Extraction Yield Prediction for the Large-Scale Recovery of Cannabinoids

Hart Plommer,[a,b] Isaiah O. Betinol,[a] Tom Dupree,[b] Markus Roggen,[b] Jolene P. Reid[a]*

Correspondence to jreid@chem.ubc.ca
markus@deliclabs.com

[a]Department of Chemistry, University of British Columbia, 2036 Main Mall, Vancouver, British Columbia, V6T 1Z1, Canada

[b]Delic Labs, 3800 Wesbrook Mall, Vancouver, British Columbia V6S 2L9, Canada

Table of Contents

Generation and Curation of CannaLit Dataset

The raw dataset was taken from the DELIC Labs electronic extraction database and various client-provided datasets. The masses for the acidic cannabinoids CBDA, THCA, and CBGA were normalized to their equivalent neutral molecule mass to account for decarboxylation of the carboxylic acid group that occurs when cannabis is subjected to heat:

$$\begin{gathered} adjusted\ CBDA\ mass = CBDA\ mass\ \times 0.8772 \\ adjusted\ THCA\ mass = THCA\ mass\ \times 0.8772 \\ adjusted\ CBGA\ mass = CBGA\ mass\ \times 0.8779 \end{gathered}$$

Then, each cannabinoid's yield was calculated as the mass of cannabinoid molecule obtained divided by the total input cannabinoid mass (i.e., sum of all cannabinoids within the input biomass):

$$cannabinoid\ yield = \frac{cannabinoid\ mass\ (g)}{total\ input\ cannabinoid\ mass\ (g)} \times 100\%$$

Where an input cannabinoid concentration of THCA, THC, CBDA, CBD, or CBN was not reported, we reported the input value as zero (e.g., input_cbn was featurized as zero). Input cannabinoid concentrations were converted to mass fractions by simply dividing their wt% values by 100. For client-provided datasets, input_cann_total was manually curated by summing the cannabinoid masses (normalized to neutral forms if necessary). The cannabinoid molecule and extractor model were one hot encoded. The corresponding extractor models are SCFN triple 24L (extractor_model1), Extrakt 140 (extractor_model2), and SCFN dual 12L (extractor_model3).

The density feature was calculated using the extraction temperature and pressure with the tool provided at https://www.peacesoftware.de/einigewerte/co2_e.html

In cases where flow rate information was missing for some extractor_model2 entries, we calculated the flow rates based on a 60 Hz pump speed achieving 600 mL/min flow rate (36 kg/h). Since flow rate is directly proportional to pump speed, this meant a 50 Hz pump speed generated a flow of 30 kg/h.

Entries that showed production of greater amounts of cannabinoid on output than input (i.e., more than theoretically possible even if considering both neutral and acid cannabinoid forms) were removed from the dataset. In some cases, negative recoveries were observed for CBN when utilizing recoveries calculated from spent and unspent biomass rather the cannabis extracts. These entries were removed as well.

Input_xxx (e.g., input_cbd, input_thc) are the weight fraction of each respective input cannabinoid (e.g. an input_cannabinoid value of 0.0645 is 6.45 %w/w), without any normalization done for acidic cannabinoids (considering CO2 lost), while the input_mass feature is the mass of the entire biomass used for extraction.

Calculation of Possible Process Conditions

To generate all the possible forms a variable could take, the difference of the minimum and maximum values of each continuous process variable (extraction time, temperature, pressure, and flow rate) within the CannLit dataset, subject to certain step sizes, was calculated (shown in Table 1 below). Minimum values were rounded to the nearest smaller integer while maximum values were rounded to the nearest larger integer. Multiplying each column's step size by one another (i.e. 158.5 × 258 × 20.5 × 52) yields 4.4×10^7 possible process conditions to be tried.

Table 1. Calculation of Accessible Process Space.

	Extraction Time	**Pressure**	**Temperature**	**Flow Rate**
Minimum value	42	94	44	9
Maximum value	360	353	85	115
Difference (rounded)	317	258	41	104
Steps	158.5	258	20.5	52

Cultivar Legend

Cultivars were given standard names in the main text for clarity. Table 2 shows the original names for each cultivar.

Table 2. Cultivar legend.

Cultivar 1	Emblem cultivar CBD
Cultivar 2	Emblem cultivar NN
Cultivar 3	Emblem cultivar BW
Cultivar 4	Emblem cultivar N2
Cultivar 5	Emblem cultivar SB
Cultivar 6	Emblem cultivar NL
Cultivar 7	Emblem cultivar TI
Cultivar 8	CBD cultivar from Africa
Cultivar 9	Zenabis 24k Gold
Cultivar 10	Indica blend
Cultivar 11	Emblem cultivar HB
Cultivar 12	Emblem cultivar SK
Cultivar 13	Indica Princess
Cultivar 14	Lifter

ML Model Development

Unsupervised machine learning

Uniform manifold approximation and projection (UMAP)[1] was chosen to reduce the extraction conditions to 2-dimensions suitable for plotting. All plots shown have a min_dist = 0.5 to ensure separation of data entries and n_neighbours = 50 such that global structure is emphasized while still retaining local neighbourhoods.

Supervised machine learning

We tested multiple linear and non-linear machine learning models to predict extraction yields from the corresponding conditions. The following workflow was carried out using the Scikit-learn package in python.[2] All scripts can be found in the accompanying repository.

All models used the same 80:20 train:test split generated pseudorandomly using the built-in train_test_split function and a random state of 25. Parameters were first normalized using the StandardScaler function according to the formula:

$$P_{norm} = \frac{P - u_p}{\sigma_p}$$

where the scaler is trained only on the training data. Hyperparameters were tuned using a random search method (150 models created) where the best performing model was chosen based on the 10-fold cross-validation R^2. The final models were evaluated using R^2, 10-fold cross-validation repeated 10 times, leave-one-out cross-validation, and test set statistics. Random forest, XGBoost, and *k*-nearest neighbour models provided remarkable training set statistics with R^2 values nearing unity. Ultimately, we chose the random forest regressor for future modelling as the predictive performance of the test set most closely matched that of the training set as compared to the XGBoost and *k*-nearest neighbour regressors.

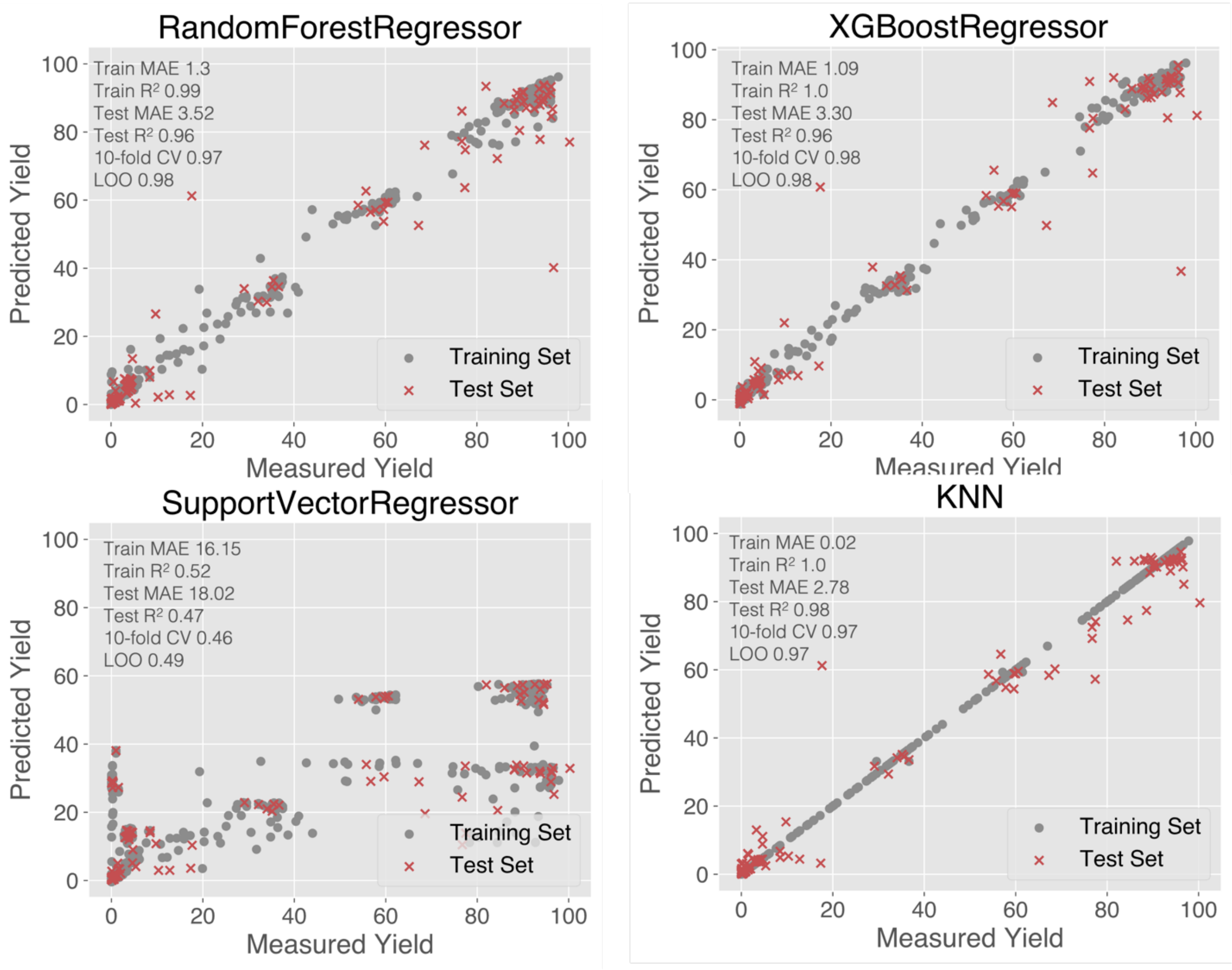

Figure S1. Random forest, XGBoost, support vector, and *k*-NN regressors tested for predicting extraction yields. MAE = mean average error, CV = cross-validation, LOO = leave-one-out cross validation.

Various linear models with different regularizations were also tested with none achieving adequate results.

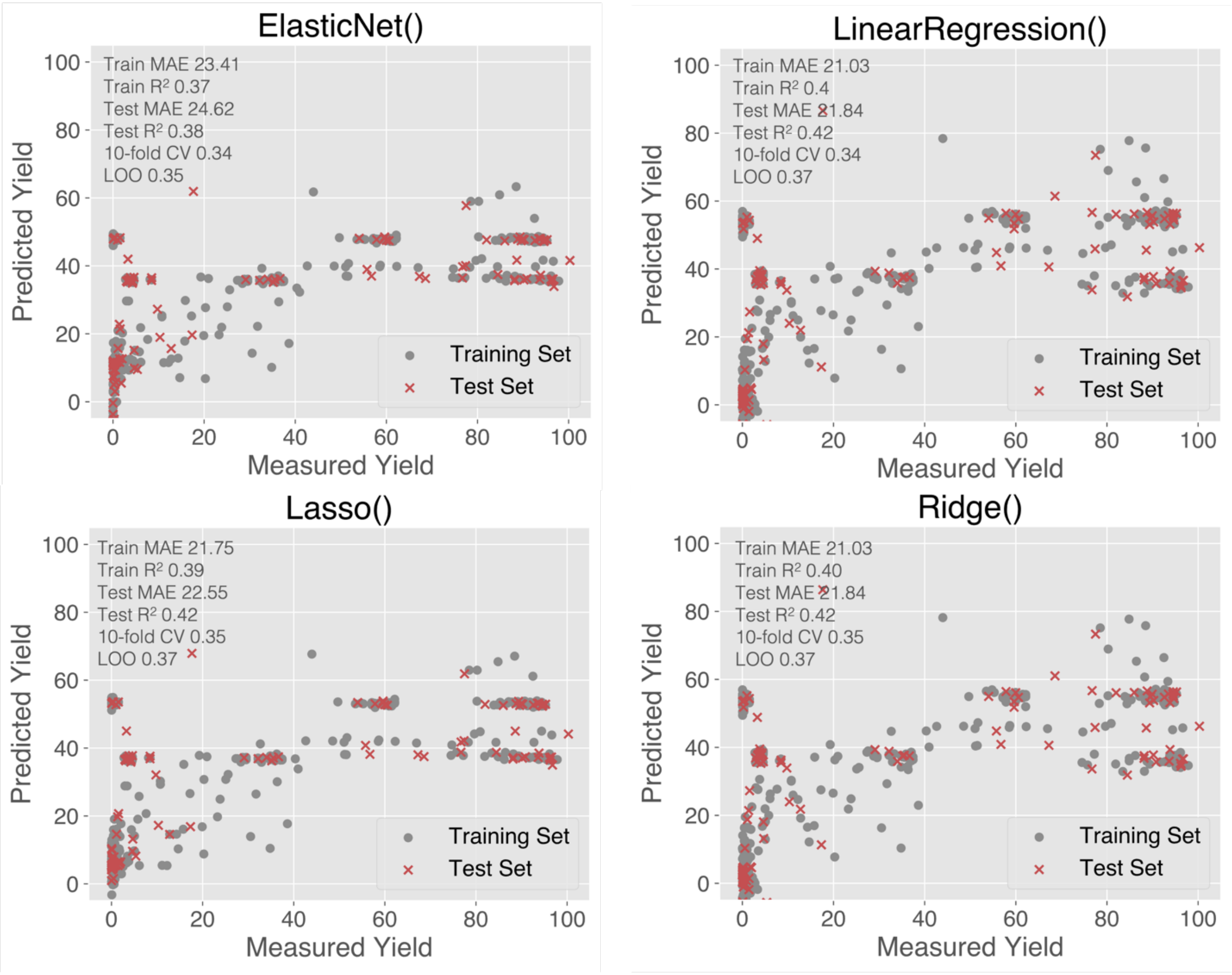

Figure S2. Linear models tested for predicting extraction yields. MAE = mean average error, CV = cross-validation, LOO = leave-one-out cross validation.

Model Validations

Given the remarkable prediction statistics observed with various non-linear regression models, we first questioned how much training data was necessary to accurately predict the test set. The same 20% of data was removed and held out for evaluation and the training set varied by randomly sampling a proportion of the original training set. The learning curve for the random forest model with ideal hyperparameters is shown in Figure S3 and demonstrates that only 20% of the original training set data (16% of all available data) is needed to predict both the training set and test set with adequate performance.

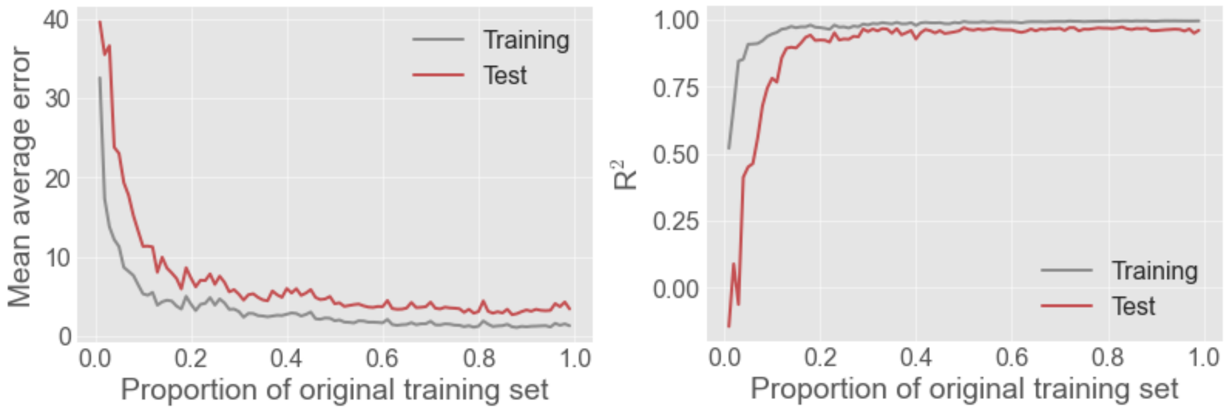

Figure S3. Random forest learning curves with either mean-average-error (left) or R^2 (right) scoring.

In addition to standard model evaluation (e.g., R^2, MAE, test set performance) we performed experiments to support the chemical validity of our descriptors. The first test was determining if the extraction features for each run were chemically relevant or if the machine learning models were simply capturing trends in the data with respect to the extraction component. Here, the 16 features describing the extraction conditions were replaced with a random number for each extraction and used alongside the categorical features describing the extraction component.[3] We find that model performance sharply drops thus supporting the chemical validity of the extraction conditions in predicting extraction yield. Specifically, the test set and cross-validation statistics are especially poor with these random features, though there is some ability to predict the training data.

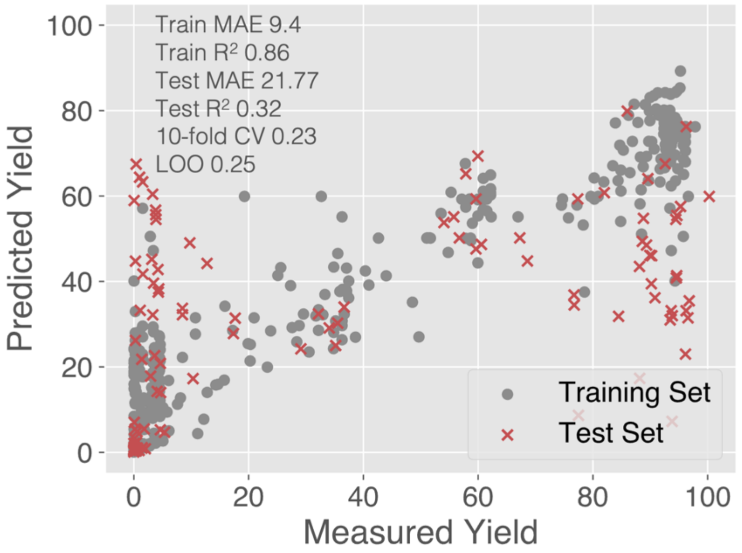

Figure S4. Random forest model built on the CannaLit database where all features other than cannabinoid identity are replaced with random numbers.

Replacing all 24 features with random numbers, thus creating a fully random barcode for each extraction, also showed poor model performance.

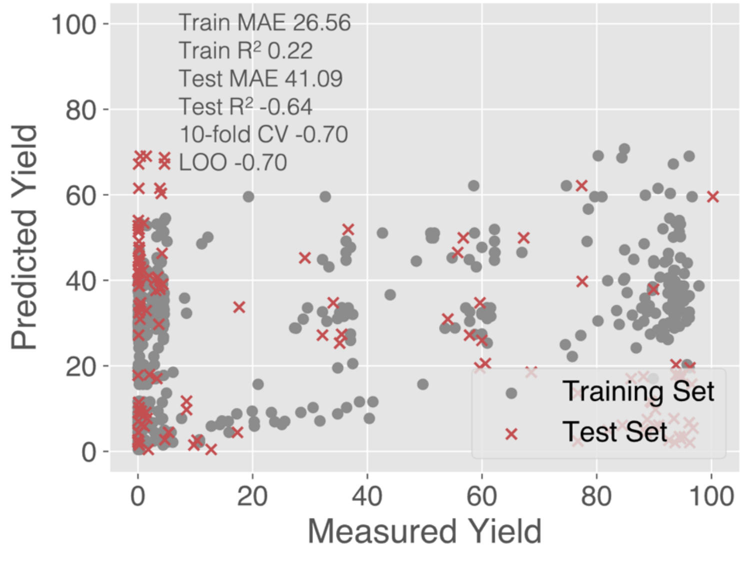

Figure S5. Random forest model built on the CannaLit database where all features including cannabinoid identity are replaced with random numbers.

Feature Importance

We performed supplemental analysis to determine which features are most important for the random forest regressor to confirm appropriate model logic. We obtained the Gini importance values[4] using the built in feature_importances_ attribute and found that the input cannabinoid mass has the greatest importance in determining the separation yield, a testament to the ability of tree-based models to uncover non-linear trends as uniparameter plots show little correlation between these features and the extraction yield. Unsurprisingly, the identity of cannabinoid extracted is also deemed highly important as these features separate identical extraction runs - this observation does confirm that the model is following appropriate logic. Looking at the last notable parameter, flowRate, we see a clear cutoff between low and high flow rates where increasing flow rates appears to severely hinder the extraction yield.

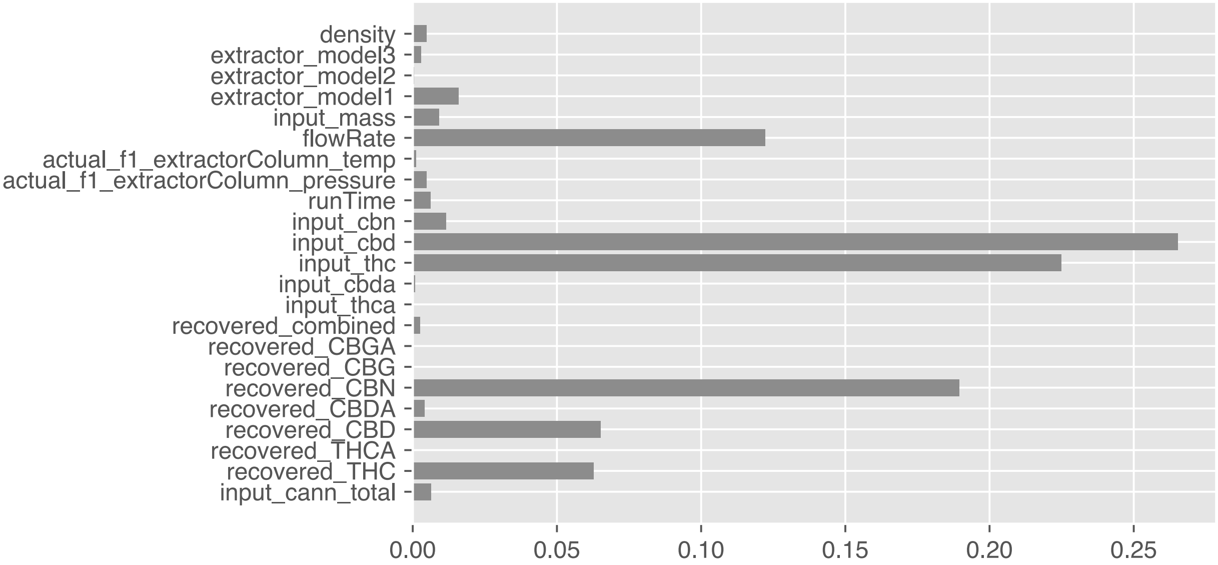

Figure S6. Feature importance for the random forest model.

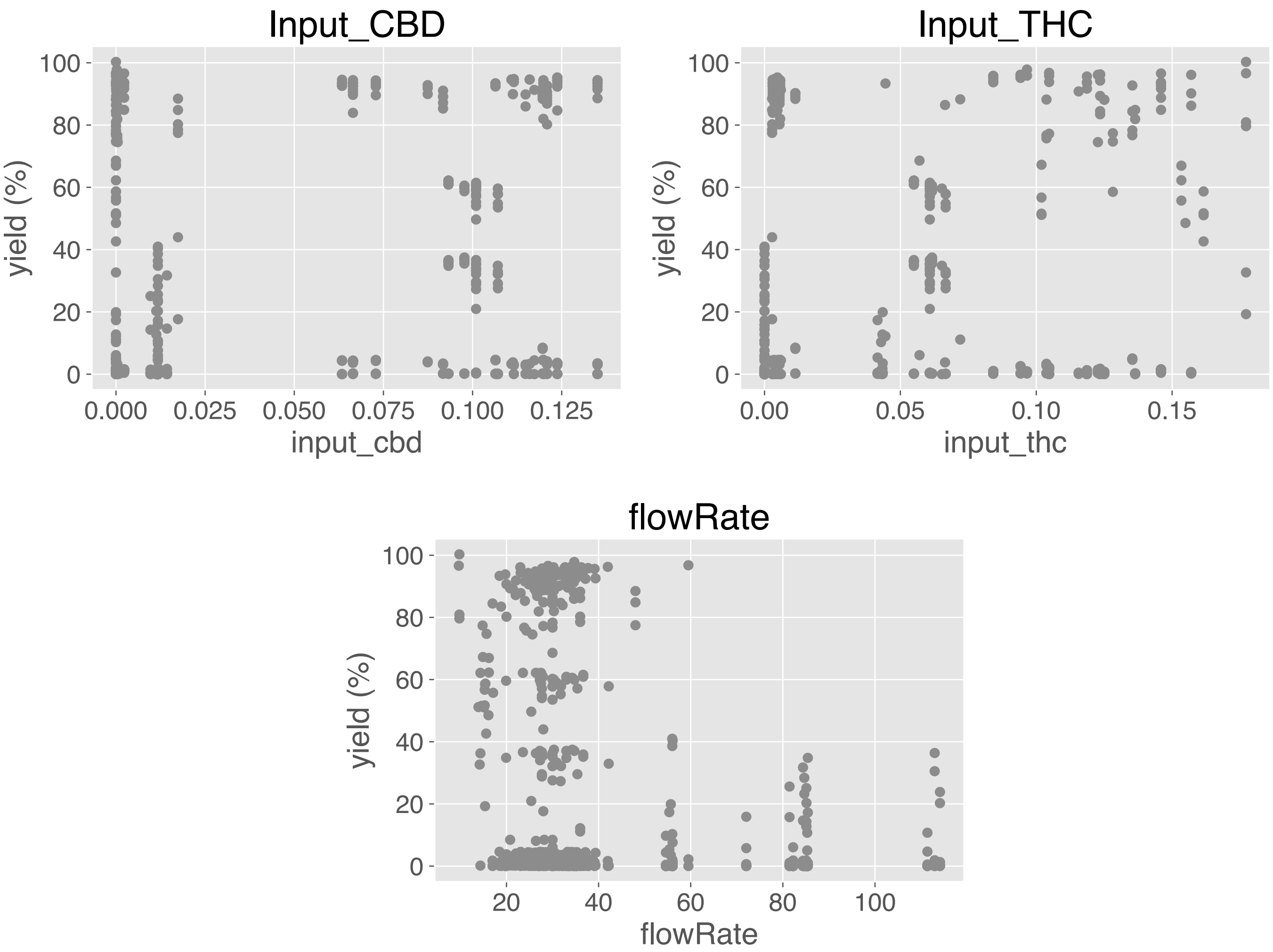

Figure S7. Uniparameter plots relating individual features to yield.

Out-of-Cultivar Study

We performed leave-one-cultivar-out analysis to test the robustness of the model. 13 of 14 strains were predicted with a mean average error below 20%, and 8 out of 14 strains predicted with a mean average error below 10%.

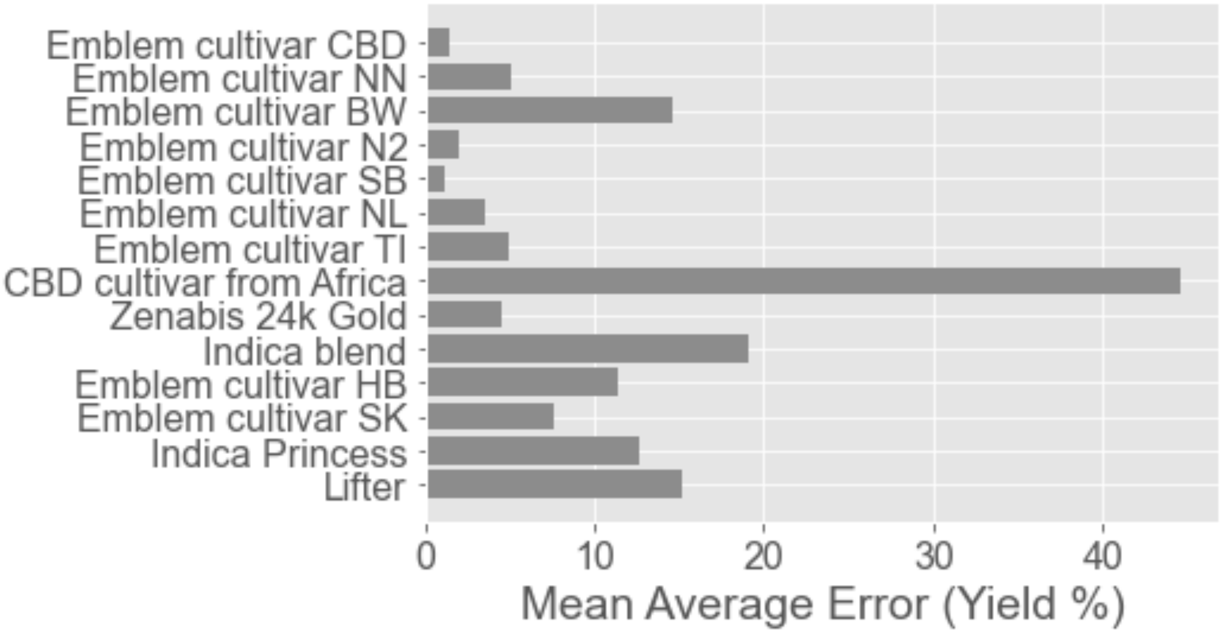

Figure S8. Mean average error obtained when holding out individual cultivars and predicting from the rest of the data.

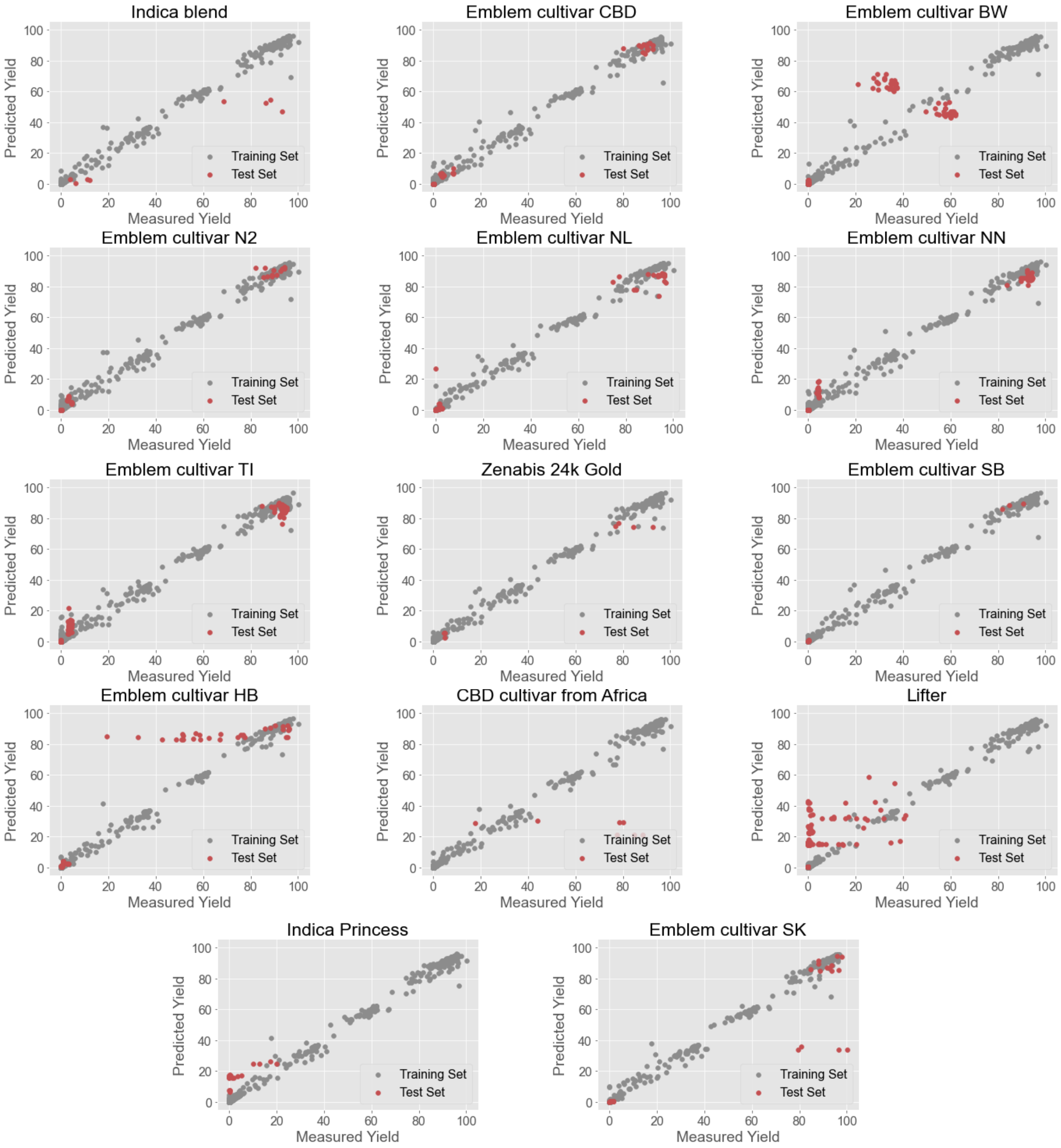

Figure S9. Individual plots from leave-one-cultivar-out analysis

Single Cannabinoid Comparisons

CannaLit was collected with the intent of maximizing data from difficult extraction processes. A consequence of this approach is that extraction conditions from one cannabinoid influence predictions of other cannabinoids. To ensure that model generalizability is retained or improved by including all cannabinoids, we compared CannaLit to a THC-constrained CannaLit for the prediction tasks included in the main text.

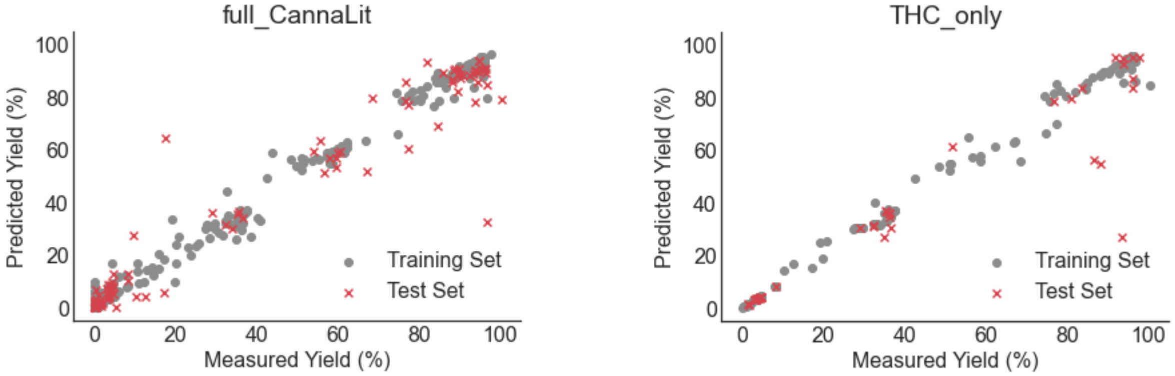

Figure S10. Comparison of the random forest model with full CannaLit (left, $R^2 = 0.99$, Test $R^2 = 0.96$, 10-fold CV = 0.97, LOO = 0.98) and a model from a THC only dataset (right, $R^2 = 0.99$, Test $R^2 = 0.88$, 10-fold CV = 0.95, LOO = 0.96).

The model built only on THC recovery exhibits similar training set statistics to the full CannaLit model with test set statistics slightly worse (Figure S10). An out-of-cultivar test was then performed to compare model generalizability across unseen cultivars (Figure S11). We find that the model built on full CannaLit outperforms the THC model with no cultivars having a MAE over 50% while the THC model has 3 such cultivars.

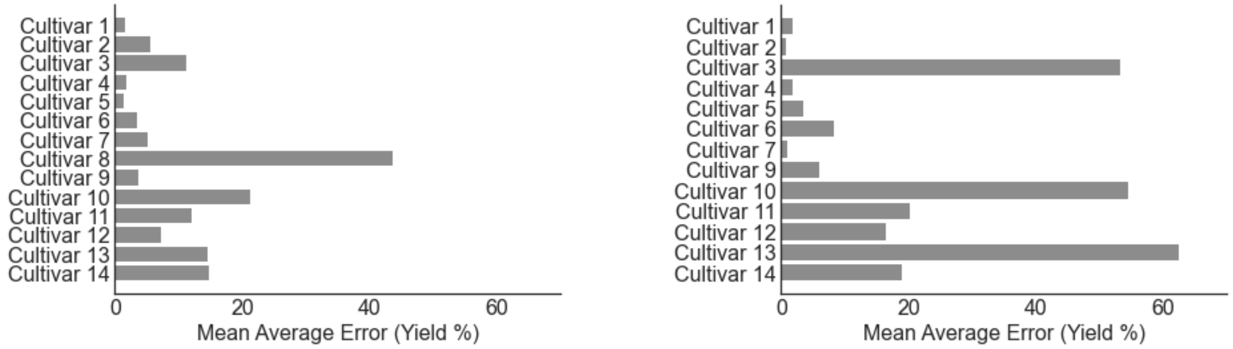

Figure S11. Mean average error when predicting cultivars left out of training for the full CannaLit model (left) and THC-only model (right).

Similar discrepancies were found when comparing the performance of predicting high-scale reactions or a held-out set cultivar 13 (originally used to compare to a single DoE designed dataset).

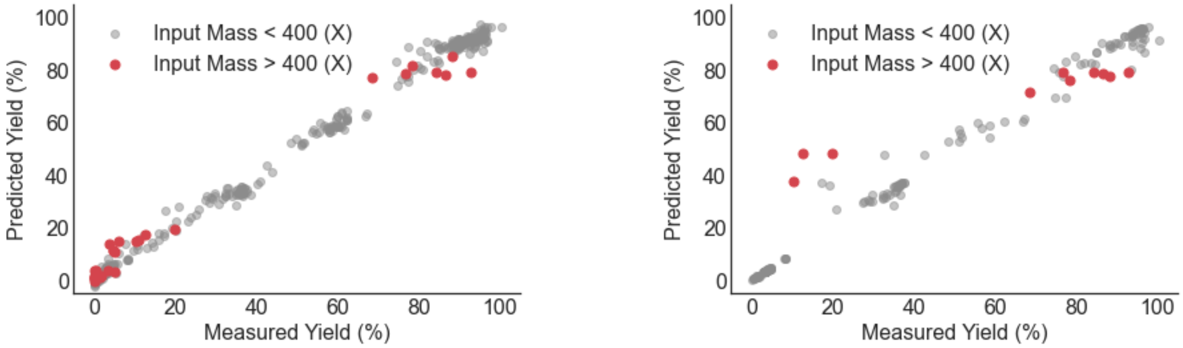

Figure S12. Comparing the ability of a model trained on full CannaLit (left, test $R^2 = 0.98$, test MAE = 3.88) and THC-only (right, test $R^2 = 0.68$, test MAE = 13.63) extractions with an input biomass <400g to predict extractions with an input biomass >400g.

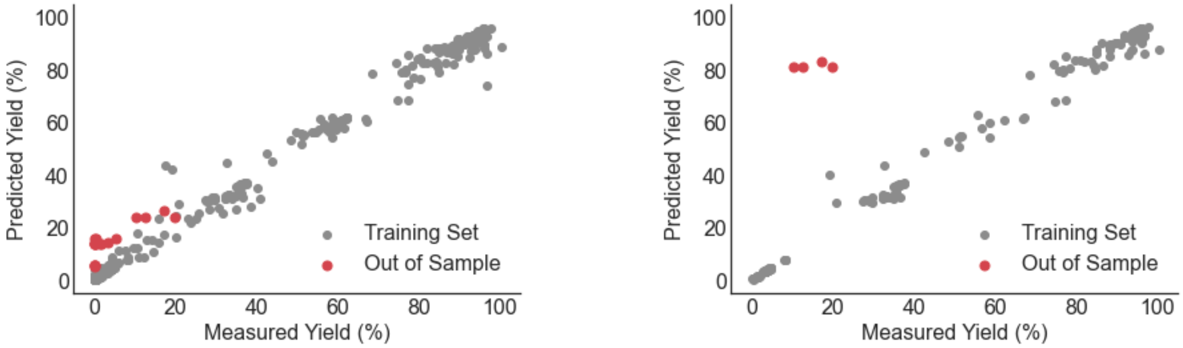

Figure S13. Prediction of cultivar 13 from a full CannaLit model (left, test MAE = 11.8) vs a THC-only model (right, test MAE = 68.1)

Molecular Fingerprint Comparisons

As noted in the main text, different cannabinoids can have different extraction yields based only on its inherent chemical properties and not the extraction conditions (e.g., CBD is more soluble in CO_2 than THC). We tested if a predictive model could be obtained from only the chemical structures. Morgan fingerprints[5] with 2048 bits and radius = 2 were obtained for each cannabinoid using RDKit[6] and a random forest model trained using only the fingerprints of the respective cannabinoids (Figure S14).

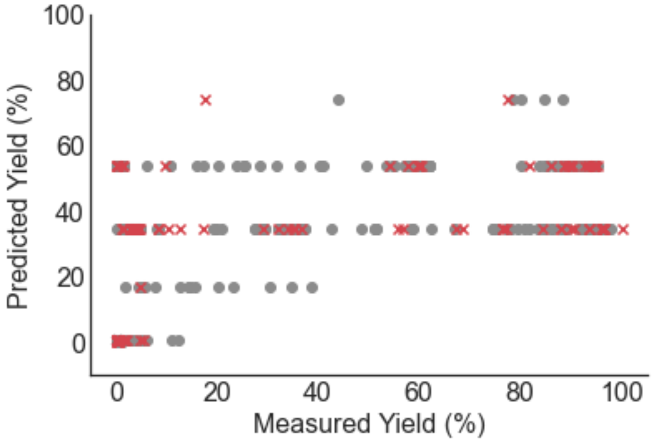

Figure S14. Random forest model predicting extraction yields for cannabinoids from the chemical fingerprints. $R^2 = 0.36$, MAE = 21.0, test $R^2 = 0.38$, test MAE = 21.7.

We find that this model is unable to find strong correlations and conclude that inclusion of extraction conditions is necessary for predicting yields.

References

(1) McInnes, L.; Healy, J.; Melville, J. UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction. *arXiv* **2020**. https://doi.org/10.48550/arXiv.1802.03426.

(2) Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel, M.; Prettenhofer, P.; Weiss, R.; Dubourg, V.; Vanderplas, J.; Passos, A.; Cournapeau, D.; Brucher, M.; Perrot, M.; Duchesnay, E. Scikit-Learn: Machine Learning in Python. *J. Mach. Learn. Res.* **2011**, *12*, 2825–2830.

(3) Chuang, K. V.; Keiser, M. J. Comment on "Predicting Reaction Performance in C–N Cross-Coupling Using Machine Learning." *Science* **2018**, *362* (6416), eaat8603. https://doi.org/10.1126/science.aat8603.

(4) Breiman, L. Random Forests. *Mach. Learn.* **2001**, *45* (1), 5–32. https://doi.org/10.1023/A:1010933404324.

(5) Morgan, H. L. The Generation of a Unique Machine Description for Chemical Structures-A Technique Developed at Chemical Abstracts Service. *J. Chem. Doc.* **1965**, *5* (2), 107–113. https://doi.org/10.1021/c160017a018.

(6) Landrum, G.; Tosco, P.; Kelley, B.; Ric; Sriniker; Gedeck; Vianello, R.; NadineSchneider; Kawashima, E.; Dalke, A.; N, D.; Cosgrove, D.; Cole, B.; Swain, M.; Turk, S.; AlexanderSavelyev; Jones, G.; Vaucher, A.; Wójcikowski, M.; Take, I.; Probst, D.; Ujihara, K.; Scalfani, V. F.; Godin, G.; Pahl, A.; Berenger, F.; JLVarjo; Strets123; JP; DoliathGavid. RDKit: Open-Source Cheminformatics, 2022. https://doi.org/10.5281/ZENODO.6330241.