For each discrete feature , we calculate the probability that it overlaps with common polymorphisms. We then calculate the information content to denote its weighted value .

The situation is more complex for continuous features, as different feature values have different probabilities of being observed in natural polymorphisms. Thus, one weight cannot suffice for varied feature values. For a continuous feature , which is associated with a score , we will calculate feature weights for each. In particular, we discretize at each value and compute . Then we fit a smooth curve for all to obtain continuous . When we evaluate the continuous featurefor a particular variant, we calculate its weighted value using the fitted function.

We score each variant by summing up the weighted values of all its features. We will also consider the feature dependency structure when calculating the scores (e.g., removing redundant features or performing dimension reduction techniques).

**D-2-b Research plan for Aim 2**

**D-2-b-i Statistical framework for parameter tuning using Bayesian updates**

The initial feature weights () (given number of features) assigned in D-1-b-v will be further optimized with newly available “gold standard” datasets. We plan to tune these parameters using an incremental Bayesian learning strategy. For a variant , given feature values , can be rewritten as (), with functions depicts the relationship between and . All parameters inare the same for all variants. Given the eleVAR score (equation 3 in D-1-b-v), the probability that is functional ( designates a positive result, whereas denotes a negative result) follows a logistic function ( are scaling parameters). To update (more specifically, the parameters in functions ) with training data , we implement Bayes’ rule:. The probability of observing (given andfeature values corresponding to variants in ) is proportional to the probability of observing given and , multiplied by the prior probability of . Assuming independency between data points in , which can be achieved by proper training data construction, , given observations in .

Using the training data, we will maximize this function to find the most probable functions , and these will be used as our updated parameters. The updated will then be used as tuned parameters in eleVAR to prioritize variants. The procedure will be iterated in several rounds. In the first round of tuning, feature weights obtained in D-1-b-v will be used to construct priors . In subsequent rounds, the updated weights will be set as new priors.