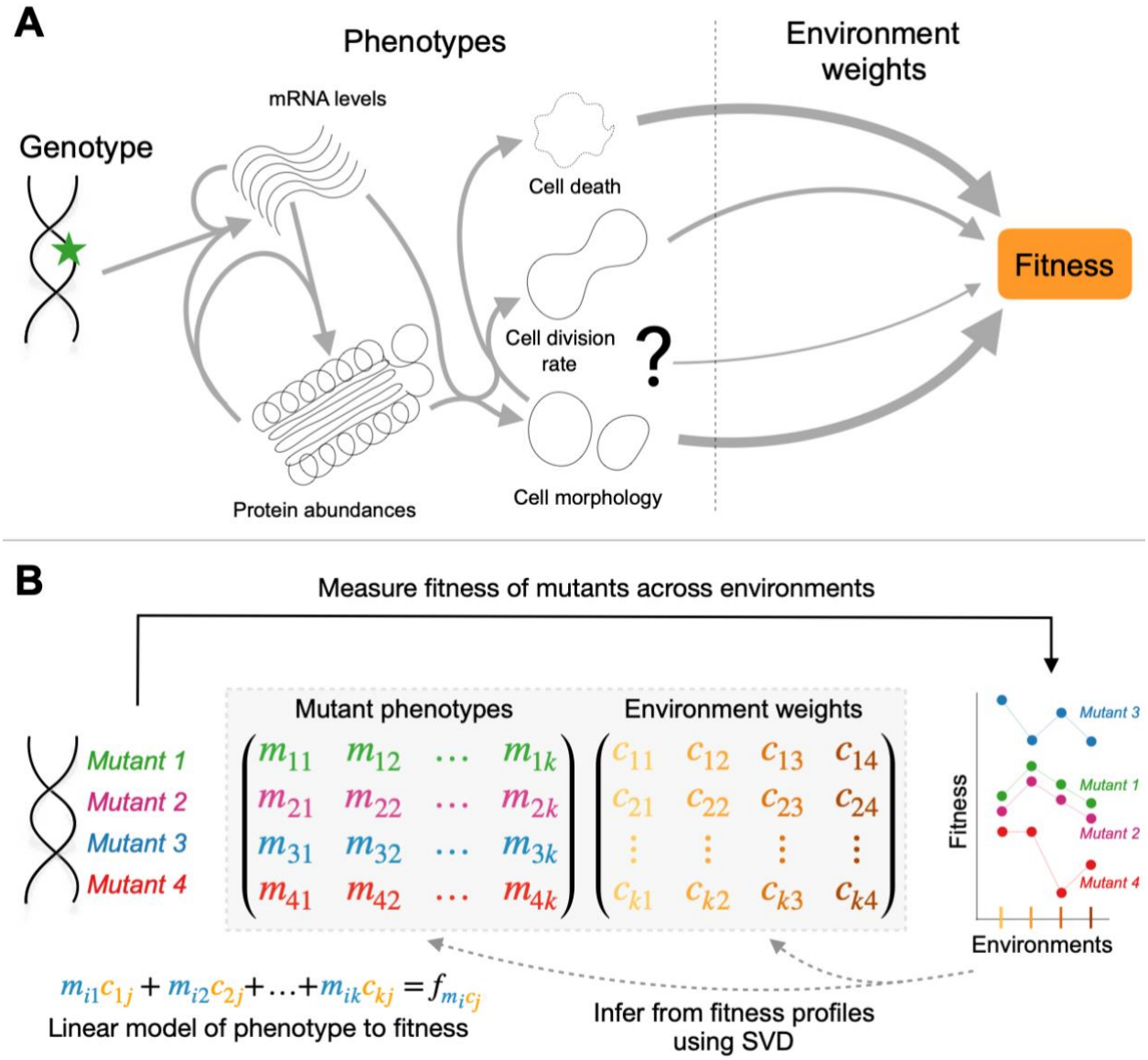

Figure 1. Environmental perturbations can identify fitness-relevant phenotypes. (A) Uncovering the genotype to phenotype to phenotype to fitness map is a difficult problem. Mutations can have several phenotypic effects, and these phenotypic effects can percolate throughout the organism. Despite these phenotypic effects, only some of these phenotypes (here, cell death and cell morphology) have substantial importance in this environment and will contribute most to the fitness effect of this mutation. Our approach aims to identify the number of fitness-relevant phenotypes, and their importance to fitness in a given environment from fitness measurements. **(B)** We measure the fitness of a collection of mutants across a set of environmental perturbations to construct their "fitness profiles". These profiles inform us of the phenotypes of these mutants, such that mutants with similar fitness profiles, in this case mutants 1 and 2, are identified as having similar effects on fitness-relevant phenotypes. We explicitly decompose the profiles of these mutants into a set of *k* mutant phenotypes and corresponding *k* environmental weights using singular value decomposition such that the linear combination of these phenotypes and environmental weights determines the expected fitness of each mutant in each environment.

RESULTS

Mutants that improve fitness under glucose limitation vary in their responses to environmental perturbations

A previous evolution experiment generated a collection of hundreds of independent mutations that provide a benefit to yeast cells growing in a glucose-limited environment (Levy et al., 2015). Many of these mutants, which began the evolution experiment as haploid, underwent whole-genome duplication to become diploids and to achieve a fitness benefit of ~30% per cycle (Fig 2B). Some of the diploids acquired additional mutations, including amplifications of either chromosome 11 or 12 as well as point mutations, which generated additional fitness benefits. The adaptive mutants that remained haploid primarily acquired loss-of-function mutations in nutrient-response pathways (RAS/PKA and TOR/SCH9) as well as a handful of mutations not classified as falling into these pathways, including a mutation in the HOG pathway gene *SSK2* (Fig 2B) (Venkataram et al., 2016). Do these diverse mutations represent different strategies for adapting to this environment? Or instead, are the underlying fitness-relevant phenotypic effects of these mutations the same?

To understand the mutants' diversity of phenotypes that contribute to a fitness effect in the glucose-limited environment they evolved in, we implement the strategy described in Figure 1B. Namely, we measure the fitness of adaptive mutants in a collection of environments. We then decompose the fitness profiles to identify the set of fitness-relevant phenotypic components represented by these mutants. Because the clones are barcoded, we can use previously-established methods to measure the fitness of this set of mutants in bulk and with high-precision (Venkataram et al., 2016). Specifically here, we compete a pool of the barcoded mutants against a reference ancestral strain over the course of several serial dilution cycles. We take timepoints at each transfer to construct barcode frequency trajectories that we then use to calculate the fitness of each barcode relative to the ancestor per cycle (Fig 2A).

In the end, we focus on a set of 262 adaptive mutants that have been sequenced, show clear adaptive effect in the evolution condition, and have fitness measurements in all of our collection of 45 environments. These 45 environments, which include experiments from previously published work (Li et al., 2018; Venkataram et al., 2016), range from batches of the evolution condition done on different days, to making subtle shifts to the amount of glucose, e.g. from 1.5% to 1.6%, to changing the shape of the flask (baffled vs non-baffled), all the way to non-fermentable carbon sources or increasing the salt concentration to induce osmotic stress (Table S1).

Our approach relies on using environments that reflect small perturbations from the evolution condition, as these are likely to capture the phenotypes relevant to fitness in the evolution condition. Thus, we must classify the 45 conditions into a set of "subtle" perturbations, from which we will detect the phenotypes relevant to fitness in the evolution condition, and "strong" perturbations, which we will use to test our model and understand how the importance of phenotypic components varies across environments. We measure the distance of an

environment using the average shift in fitness of all tested adaptive mutants and comparing it to the shifts we see across nine batches of fitness remeasurements in the ostensibly the original evolution condition (EC; M3 1.5% glucose, Materials and Methods). Note that the variation in fitness across the EC batches is much larger than the variation observed between the replicates for each given batch ($p < 1e-5$ from permutation test, Fig S2), suggesting that batch fitness variation is likely due to environmental variability that we were unable to control (e.g. shifts of shaker temperature across multiple days or subtle uncontrolled changes in the media), rather than technical noise. Therefore, this variation in fitness across the EC batches serves as a natural benchmark for the strength of environmental perturbations. For 16 non-EC environments, the average deviation of fitnesses from the evolution condition was similar to the variation we observe across the EC batches (Z-score less than two). These conditions, together with the 9 EC batches, make up a set of "subtle" environmental perturbations. The remaining 20 environmental perturbations, where the average deviation is larger than that observed across batches (Z-score greater than two), were classified as "strong" perturbations (Fig 2C, top).

We find that, while the rank order of the fitness effect of common mutations is relatively preserved across the 25 subtle perturbations, strong perturbations indicate that the common mutations do reflect different behaviors (Fig 2C, bottom). For example, *IRA1* nonsense mutants, the most adaptive in the EC batches, generally remain the most adaptive across the subtle perturbations, except a few conditions (e.g. Baffle) where they have similar or even decreased fitness compared to the *GPB2* mutants. Additionally, the *GPB2* and *PDE2* mutants have more or less similar fitness effects in the EC batches and only occasionally switch order across these subtle perturbations. In contrast, the 20 strong perturbations show clear differences in response for the common mutations (Fig 2C, bottom). For example, the 1 Day environment seems to affect *GPB2* mutants strongly, which have decreased fitness in this environment despite the remaining mutants behaving similar to their EC fitness. The strongest of these perturbations also reveal clear tradeoffs for some of these adaptive mutants. For example, *PDE2* and *IRA1* nonsense mutants are particularly sensitive to osmotic stress (shown NaCl, KCl environments), and *IRA1* nonsense mutants are very deleterious in the long transfer conditions (5-, 6-, 7-Day environments) (Li et al., 2018) . In contrast to complex behavior exhibited by the adaptive haploids, the diploids appear to be relatively robust to strong tradeoffs, appearing similarly adaptive across all perturbations, subtle and strong.

The observations that different mutants have different behaviors across environments, particularly those very different from the evolution condition, suggests that they do not influence identical phenotypes. However, it remains to be seen how complex this apparent phenotypic diversity is and whether variation in fitness across subtle conditions is predictive of fitness variation due to strong perturbations. In the next section, we will follow the procedure outlined in Fig. 1B to reconstruct the phenotypic space of fitness-relevant traits using only subtle perturbations and assess its complexity and predictive power.

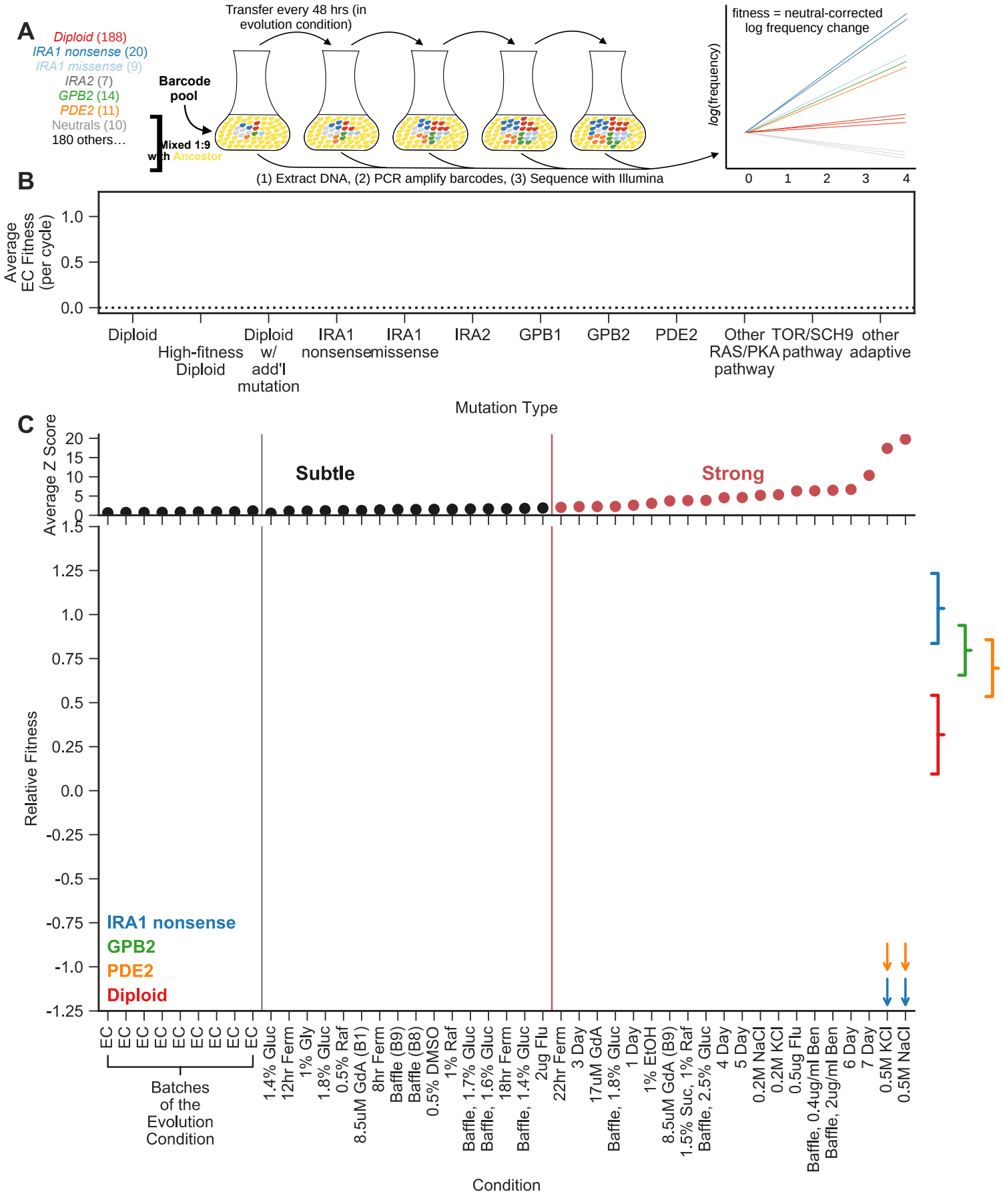
A
Diploid (188)
IRA1 nonsense (20)
IRA1 missense (9)
IRA2 (7)
GPB2 (14)
PDE2 (11)
Neutrals (10)
180 others…
Barcode pool
Mixed 1:9 with Ancestor
Transfer every 48 hrs (in evolution condition)
(1) Extract DNA, (2) PCR amplify barcodes, (3) Sequence with Illumina
log(frequency)
fitness = neutral-corrected log frequency change
0 1 2 3 4
B
Average EC Fitness (per cycle)
1.0
0.5
0.0
Diploid
High-fitness Diploid
Diploid w/ add'l mutation
IRA1 nonsense
IRA1 missense
IRA2
GPB1
GPB2
PDE2
Other RAS/PKA pathway
TOR/SCH9 pathway
other adaptive
Mutation Type
C
Average Z Score
20
15
10
5
0
Subtle
Strong
Relative Fitness
1.5
1.25
1.0
0.75
0.5
0.25
0.0
-0.25
-0.5
-0.75
-1.0
-1.25
IRA1 nonsense
GPB2
PDE2
Diploid
EC
Batches of the Evolution Condition
1.4% Gluc
12hr Ferm
1% Gly
1.8% Gluc
0.5% Raf
8.5uM GdA (B1)
8hr Ferm
Baffle (B9)
Baffle (B8)
0.5% DMSO
1% Raf
Baffle, 1.7% Gluc
Baffle, 1.6% Gluc
18hr Ferm
Baffle, 1.4% Gluc
2ug Flu
22hr Ferm
3 Day
17uM GdA
Baffle, 1.8% Gluc
1 Day
1% EtOH
8.5uM GdA (B9)
1.5% Suc, 1% Raf
Baffle, 2.5% Gluc
4 Day
5 Day
0.2M NaCl
0.2M KCl
0.5ug Flu
Baffle, 0.4ug/ml Ben
Baffle, 2ug/ml Ben
6 Day
7 Day
0.5M KCl
0.5M NaCl
Condition

Figure 2. Measuring fitness for a collection of adaptive mutants across many environments has the potential to reveal biological signal. (A) Schematic of fitness measurement experiment with DNA barcodes. Mutants tagged with DNA barcodes are pooled with the ancestor such that, at the initial timepoint of the experiment, the pool makes up 10% of the population. The pool is then propagated for several transfers. DNA is extracted from each timepoint, and the barcode region is PCR amplified and then sequenced with Illumina sequencing. Barcode frequencies are then quantified based on the relative counts of the barcodes, and fitness is calculated based on the rate of change of the barcode frequency, corrected for the frequency change of known neutral barcodes. **(B)** Average fitness in the evolution condition (per cycle) of each mutant, calculated as the average across all 9 batches of the evolution condition. **(C) (top)** Conditions are ordered based on similarity to the average across all batches. Conditions where mutants are less than two standard deviations different from the evolution condition (EC batches mean) are denoted in black and make up the subtle perturbation set. Conditions where the aggregate behavior exceeds two standard deviations are shown in red and make up the strong perturbations. **(bottom)** For each common mutation type, we take the average fitness across the evolution condition batches and the standard deviation of this behavior – brackets on the right represent two standard deviations away from the mean amongst the batch conditions per mutation type. Arrows indicate relative fitness values below -1.25. Grey shading for eye-guiding purposes only.

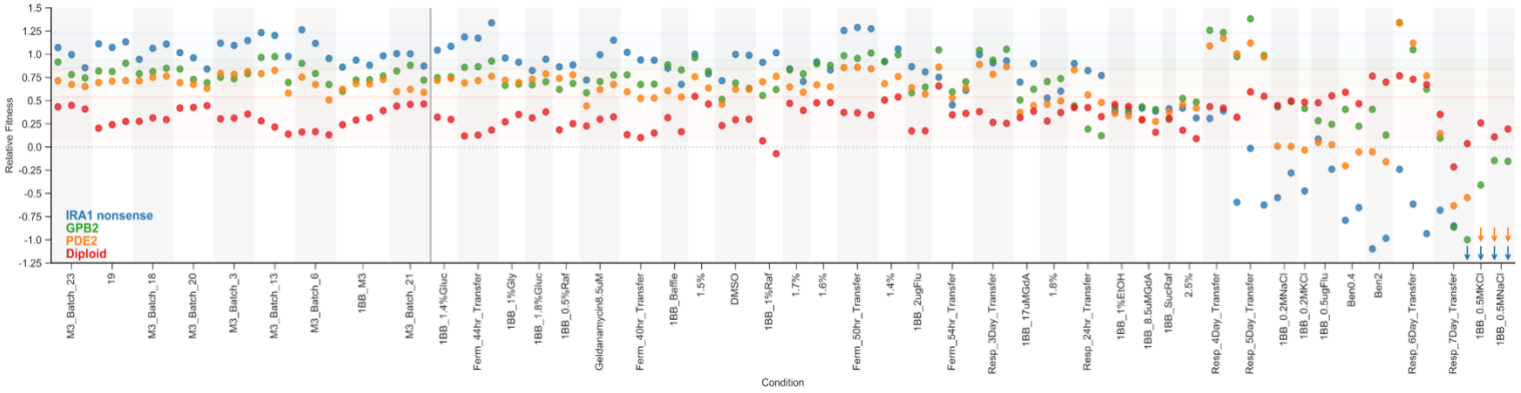

Fig S1. Barcoded fitness measurements provide precise estimates of fitness. Showing all the replicates for all the non-batch conditions.

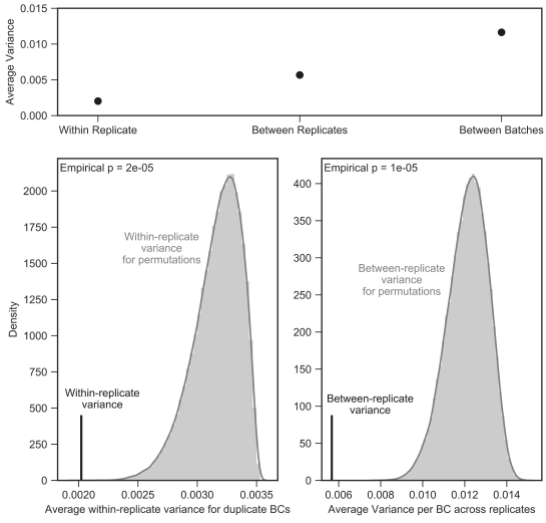

Fig S2. Variance in estimated fitness within replicates for identical mutants tagged with different barcodes is smaller than the variance in estimated fitness across replicates (both top panel “within replicate” vs “between replicate” comparison and bottom left panel via permutation test). Variance in estimated fitness across replicates is also smaller than that observed between batches (top panel and bottom right comparison via permutation test).

Table S1. All mutants included. Show number of mutants, including number included in canonical training and test sets.

8-dimensional phenotypic space predicts fitness variation across subtle environmental perturbations and known genetic features of adaptive mutants

After we use SVD to decompose the fitness variation across all the subtle environments, our next key question is which of the SVD dimensions reflect true biological variation and which reflect technical noise and thus lead to overfitting. To do this we use two independent approaches. The first approach takes advantage of estimates of the amount of measurement error given by our noise model, which estimates the uncertainty of a given measurement based on finite coverage and between-replicate variation. A matrix composed solely of random error, with no true signal, will have some largest component that Singular Value Decomposition (SVD) detects, representing the largest signal that can be generated from this noise alone. This largest noise component represents the detection threshold for real biological signal, as any signal smaller than this largest component of noise will be overwhelmed by error. We thus simulated many matrices of our estimate of measurement error and identified the detection threshold [see methods]. We then kept only components with more explanatory power than our estimate of noise (Fig. 3A). Our second method is bi-cross-validation whereby we first split the data into training and test sets. After using SVD to fit a phenotypic model on the training data, this model is then used to predict the fitness data in the test set. Because overfitting should reduce prediction accuracy in the test set, we then select the number of phenotypes that performs the best at predicting the test data across many iterations of this predictive process (Fig S4).

To validate both of these approaches, we simulated data with a known number of fitness-relevant phenotypes and used these procedures to estimate the number of phenotypes. With very small amounts of measurement error, both approaches accurately estimate the number of phenotypes. As measurement error increases, we retain the ability to detect only the largest phenotypes, as expected. Importantly, both methods largely agree on the number of detectable phenotypes in the data (Fig. S3).

Our analysis revealed that there is a small number of detectable fitness-relevant phenotypes represented by these adaptive mutants in the subtle environmental perturbations. The first component is very large, explaining 95% of variation in this data (Fig 3A), and effectively represents each mutant's average fitness across the subtle perturbations and the fitness of the average mutant in each condition. This is not surprising, as the fitness of mutants in the evolution condition should be predictive of fitness in the subtle perturbations. The next seven components explain the remaining genotype by environment interactions detectable with our approach. Of these, the first four capture 87% of the remaining variation (67.8%, 8.3%, 5.6%, and 5.3%, respectively) - these represent "major" components of interaction. Our approach also detects three additional "minor" interaction components that each capture less than 2% of the remaining variation (Fig 3A). Additional smaller components beyond these eight are below the level of detection.

We further validated this space using bi-cross-validation. Specifically, we designated a balanced set 60 of the 292 mutants as a training set, chosen such that each recurrent mutation type (diploid, high-fitness diploid, RAS/PKA mutants) is roughly equally represented (see Methods). The remaining 232 mutants comprise the test set. This set contains all mutation types represented by only a single mutant, including all TOR/SCH9 (*TOR1, SCH9, KOG1)* and HOG (*SSK2)* pathway representatives. We iteratively constructed 25 phenotype spaces by leaving out each subtle perturbation, creating the space with the remaining 24 conditions and the training mutants. We then predicted the fitness of the held-out testing mutants in this held-out condition (see Methods) with each number of components, using a measure of variance explained that weights the contribution of each mutant to overall variance explained based on the number of mutants that share its mutation type (see Methods). Averaging across the 25 iterations, a model with a single component, which again represents the average fitness for each mutant and the effect of the average mutant for each condition (1-component model), explains 85% of weighted variance for the test mutants in the left-out conditions. A 5-component model using this mean effect as well as the four major interaction components shows substantial improvement, explaining 95.1% of weighted variance. An 8-component model, which further includes three minor interaction components, shows additional incremental improvement and explains 96.2% of the weighted variance. Additional, smaller components do not provide consistent improvement of fit. Altogether, this shows that all eight components do in fact reflect detectable and predictive signals, despite the small size of the three minor interaction components. While there may be additional signal in the data, we are not confident in our ability to capture it with even smaller components.

We then asked if the 8-dimensional phenotype space constructed with fitness profiles of the training mutants and all the subtle perturbations clusters the mutants by pathway, genes, or another obvious pattern indicative of phenotypic variation despite labelled information being hidden from the inference procedure. We also inferred the location of the remaining 232 test mutants by allowing them to find the best phenotypes based on the weights of the conditions. We used Uniform Manifold Approximation and Projection (UMAP) to visualize the distance between mutants in the 8-dimensional phenotype space (Fig. 3B).

The map shows that many mutants do cluster by type and by gene. Specifically, the diploids, *IRA1 nonsense, GPB2,* and *PDE2* mutants all form distinct clusters. Interestingly, six high-fitness diploids (diploids with higher than average diploid fitness in the EC) also form a distinct cluster despite being whole genome sequenced and sharing no clear functional mutations. We confirmed these clusters by explicitly calculating the median pairwise distance for these groups of mutants and finding that they are indeed more closely clustered than randomly chosen groups of mutants (see Fig S5). In addition to mutants clustering by gene, there is evidence that mutants also cluster by pathway. In particular, TOR/SCH9 pathway mutants are closer to each other than randomly-selected groups of adaptive haploids ($p = 0.0191$). This further argues that the phenotypic space we created is biologically meaningful as none of the TOR/SCH9 mutants were in the training set and thus not used to create the phenotypic space.

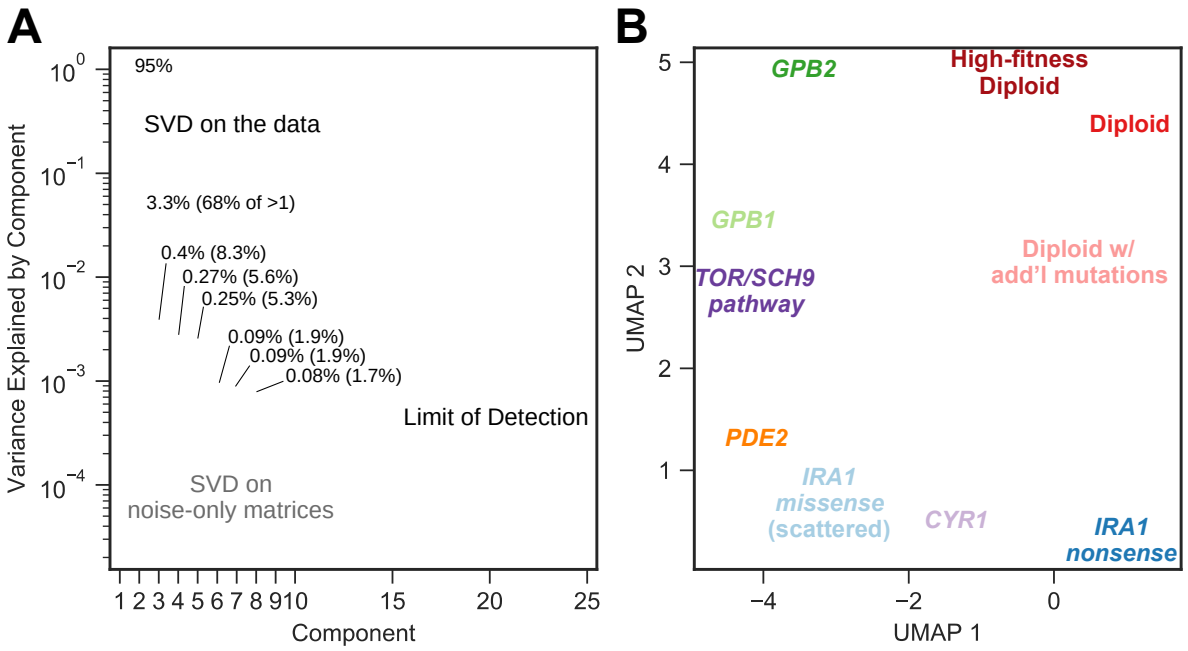

Figure 3. Subtle environmental perturbations reveal a 8-component phenotype space that reflects known biological features. (A) We use SVD to decompose the fitness profiles of 292 adaptive mutants in the subtle environmental perturbations. The first component represents 95% of variation across the subtle perturbations. The signal in the data reflected by subsequent components declines (black dots, connected by line). Variance explained is indicated by the percentage, in parentheses is the percent of variation beyond the first component explained by each of these components. SVD on data simulated from the estimated measurement noise (repeated 1000 times, gray lines) reveals the limit of detection (dashed line), represented by the largest signal from the noise, indicating that there are 8 detectable components in this data. **(B)** We construct a 8-component phenotype space from the fitness profiles of the balanced training mutants across the subtle perturbations and then place the testing mutants based on the condition weights. Uniform Manifold Projection and Approximation (UMAP) on the resulting space visualizes the relationship between mutants in this 8-dimensional space. Mutants that are close together have similar inferred phenotypic responses. Mutants with mutations in the same gene tend to be closer together than random, in particular *IRA1 nonsense* mutants in dark blue, *GPB2* mutants in dark green, *PDE2* mutants in dark orange, and Diploid mutants in red. Six high-fitness diploid mutants also form a cluster despite no known genetic similarities. Note that for visualization purposes and to prevent imbalance in the UMAP procedure, we have only included the pure diploids that are in the training set.

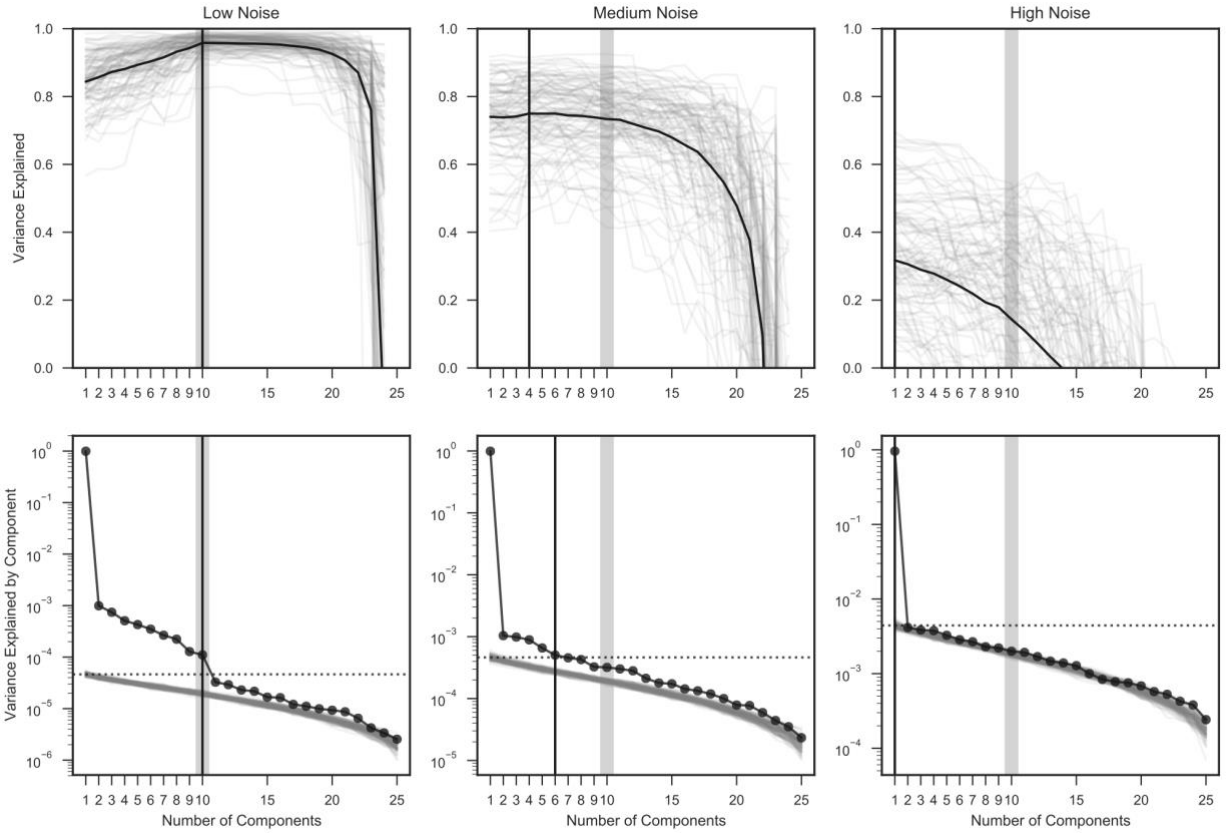

Fig S3. Simulations show that our overfitting procedures work. Simulating fitness matrices with a known number of phenotypic components (10, represented by the vertical gray line), we test our methods to detect when we are overfitting measurement error. With very low simulated measurement error, both methods agree and accurately detect the correct number of components. As measurement noise decreases, the methods lose the ability to detect small components and, if measurement noise is too large, can only detect a single component. Both methods (bi-cross-validation with leave-one-out above and SVD on simulated measurement noise) roughly agree on the number of detectable components. Vertical gray lines indicate the simulated number of components. Vertical black lines denote the inferred number of phenotypes for each approach and amount of measurement error.

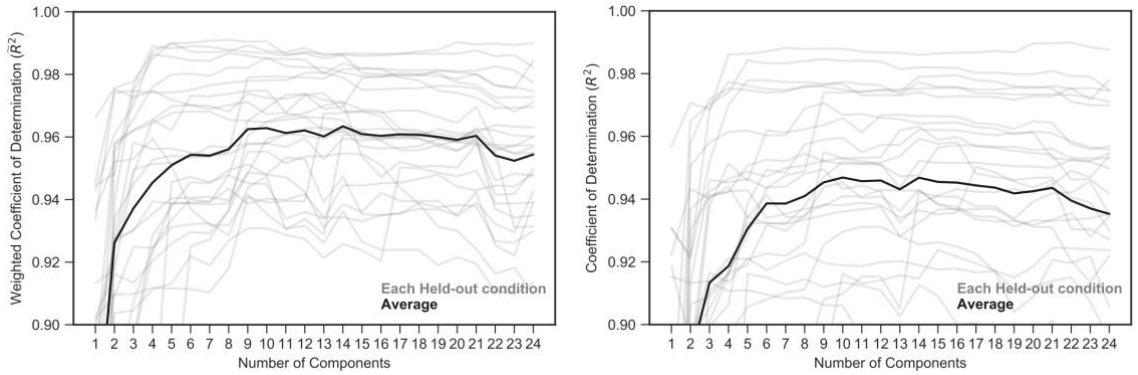

Figure S4. Leave-one-out bi-cross-validation confirms 3 minor components contain detectable biological signal. (A) Prediction of fitness for the held-out mutants in the held-out condition for each held-out subtle perturbation is shown in gray. The average is shown in black. Predictive power increases until component 9 and then flattens out. Note that component 14 has slightly better predictive power than component 9, on average. **(B)** Same as **(A)** but with standard R^2metric.

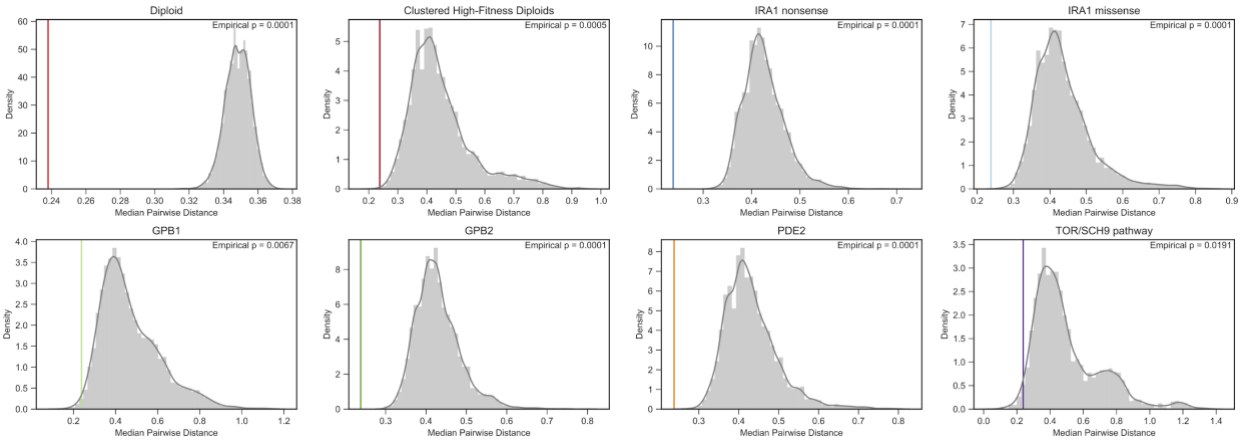

Fig S5. Mutants are closer to others that share their mutation type than expected by chance. The median pairwise distance for each group of mutants is closer than for randomly chosen mutants with the same number of mutants. Colored vertical line indicates the realized median pairwise distance for each group of mutants. Gray histogram shows the distribution of median pairwise distances for randomly chosen mutants with the same number of mutants. Null distributions were chosen from all mutants for the diploid comparison and all mutants that are not pure diploids for the remaining comparisons. This was done because diploids comprise the majority of the mutants, and it then becomes very likely to select a set of all diploids, which will be anomalously close together.

An 8-component model predicts mutant fitness in novel and substantially different environments

In the previous section, we showed that the phenotype space, constructed from the subtle perturbations, reflects known genetic features of the adaptive mutants. Additionally, across the 25 phenotype spaces constructed by leaving out each subtle perturbation, there was a general decline in the contribution of each subsequent component to predictive power of the held-out conditions. We next investigated if this phenotype space could predict the fitness of the testing mutants in the strong environmental perturbations.

For comparison, we first show our ability to predict fitness in each one of the held-out subtle perturbations (Figure 4A top, left side). As expected, the 1-component model does a reasonable job at predicting the fitness of the held-out mutants in each held-out condition, ranging from ~60% to 97% of weighted variance explained. The 5-component model, with the four major additional components, shows an improvement in predictive power ($\tilde{R}^2$ ranging from 86% to 99%). The full 8-component model does incrementally better. The bottom panel of Figure 4 explicitly shows improvement in predictive power from the inclusion of the three minor components. Across the subtle environmental perturbations, this improvement is small, with the largest improvement coming from one EC batch, where the three minor components add 10% of the total predictive power.

We next found that the 8-dimensional phenotypic space built using only subtle perturbations is surprisingly predictive of the fitness of test mutants in novel environments that represent strong perturbations from the evolution condition (Figure 4). The first component predicts between -42% and 79% of weighted variance in these environments, representing strong differences in baseline predictability amongst these different environments (Fig 4A, top). Note that here, the negative explained variance indicates that the model has very poor predictive power, explicitly due to predicted fitness that is worse than choosing the average fitness for a given condition. In all cases, the full 8-component phenotype space does a much better job of capturing true behavior (ranging from 50% to 95% of weighted variance explained), even in cases where the first component does very poorly. Most strikingly, using just the five largest components often does substantially worse at predicting fitness in these strong perturbations (Fig 4A). The bottom panel of Figure 4A explicitly shows the improvement in predictive power due to the inclusion of the three minor components. For example, the three minor components add 20% of the weighted variance explained by the full phenotype space for the 1-Day condition ($\tilde{R}^2$ = 0.6 for 5-component, $\tilde{R}^2$ = 0.73 for 8-component). This pattern is particularly strong for the 6-Day environment, with the three minor components adding 43% of the full model's explained variation. As shown in Figure 4A, these observations hold across all phenotype spaces constructed by leaving out each subtle perturbation, indicating that this pattern is not specific to small phenotypes detected solely due to the inclusion of any single training condition.

The importance of the smallest phenotypic components is, however, condition-specific. For example some strong conditions, including Baffle- 1.8% Glucose, 8.5uM GdA (B9) and Baffle - 2.5% Glucose, have similar prediction patterns to that of the small perturbations (Fig 4). In these

cases, the 5 largest components capture roughly the same amount of variation as the full phenotype space with all 8 components. Why is it that some conditions are seemingly well-captured by the largest components and others show significant improvement? Moreover, is this improvement in predictive power driven by a general, shared phenotypic response for many mutants or, instead, reflective of specific phenotypic effects of a handful of mutants?

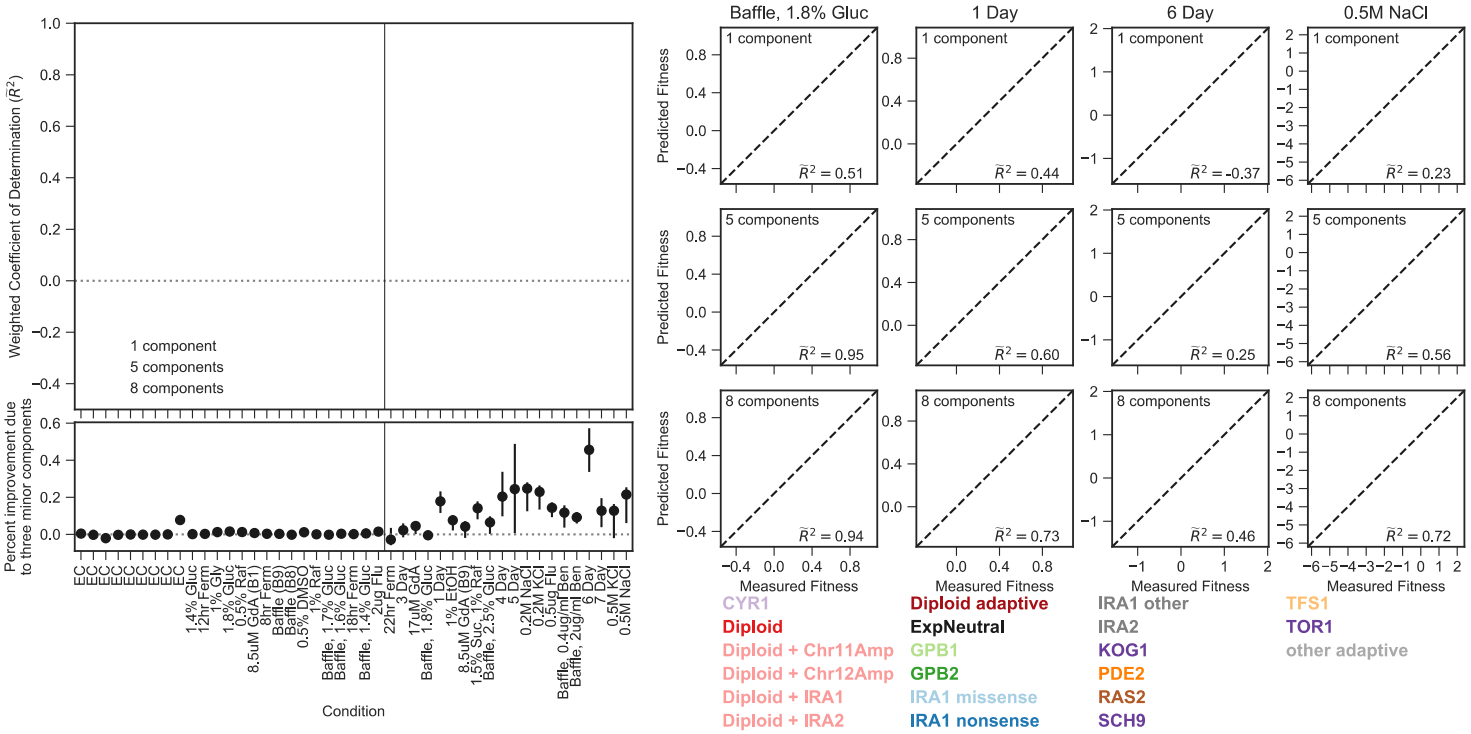

Figure 4. Phenotypic components learned from subtle conditions can accurately predict fitness of held-out mutants in strong conditions. (A) Predictions from the 8-component model (red circle) are typically better than the 1-component mode (open circle) and the average of 1000 permutations (black line, each permutation shown in gray). The 5-component model (black dot) has similar predictive power for most conditions. However, particular conditions (e.g. 1-, 4-, 6-Day) gain significant predictive power from the additional three minor components. Bottom shows percent of the full 8-component model resulting from the addition of the three minor components. Error bars denote the full range of values observed for each prediction across the 25 leave-one-out models. **(B)** Comparison of predictions of the 1-, 5-, and 8- component models for all held-out mutations in Baffle + 1.8% Glucose,1 Day, 6 Day, and 0.5M NaCl conditions. Note that $\tilde{R}^2$ less than zero indicates that the prediction is worse than using the mean fitness in that condition and that $\tilde{R}^2$ is weighted according to the number of each mutation type present in the held out data (see Methods for details). Points in **(B)** colored by the mutation type. For a full set of prediction comparisons see Supplement. Gray and white background in **(A)** for eye-guiding purposes only.

Fig S?. Robust to choice of mutants.
How robust are these results to our choice of mutants? 1) inference of 9 components, 2) predictive power, and 3) the “latent” phenotype idea

Fig S?. Figure without weighted R^2?

Fig S?. All 1-, 4-, 9- component comparisons by condition. (for all of them).

Fig S?. UMAP of conditions? [not really sure where this fits]

To further understand the source of the additional predictive power for the smallest three components, we explicitly looked at the contribution of each component to the ability to predict each condition. We did this by comparing the predictive power of the space with only n fitness-relevant phenotypes to that of the n+1-component model (e.g. 5- versus 6-component models). This change represents the contribution of the n+1th (e.g. 6th) component to predictive power.

As expected, the additional predictive power contributed by each component to each of the subtle perturbations declines with the overall size of the component. However, for some strong conditions some of the small components' contribution can be relatively large. For instance, component 6 contributes less than 0.01 to additional weighted explained variation for the subtle perturbations on average, but adds more than 0.06 to the predictive power for the 6-Day environment (Fig 5A). There are similar effects for the other small components, with the 7th component adding over 0.075 to weighted r-squared in 0.5M NaCl and the 8th adding over 0.15 to 6-Day.

Furthermore, this increase in explanatory power seems to be driven by specific phenotypic effects of a subset of mutants. For example, diploids with known additional mutations, including those with chromosome 11 amplifications, chromosome 12 amplifications, *IRA1* mutations, and *IRA2* mutations, see a particular increase in their prediction improvement driven by the inclusion of the 6th component in the 6-Day environment (Fig 5B). Similarly, *GPB2* mutants drive the improvement seen from the inclusion of the 7th component in the 0.5M NaCl environment, and several adaptive haploids drive this pattern for the 8th component in the 6-Day environment.

This suggests that, although some phenotypes may have limited contribution to fitness in the evolution condition, and appear to be minor based on their effects in subtle environmental perturbations, the importance of these phenotypes is context-dependent and can be large in some contexts. Specifically, there are particular conditions (i.e. 0.5M NaCl environment) where these phenotypes have substantial contribution to fitness. Thus, these phenotypes of small effect in the evolution condition are in fact "latent", and their contribution to fitness can be revealed in new environments. Moreover, these phenotypes appear to be driven by the particular effects of a subset of mutants (i.e. *GPB2* mutants), suggesting that these latent phenotypes reflect pleiotropic effects uniquely affected by these mutants.

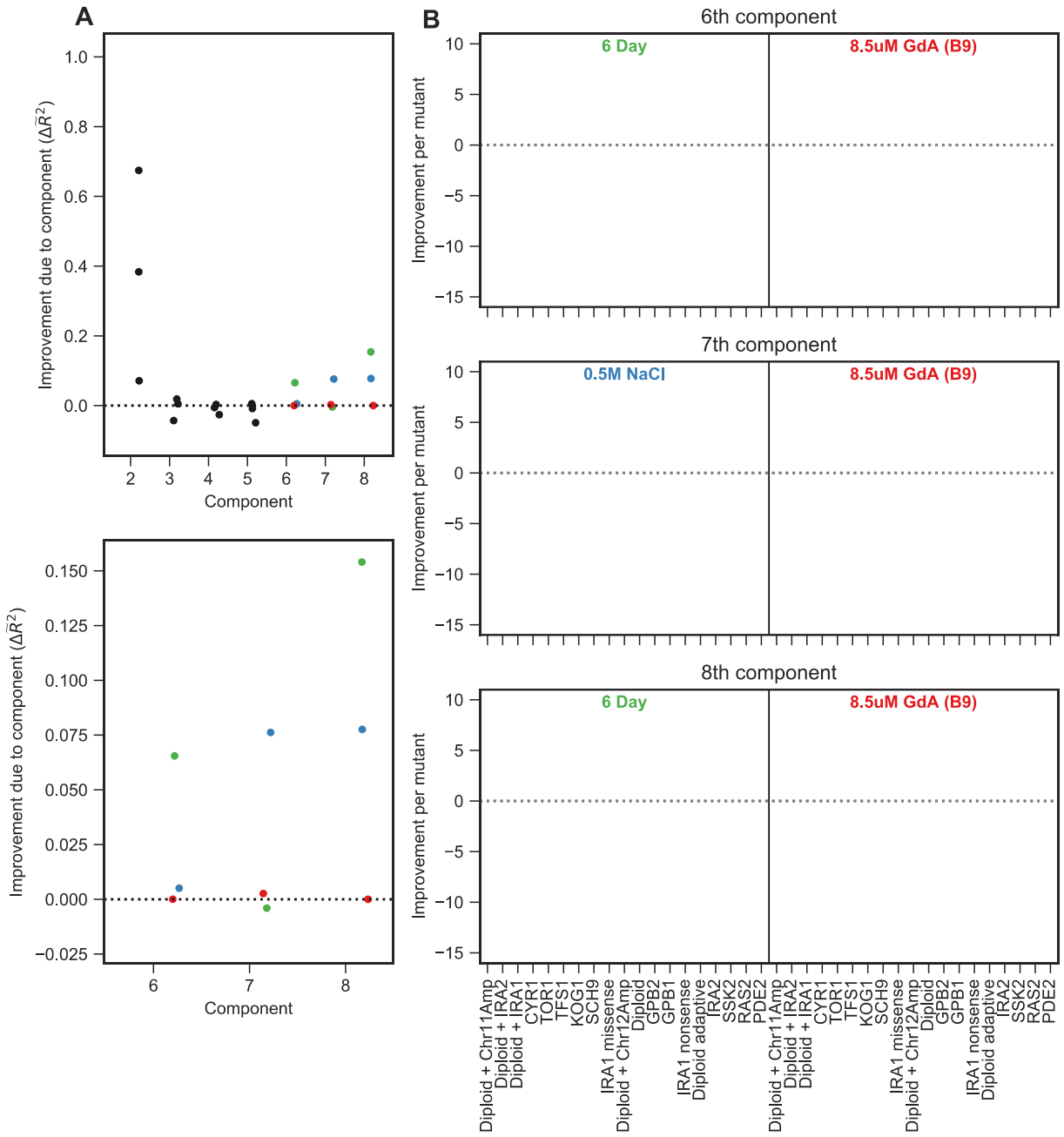

Figure 5. Phenotypes with minor contribution in the subtle conditions have large, specific biological relevance in other, strong conditions. (A) Particular conditions have substantial improvement from the addition of specific components. Magnification shows components with minor contribution to explanatory power across the subtle environments (gray points). 7th component has large contribution to explained variation for 6-Day condition, 8th for 0.5M NaCl, and 9th for 6-Day condition. **(B)** Improvement in predictive power (units of measurement error) by mutant for the 7th, 8th, and 9th components in the condition with largest improvement (white background) and a condition without a large benefit (8.5uM GdA (B9) with gray background). Improvement is particularly strong for particular mutants and is specific to particular environments. Ordered by average improvement from 7th component in 6-Day condition.

DISCUSSION [just ideas/outline right now]

1. Brief Recap of our main claim and why it's important

- Detecting fitness-relevant phenotypes is a big and difficult problem…
- our new approach allows us to identify phenotypic components that fitness-relevant
- we find a small number of large-effect FRPs in subtle, with some FRPs of small effect having substantial effect in new environments.

2. big consequences for evolution

Context dependency:
- changing environments (has particular consequences for particular mutants with latent phenotypic effects)
- epistasis (these aren't truly "redundant" mutants.. and can affect what subsequent mutations might arise

3. why it was now possible?
    a. precision of measurement
    b. why subtle?

Importance of precision of measurement:
- With less precision, we wouldn't detect the latent phenotypes in the subtle perturbations (a 2-fold increase in measurement noise would prevent us from detecting any of these latent phenotypes!!)
- We know there are actually more than 9 fitness-relevant phenotypes, too! We fail to capture all behavior in most of the strong perturbations, suggesting there must be more, small components not captured by our subtle perturbations and/or our level of measurement error

Importance of subtlety:
- Do we get the same result if we were to use the strong environments? Why is subtlety important?

4. what does this really mean? what are the things we're calling "phenotypes"?

- "phitnotypes", "causal properties of the organism affected by adaptive mutations"
- we treat mutants as having fixed phenotypic effects - strong perturbations have the possibility of exerting an environmental influence on the phenotypes themselves, which would violate this assumption.
- can we actually identify what measurable phenotypes map to these phitnotypes?

5. drawbacks: one specific example - maybe this experiment was weird in some way (too complex/too simple - how might this differ in other environments?

Dependency on this evolution experiment:
- Are latent phenotypes generically observed across evolution to other environments?
- How do more complex environments affect this result?

6. some other possible uses? [

A new approach and tool that could be useful for:

- Ecology
- identifying functional units using perturbations
- detecting relevant phenotypes for any objective function (tumor growth, drug resistance)

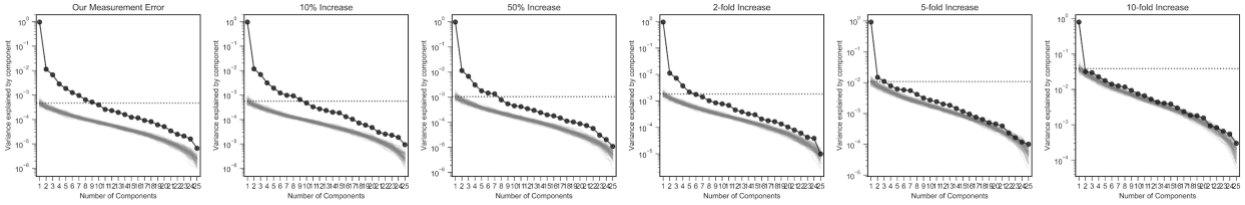

Fig S?. Measurement precision allows us to detect latent phenotypes. If we simulate less precise measurements, we see that we only detect 2-4 anyways…

Fig S?. Why subtle? What does doing SVD on ALL conditions give us? Why is subtlety important?

REFERENCES

Josse, J., Sardy, S., 2014. Adaptive Shrinkage of singular values. ArXiv13106602 Stat.

Levy, S.F., Blundell, J.R., Venkataram, S., Petrov, D.A., Fisher, D.S., Sherlock, G., 2015. Quantitative evolutionary dynamics using high-resolution lineage tracking. Nature 519, 181–186. https://doi.org/10.1038/nature14279

Li, Y., Venkataram, S., Agarwala, A., Dunn, B., Petrov, D.A., Sherlock, G., Fisher, D.S., 2018. Hidden Complexity of Yeast Adaptation under Simple Evolutionary Conditions. Curr. Biol. 28, 515-525.e6. https://doi.org/10.1016/j.cub.2018.01.009

Venkataram, S., Dunn, B., Li, Y., Agarwala, A., Chang, J., Ebel, E.R., Geiler-Samerotte, K., Hérissant, L., Blundell, J.R., Levy, S.F., Fisher, D.S., Sherlock, G., Petrov, D.A., 2016. Development of a Comprehensive Genotype-to-Fitness Map of Adaptation-Driving Mutations in Yeast. Cell 166, 1585-1596.e22. https://doi.org/10.1016/j.cell.2016.08.002

Wagner, G.P., Altenberg, L., 1996. Perspective: Complex Adaptations and the Evolution of Evolvability. Evolution 50, 967–976. https://doi.org/10.2307/2410639

METHODS

LEAD CONTACT AND MATERIALS AVAILABILITY

Further information and requests for resources and reagents should be directed to and will be fulfilled by the Lead Contact, Jane Doe (janedoe@qwerty.com).

EXPERIMENTAL MODEL AND SUBJECT DETAILS

The yeast strains used in this study can be grown and maintained using standard methods (e.g. YPD media in test tubes, glycerol stocks for long term storage at -80°C), but should be propagated in the appropriate selection environment (a glucose-limited minimal media - M3 medium for the evolution condition) for comparable fitness and phenotypic measurements.

Experiments were performed with either one of two pools of barcoded mutants isolated from a previous evolution experiment (Levy et al., 2015). The first pool of 4,800 mutants was constructed by plating out frozen samples of the two replicate evolution experiments [cite levy blundell] and picking and pooling colonies into a single culture. This pooled culture was grown up in YPD and then aliquoted into 1.5ml glycerol stocks and stored at -80°C. The second pool of 500 mutants was constructed similarly, with the additional stipulation that each barcode be represented equally in the pool. This was done by sequencing the barcode region of several plates of mutants, growing up a single representative of each barcode in YPD individually, and then pooling these cultures together. The pooled culture was then aliquoted into 1.5ml glycerol stocks and stored at -80°C.

For one batch of experiments, we also included two sets of re-barcoded mutants…

Restriction-site ancestor?

METHOD DETAILS

Conducting the barcoded fitness measurements

Fitness measurement experiments were performed as described previously (Li et al., 2018; Venkataram et al., 2016), where a pool of barcoded mutants (see EXPERIMENTAL MODEL AND SUBJECT DETAILS for more information on the two pools used) was competed against with a reference strain. We conducted fitness measurements under a variety of conditions that are perturbations of the evolution condition (for details on the differences induced in each perturbation see Table S1). Here, we describe the procedure for the evolution condition.

To begin the experiment, we separately grew up an overnight culture of the barcode pool and the ancestral strain in M3 (minimal, glucose-limited) medium. We then mixed the saturated barcode pool culture with the saturated ancestor culture at a 1:9 ratio by optical density. This ratio allows for each mutant to effectively compete individually against the ancestor. In addition, the ancestral strain, being the majority of the culture, effectively sets the rate of metabolism and environmental change during a single transfer cycle. We then inoculated 400μL of this pooled culture ($\sim 5 \times 10^7$ cells) into 100mL of M3 in 500mL DeLong flasks. The culture was then grown at 30°C in an incubator shaking at 223 RPM for 48 hours. After 48 hours of growth, 400μL of

saturated culture was transferred into fresh M3 medium. This serial dilution was continued over several transfers. Each transfer, including after the initial mixing, the remaining culture was transferred to 50mL conicals, spun down at 3000 rpm for 5 minutes, resuspended in 5mL of sorbitol freezing solution (0.9M sorbitol, 0.1M Tris-HCL pH 7.5, 0.1M EDTA pH 8.0), aliquoted into three 1.5mL tubes, and was then stored at -80°C.

For batch 9, where additional neutral lineages and the re-barcoded IRA1 nonsense and IRA1 missense lineages were included, the initial inoculation mix consisted of 90% ancestral strain, 9.4% 500 barcode mutant pool, 0.2% additional neutral spike-in pool, 0.2% re-barcoded IRA1 nonsense pool, and 0.2% re-barcoded IRA1 missense pool.

Condition details.

In this study, we present fitness measurement data from a collection of 45 conditions. Several of these included conditions are from previous studies (Li et al., 2018; Venkataram et al., 2016). Please see Table S1 for specific details on each condition and the differences from the evolution condition. Note that some conditions, including the Fluconazole conditions and some of the Geldanamycin conditions have unexpected orderings in the strength of perturbation (i.e. the smaller drug concentration shows a larger difference in fitness or similar concentrations seem to have different effects). These could be reflective real biological patterns, but also possibly due to technical reasons, such as degradation of the drug or poor solubility. Despite this possibility, we include these conditions because the exact perturbation does not matter for our purposes. We do not rely on the identity of the environmental perturbations and use only the effect of the realized perturbation on fitness.

#	Condition Name	Source	Number of barcodes	Batch Number	Batch Date	Number of replicates	Description of Manipulation
1	EC	This study	500	10	9/4/15	3	No preculture
2	EC	This study	4800	4	5/9/15	3	No barcodeless ancestor
3	EC	Li et al (2018)	4800	4	5/9/15	3	
4	EC	This study	500	5	8/17/15	3	No preculture
5	EC	Venkataram et al (2016)	4800	1	11/24/14	3	
6	EC	Li et al (2018)	4800	3	5/1/15	3	
7	EC	Venkataram et al (2016)	4800	2	12/26/14	3	
8	EC	This study	500	9	12/10/17	4	
9	EC	This study	500	5	8/17/15	3	
10	1.4% Gluc	This study	500	9	12/10/17	2	1.4% glucose concentration
11	12hr Ferm	Li et al (2018)	4800	3	5/1/15	3	44 hours of growth, 2*10^8 cells transferred, resulting in ~12 hours of fermentation phase
12	1% Gly	This study	500	9	12/10/17	2	Added 1% glycerol0
13	1.8% Gluc	This study	500	9	12/10/17	2	1.8% glucose concentration
14	0.5% Raf	This study	500	9	12/10/17	2	Added 0.5% raffinose
15	8.5uM GdA (B1)	This study	4800	1	11/24/14	3	Added 8.5uM Geldanamycin
16	8hr Ferm	Li et al (2018)	4800	3	5/1/15	3	40 hours of growth, 8*10^8 cells transferred, resulting in ~8 hours of fermentation phase
17	Baffle (B9)	This study	500	9	12/10/17	2	Used baffled flask
18	Baffle (B8)	This study		8		2	Used baffled flask
19	0.5% DMSO	This study	4800	1	11/24/14	3	Included 0.5% DMSO
20	1% Raf	This study	500	9	12/10/17	2	Added 1% raffinose
21	Baffle, 1.7% Gluc	This study		8		2	1.7% glucose concentration, used baffled flask
22	Baffle, 1.6% Gluc	This study		8		2	1.6% glucose concentration, used baffled flask
23	18hr Ferm	Li et al (2018)	4800	4	5/9/15	3	50 hours of growth, 2.5*10^7 cells transferred, resulting in ~18 hours of fermentation phase
24	Baffle, 1.4% Gluc	This study		8		2	1.4% glucose concentration, used baffled flask
25	2ug Flu	This study	500	9	12/10/17	2	Added 2ug Fluconazole
26	22hr Ferm	Li et al (2018)	4800	4	5/9/15	3	54 hours of growth, 6.25*10^6 cells transferred, resulting in ~22 hours of fermentation phase
27	3 Day	Li et al (2018)	4800	2	12/26/14	3	3 days of growth
28	17uM GdA	This study	500	9	12/10/17	2	Added 17uM Geldanamycin
29	Baffle, 1.8% Gluc	This study		8		2	1.8% glucose concentration, used baffled flask
30	1 Day	Li et al (2018)	4800	2	12/26/14	3	24 hours of growth
31	1% EtOH	This study	500	9	12/10/17	2	Added 1% ethanol
32	8.5uM GdA (B9)	This study	500	9	12/10/17	2	Added 8.5uM Geldanamycin
33	1.5% Suc, 1% Raf	This study	500	9	12/10/17	2	No glucose, 1.5% Sucrose, 1% Raffinose
34	Baffle, 2.5% Gluc	This study		8		2	2.5% glucose concentration, used baffled flask
35	4 Day	Li et al (2018)	4800	2	12/26/14	3	4 days of growth
36	5 Day	Li et al (2018)	4800	6	9/16/15	3	5 days of growth
37	0.2M NaCl	This study	500	9	12/10/17	2	Added 0.2M NaCl
38	0.2M KCl	This study	500	9	12/10/17	1	Added 0.2M KCl
39	0.5ug Flu	This study	500	9	12/10/17	2	Added 0.5ug Fluconazole
40	Baffle, 0.4ug/ml Ben	This study		7		2	Added 0.4ug/mL Benomyl, used baffled flask
41	Baffle, 2ug/ml Ben	This study		7		2	Added 2ug/mL Benomyl, used baffled flask
42	6 Day	Li et al (2018)	4800	6	9/16/15	3	6 days of growth
43	7 Day	Li et al (2018)	4800	6	9/16/15	3	7 days of growth
44	0.5M KCl	This study	500	9	12/10/17	1	Added 0.5M KCl
45	0.5M NaCl	This study	500	9	12/10/17	1*	Added 0.5M NaCl

716

Table S1. List of all conditions used in this study. List of conditions ordered by similarity to the average across the evolution condition (same ordering as in the main text figures). Note that 0.5M NaCl was started with 2 replicates. The second replicate was excluded from analysis due to very low growth and/or failed transfer (media still transparent after 48 hours), during the second transfer. Note al

DNA Extraction of each sample

We extracted DNA from each sample (representing a time-point for a particular replicate). Batches 1-6 and 10 were conducted with the protocol described in (Venkataram et al., 2016). To improve the ease and yield extraction, we made some slight modifications to the DNA extraction for batches 7, 8, and 9. This protocol is detailed here.

A single tube for each sample is removed from the freezer and thawed at room temperature. We then extract DNA using modification of the Lucigen MasterPure yeast DNA purification kit (#MPY80200). We transfer the thawed cells into a 15mL conical and centrifuge for 3 min at 4000 RPM. After discarding the supernatant, the pellet is then resuspended with 1.8mL of the MasterPure lysis buffer, and 0.5mm glass beads are added to help with disruption of the yeast cell wall. The mix of pellet, lysis buffer, and beads is then vortexed for 10 seconds and incubated for 45 minutes at 65°C, with periodic vortexing. The solution is then put on ice for 5 min and then 900μL of MPC Protein Reagent is mixed with the solution. We then separated protein and cell debris by centrifugation at 4000 RPM, transferring 1900μL of supernatant to 2mL centrifuge tubes. We further separate remaining protein and cell debris by centrifuging at 13200 RPM for 5 min. The supernatant is then transferred into two 2mL centrifuge tubes, with 925μL of the supernatant into each. Next, we add 1000μL of isopropanol to each tube, mix by inversion, centrifuge at 13200 RPM for 5min, and discard the supernatant. The pellet, containing the DNA is then resuspended in 250μL of Elution Buffer, and 10μL of 5ng/μL RNAase A is added. This is either left at room temperature overnight or incubated at 60°C for 15 min. Next the two tubes per sample are combined into a single tube and 1500μL of ethanol is added. This is then mixed by inversion, and strands of precipitating DNA should appear. This is centrifuged at 13200 RPM for 2 min, and the supernatant is discarded. We again precipitate the DNA by resuspending with 750μL of ethanol, and collect the DNA by centrifuging 13200 RPM for 2 min. The supernatant is discarded, and we let the tubes air dry. Finally, we resuspend the pellet with to 50ng/ul in Elution Buffer for the PCR reactions (approximately 3600ng of DNA are used for the PCR reactions).

PCR Amplification of the Barcode Locus

After extracting DNA, we PCR-amplified the barcode locus for each sample. Batches 1-6 and 10 were conducted with the protocols described in (Li et al., 2018; Venkataram et al., 2016). We made some slight modifications to this protocol, including using a new set of primers to allow for nested-unique-dual index labeling, for batches 7, 8, and 9. We detail this modified protocol here.

We used a two-step PCR protocol to amplify the barcodes from the DNA. The first PCR cycle uses primers with “inline indices” to label samples (see *Mitigating the effects of index hopping* section for details). Each primer also contains a Unique Molecular Identifier (UMI) - denoted by the sequence of “N” nucleotides in the primer - which is used to determine if sequences ultimately sequence result from distinct molecules during this first step of PCR (see []). Primers are HPLC purified to ensure they are the correct length.

Forward primers

Primer Name	Sequence
F201	TCGTCGGCAGCGTC AGATGTGTATAAGAGACAG NNNNNNNN CGATGTT TAATATGGACTAAAGGAGGCTTTT
F202	TCGTCGGCAGCGTC AGATGTGTATAAGAGACAG NNNNNNNN ACAGTGT TAATATGGACTAAAGGAGGCTTTT
F203	TCGTCGGCAGCGTC AGATGTGTATAAGAGACAG NNNNNNNN TGACCAT TAATATGGACTAAAGGAGGCTTTT
F204	TCGTCGGCAGCGTC AGATGTGTATAAGAGACAG NNNNNNNN GCCAATT TAATATGGACTAAAGGAGGCTTTT
F205	TCGTCGGCAGCGTC AGATGTGTATAAGAGACAG NNNNNNNN ATCACGT TAATATGGACTAAAGGAGGCTTTT
F206	TCGTCGGCAGCGTC AGATGTGTATAAGAGACAG NNNNNNNN CAGATCT TAATATGGACTAAAGGAGGCTTTT
F207	TCGTCGGCAGCGTC AGATGTGTATAAGAGACAG NNNNNNNN GGCTACT TAATATGGACTAAAGGAGGCTTTT
F208	TCGTCGGCAGCGTC AGATGTGTATAAGAGACAG NNNNNNNN TAGCTTT TAATATGGACTAAAGGAGGCTTTT
F209	TCGTCGGCAGCGTC AGATGTGTATAAGAGACAG NNNNNNNN TTAGGCT TAATATGGACTAAAGGAGGCTTTT
F210	TCGTCGGCAGCGTC AGATGTGTATAAGAGACAG NNNNNNNN ACTTGAT TAATATGGACTAAAGGAGGCTTTT
F211	TCGTCGGCAGCGTC AGATGTGTATAAGAGACAG NNNNNNNN GATCAGT TAATATGGACTAAAGGAGGCTTTT
F212	TCGTCGGCAGCGTC AGATGTGTATAAGAGACAG NNNNNNNN CTTGTAT TAATATGGACTAAAGGAGGCTTTT

Reverse primers

Primer Name	Sequence
R301	GTCTCGTGGGCTCGG AGATGTGTATAAGAGACAG NNNNNNNN TATATACGC TCGAATTCAAGCTTAGATCTGATA
R302	GTCTCGTGGGCTCGG AGATGTGTATAAGAGACAG NNNNNNNN CGCTCTATC TCGAATTCAAGCTTAGATCTGATA
R303	GTCTCGTGGGCTCGG AGATGTGTATAAGAGACAG NNNNNNNN GAGACGTCT TCGAATTCAAGCTTAGATCTGATA
R304	GTCTCGTGGGCTCGG AGATGTGTATAAGAGACAG NNNNNNNN ATACTGCGT TCGAATTCAAGCTTAGATCTGATA
R305	GTCTCGTGGGCTCGG AGATGTGTATAAGAGACAG NNNNNNNN ACTAGCAGA TCGAATTCAAGCTTAGATCTGATA
R306	GTCTCGTGGGCTCGG AGATGTGTATAAGAGACAG NNNNNNNN TGAGCTAGC TCGAATTCAAGCTTAGATCTGATA
R307	GTCTCGTGGGCTCGG AGATGTGTATAAGAGACAG NNNNNNNN CTGCTACTC TCGAATTCAAGCTTAGATCTGATA
R308	GTCTCGTGGGCTCGG AGATGTGTATAAGAGACAG NNNNNNNN GCGTACGCA TCGAATTCAAGCTTAGATCTGATA

For the first step of PCR, we performed 8 reactions per sample. For each set of 8 reactions, we used the master mix:

- 200μL OneTaq HotStart Polymerase Master Mix
- 8μL 10uM Forward primer
- 8μL 10uM Reverse primer
- 72μL sample genomic DNA (diluted to 50ng/μL or all of sample if between 25-50ng/μL)
- 16μL 50mM MgCl2
- 96μL Nuclease Free Water

50μL of the master mix was then aliquoted into 8 PCR tubes, and run with the following PCR reaction on the thermocycler:

1. 94°C for 10 min
2. 94°C for 3 min
3. 55°C for 1 min
4. 68°C for 1 min

5. Repeat steps 2-4 2x (for a total of 3 cycles)
6. 68°C for 1 min
7. Hold at 4°C

We then add 100μL of binding buffer from the ThermoScientific GeneJET Gel Extraction Kit (R1331) to each PCR reaction, and perform a standard PCR purification protocol in one column per sample. In the final step, we elute into 80μL of elution buffer.

For the second step of PCR, we use standard Nextera Index XT v2 primers. We use specific indices to uniquely dual-index each sample using our nested scheme (see *Mitigating the effects of index hopping* section for details). We performed 3 reactions of the second step PCR per sample, using the master mix:

- 1.5μL Q5 Polymerase
- 30μL Q5 Buffer
- 3μL 10mM dNTP
- 6.25μL i7 Nextera XT Primer ("N" primer)
- 6.25μL i5 Nextera XT Primer ("S" primer)
- 78μL purified step 1 PCR product
- 25μL Nuclease Free Water

This master mix was then divided into 3 PCR tubes per reaction, and run with the following protocol on a thermocycler:

1. 98°C for 30 sec
2. 98°C for 10 sec
3. 62°C for 20 sec
4. 72°C for 30 sec
5. Repeat steps 2-4 21 times (for a total of 22 cycles)
6. 72°C for 3 min
7. Hold at 4°C

We then added 100μL of binding buffer from the ThermoScientific GeneJET Gel Extraction Kit and purified the PCR product, eluting into 43μL.

Removal of the Ancestral Strain via Digestion and Gel Purification

To avoid the vast majority of our sequencing reads mapping only to the ancestor (and thus not being informative to relative fitness of the mutants), we use restriction digest to cut the ApaLI restriction site in the middle of the ancestor's barcode region. We mix 43μL of the second step PCR product with 2μL of ApaLI and 5μL of 10X Cutsmart incubate at 37°C for at least 2 hours (up to overnight). After digestion, we conducted size selection by running the digested sample on a gel, selecting the region between 350bp and 1000bp, and isolating the DNA using a standard ThermoScientific GeneJET Gel Extraction protocol for sequencing. Longer sequences were kept because of the possibility that some barcode sequences may selectively form complexes with themselves or other barcodes.

Note that for some samples, we also digested the ancestor before PCR, in addition to after PCR, to increase the yield of the library preparation . For these samples, we mixed 80ul of genomic DNA (at concentration 50ng/μL) with 10μL of 10X Cutsmart and 2μL of ApaLI and incubated 37°C for at least 2 hours (up to overnight). This product was then used as the template for PCR step 1 (with appropriate water volume adjustments to ensure 50μL reactions).

Sample pooling and Amplicon Sequencing

We used Qubit High Sensitivity to quantify the concentration of the final product for each sample, and pooled them in equal frequency for sequencing. Our samples were then sent to either Novogene (https://en.novogene.com/) or Admera Health (https://www.admerahealth.com/) for quality control (qPCR and either Bioanalyzer or TapeStation) and sequencing. We used 2x150 paired-end sequencing along with index sequencing reads on Illumina HiSeq machines using patterned flow cell (either HiSeq 4000 or HiSeq X, and see *Mitigating the effects of index hopping* for a discussion of our reduction of the effects of index hopping despite using these patterned flow cell machines). Samples were sequenced with at least 20% genomic DNA (either whole genomes from an unrelated project or phi-X) including in the lane to ensure adequate diversity on the flow cell.

Technical replicates

We performed several technical replicates for …

Mitigating the effects of index hopping

To reduce the effects of index hopping observed on Illumina patterned flow cell technology (including HiSeq 4000, HiSeq X, and Novaseq machines) , we devise a nested unique-dual-indexing approach using the combination of inline indices from our first step PCR primers and Illumina indices read in a separate Index Read (in our case Nextera indices) to uniquely label all samples fun on a flow cell.

In this approach, each inline index is always paired with an associated Nextera index on the other end of the amplicon. For example, the F1 primer is paired with the first i7 index, and the R1 primer is paired with the first i5 index. We then combinatorially combine these fixed pairs (twelve F-i7 pairs and eight R-i5 pairs) to label up to 96 samples that can be run on the same lane of sequencing. From this approach, we can identify and remove any single-index swaps from primers not removed from the library (the dominant source of index hopping according to the Illumina whitepaper) as well as template swapping events, as these events would result in an unpairing of our fixed F-i7 (or R-i5) pairs of primers.

To reduce the effect of index hopping contamination on our results, we included only samples that were sequenced on non-patterned flow cell technology (HiSeq 2000 and 2500 for samples in batches 1-6, 10, NextSeq for samples in batch 9) or were sequenced on patterned flow cell technology (patterned flow cell HiSeq) with nested unique-dual indexing for samples in batches 7, 8, and 9.

Processing of Amplicon Sequencing Data

We initially processed the amplicon sequencing data by de-multiplexing the data by identifying indices to match reads to their samples, mapping reads to a known list of barcodes generated by Venkataram et al, removing PCR duplicates using the UMIs from the first-step primers, and counting the number of reads for each barcode in each sample. The source code for this step can be found at []. We processed all raw data for this study using the same pipeline, including re-processing the raw sequencing files for data from previous studies (Li et al., 2018; Venkataram et al., 2016).

Briefly, we

QUANTIFICATION AND STATISTICAL ANALYSIS

Fitness Estimate Inference

[see venkataram et al]

Inference of selection coefficient from barcode frequencies.

[see venkataram et al]

Noise model

[essentially same as Venkataram et al, additional check with spike-ins]

Classifying mutants by mutation type

Mutants were classified into various mutation types based on previous whole genome sequencing [cite venkataram]. Most mutants were classified based on the gene of the putative causal mutation (i.e. mutation in a recurrently-hit gene and/or gene in either the RAS/PKA or TOR/SCH9 pathways). Because there are clear differences in fitness between missense and nonsense/frameshift/indel mutations in IRA1, these mutant were divided into a “missense” and “nonsense” classes, where mutants with frameshift and indel mutations were classified as “nonsense”. Additionally, diploid mutants were divided into two groups based on their average fitness effect across the 9 batches of the evolution condition - those with average fitness greater than 2 standard deviations above the average of the diploids were classified as “high-fitness diploids”.

Calculation of Weighted Average Z Score

To classify conditions as subtle or not-subtle perturbations from the evolution condition, we empirically quantify the typical difference in fitness for each condition from the average across the 9 batches of the evolution condition. We first quantify the typical difference in fitness for each batch condition for each mutant:

$$\sigma_i = \sum_{j}^{batches} \left| f_{ij} - \overline{f_i} \right|$$

where σ_i^2 represents the variance in fitness across the batches for mutant *i*, and $\overline{f_i}$ represents the average fitness of mutant *i* across the batch conditions. To ensure that each mutation type contributes equally to our classification of how different each condition is from the evolution condition, we weight each mutant’s contribution to the overall condition difference based on the number of mutants with the same mutation type, such that the mutation-type-weighted average Z score for a given condition *j* is given by:

$$z_j = \sum_{i}^{mutants} \frac{\left| f_{ij} - \overline{f_i} \right|}{n_{type(i)} \sigma_i}$$

where $n_{type(i)}$ represents the number of mutants that are the same mutation type as mutant *i.*

Model of Fitness-Relevant Phenotypes

We explicitly model each mutant as having a fixed phenotypic effect, represented by a vector of *k* phenotypes. Each of the *k* phenotypes has a fixed weight in each environment, such that the linear combination

$$m_{i1}c_{1j} + m_{i2}c_{2j} + m_{i3}c_{3j} + \ldots + m_{ik}c_{kj} = f_{ij}$$

represents the fitness of mutant *i* in condition *j*.

Using Singular Value Decomposition to decompose the fitness matrix

This linear representation of the combination of mutant phenotype and weights in each environment allow us to use use Singular Value Decomposition (SVD) to decompose the fitness matrix Fas:

$$M\Sigma C^T = F$$

The left hand side of this equation consists of three matrices:M, which represents the positions of the mutants in the phenotype space, C^T, which represents the phenotypic weights of the environments, and Σ, a diagonal matrix representing the singular values of the fitness matrix F. Though the singular values are informative in this separation of three matrices, we can also think of this decomposition into two matrices, where we fold the singular values into either the mutant phenotypes or the environment weights. This decomposition fully captures the data represented in the fitness matrix F, including data generated by measurement error rather than underlying biological signals.

Importantly, the Eckart-Young-Mirsky theorem states that taking the first *k* singular values and the corresponding values of first *k* phenotypic components for each mutant and environment represents the best possible rank *k* approximation of the matrix Fif evaluated via the sum of least squares [cite]. This means that, for example, that the best possible approximation of the matrix $F$with only one parameter for each mutant and each condition (along with scaling factor given by the singular value) is given by the first entry for each mutant, each condition, and the first singular value. This also means that the contribution of each singular value is such that the *k*th singular value is greater than or equal to the *k+1*th singular value.

In order to study only components generated by biological signals and not measurement noise, we remove very small components that are likely to be noise (this is known as “hard thresholding”). However, we must also keep enough components such that we capture as much of the detectable signal as possible. This overfitting problem is common in statistics, and several methods have been devised to select the appropriate number of components to include.

Estimating the detection threshold using measurement error

One method to select the appropriate number of components to include in the model and prevent overfitting is to use known measurement error to estimate the size of the singular values generated from measurement error alone. The distribution for well-behaved noise distributions (i.e. the same for all entries of the matrix) is well-characterized [cite Sengupta and Mitra 1999]. Despite this not being the case for our data, we can estimate the noise-derived singular values empirically. Explicitly, we simulate many matrices, with each entry pulled from a normal distribution centered at zero and with variance given by the estimated measurement variance of the corresponding fitness measurement for that entry of the matrix. We then apply SVD to this noise-only matrix, which will give us a set of singular values generated only by noise. From many such simulations, we can take the average size of the largest singular value, which corresponds to the expected size of the largest component of noise, as our threshold of detection. This approach is analogous to identifying a threshold when measurement noise is known but not identical for all entries in the matrix suggested in (Josse and Sardy, 2014).

Estimating detection threshold using bi-cross-validation

Another method for identifying the appropriate number of components is to use predictive power for selection. This method relies on the intuition that measurement error is uncorrelated, so the

inclusion of a component that represents some of this measurement error should actually hurt the predictive power of new data not included in the estimation of the component. We use a bi-cross-validation scheme of the SVD devised by Owen and Perry (2009), which divides the mutants and conditions into distinct groups of training and testing sets. This subsequently divides the matrix of fitness measurements into 4 submatrices: the fitness of the training mutants in the training conditions (*D)*, the fitness of the training mutants in the testing conditions (*C)*, the fitness of the testing mutants in the training conditions (*B),* and the fitness of the testing mutants in the testing conditions (*A*).

$$F = \begin{pmatrix} A & B \\ C & D \end{pmatrix}$$

SVD is then carried out on the training data (submatrix *D*), which returns a set of singular values and corresponding singular vectors which perfectly captures the fitness data in *D*. We then use each of the increasingly complex models represented by including more of the returned singular values to predict the fitness of the testing mutants in the testing conditions (submatrix *A*). We then select the number of components to include in the model that has the best predictive power on average across choices which mutants and conditions to designate as the test set.

Explicitly, we use the formulation proposed by Owen and Perry 2009 for the prediction of the held-out submatrix *A*:

$$\hat{A} = B\left(\hat{D}^{(k)}\right)^{+} C$$

where $\left(\hat{D}^{(k)}\right)^{+}$ denotes the Moore-Penrose inverse of the rank *k* approximation of sub-matrix *D*. This prediction is equivalent to fixing the locations of the training mutants and conditions, independently using least-squares regression to identify the best possible estimate for the testing mutants and conditions, and then using these estimated locations to predict the fitness of the testing mutants in the testing conditions (Owen and Perry 2009).

Simulating phenotype space

Division of Mutants into Training and Testing Sets

In order to train our phenotype space

Calculation of Weighted Coefficient of Determination

To evaluate the model's predictive power, we used a measure of predictability ($\tilde{R}^2$) that weights the contribution of each mutant to overall variance explained based on the number of mutants that share its mutation type (diploids, IRA1 nonsense, IRA1 missense, GPB2, etc.). The use of this weighting is conservative, as it reduces the inflation of prediction that could be caused by the inclusion of similar mutants (with the same mutation type) in both the training and testing sets. We also note that the prediction results are qualitatively similar when using a standard variance explained measure (see Supplemental Figures). For overall predictive power across all mutants and conditions, we use the measure:

$$\tilde{R}^2 = 1 - \frac{\sum_i^{mutants} \sum_j^{conditions} \frac{1}{n_{type(i)}} \left(f_{ij} - \widehat{f_{ij}}\right)^2}{\sum_i^{mutants} \sum_j^{conditions} \frac{1}{n_{type(i)}} \left(f_{ij} - \bar{f}\right)^2}$$

where $\bar{f}$ denotes the average fitness for all evaluated mutants and evaluated conditions. For predictive power per condition *j*, we use a similar measure:

$$\widetilde{R_j}^2 = 1 - \frac{\sum_i^{mutants} \frac{1}{n_{type(i)}} \left(f_{ij} - \widehat{f_{ij}}\right)^2}{\sum_i^{mutants} \frac{1}{n_{type(i)}} \left(f_{ij} - \bar{f}_j\right)^2}$$

where $\bar{f}_j$ denotes the average fitness across all evaluated mutants in condition *j.*

Note that this measure has a range between negative infinity and 1, where negative values indicate that the weighted squared error from the prediction is larger than the weighted squared error from the average fitness in the condition.

DATA AND CODE AVAILABILITY

Data Resource

The raw Illumina sequencing data for the fitness measurement assays can be found at [].

Code

The software repository for the barcode counting code can be found at []
The software repository for the fitness estimate inference can be found at [].
The code for all downstream analysis, including figure generation can be found at https://github.com/grantkinsler/1BigBatch.

KEY RESOURCES TABLE