Supporting Information for Active Learning High Coverage Sets of Complementary Reaction Conditions

Sofia L. Sivilotti,[†] David M. Friday,[†] and Nicholas E. Jackson[*,†]

†*Department of Chemistry, University of Illinois at Urbana-Champaign, 505 S Mathews Avenue, Urbana, Illinois, 61801, USA*

‡*Department of Computer Science, University of Illinois at Urbana-Champaign, Urbana, Illinois 61801, USA*

E-mail: jacksonn@illinois.edu

Contents

Dataset Analysis **1**

Active Learning Strategy Comparisons **4**

Exploit Function Example **9**

Latin Hypercube Sampling **11**

Dataset Size Impact with B-H Dataset **11**

Featurization with Molecular Fingerprints **12**

Dataset Analysis

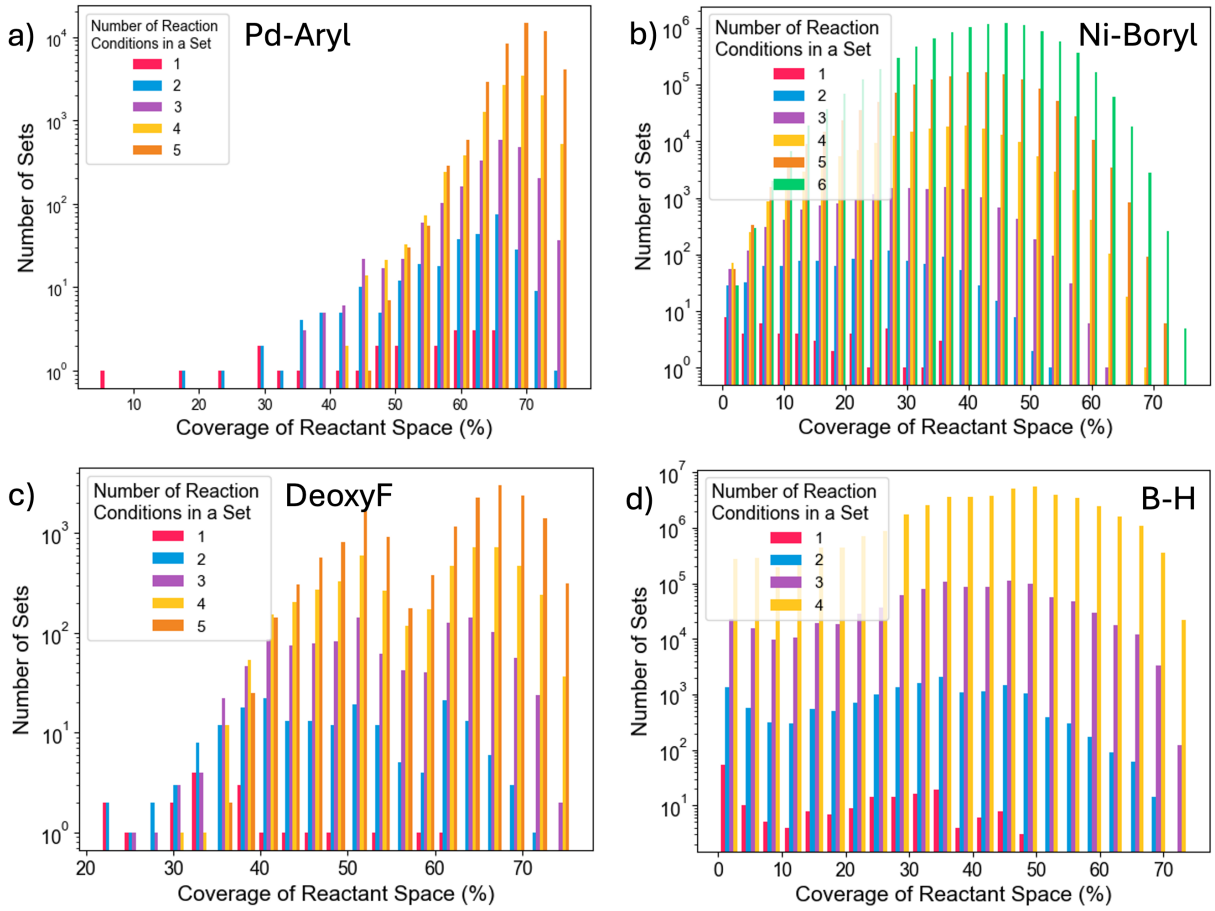

Figure S1: Histograms of the coverage of the reactant space by sets of reaction conditions of different sizes for the a) Pd-Aryl, b) Ni-Boryl, c) DeoxyF, and d) B-H datasets. Figure S1c represents the same data as Figure 2 in the main text, but with the full range of reaction space coverage on the x-axis.

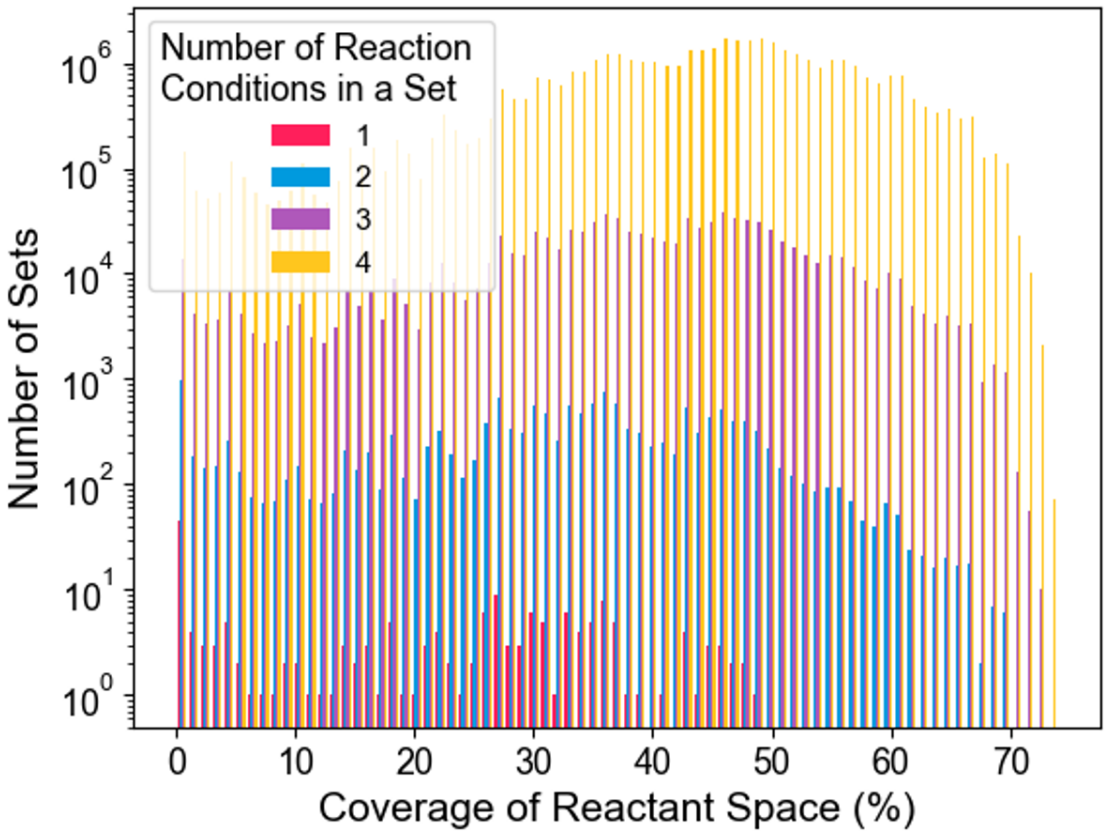

Figure S2: Histogram of the coverage of reactant space by sets of reaction conditions of up to size 4 on the B-H dataset with finer binning than Fig S1d, showing the improvement from 3 to 4 conditions in a set. Note that the maximum coverage of the space with a set size of 4 is 73%, while the maximum coverage with all sets is 75%. Enumeration of all sets of size > 4 was not performed due to the size of the B-H dataset.

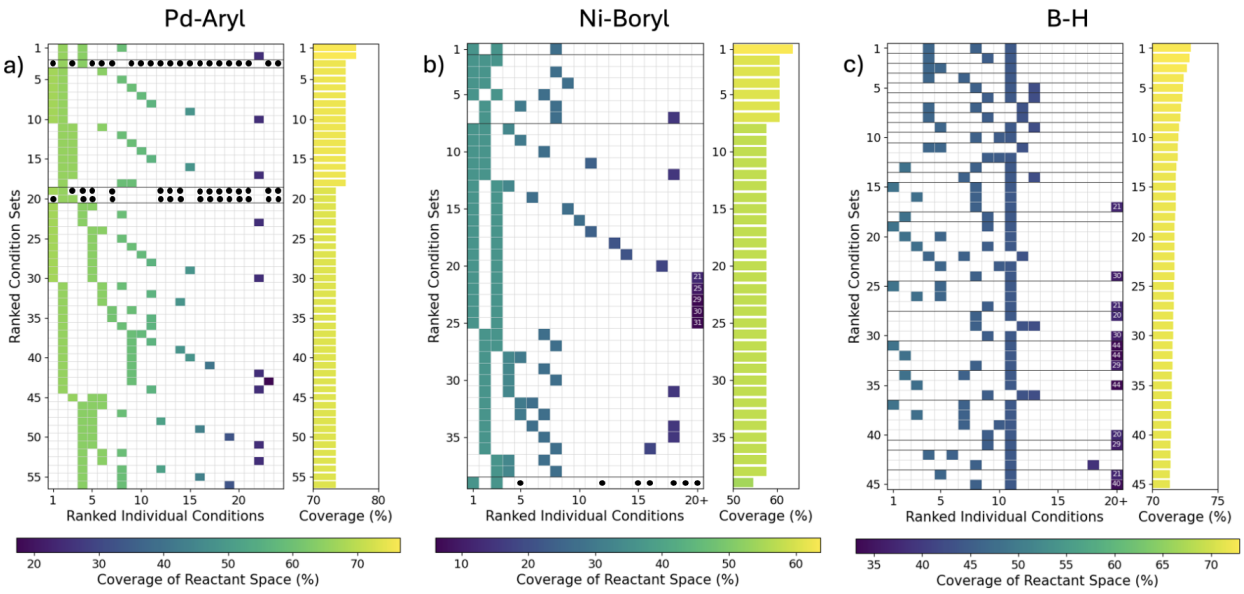

Figure S3: Grids showing the specific reaction conditions (ranked by coverage) within the highest coverage sets of reaction conditions for the a) Pd-Aryl, b) Ni-Boryl, and c) B-H datasets, with the coverage of each set on the right. The color of each grid point corresponds to the coverage of the individual reaction condition. For sets of two conditions, any 3rd reaction condition added to the set would yield the same coverage, unless listed in a higher-ranking set.

Active Learning Strategy Comparisons

Several Active Learning (AL) strategies were tested in this work, shown in Figures S4 - S6. Figure S4 shows the performance of several candidate Exploit acquisition functions using an RFC. The simplest and highest performing Exploit function is the Exploit function (Eq S1, also Eq 2, main text). This was compared to the Exploit* function (Eq S2), which also accounts for the likelihood of success of a selected reaction. This function proved to be lower performing in all simulations. In toy models, the Exploit* algorithm strongly avoided reactants which had failed under a few reactions (due to a lower $\phi_{r,c}$) preventing the algorithm from finding complementary conditions. This algorithm was also compared to the Exploit** function (Eq S3) which accounts for complementarity beyond pairs of 2 conditions, accounting for the coverage of sets of conditions (γ_S) up to size m. This algorithm was

much more computationally expensive because it needed to generate all possible sets of 3+ conditions, and did not perform substantially better than the Exploit* function. To reduce the computational cost of the Exploit functions, a method of stochastic batching was tested, where a random subset of conditions was considered when calculating the complementarity of each reaction. This approach suffered degraded performance on the three smaller datasets when a subset of only 5 conditions was considered.

$$\text{Exploit}_{r,c} = \frac{1}{|C|}[\gamma_{\{c\}} + \sum_{c_i \in C \setminus \{c\}} \gamma_{\{c,c_i\}}(1 - \phi_{r,c_i})] \tag{1}$$

$$\text{Exploit*}_{r,c} = \phi_{r,c}\frac{1}{|C|}[\gamma_{\{c\}} + \sum_{c_i \in C \setminus \{c\}} \gamma_{\{c,c_i\}}(1 - \phi_{r,c_i})] \tag{2}$$

$$\text{Exploit**}_{r,c} = \phi_{r,c}\frac{1}{|C_S|}\sum_{S \in C_S} \gamma_S \frac{1}{|S| - 1}\sum_{c_i \in S \setminus \{c\}} (1 - \phi_{r,c_i}) \tag{3}$$

$$C_S = \{\{c\} \cup S_n | S_n \in C^n \text{ and } n < m\}$$

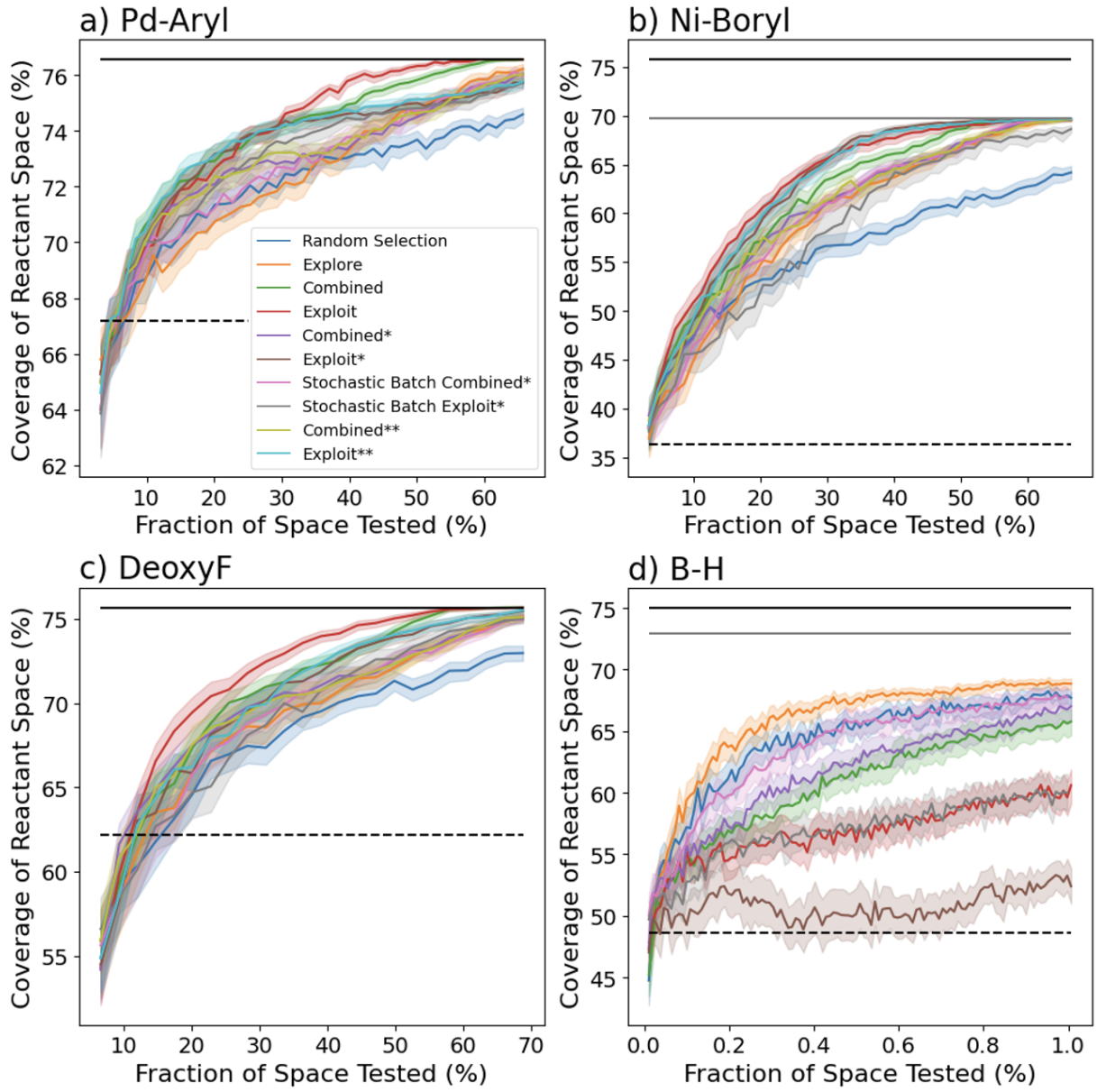

Figure S4: Coverage of the highest predicted set of three (or four for Ni-Boryl) reaction conditions using AL guided by several acquisition functions when using a Random Forest Classifier across all four datasets. Each line is the average of 100 runs, and the shaded region shows the 95% confidence interval of the mean. Dashed horizontal lines mark the coverage of the most general individual condition. Black solid lines mark the maximum possible coverage of the entire dataset while gray solid lines indicate the maximum coverage of a set of three (or four for Ni-Boryl) conditions. The B-H dataset run was terminated at 1% of the space as this corresponded to 4,500 reactions. The data presented is the same as Figure 4, main text, with additional acquisition functions.

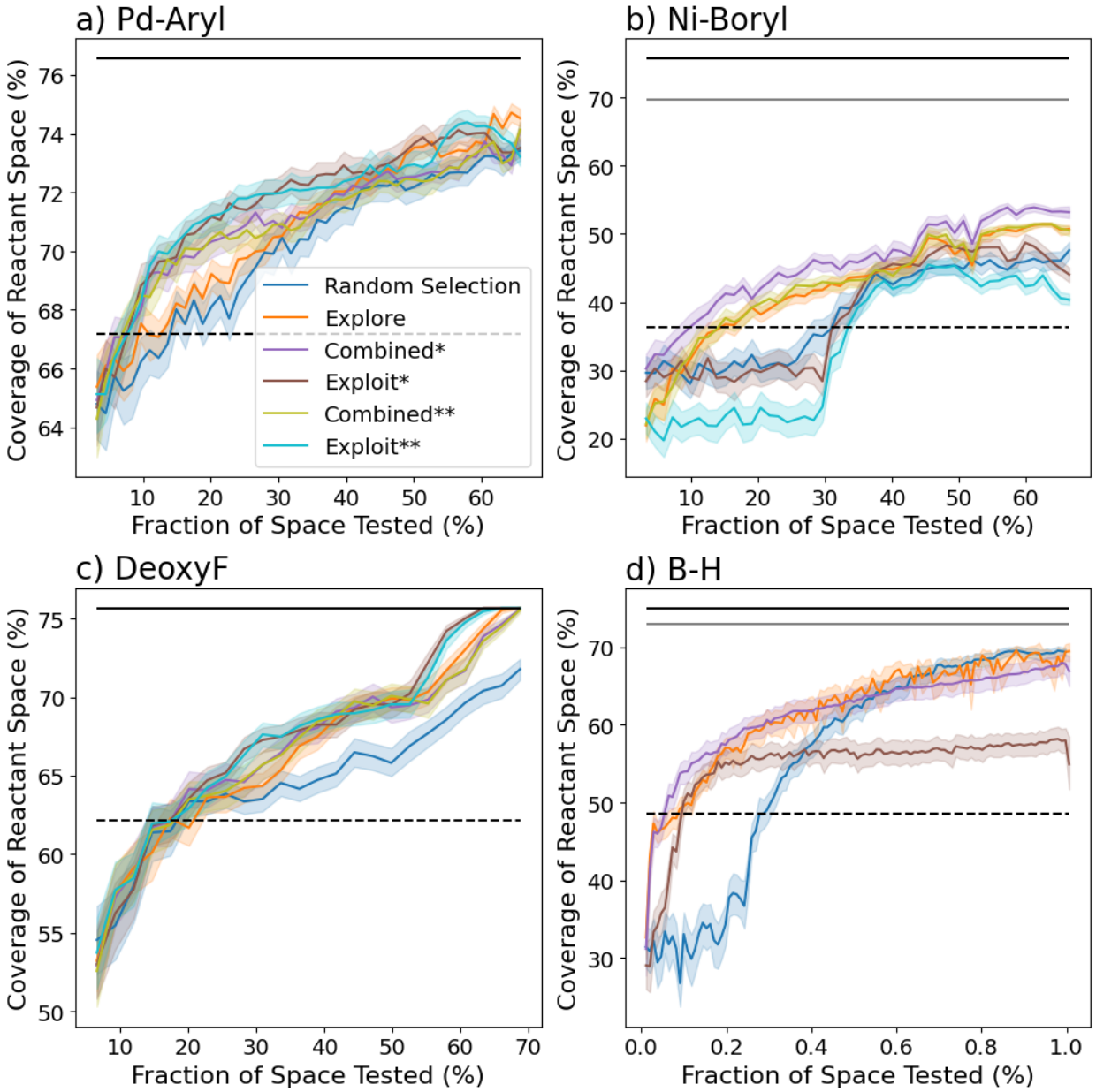

Figure S5: Coverage of the highest predicted set of three (or four for Ni-Boryl) reaction conditions using AL guided by several acquisition functions when using a Gaussian Process Classifier across all four datasets. Each line is the average of 100 runs, and the shaded region shows the 95% confidence interval of the mean. Dashed horizontal lines mark the coverage of the most general individual condition. Black solid lines mark the max possible coverage of the entire dataset while gray solid lines indicate the maximum coverage of a set of three (or four for Ni-Boryl) conditions. The B-H dataset run was terminated at 1% of the space as this corresponded to 4,500 reactions.

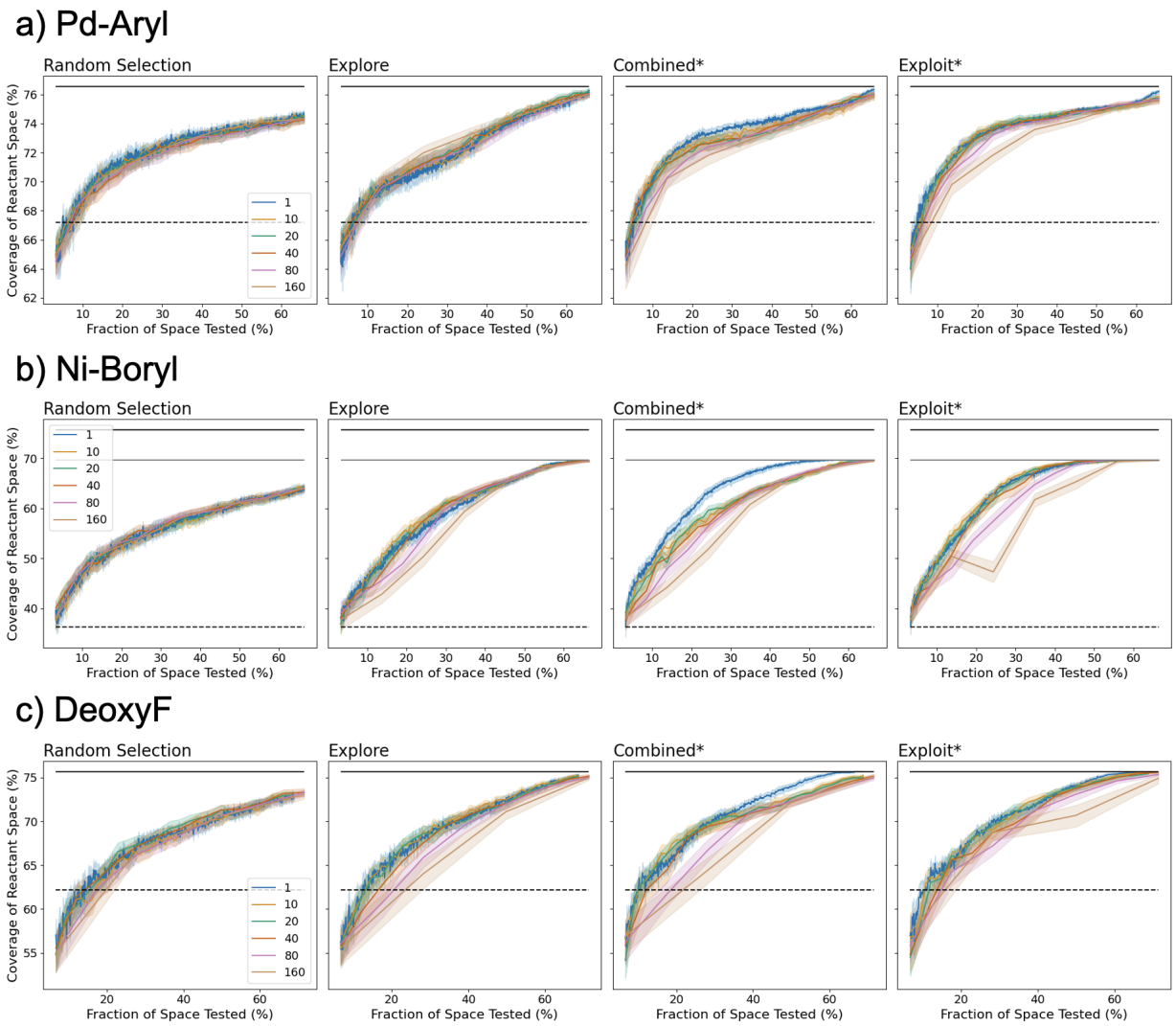

Figure S6: Performance of AL method using several batch sizes and several acquisition functions on the a) Pd-Aryl, b) Ni-Boryl, and c) DeoxyF datasets. AL used a RFC for all plots. Each line is the average of 100 runs, and the shaded region shows the 95% confidence interval of the mean. Dashed horizontal lines mark the coverage of the most general individual condition. Black solid lines mark the maximum possible coverage of the entire dataset while gray solid lines indicate the maximum coverage of a set of three (or four for Ni-Boryl) conditions. For each dataset, 100 batches of initial seed points were selected using Latin Hypercube Sampling and used at each batch size and acquisition function.

Exploit Function Example

The Exploit function attempts to suggest reactions (r, c pairs) that meet 2 criteria:

1. Select reactant(s) (r) likely not covered by other conditions (e.g. high $1 - \phi_{(r,c_i)}$)

2. Select a condition (c) that would be complementary to other conditions for sets with high predicted coverage (high $\gamma_{\{c,c_i\}}$ values)

These goals are combined in the Exploit function (Eq S2 or Eq 3 main text). To illustrate how this function works, a toy example is provided in Figure S7 along with the explicit math used to calculate $\text{Exploit}_{r,c}$. This toy example contains a space of 4 reactants (A, B, C, D), and 4 conditions (1, 2, 3, 4). Two reactions have been measured (A2 = success, C4 = fail), and the rest of the $\phi_{(r,c)}$ were manually set to roughly mimic a Gaussian Process Classifier (GPC) on One Hot Encoded vectors. Using these values, the predicted coverage of all pairs of conditions were calculated, where a $\phi_{(r,c)} \geq 0.5$ is considered a success. These values are then used to calculate all $\text{Exploit}_{r,c}$ values, shown in the bottom grid. The calculations for Exploit_{C2} are shown explicitly on the right.

To demonstrate the performance of the Exploit function, six rounds of a campaign on this toy function are shown in Figure S8. This campaign shows how the Exploit function consistently selects reactions that provide information for discovering a complementary set of reaction conditions. In rounds 1-3, it prioritizes finding a complementary condition that could succeed on the hardest known reactant (C), followed by searching to discover how well the highly complementary condition #3 complements the best known condition #2 in rounds 4-6.

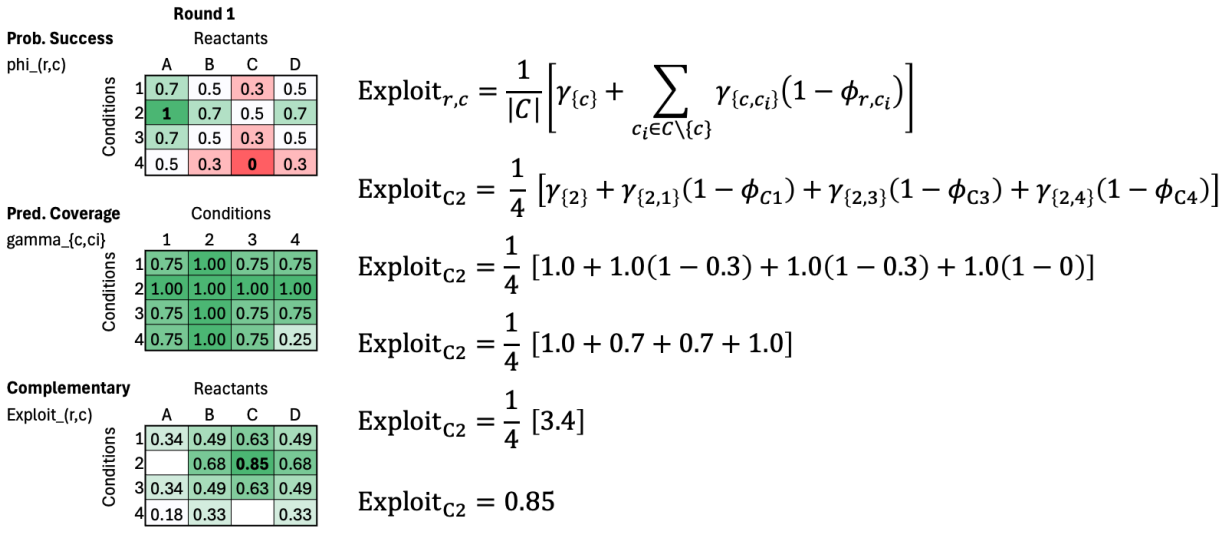

Figure S7: Example calculation of the Exploit acquisition function for a toy dataset.

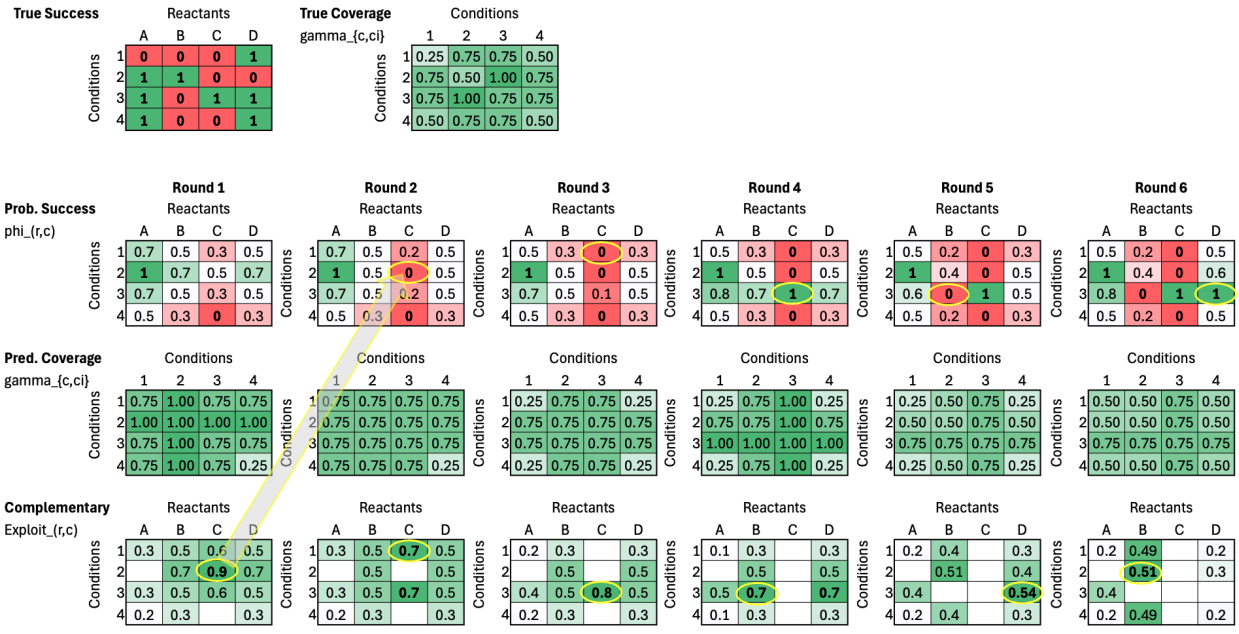

Figure S8: Example campaign guided by the Exploit acquisition function. The true outcome of each reaction and true coverage of each set of two conditions are shown at the top. Six rounds of a simulated campaign are shown below. The top row shows predicted probability of success ($\phi_{r,c}$) with approximate GPR values based on the collected data (bolded 0s or 1s). The middle row shows the predicted coverage ($\gamma_{\{c,c_i\}}$) given $\phi_{r,c}$. The bottom row shows the calculated $\text{Exploit}_{r,c}$ values, with the maxima bolded and the selected next reaction circled in yellow.

Latin Hypercube Sampling

Each AL campaign is initialized using Latin hypercube sampling. Latin hypercube sampling is a method for selecting near random points from a multi-dimensional space such that no two points share the same value in any dimension.[1] This method is used to select initial points to ensure there is diversity in the reactants and conditions used to seed the model. Additionally, since classifiers need at least one sample from each label to learn a classification function, the Latin hypercube sampling was repeated until the selected points included at least one instance of a successful reaction and one instance of a failed reaction. For each dataset, 100 sets of seed points were selected and used across batch and acquisition function testing.

Dataset Size Impact with B-H Dataset

Given the unique performance of the AL algorithms on the B-H dataset and its unique properties (size, dimensions, ML-interpolated data) we explored how reducing the dataset size and dimensions impacted the AL algorithms' performance. Simulations on six subsets of the B-H dataset were run using all four reaction selection methods, shown in Figure S9. The original dataset contained 50 nucleophiles (reactants), 50 aryl-halides (reactants), 3 bases (conditions), and 3 solvents (conditions), and 20 catalyst ligands (conditions). Reactant subsets of 25 molecules were randomly selected once for all runs. Base and solvent subsets selected the single best base and/or solvent once for all runs. Catalyst subsets of 14 catalyst ligands were randomly selected once for all runs. The simulations show similar trends to the full B-H, with the exception that the Combined algorithm outperforms the Explore algorithm in several cases.

We interpret all these results as follows: over large, multi-dimensional spaces, Explore is initially far more effective than Exploit because it prioritizes reducing RFC model uncertainty, helping it identify broad trends in reaction outcomes across the reactant-condition

space quickly. However, Explore doesn't favor understanding high-coverage reaction conditions, rapidly converging to a decent (but often sub-optimal) result, improving slowly with time. In contrast, the Exploit algorithm exclusively focuses on what appear to be the highest coverage conditions and sets. This yields poor performance when the space is large and general trends are unknown, but can provide more tailored recommendations as the trends are uncovered. The Combined algorithm employs both of these advantages, allowing for exploration at earlier times to learn the trends in the data, and more focused exploitation at later times, consistently surpassing Explore by the time 12% of the space has been tested.

Featurization with Molecular Fingerprints

With the success of the AL algorithm over reactant-condition space described with only One Hot Encoded (OHE) vectors, we attempted to further improve the performance using more descriptive molecular features using molecular fingerprints as they are high-throughput dataset-agnostic features for datasets containing SMILES strings which may correlate with reaction yield. Therefore, we calculated the molecular fingerprints (using the RDKit command Chem.RDKFingerprint(), finding Daylight-like fingerprints with path lengths 1-7) for all molecules in the DeoxyF and Pd-Aryl datasets and reran the same analyses. Note that using Fingerprints only impacts the Random Forest Classifier (RFC) and its prediction of if a reaction will be successful (0 or 1), and does not directly inform the AL algorithm, which only considers the RFC's predictions and uncertainties (see Eq's 1-3, main text).

For the DeoxyF dataset, SMILES strings were provided for the Substrates (37 reactants), Base (4 conditions), and Fluoride (5 conditions). We therefore tested and compared the performance of the 4 AL algorithms (Random, Explore, Exploit, and Combined), with the reactant-condition space described as fingerprints for the following molecules (keeping all other parameters OHE): Base only, Fluoride only, Substrate only, Conditions (Base + Fluoride), All, and None. The results are shown in Figure S10.

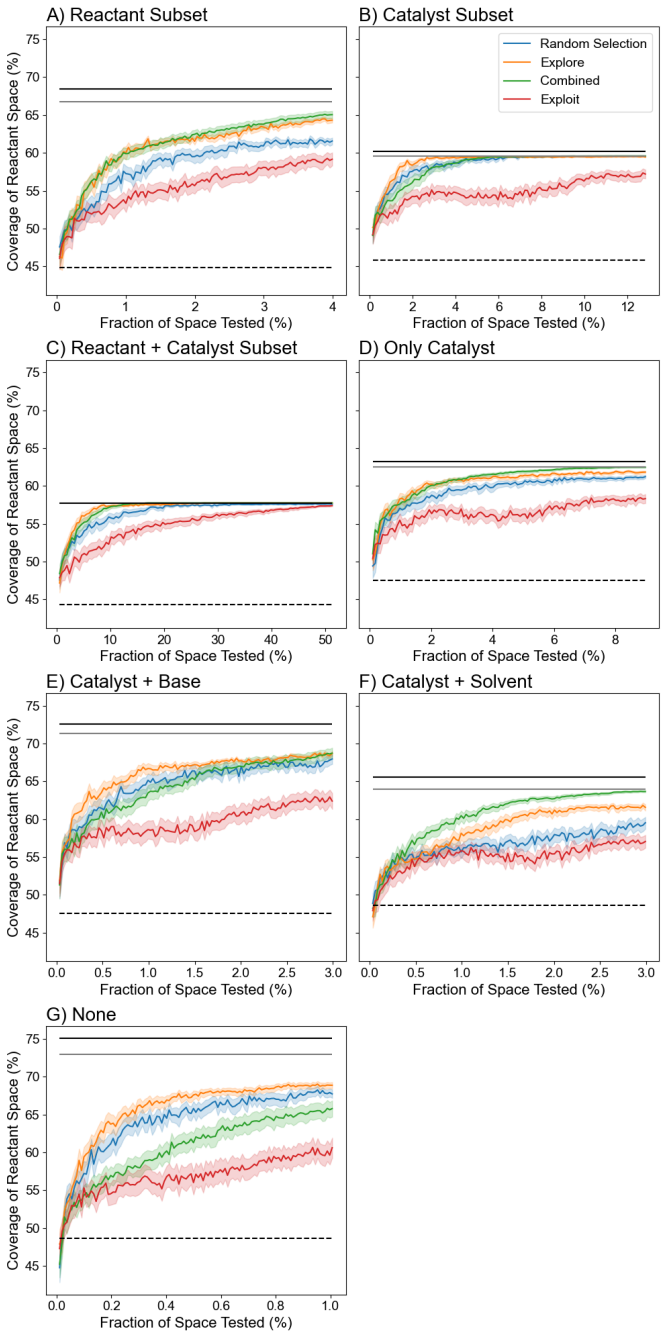

Figure S9: Simulations of the four reaction selection options on subsets of the B-H dataset. The subsets and the corresponding OHE space dimensions are listed, and the dimension of the space listed in the following order: (nucleophiles × aryl-halides × bases × solvents × catalyst ligands). A) Reactant Subset ($25 \times 25 \times 3 \times 3 \times 20$). B) Catalyst Subset ($50 \times 50 \times 1 \times 1 \times 14$). C) Reactant + Catalyst Subset ($25 \times 25 \times 1 \times 1 \times 14$). D) Only Catalyst ($50 \times 50 \times 1 \times 1 \times 20$). E) Catalyst + Base ($50 \times 50 \times 1 \times 3 \times 20$). F) Catalyst + Solvent ($50 \times 50 \times 3 \times 1 \times 20$). G) None ($50 \times 50 \times 3 \times 3 \times 20$). Note that the x-axis and fraction of space varies depending on the size of the datset. Dashed horizontal lines mark the coverage of the most general individual condition. Black solid lines mark the max possible coverage of the entire dataset.

For the Pd-Aryl dataset, SMILES strings were provided for the Electrophile (8 reactants), Nucleophile (8 reactants), and Ligand (24 conditions). We therefore tested and compared the performance of the 4 AL algorithms (Random, Explore, Exploit, and Combined), with the reactant-condition space described as fingerprints for the following molecules (keeping all other parameters OHE): Electrophile only, Nucleophile only, Ligand only, Reactants (Electrophile + Nucleophile), All, and None. The results are shown in Figure S11.

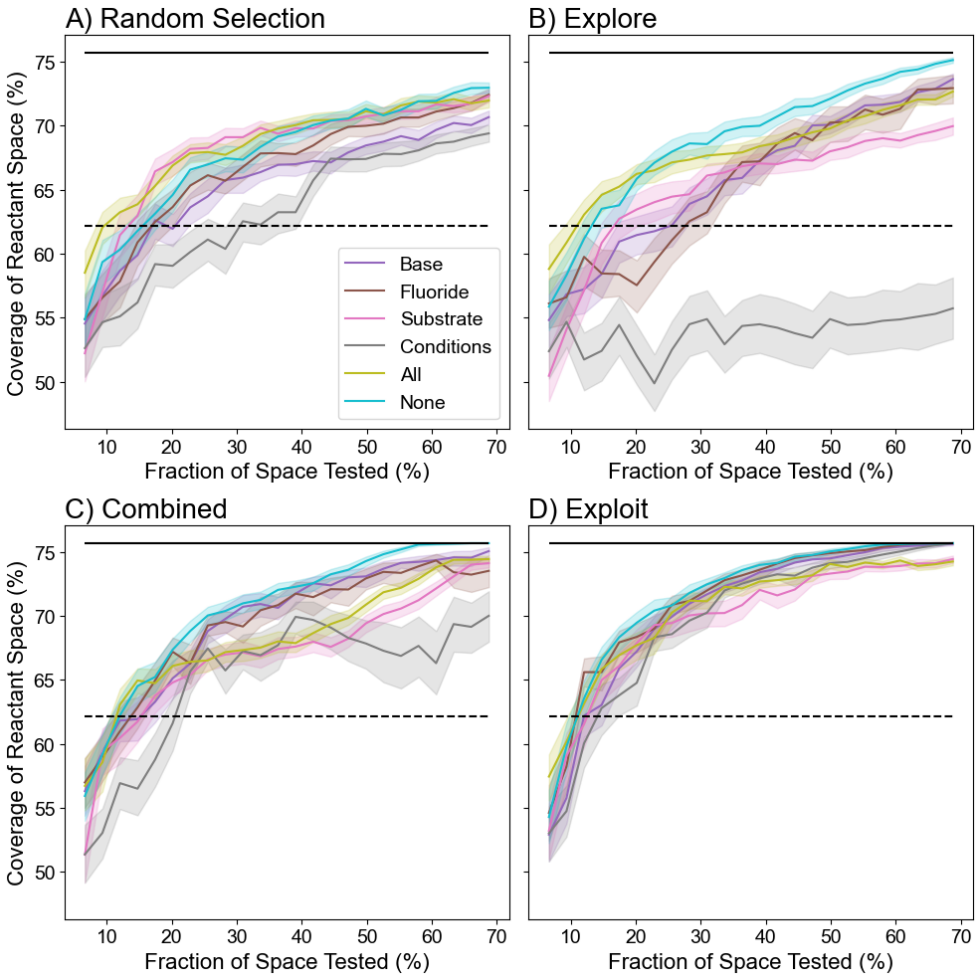

Figure S10: Plots of the performance of each reaction selection algorithm (AL and random selection) on the DeoxyF dataset using fingerprints to represent the substrate (reactant), base (condition), fluoride (condition), and combinations thereof. Dashed horizontal lines mark the coverage of the most general individual condition. Black solid lines mark the max possible coverage of the entire dataset.

These results show that OHE features ("None") consistently perform as well as those

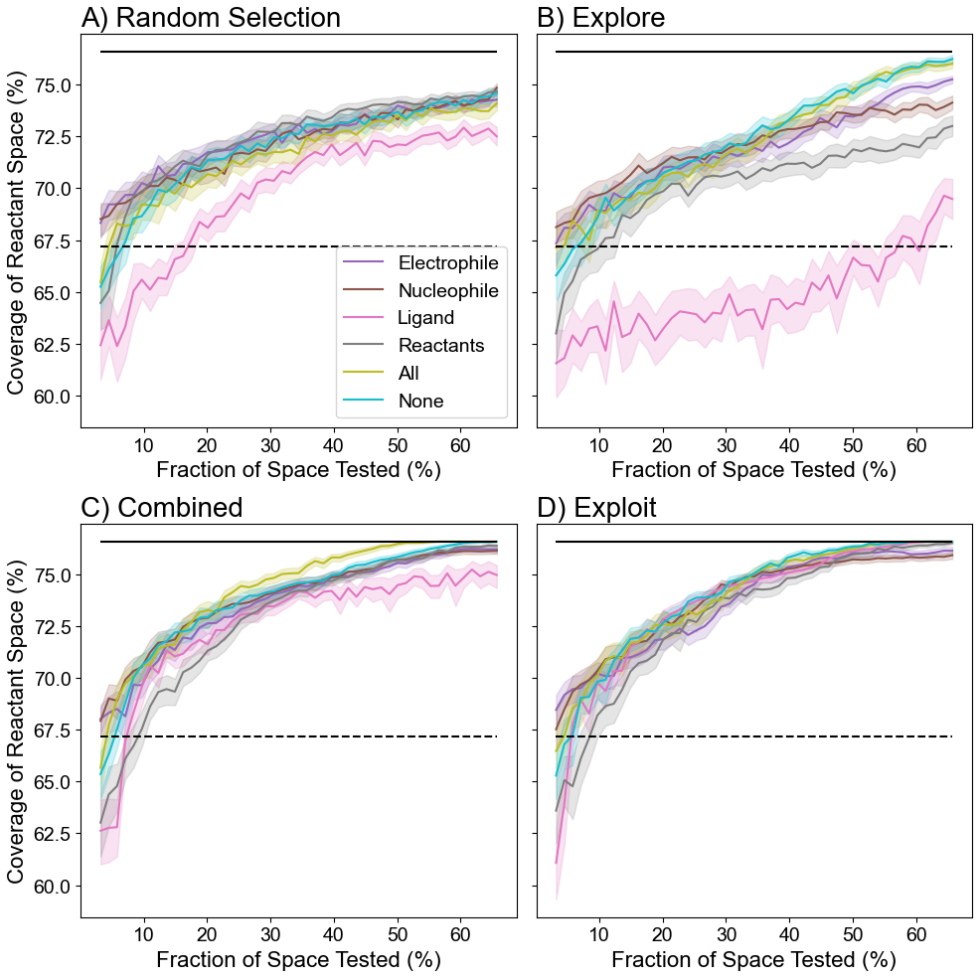

Figure S11: Plots of the performance of each reaction selection algorithm (AL and random selection) on the Pd-Aryl dataset using fingerprints to represent the electrophile (reactant), nucleophile (reactant), catalyst ligand (condition), and combinations thereof. Dashed horizontal lines mark the coverage of the most general individual condition. Black solid lines mark the max possible coverage of the entire dataset.

described by nearly any combination of fingerprints (especially when employing AL), and no set of fingerprints consistently outperforming OHE. This indicates that fingerprints do not adequately correlate with reaction yields to produce a significant advantage over OHE. Note that Random results correspond to randomly selecting successive reactions to test, and that the performance of the RFC will still vary with descriptors, as better descriptors will lead to more accurate yield predictions and therefore better selection of high-coverage reaction sets.

As a case in point in the deoxyF dataset, for the 3 molecules with the closest fingerprints (shown below), the first two molecules have similar yields, but the third produces substantially lower yields for all conditions because it is a tertiary alcohol. The high fingerprint similarity among the molecules is primarily due to their phenyl group and alkyl chain; however, the phenyl group is distant from the reactive alcohol, causing the fingerprint similarity and the reactivity to have low correlation. It is expected that more tailored molecular features (bond strengths, sterics, electron densities, etc.) could be used to construct more informative similarity metrics among the molecules.

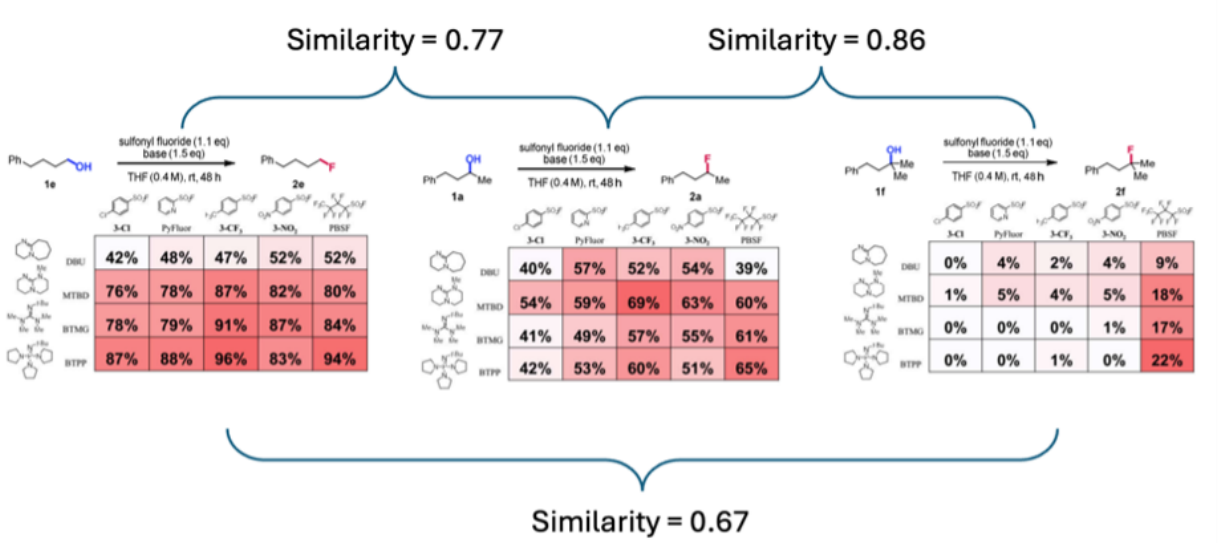

Figure S12: The most similar set of three molecules in the DeoxyF dataset, their reaction yields (from the original work's SI),[2] and their pairwise fingerprint similarities, highlighting how fingerprint similarities sometimes correlate very poorly with reaction yields.

References

(1) McKay, M. D.; Beckman, R. J.; Conover, W. J. A Comparison of Three Methods for Selecting Values of Input Variables in the Analysis of Output from a Computer Code. *Technometrics* **1979**, *21*, 239–245, Publisher: [Taylor & Francis, Ltd., American Statistical Association, American Society for Quality].

(2) Nielsen, M. K.; Ahneman, D. T.; Riera, O.; Doyle, A. G. Deoxyfluorination with Sulfonyl Fluorides: Navigating Reaction Space with Machine Learning. *Journal of the American Chemical Society* **2018**, *140*, 5004–5008, Publisher: American Chemical Society.