S1 Ternary Phase Diagram

The energies of an example ternary system were visualized in Fig 6 of the main body. Here, we show the underlying distribution of energies relative to the hull. There are 66 compositions per phase, resulting in 198 total phase-composition pairs. Seven of the phase-composition pairs are on the hull–five belong to the blue phase and other two are from the pink phase. The orange phase does not contribute to the hull.

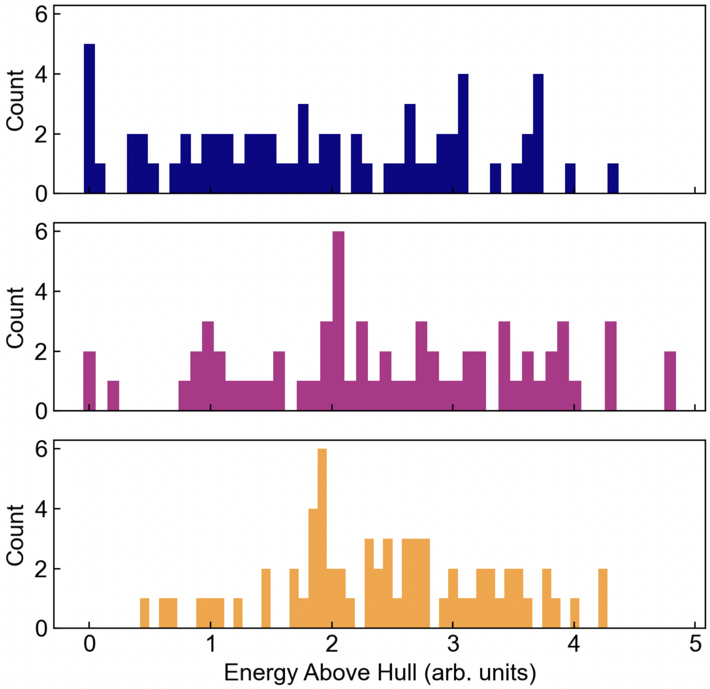

Figure S1 Here we more closely examine the energies from Fig. 6 (of the main body) by plotting the distribution of energies relative to the convex hull. Each phase is colored in the same way as it is in Fig. 6. Seven composition-phase pairs are on the hull–five from the blue phase and two from the purple phase.

S2 Converging Computational Parameters

The parameters used in CAL will affect the efficiency with which CAL learns about the convex hull. In this current implementation, the tunable parameters for the active learning process are K, m, and the hyperparameters of the Gaussian Process (GP). Effectively tuning GP hyperparameters is a rich sub-field, but for brevity, we will focus on the parameters that are specifically introduced by CAL. As a reminder, K is the number of fantasized energy-values for which we calculate the information gain. For a given phase-composition pair, the information gain is calculated K times, and those values are averaged together to estimate the expected information gain. The other parameter, m, is the number of samples from the GP that are used to calculate the convex hull distribution and its entropy.

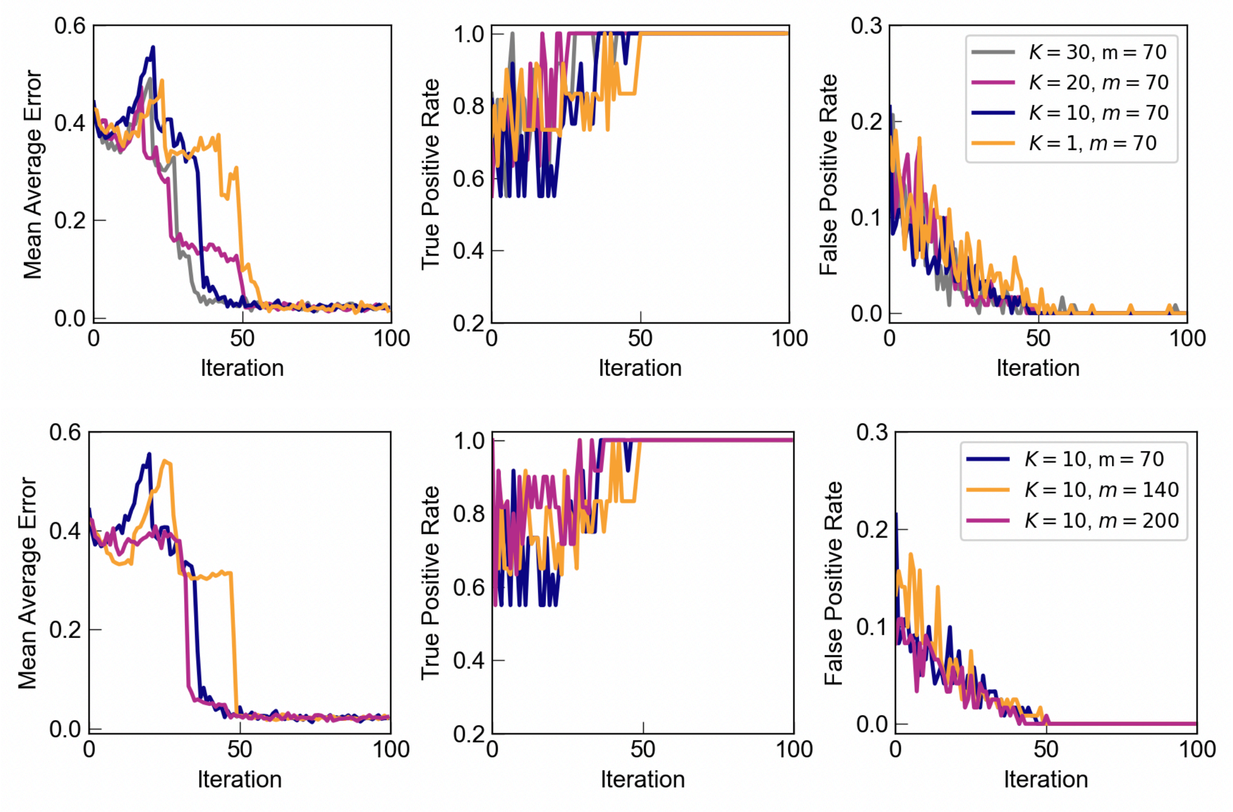

Figure S2 We find that performance converges quickly with increasing K. In the bottom row, we find that performance does not vary significantly with increasing m. To be conservative, we use a $K = 10$ and $m = 200$. Testing is conducted on a ternary composition space with three different phases. Performance metrics are averaged across three different random seeds, corresponding to varying energy surfaces.

While we provide general rules for selecting m and K in Section 6.4 of the main body, these serve only as starting points; users are encouraged to run their own convergence tests for their systems of interest. One simple way to test m and K is the following. First, use exceptionally stringent CAL parameters to obtain the EIG across all phase-composition pairs. Then, repeat the calculation with increasingly loose parameters until the general trends in the EIG are lost. Of course, the proposed test is dependent on the current state of the system. As such, one may want to conduct these tests for multiple sets of data, or different subsets of the currently known data.

In this work, we are in the somewhat unusual situation of already having access to the "ground-truth" energy surfaces. For benchmarking purposes, we run end-to-end testing, observing whether the overall performance is dependent on CAL parameters. For Fig. S2, we test on a ternary composition space with three phases, as was done in the main body. These results are averaged over three different seeds (and thus, three different sets of ternary energy surfaces) to give the performance.

To begin, we will examine the top row of panels, corresponding to testing K. While performance is theoretically dependent on K, for our example, we find this dependence is relatively insignificant.

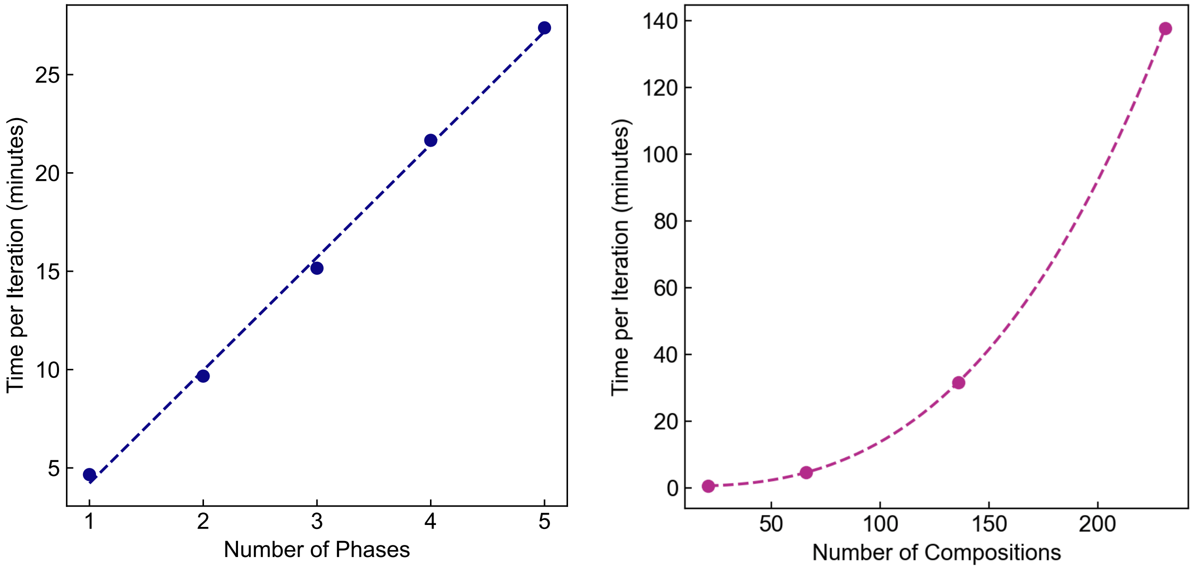

Figure S3 The computational cost of CAL, as assessed by time per iteration, changes with both the number of phases and compositions in the system. The cost scales linearly with the number of phases, while it scales cubically with the number of compositions. Dashed lines are used to show linear and cubic fits, respectively. All assessments were run for a ternary composition space.

Namely, the only measurable difference in performance comes in the mean average error of the hull when going from $K = 1$ to $K = 10$. The classification errors (i.e., true positive rate, false positive rate) seem to be similar as well. These results are in alignment with what was argued in the main text–K is used to approximate the expectation function of a one-dimensional integral, and as such, the integral should converge with fairly few samples. To balance cost and performance, we choose a $K = 10$.

In the second row, we test the parameter m. Remember that to calculate the entropy of the convex hull distribution, we need at least as many samples as the number of compositions. As such, all m values used were chosen to be above 66 since that is the number of compositions in this case example. We find that performance does not vary significantly with changing m. Admittedly, using a smaller m may have been feasible for this particular application, but to be conservative and consistent, we set $m = 200$ for all case examples in the Results.

S3 Computational Scaling

The computational cost of CAL is dependent on the complexity of the composition space. For all tests, we used a $K = 10$, just as was done in the main body. The number of samples, m, was scaled such that $m = 3c$, where c is the number of compositions. Again, this was done to match the parameters used in the main body. All evaluations were parallelized across six cores. Namely, the EIG evaluations were split such that each core received an equal (or close to equal as possible) number of EIG calculations to conduct. Once the EIG calculations were completed, they were pooled together and the phase-composition pair with the maximum EIG was chosen. The number of compositions was increased by moving to increasingly fine composition grids for a ternary space. Similarly, the number of compositions could also be increased by moving to higher dimensional composition spaces. For the purposes of evaluating computational cost, the two are equivalent. As can be seen in Fig. S3, cost scales linearly with the number of phases and cubically with the number of compositions in the space. Dashed lines correspond to linear and cubic fits, respectively.