Inferring phenotype from population structure

Carl F Schwendinger-Schreck

One of the most time-consuming steps involved in performing a genetic screen is selecting for the desired phenotype. Here, I decsribe a fast and inexpensive machine learning protocol for genetic screening in which phenotype is inferred from spatial population structure.

Even in a model organism as simple as yeast, scanning through thousands of colonies can require massive manpower, sophisticated image analysis tools, and a suite of automation. No matter how the phenotype manifests, whether in an easily-screened assay like expression/repression of a fluorescent reporter or in a more natural change in growth dynamics that may leave behind no readily-visible signature, we want to distinguish the phenotypes using as little data as possible. To highlight the value of machine learning in genetic screening, I use here bud-site selection *S.cerevisiae* (budding yeast) mutants—where the final cell morphology is normal (i.e. round), but the dynamics of growth differ (whether buds form axially or bipolarly or neither, see Fig. 1). In this case, I've chosen single timepoint images of each colony so that timelapse movies are not required. Industrial applications often wish to identify hybrid strains, generated by mixing two haploid strains. Hybrid strains, in contrast to their parents, are diploid and therefore bud using a polar program.

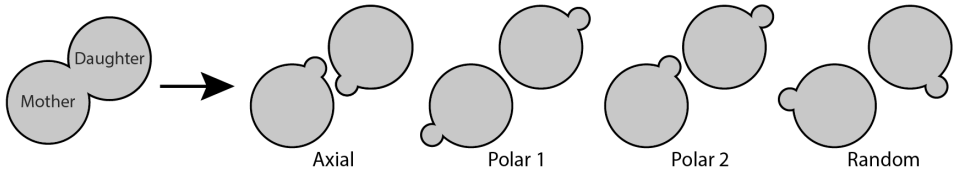

Figure 1: *Bud-site selection in budding yeast. Daughter "buds" grow until mother and daughter split and each form new buds. Axial budding occurs in haploid cells, polar budding in diploid cells, and random budding in Rsr1 mutants.*

To test our protocol, I created thousands of budding cell colonies using physics-based computational simulations. These simulated colonies are innoculated with a single cell and expand outward as cells grow and push their neighbors. In order to mimic how S. cerevisiae colonies grow in the lab, I constrain growth to 2D (as if colonies are growing on flat plate) and restrict cellular growth to a region near the colony edge (as if nutrients are diffusing in from outside the colony). Fig. 2 shows two example colonies from each budding type.

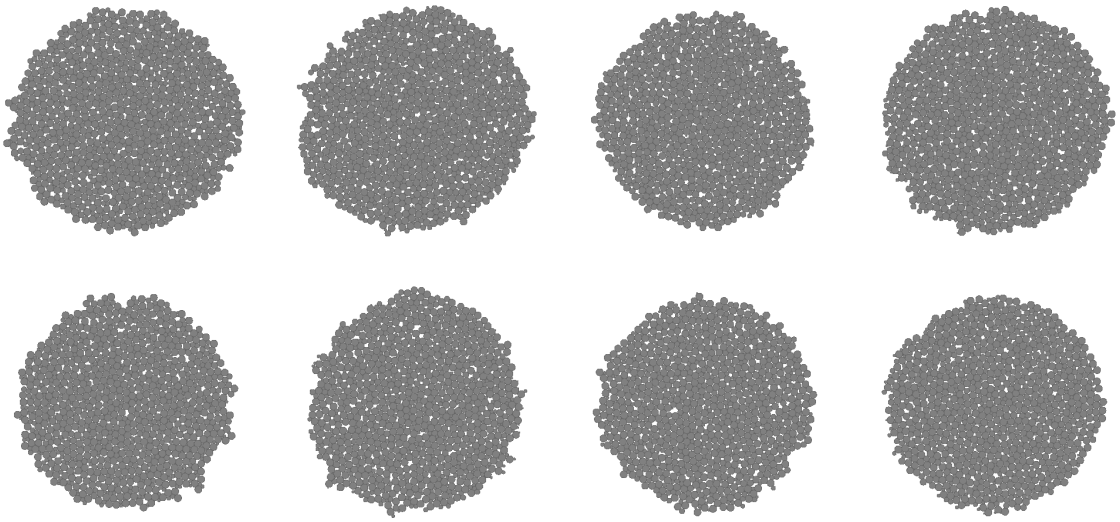

Figure 2: *Examples colonies compoased of (left to right columns): axial, polar 1, polar 2, and random types.*

Our goal is, from these images alone, to develop an algorithm to distinguish which budding type corresponds to which colony. It is not clear whether we can succeed since the colonies shown in Fig. 2 are difficult to distiguish even by eye, but such a protocol could be a very powerful method for phenotypic screening.

To train my model, I simulated 2400 colonies for each budding type. I use axial, polar 1, and random types to train and test my model, and polar 2 as a seperate test case. These three training/testing types have a total of $3 \times 2400 = 7200$ images which are randomly sorted and assigned as training (5040 images), cross-validation (1080 images), and testing (1080 images). Since I perform training on three budding types, a poor-performing learning model that randomly assigns each colony a label would have an accuracy (fraction of colony classified correctly) of around $A_0 = 1/3$, which is the baseline accuracy that I will compare my models to.

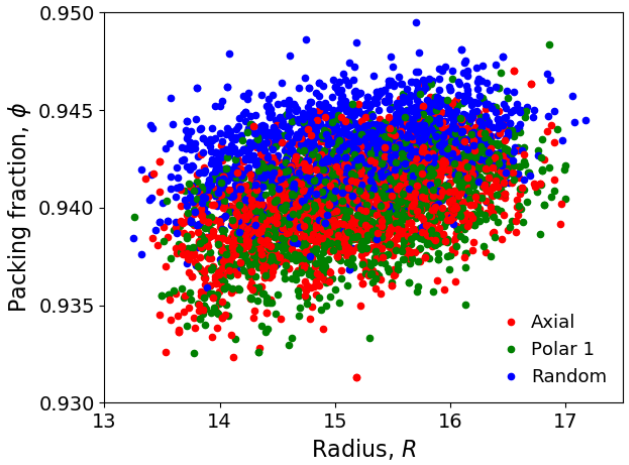

Figure 3: *Colony radii (in units of mother cell widths) and packing fractions measured from colony images for three budding types. Data is shown for the training data set, which has has* 5040 *colony images in total.*

To get a feeling for the differences between budding types, I first measure rough physical characteristics from colony images: the colony radius R and "packing fraction" ϕ (fraction of colony occupied by cells). In Fig. 3, I see similar variations in R for different budding types (due to variation in initial cell size) but that random budding cells produce slightly denser (larger ϕ) colonies. This isn't a whole lot to go on, but suggests that if we try a little harder we may find structural signatures that distinguish different budding types.

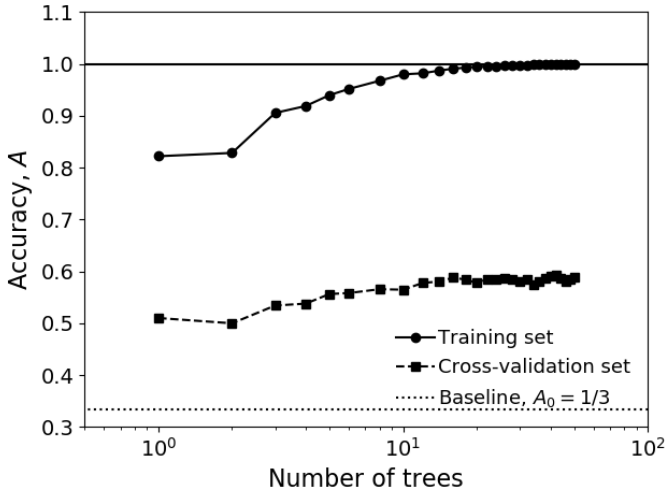

Figure 4: *Accuracy (fraction of types classified correctly) as a function of number of trees for a random forest classifier (RandomForestClassifier from sklearn in Python) based on Fig. 3 data. I cite test values for* 50 *trees.*

Before attempting to tackle more complex datasets, I train a classification model on the R/ϕ data in Fig. 3, which I refer to as "model 1". Using a random forest classifier, I find that > 20 trees is sufficient to ensure the cross-validate error is maximal (Fig. 4). The accuracy $A_1^{\text{test}} = 0.60$ for our test set is better than $A_0 = 1/3$, but still far from the ideal of $A = 1$. To see how model 1 is failing, we look at the "confusion matrix":

Predicted label				
Axial	*Polar 1*	*Random*		
0.54	0.30	0.15	*Axial*	**True label**
0.34	0.50	0.16	*Polar 1*	
0.11	0.12	0.77	*Random*	

Table 1: *Normalized confusion matrix (fraction of true labels classified as predicted labels) for model 1.*

We see that random budding type colonies are indeed distinguished from the axial and polar 1 types, as 0.77 of random colonies are predicted as such. However, as one may have guessed from Fig. 3, axial and polar 1 budding types are misidentified nearly as often as they are properly classified.

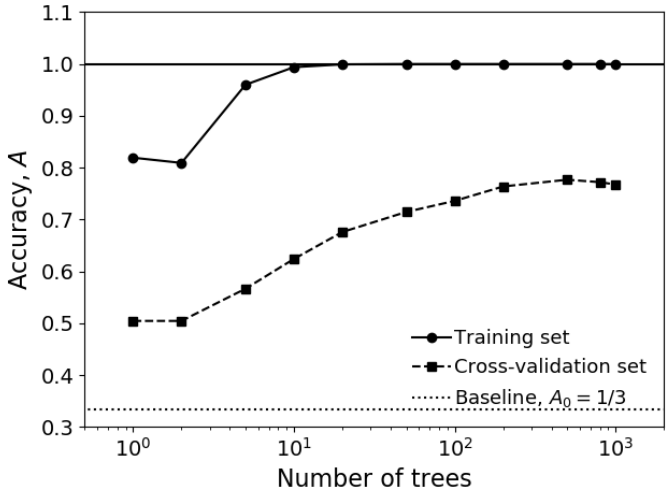

Figure 5: *Accuracy (fraction of types classified correctly) as a function of number of trees for a Random Forest classifier based on colony images (Fig. 2). I cite test values for* 1000 *trees.*

In order to improve upon model 1, I now build a model using the entire 480×480 pixel images ("model 2"). The number of features per image increase from 2 to $480^2 = 230400$ from model 1 to model 2. A random forest classifier, with 1000 trees to maximize the cross validation accuracy (Fig. 5), gives much better results ($A_2^{\text{test}} = 0.79$) than model 1. This improvement primarily stems from much better predictions of axial and polar 1 budding types, for which the fraction of correctly-classified colonies increases by ≈ 0.25.

Predicted label				
Axial	*Polar 1*	*Random*		
0.69	0.17	0.14	*Axial*	**True label**
0.11	0.75	0.14	*Polar 1*	
0.05	0.01	0.94	*Random*	

Table 2: *Normalized confusion matrix (fraction of true labels classified as predicted labels) for model 2.*

In order to further improve on model 2, I turn to neural networks. A small neural network with two hidden nodes ("model 3") and regularization paramter $\lambda = 1$ yields similar results to model 2, with an accuracy of $A_3^{\text{test}} = 0.77$. Model 3 improves upon model 2 in some regards and worsens in others (see confusion matrix

below), but it is suprising that model 3 works as well as it does given that the images in Fig. 3 are nearly indistinguishable by eye and that the neural network in model 2 uses only two hidden nodes.

Predicted label				
Axial	*Polar 1*	*Random*		
0.77	0.14	0.08	*Axial*	**True label**
0.18	0.69	0.12	*Polar 1*	
0.14	0.01	0.85	*Random*	

Table 3: *Normalized confusion matrix (fraction of true labels classified as predicted labels) for model 3.*

To get a feeling for why model 2 works so well, I show the two hidden parameters in Fig. 6. These parameters seem to be capturing spatial fluctuations in both colony radius R and packing fraction ϕ, whereas model 1 only considered mean values of R and ϕ.

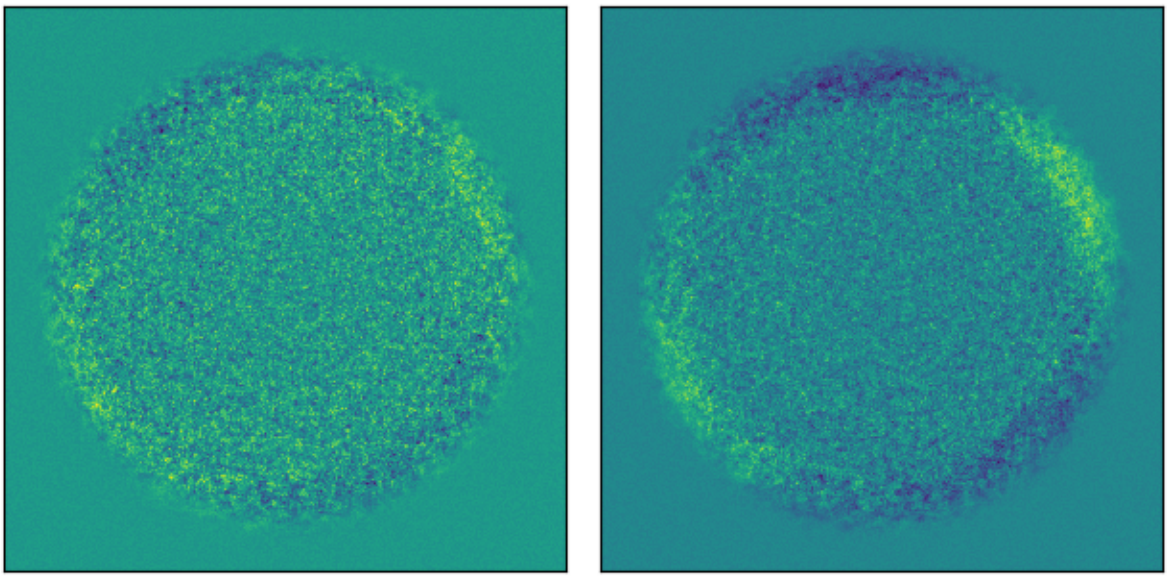

Figure 6: *Neural network hidden nodes for model 2 with two hidden paramters and regularization paramter* $\lambda = 1$.

We can further improve on model 3 by adding more hidden layer nodes ("model 4"). Using 40 nodes and a regularization paramter $\lambda = 1$ increases the accuracy to $A_4^{\text{test}} = 0.84$, stemming from improvements in all types with respect to model 3. Interestingly, model 4 does not acheive an accuracy for the random budding type as high as we have found for the random forest classifier (0.94 for model 2).

Predicted label				
Axial	*Polar 1*	*Random*		
0.84	0.09	0.08	*Axial*	**True label**
0.10	0.80	0.10	*Polar 1*	
0.08	0.05	0.87	*Random*	

Table 4: *Normalized confusion matrix (fraction of true labels classified as predicted labels) for model 4.*

Many of parameters of model 4 (Fig. 7) capture fluctations in R and ϕ similar to those of model 2 (Fig. 6), athough other parameters do not have easily-discernable structure. It is not clear to me that model 4 (or even model 3) is has found the best parameter set since so many of the parameters lack structure, so the neural network models would benefit from careful regularization, different cost function minimization routines, or an entirely different network architecture.

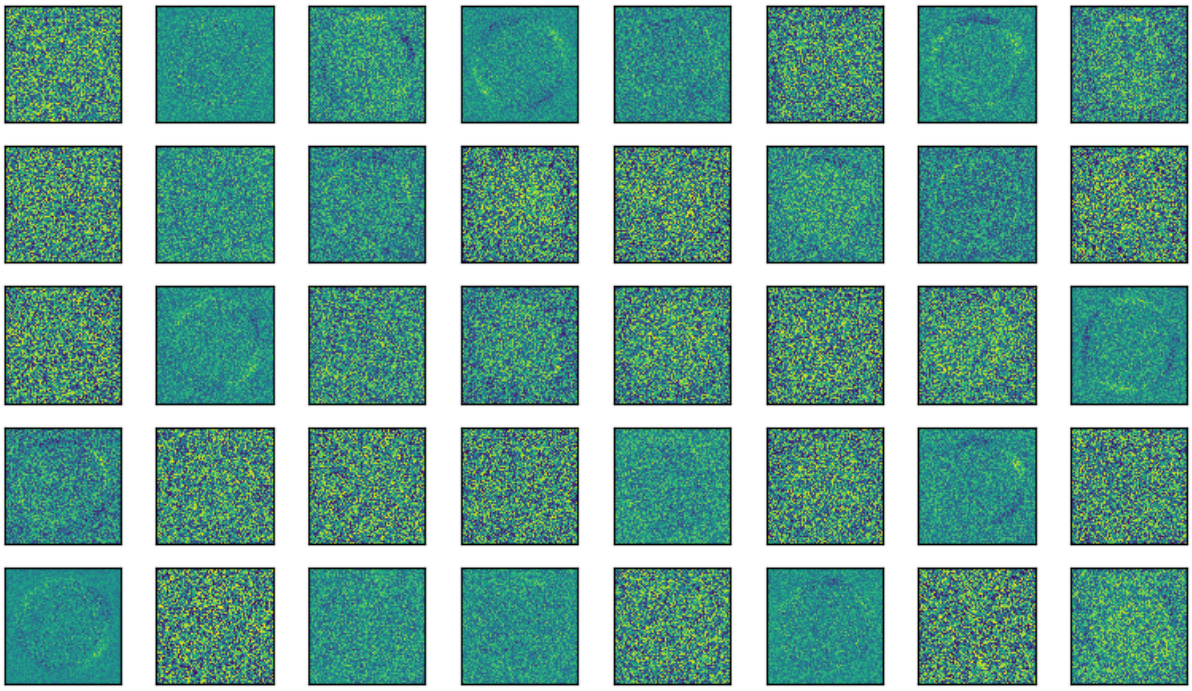

Figure 7: *Neural network hidden nodes for model 3 with 40 hidden paramters and regularization paramter* $\lambda = 1$.

To summarize what I have found, the growth of simulated budding cells in spatial settings produce phenotypic-depedent population-level patterns that allow us to determine (up to an accuracy of 0.84 for model 4) which phenotype a cell colony is composed of. We can acheive this high degree of accuracy even though the colonies have no readily-visible distinguishing features to the naked eye. This method may provide both tremendous cost savings as it circumvents additional time consuming and expensive phenotypic screens.