Supplementary Information (SI) for Digital Discovery.

Meta-Learning Linear Models for Molecular Property Prediction. Supporting Information.

Yulia Pimonova,[*,†] Michael G. Taylor,[‡] Alice Allen,[‡,¶,§] Ping Yang,[*,‡] and Nicholas Lubbers[*,†]

†*Computing and Artificial Intelligence Division, Los Alamos National Laboratory, Los Alamos, NM 87545, USA*

‡*Theoretical Division, Los Alamos National Laboratory, Los Alamos, NM 87545, USA*

¶*Center for Nonlinear Studies, Los Alamos National Laboratory, Los Alamos, NM 87545, USA*

§*Max Planck Institute for Polymer Research, Ackermannweg 10, 55128 Mainz, Germany*

E-mail: ypimonova@lanl.gov; pyang@lanl.gov; nlubbers@lanl.gov

Contents

1 Experimental Setup and Data Splits

We evaluate all methods in a low-data setting using an N-shots protocol. For each target task, we (i) define a fixed held-out test set, (ii) restrict training to N labeled samples (shots) drawn from the remaining data, and (iii) repeat this subsampling procedure across 10 $\{0, 1, \ldots, 10\}$ random seeds to report mean performance and uncertainty. Parameter selection for ridge regularization is performed using cross-validation on the training data only, and the test set is never used for tuning. We consider $N \in \{5, 10, \ldots, 160\}$. For each N, we sample N training points without replacement from the target-task training pool and evaluate on the fixed target-task test split. We use fixed, precomputed graphlet fingerprints as the molecular representation. We restrict the maximum substructure (graphlet) size to ≤ 7. This cap is chosen to keep feature dimensionality and runtime manageable, since the number of possible graphlets grows rapidly with increasing substructure size. We report the exact graphlet-size range and fingerprint dimensionality used for each dataset in Table 1 in the main text.

1.1 Dataset-specific split definitions

- **Boobier solubility**, **BigSolDB 2.0.** We define tasks as solvents. The train/test splits are constructed 80/20.
- **QM9-MultiXC.** We define tasks as computational chemistry methods (DFT functional + basis set).The train/test splits are constructed 80/20.

For the evaluation metric we report MAE/RMSE/R^2 on the target-task test set. Error bars in figures denote standard error of the mean calculated as $\mathrm{SE}_{\bar{x}} = \frac{s}{\sqrt{n}}$ over random subsampling seeds.

1.2 Hyperparameter selection

All ridge-regression components in this work use the `scikit-learn` regularization parameter α. Unless otherwise stated, α is selected by cross-validation on the training split only using a log-spaced grid $\alpha \in \text{logspace}(-6, 6, 26)$ with 5-fold CV. For LAMeL, the same selection procedure is applied to (i) per-support-task ridge fits and (ii) the target-task adaptation step(s) using only the N labeled target samples. For the solvent similarity analysis, cross-validation is used to select α for the ridge fits, but the underlying random 80/20 split is performed once.

2 Results for Boobier et al. Dataset

2.1 Individual Solvents Solubilities under Varying Maximum Substructure Size

We evaluated solvent-specific solubility predictions using the Boobier et al. dataset across acetone, benzene, ethanol, and water. In all experiments all available remainging tasks were used as support, i.e. if solubility in acetone is the target task, solubilities in benzene, water and ethanol are used as support tasks. Figures 1–4 illustrate performance trends as the maximum substructure size used in molecular fingerprinting increases (3, 5, 7, from left to right).

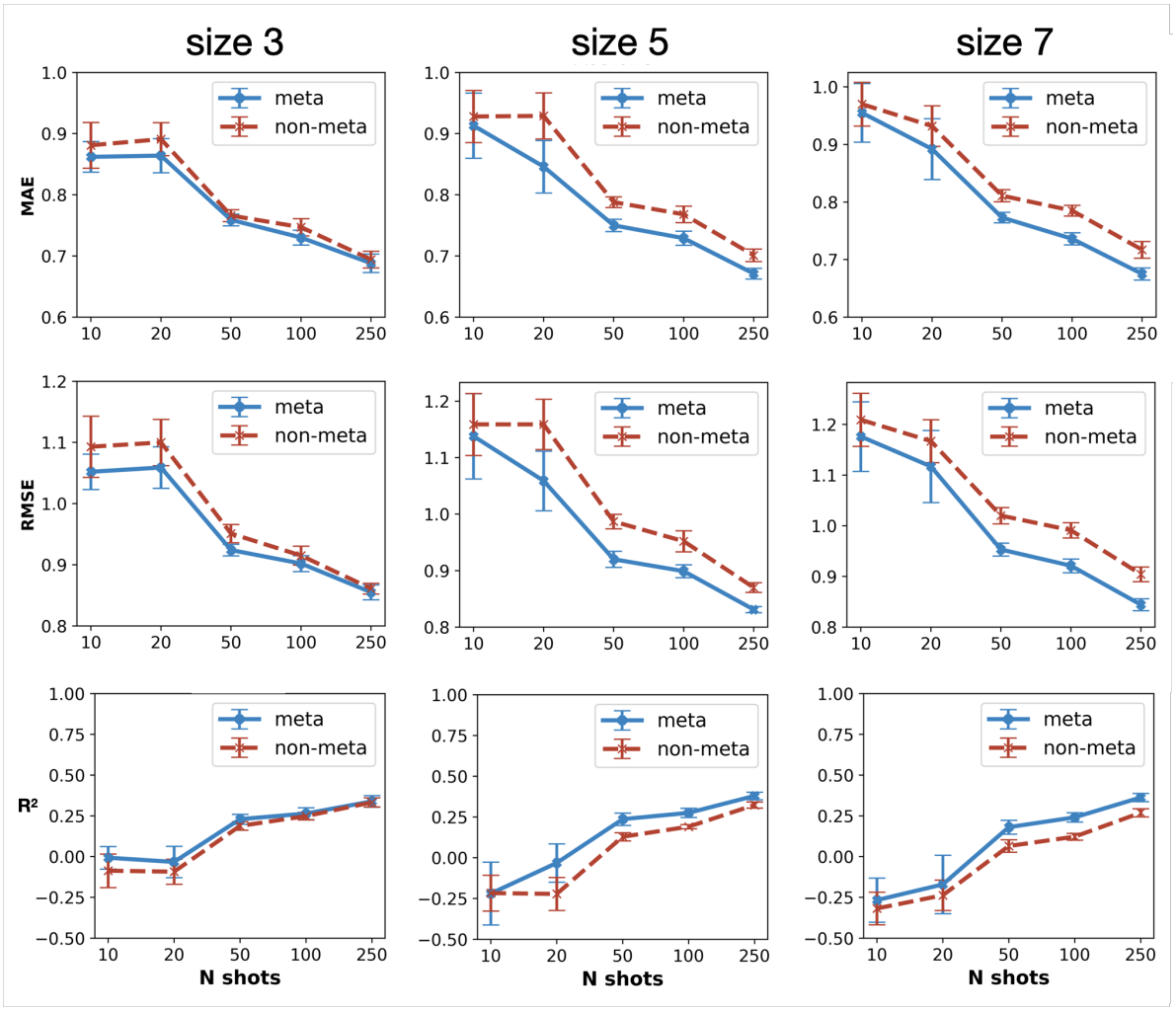

Figure 1: *LAMeL* meta-learning results for predicting solubilities of small organic molecules in acetone with maximum substructure size 3, 5, 7 (columns) being used in the fingerprinting process. The data is averaged over 10 random initializations.

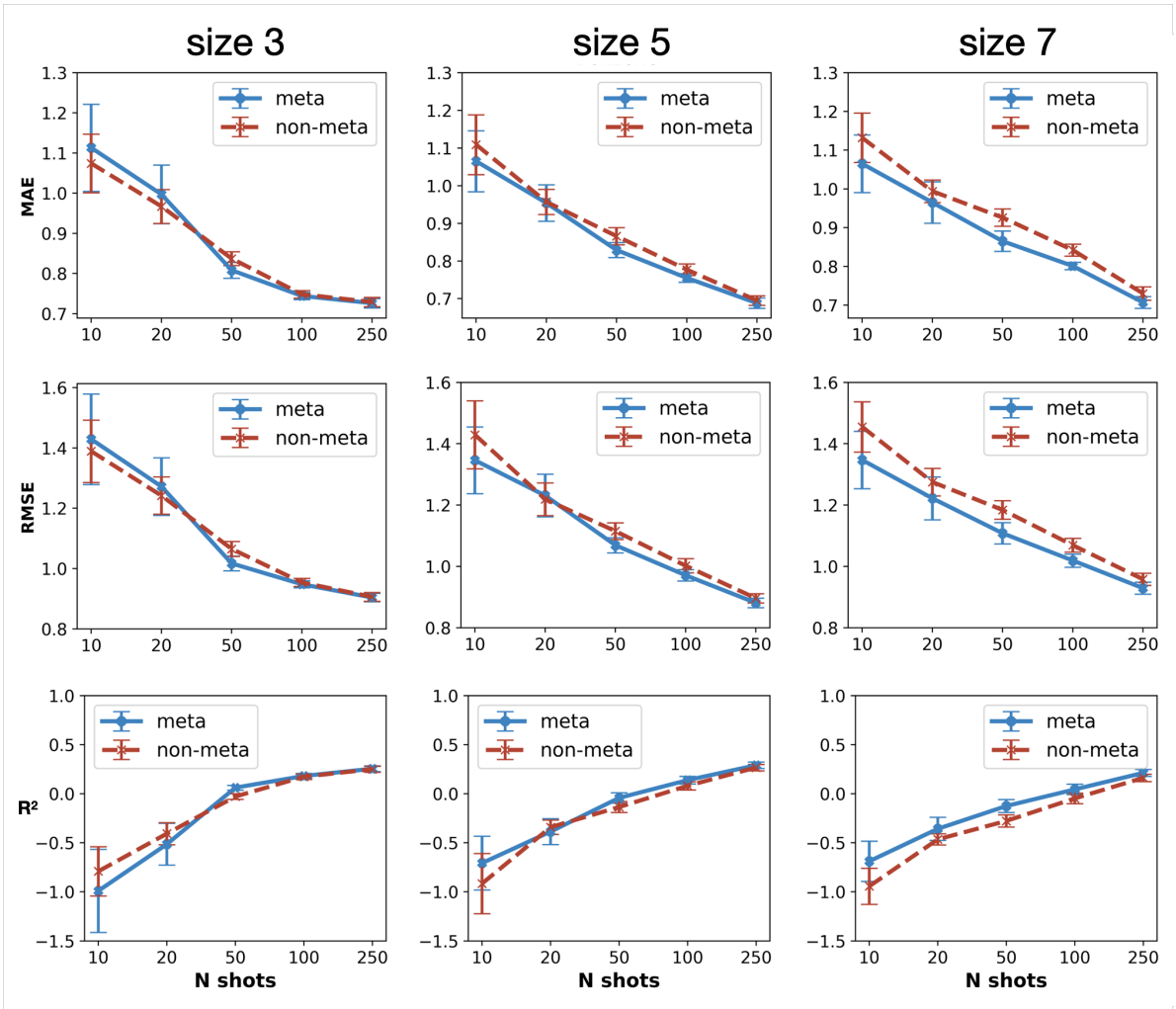

Figure 2: *LAMeL* meta-learning results for predicting solubilities of small organic molecules in benzene with maximum substructure size 3, 5, 7 (columns) being used in the fingerprinting process. The data is averaged over 10 random initializations.

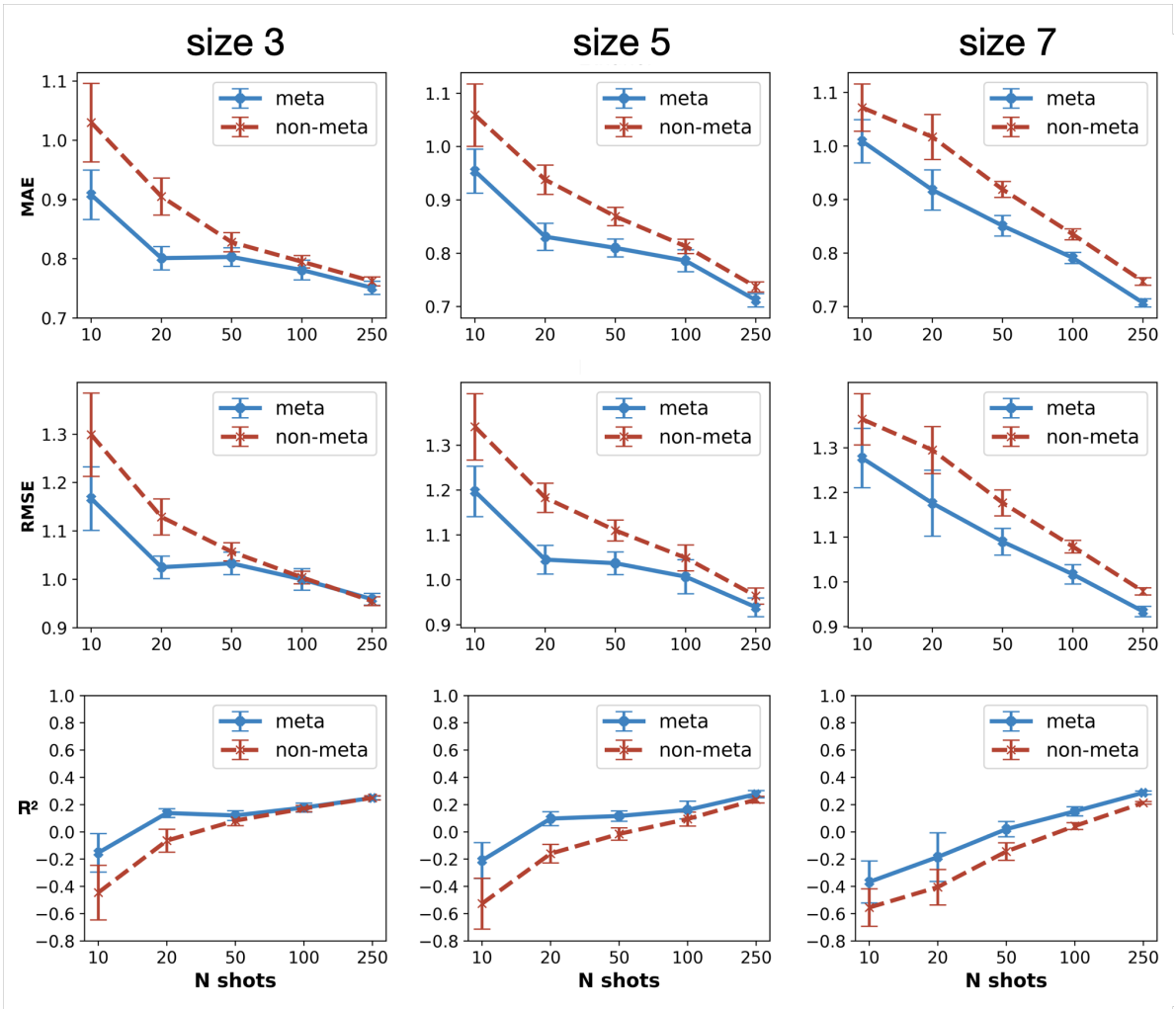

Figure 3: *LAMeL* meta-learning results for predicting solubilities of small organic molecules in ethanol with maximum substructure size 3, 5, 7 (columns) being used in the fingerprinting process. The data is averaged over 10 random initializations.

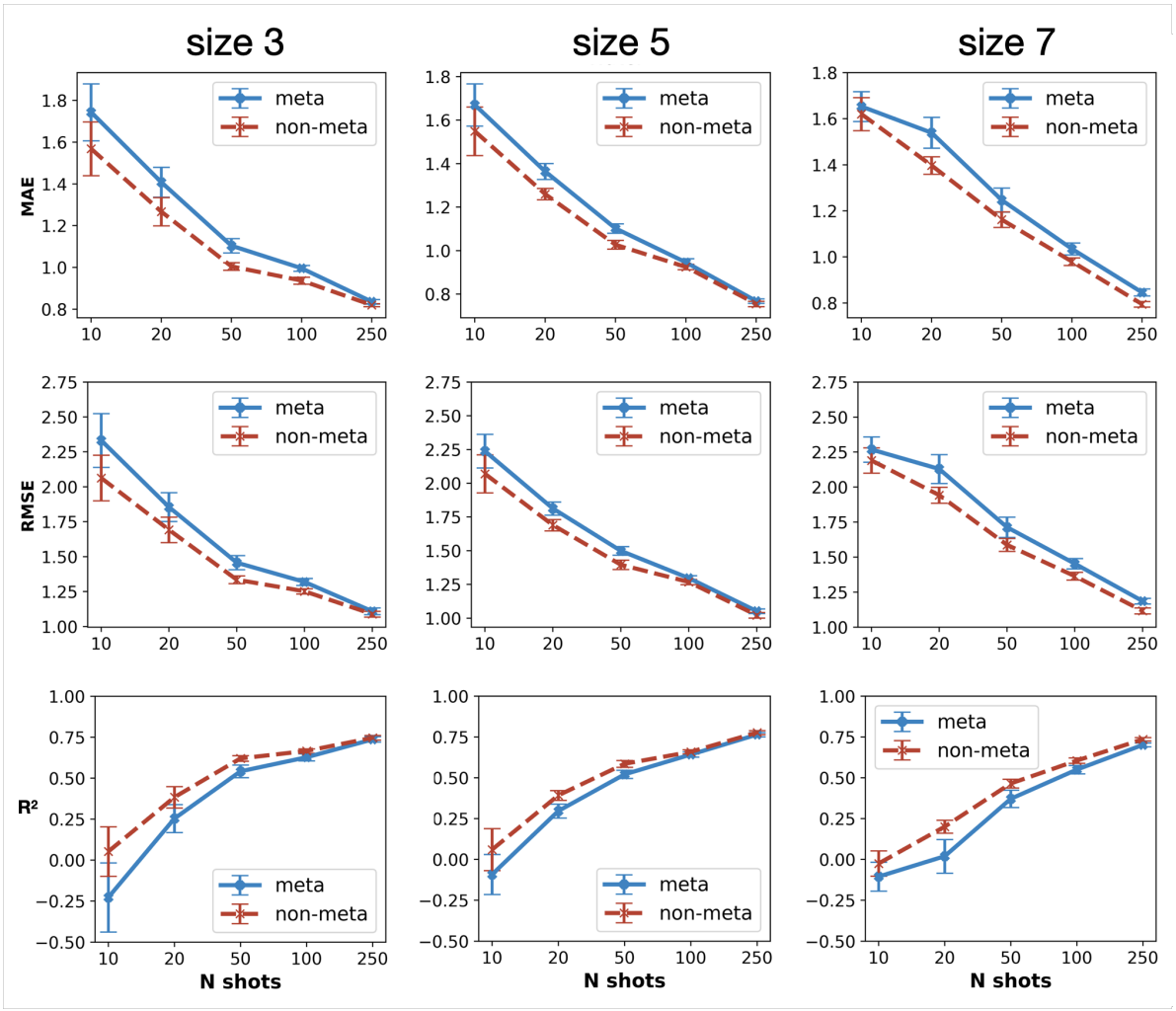

Figure 4: *LAMeL* meta-learning results for predicting solubilities of small organic molecules in water with maximum substructure size 3, 5, 7 (columns) being used in the fingerprinting process. The data is averaged over 10 random initializations.

3 Results for BigSolDB2.0 Dataset

In BigSolDB2.0, dataset filtering significantly impacts the number of solvents retained for analysis. The original BigSolDB 2.0 dataset contains 103,944 solubility datapoints. Table 1 summarizes the effect of our preprocessing on the number of retained datapoints. The majority of excluded entries correspond to repeated measurements for the same solute–solvent pair under different experimental conditions (e.g., temperature). In this work, we retain a single representative datapoint per unique solute–solvent pair.

Table 1: Effect of BigSolDB 2.0 preprocessing on dataset size.

Minimum # solvents per solute	**Datapoints retained**	**Datapoints removed**	**Removed (%)**
No minimum (0)	12,009	91,935	88.45
5	12,003	91,941	88.45
10	11,948	91,996	88.50
20	11,803	92,141	88.64
100	10,786	93,158	89.62
200	8,988	94,956	91.35
400	7,769	96,175	92.53
500	7,769	96,175	92.53

The relationship in Fig. 5 shows that as the minimum required number of datapoints per solvent increases, the number of solvents available for analysis decreases sharply. Fig. 6 highlights that smaller datasets produce more variable models, while tasks with ≥ 200 datapoints yield more stable performance metrics.

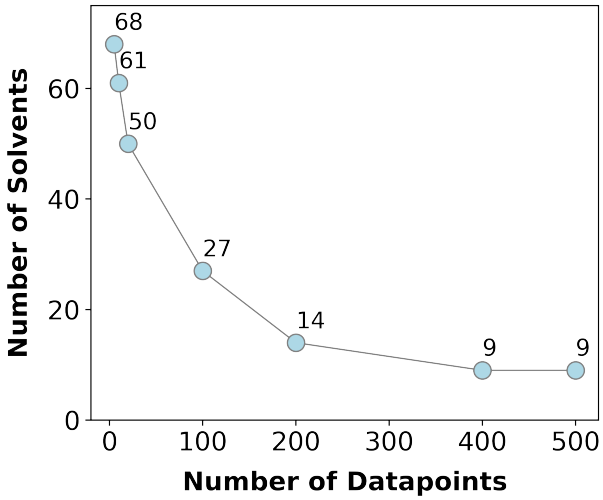

Figure 5: Relationship between the limits on the number of datapoints and the number of available solvents.

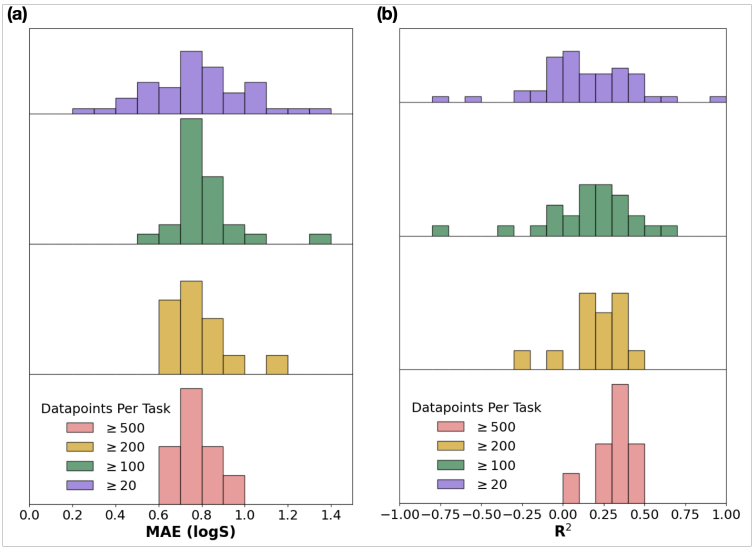

Figure 6: The distribution of performance metrics MAEs (a) and R^2 (b) for the individual independent ridge regression models depending on the size limitation imposed on solvent-specific datasets.

3.1 Absolute Performance in BigSolDB2.0 Dataset

To complement the curve plots of relative improvement in the main text, Tables S2-5 showcase absolute test MAE values for three representative BigSolDB2.0 target solvents across the N-shot regime. For each N, we summarize performance as the mean ± SEM over 10 independent random seeds, together with 95% confidence intervals, for both LAMeL (meta) and the per-task ridge baseline (non-meta). These tables provide exact values underlying the trends discussed in the main text and help readers compare meta and non-meta performance.

Table 2: Absolute MAE values for LAMeL performance in BigSolDB2.0. Target solvent: **water**.

N shots	**meta**		**non-meta**	
	mean ± SEM	CI (95%)	mean ± SEM	CI (95%)
10	1.83 ± 0.15	$(1.56, 2.14)$	1.56 ± 0.06	$(1.44, 1.69)$
20	1.45 ± 0.14	$(1.25, 1.77)$	1.38 ± 0.04	$(1.29, 1.46)$
30	1.26 ± 0.02	$(1.22, 1.31)$	1.29 ± 0.04	$(1.21, 1.36)$
40	1.32 ± 0.06	$(1.22, 1.43)$	1.3 ± 0.05	$(1.2, 1.4)$
50	1.26 ± 0.04	$(1.18, 1.34)$	1.24 ± 0.04	$(1.17, 1.3)$
60	1.26 ± 0.04	$(1.18, 1.33)$	1.27 ± 0.06	$(1.17, 1.38)$
70	1.22 ± 0.03	$(1.17, 1.28)$	1.24 ± 0.05	$(1.15, 1.34)$
80	1.18 ± 0.03	$(1.12, 1.24)$	1.21 ± 0.04	$(1.13, 1.28)$
90	1.14 ± 0.03	$(1.09, 1.2)$	1.14 ± 0.02	$(1.1, 1.19)$
100	1.11 ± 0.02	$(1.08, 1.15)$	1.12 ± 0.02	$(1.08, 1.16)$

Table 3: Absolute MAE values for LAMeL performance in BigSolDB2.0. Target solvent: **toluene**.

N shots	**meta**		**non-meta**	
	mean ± SEM	CI (95%)	mean ± SEM	CI (95%)
10	0.94 ± 0.05	$(0.86, 1.05)$	1.21 ± 0.06	$(1.11, 1.33)$
20	0.86 ± 0.04	$(0.79, 0.93)$	1.18 ± 0.06	$(1.06, 1.31)$
30	0.84 ± 0.03	$(0.78, 0.9)$	1.08 ± 0.03	$(1.04, 1.13)$
40	0.85 ± 0.02	$(0.8, 0.89)$	1.06 ± 0.05	$(0.97, 1.16)$
50	0.83 ± 0.02	$(0.79, 0.88)$	1.04 ± 0.04	$(0.96, 1.12)$
60	0.83 ± 0.03	$(0.78, 0.88)$	1.0 ± 0.04	$(0.92, 1.08)$
70	0.82 ± 0.03	$(0.76, 0.88)$	0.98 ± 0.04	$(0.91, 1.05)$
80	0.81 ± 0.03	$(0.76, 0.86)$	0.99 ± 0.03	$(0.93, 1.06)$
90	0.82 ± 0.02	$(0.77, 0.86)$	0.97 ± 0.04	$(0.89, 1.05)$
100	0.81 ± 0.03	$(0.76, 0.86)$	0.98 ± 0.04	$(0.91, 1.05)$

Table 4: Absolute MAE values for LAMeL performance in BigSolDB2.0. Target solvent: **methyl acetate**.

N shots	**meta**		**non-meta**	
	mean ± SEM	CI (95%)	mean ± SEM	CI (95%)
10	0.89 ± 0.16	$(0.64, 1.23)$	0.92 ± 0.05	$(0.83, 1.04)$
20	0.64 ± 0.03	$(0.59, 0.71)$	0.94 ± 0.07	$(0.82, 1.08)$
30	0.63 ± 0.04	$(0.57, 0.71)$	0.88 ± 0.05	$(0.79, 0.97)$
40	0.58 ± 0.02	$(0.54, 0.62)$	0.87 ± 0.04	$(0.79, 0.95)$
50	0.57 ± 0.02	$(0.54, 0.6)$	0.86 ± 0.05	$(0.77, 0.96)$
60	0.58 ± 0.02	$(0.55, 0.61)$	0.87 ± 0.04	$(0.8, 0.95)$
70	0.57 ± 0.01	$(0.55, 0.6)$	0.83 ± 0.03	$(0.77, 0.89)$
80	0.56 ± 0.02	$(0.53, 0.6)$	0.83 ± 0.03	$(0.76, 0.89)$
90	0.57 ± 0.02	$(0.53, 0.6)$	0.8 ± 0.02	$(0.75, 0.84)$
100	0.55 ± 0.02	$(0.51, 0.59)$	0.78 ± 0.02	$(0.74, 0.82)$

3.2 Dielectric Constants for Solvents in BigSolDB2.0 Dataset

We provide dielectric constant values for solvents in BigSolDB2.0 after preprocessing (Table 1). Dielectric constants for solvents in the BigSolDB2.0 dataset were primarily retrieved using the chemicals Python package.[1] For solvents not available through the package, values were manually curated from the reference tables.[2] These values span a wide range, from non-polar solvents such as n-hexane (1.9) to highly polar solvents such as water (80.1). Fig. 7 visualizes solvents with ≥ 20 datapoints, only showing the solvents used in the meta-learning experiments in the main text.

Table 5: Dielectric constants for all solvents in BigSolDB2.0 after preprocessing

Solvent	Dielectric constant, ε	Temperature, K
n-hexane	1.8865	293.20
n-heptane	1.9209	293.20
cyclohexane	2.0243	293.20
1,4-dioxane	2.2189	293.20
tetrachloromethane	2.2379	293.20
benzene	2.2825	293.20
toluene	2.3790	296.35
diethyl ether	4.2666	293.20
MTBE	4.5000	293.15
n-pentyl acetate	4.7900	293.20
chloroform	4.8069	293.20
isobutyl acetate	5.0680	293.20
n-butyl acetate	5.0700	293.20
n-propyl acetate	5.6200	293.20
chlorobenzene	5.6895	293.20
isopropyl acetate	5.7600	298.15
ethyl acetate	6.0814	293.20
acetic acid	6.2000	293.20
methyl acetate	7.0700	288.20
THF	7.5800	298.15
ethyl formate	8.5700	288.20
dichloromethane	8.9300	298.00
n-octanol	10.3000	293.20
1,2-dichloroethane	10.4200	293.20
n-heptanol	11.7500	293.20
2-propoxyethanol	11.7600	293.15
n-hexanol	13.0300	293.20
2-ethoxyethanol	13.3800	298.20
transcutol	13.4000	298.15
n-pentanol	15.1300	298.20
isopentanol	15.6300	293.20

Continued on next page

Solvent	Dielectric constant, ε	Temperature, K
cyclohexanone	16.1000	293.00
2-methoxyethanol	17.2000	298.20
sec-butanol	17.2600	293.20
n-butanol	17.8400	293.20
isobutanol	17.9300	293.20
2-butanone	18.5600	293.20
isopropanol	20.1800	293.20
n-propanol	20.8000	293.20
acetone	21.0100	293.20
ethanol	25.3000	293.20
propylene glycol	27.5000	303.20
NMP	32.5500	293.20
methanol	33.0000	293.20
acetonitrile	36.6400	293.20
DMF	38.2500	293.20
DMAc	38.8500	294.20
ethylene glycol	41.4000	293.20
DMSO	47.2400	293.20
water	80.1000	293.20

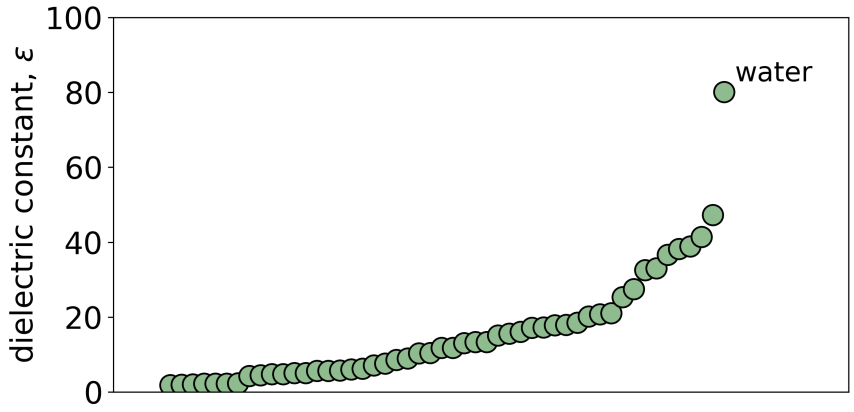

Figure 7: Dielectric constant values for solvents with $\geq$ 20 entries in the BigSolDB 2.0 dataset

3.3 Interpretability Analysis

This section provides additional interpretability analyses supporting the claim that LAMeL preserves coefficient-level interpretability and enables understanding of learned structure–property relationships. All analyses in this section use fixed MinervaChem graphlet fingerprints, for which each feature corresponds to an explicit substructure. As a result, any linear coefficient vector including LAMeL's meta-adapted coefficients and intermediate support-task coefficient vectors can be mapped back to specific substructures and visualized on molecules.

Interpretability results in this section were generated on the BigSolDB 2.0 benchmark using graphlet fingerprints with maximum substructure size 5. For the interpretability tests solubility in acetone was selected to be the target task with all remaining solvents used as support tasks. The support and target splits, N-shot sampling protocol, and ridge regularization selection follow the procedures described in the main text.

Because graphlet fingerprinting results in explicit tie between a particular substructure and it's coefficient, for a molecule with fingerprint vector $x \in \mathbb{R}^d$, the predicted property is $\hat{y} = x^\top \beta$. To visualize the contribution of substructures to $\hat{y}$ at the atom/bond level, we use the respective functionality in `minervachem`.[3] Each graphlet occurrence in the molecule inherits the coefficient weight of the corresponding feature, and the signed contributions are accumulated over atoms/bonds found in the graphlets.$LAMeL$ learns a shared low-dimensional manifold in coefficient space from support tasks. To make this shared structure interpretable, we analyze the collection of support-task coefficient vectors in coefficient space and visualize dominant directions of variation.

Let $\beta_\tau \in \mathbb{R}^d$ denote the ridge regression coefficient vector for support task τ We assemble a coefficient matrix $B \in \mathbb{R}^{T \times d}, \quad B_{\tau,:} = \tilde{\beta}_\tau^\top$, where $\tilde{\beta}_\tau$ denotes a β_τ weight-adjusted through step 2 of the $LAMeL$ workflow. We perform principal component analysis (PCA) on the centered coefficient matrix to obtain principal directions $\{v_k\}_{k=1}^K$, where $v_k \in \mathbb{R}^d$ is a direction

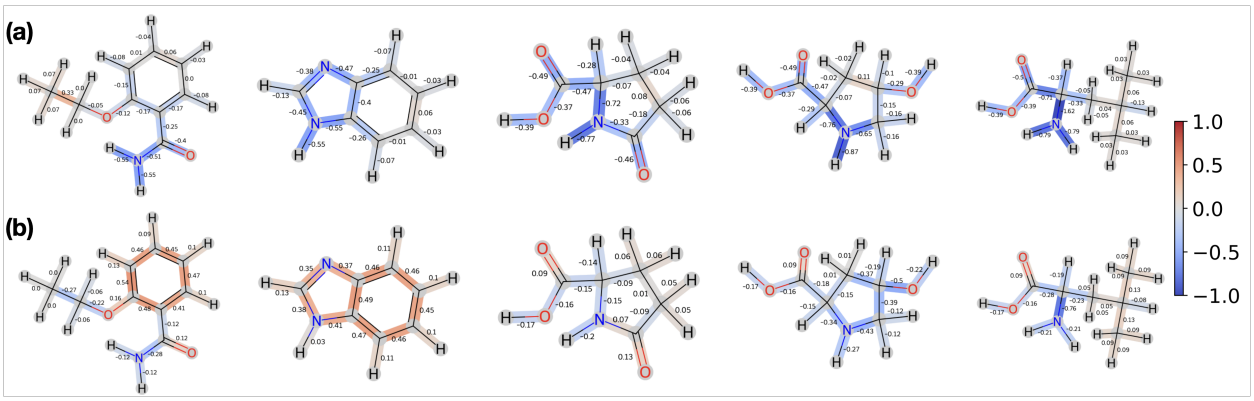

Figure 8: Projections of principal components (PC) of shared support tasks subspace onto molecular bonds : (a) PC1, (b) PC2. Colors indicate signed contributions after normalization to [-1,1] for visualization.

in feature space containing a dominant mode of cross-task variation in linear coefficients. We then treat v_k as a coefficient vector and project it onto representative molecules using the bond attribution mapping described above.

Figure 8 shows example projections of the first two principal directions onto representative molecules. Different principal components emphasize distinct chemical motifs: polar functional groups in PC1 and aromatic/ring fragments in PC2, suggesting that the support task manifold captures chemically recognizable solubility patterns.

4 QM9-MultiXC results

For comparison, we applied non-meta ridge regression to the QM9-MultiXC benchmark. Fig. 9 reports results for three density functional approximations (M06-L SZ, TPSSH DZP, and MPBE0KCIS TZP). As expected, model errors decrease with increasing training shots; however, the performance remains consistently inferior to that achieved with the meta-learning approach.

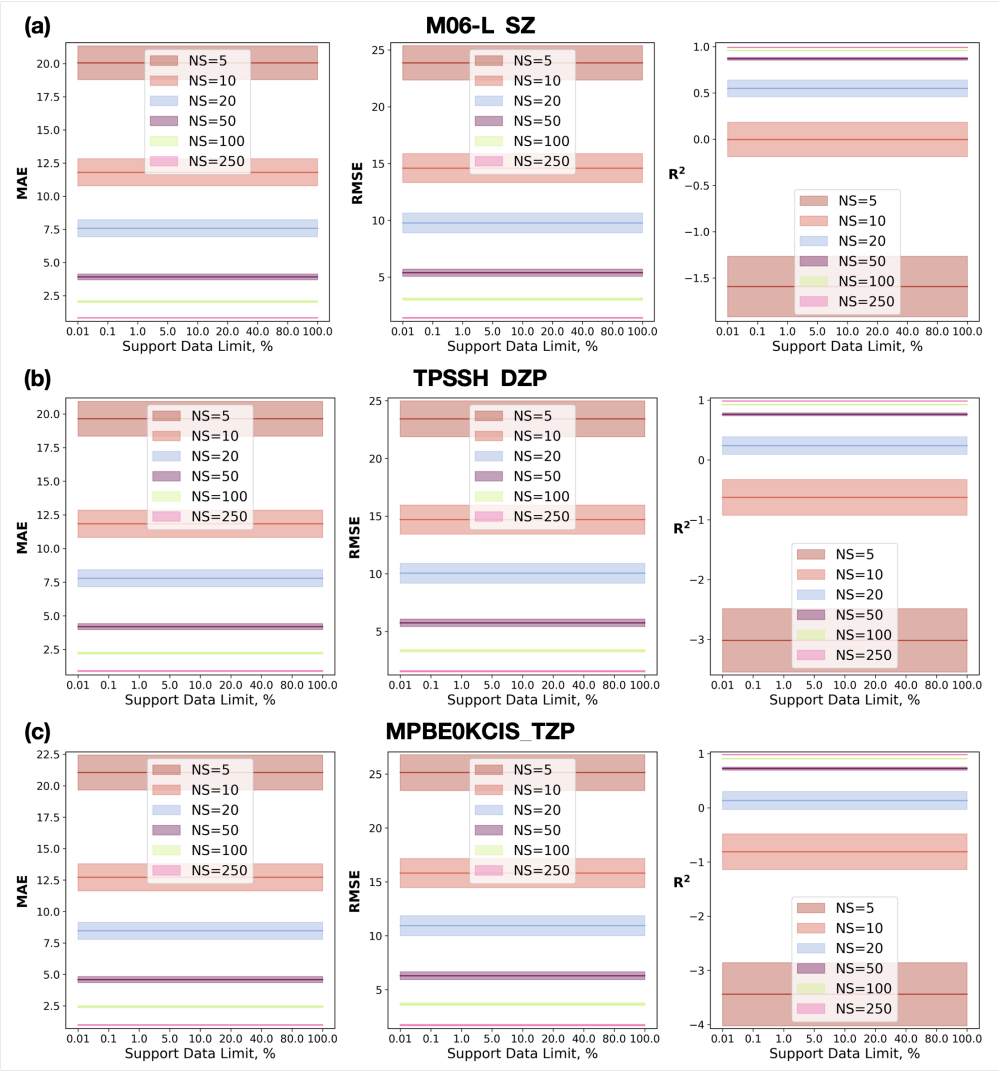

Figure 9: Non-meta ridge regression performance for three target tasks: (a) M06-L_SZ, (b) TPSSH_DZP, and (c) MPBE0KCIS_TZP. The legend indicates the number of shots (NS) used during training.

4.1 Absolute Performance in QM9-MultiXC Dataset

Tables S6-9 report absolute test MAE values for representative target tasks from the QM9-MultiXC benchmark under the same N-shot evaluation protocol used throughout this work. For each N, we summarize performance for LAMeL (meta) and the per-task ridge baseline (non-meta) as mean ± SEM over 10 random seeds, together with 95% confidence intervals. Here, the support regime uses all available support-task data and a fixed set of five support tasks drawn at random.

Table 6: Absolute MAE values for LAMeL performance in QM9-MultiXC. The results are shown for all available data used in the support regime and 5 support tasks. Target task: **M06-L_SZ**.

N shots	**meta**		**non-meta**	
	mean ± SEM	CI (95%)	mean ± SEM	CI (95%)
5	0.76 ± 0.2	$(0.36, 1.18)$	20.05 ± 1.27	$(17.63, 22.53)$
10	0.33 ± 0.02	$(0.3, 0.36)$	11.8 ± 1.02	$(9.95, 13.88)$
20	0.28 ± 0.01	$(0.26, 0.31)$	7.58 ± 0.65	$(6.42, 8.89)$
50	0.23 ± 0.01	$(0.22, 0.25)$	3.91 ± 0.22	$(3.48, 4.3)$
100	0.22 ± 0.0	$(0.21, 0.23)$	2.05 ± 0.08	$(1.9, 2.2)$
250	0.21 ± 0.0	$(0.21, 0.22)$	0.83 ± 0.02	$(0.79, 0.87)$

Table 7: Absolute MAE values for LAMeL performance in QM9-MultiXC. The results are shown for all available data used in the support regime and 5 support tasks. Target task: **TPSSH_DZP**.

N shots	**meta**		**non-meta**	
	mean ± SEM	CI (95%)	mean ± SEM	CI (95%)
5	2.95 ± 0.82	$(1.71, 4.73)$	19.64 ± 1.3	$(17.18, 22.18)$
10	1.6 ± 0.1	$(1.42, 1.82)$	11.83 ± 1.01	$(9.99, 13.86)$
20	1.24 ± 0.05	$(1.13, 1.34)$	7.78 ± 0.64	$(6.68, 9.09)$
50	0.76 ± 0.02	$(0.73, 0.79)$	4.19 ± 0.22	$(3.76, 4.6)$
100	0.55 ± 0.01	$(0.52, 0.58)$	2.22 ± 0.08	$(2.08, 2.38)$
250	0.43 ± 0.01	$(0.41, 0.45)$	0.89 ± 0.04	$(0.83, 0.99)$

Table 8: Absolute MAE values for LAMeL performance in QM9-MultiXC. The results are shown for all available data used in the support regime and 5 support tasks. Target task: **MPBE0KCIS_TZP**.

N shots	**meta**		**non-meta**	
	mean ± SEM	CI (95%)	mean ± SEM	CI (95%)
5	3.49 ± 0.9	$(2.06, 5.48)$	21.05 ± 1.39	$(18.41, 23.78)$
10	2.07 ± 0.15	$(1.78, 2.34)$	12.72 ± 1.08	$(10.76, 14.89)$
20	1.41 ± 0.06	$(1.29, 1.51)$	8.46 ± 0.68	$(7.28, 9.88)$
50	0.85 ± 0.02	$(0.81, 0.88)$	4.59 ± 0.25	$(4.11, 5.03)$
100	0.61 ± 0.01	$(0.59, 0.64)$	2.43 ± 0.09	$(2.27, 2.6)$
250	0.47 ± 0.01	$(0.45, 0.5)$	0.98 ± 0.04	$(0.91, 1.07)$

5 Baseline implementation details and hyperparameters

This section provides implementation details for additional baselines: a nonlinear model (LightGBM) and joint linear regression with task conditioning. Unless otherwise stated, all baselines use the same molecular representations, data splits, and evaluation protocol as the main-text LAMeL experiments. For each target task, we subsample the training data to obtain $N \in \{5 \dots 160\}$ labeled datapoints ("shots") for model fitting, and report performance on the same held-out test split as in the main text. For each N, we repeat each run 10 times with different random seeds and report the mean and standard error across repeats.

5.1 Nonlinear Baseline

For the nonlinear baseline (LightGBM), we train only on the target task data at each N (i.e., no incorporation of support-task data), to isolate whether improvements can be attributed simply to nonlinear modeling in the same low-data regime. We selected LightGBM[4] as a lightweight nonlinear baseline because it has shown the best results among non-linear models in our previous work with molecular fingerprints. We evaluated both fixed hyperparameters and FLAML AutoML tuning.[5]

Fixed hyper parameters:

- `num_leaves` = 31
- `learning_rate` = 0.003
- `n_estimators` = 300
- `min_child_samples` = 20
- `subsample` = 0.8

- `colsample_by_tree` = 0.8

FLAML setup:

- Estimator: `lgbm`
- Time budget per run: 45 seconds
- Optimization metric: MAE
- Search space: default FLAML

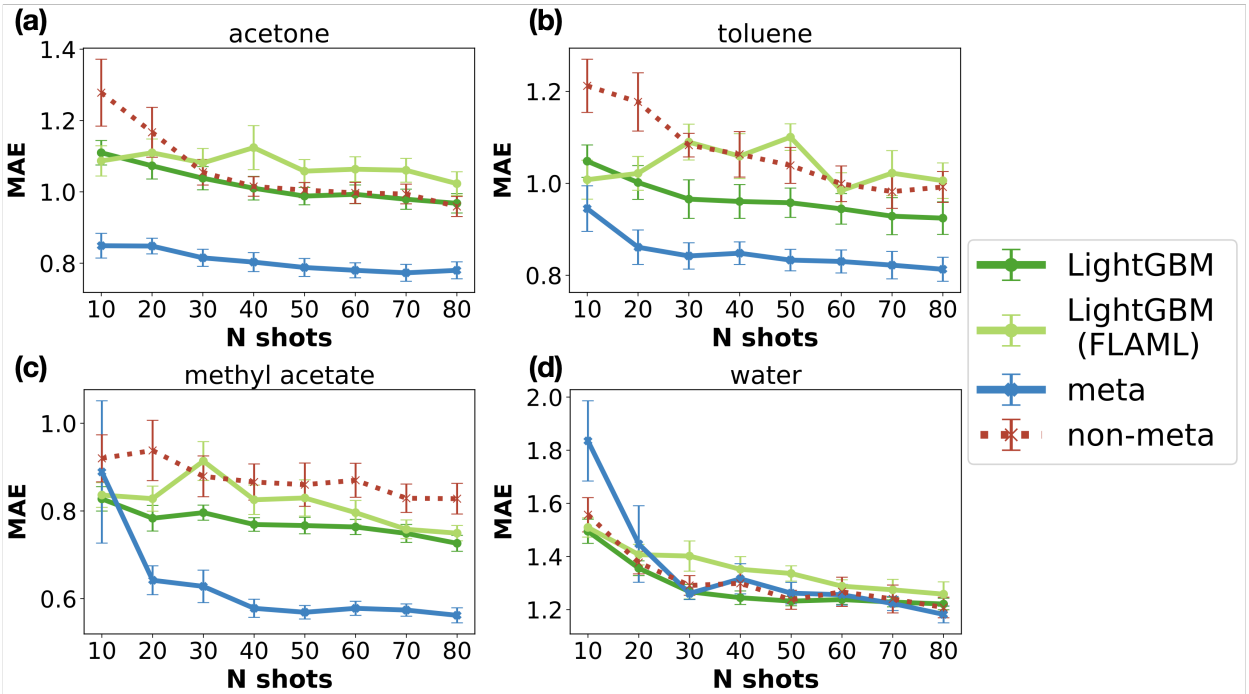

Figure 10: Comparison of LAMeL with nonlinear LightGBM baselines in the low-data regime for four representative BigSolDB2.0 target solvents: (a) acetone, (b) toluene, (c) methyl acetate, and (d) water.

5.2 Joint Task-Conditioned Ridge Regression

To test whether LAMeL's gains can be reproduced by simple task pooling, we trained a single ridge regression model on the union of all training datapoints across tasks, with a feature representation that encodes task identity via one-hot encoding.

Let $x \in \mathbb{R}^d$ be the molecular fingerprint for a datapoint, and let t denote its task index. The joint baseline constructs an augmented feature vector $\tilde{x} = [x; \phi(t)]$, where $\phi(t)$ is a task encoding. The resulting model is trained by minimizing:

$$\min_{\tilde{\beta}} \ \|\tilde{X}\tilde{\beta} - y\|_2^2 + \lambda\|\tilde{\beta}\|_2^2. \tag{1}$$

Figure 11 shows that the joint task-conditioned ridge baseline exhibits heterogeneous performance across target solvents. For acetone and methyl acetate, joint regression achieves lower MAE than LAMeL under the evaluated default support configuration, whereas for toluene it is comparable and for water it is notably worse. The observed pattern is consistent with the role of task relatedness: when the target task is well represented by the pooled training distribution, joint regression can benefit from the additional data through task-specific offsets. It is worth noting, LAMeL can narrow this gap by selecting support tasks that are more similar to the target(see section 6.2 in the SI).

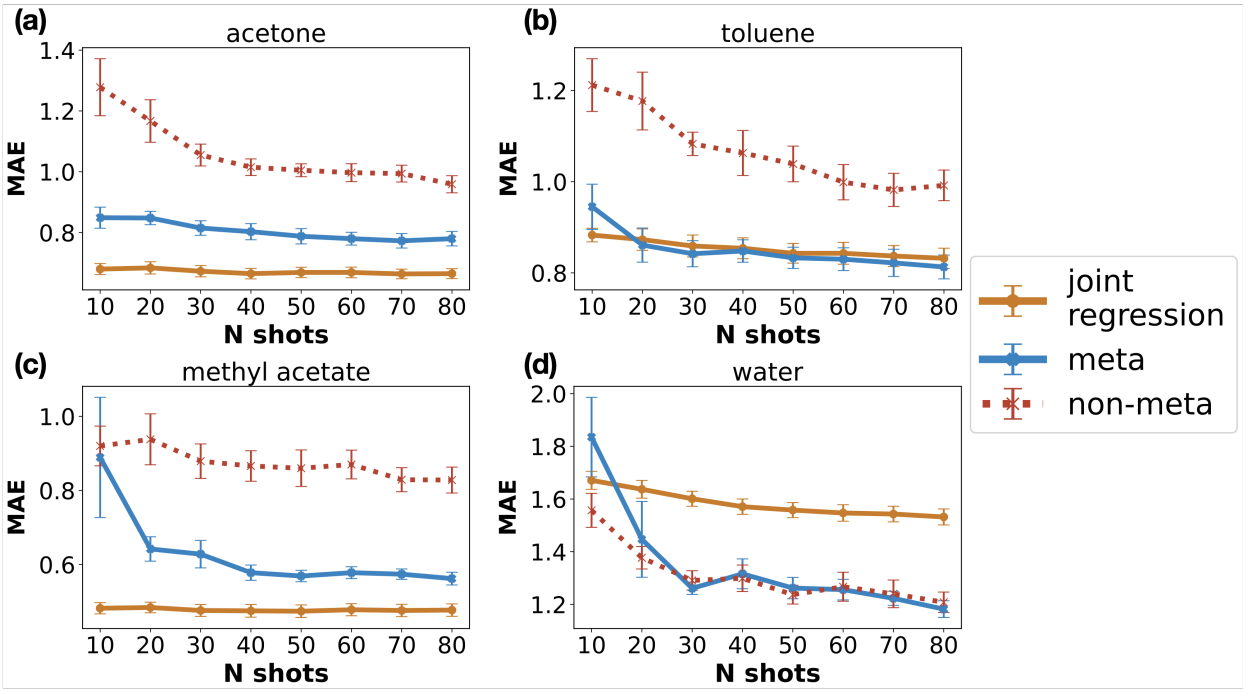

Figure 11: Joint task-conditioned ridge regression (one-hot task indicator) compared with LAMeL for four representative BigSolDB2.0 target solvents: (a) acetone, (b) toluene, (c) methyl acetate, and (d) water.

Interpretability and extensibility considerations. Although this baseline is linear, it introduces task-identity features that can absorb predictive signal and thereby reduce substructure-level interpretability. Moreover, because the task identity is encoded explicitly, extending the model to a new unseen task requires introducing a new indicator feature and refitting the pooled model, whereas LAMeL is designed to adapt to unseen tasks without retraining on prior tasks.

Note on Multi-Task Regression Many classic multi-task regression formulations assume aligned samples across tasks or require dedicated optimization to handle arbitrary missing labels and structured task–task covariance. Because our datasets contain limited molecule overlap across tasks and substantial per-entry missingness, implementing a fully coupled multi-task ridge model with structured task dependence would require a separate optimization procedure beyond the scope of this work.

6 Sensitivity Analyses on Different Components of *LAMeL*

We have tested isolated the contributions of the parallel ($\beta^{\parallel}$) and perpendicular ($\beta^{\perp}$) components of LAMeL, and the support-task set (size T and similarity-based selection). LAMeL decomposes the target-task coefficient vector into components parallel and perpendicular to the learned support-task subspace $\beta^{\star} = \beta^{\parallel} + \beta^{\perp}$ (see main text, Section 2.4).

Figure 12 reports two main components tested individually alongside full LAMeL implementation and non-meta performance for different target tasks. The perpendicular-only option performs poorly in the low-data regime, consistent with the fact that $\beta^{\perp}$ is estimated from the N target task datapoints. In contrast, the parallel-only variant typically captures most of the improvement over non-meta ridge, indicating that the learned support task contributions is the dominant factor when the target task is aligned with supports. Nevertheless, we observe full model $\beta^{\star} = \beta^{\parallel} + \beta^{\perp}$ improving over $\beta^{\parallel}$ alone, consistent with $\beta^{\perp}$ providing a target-data-driven correction when the support tasks manifold is very dissimilar to the target task.

6.1 Effect of the Number of Support Tasks T

To assess whether *LAMeL*'s performance depends on the number of support tasks alone, we repeated the experiments with varying support-set size $T \in \{3, 5, 10, T_{\text{full}}\}$, where T_{full} denotes using all available support tasks. For each random trial and each T, we sampled T support tasks at random without replacement from the pool of available support tasks, and then ran the full LAMeL pipeline.

Figure 13 shows *LAMeL* performance with different support tasks. We have not identified any distinguishable differences, which indicated limited to no dependence on the number of support tasks under random selection.

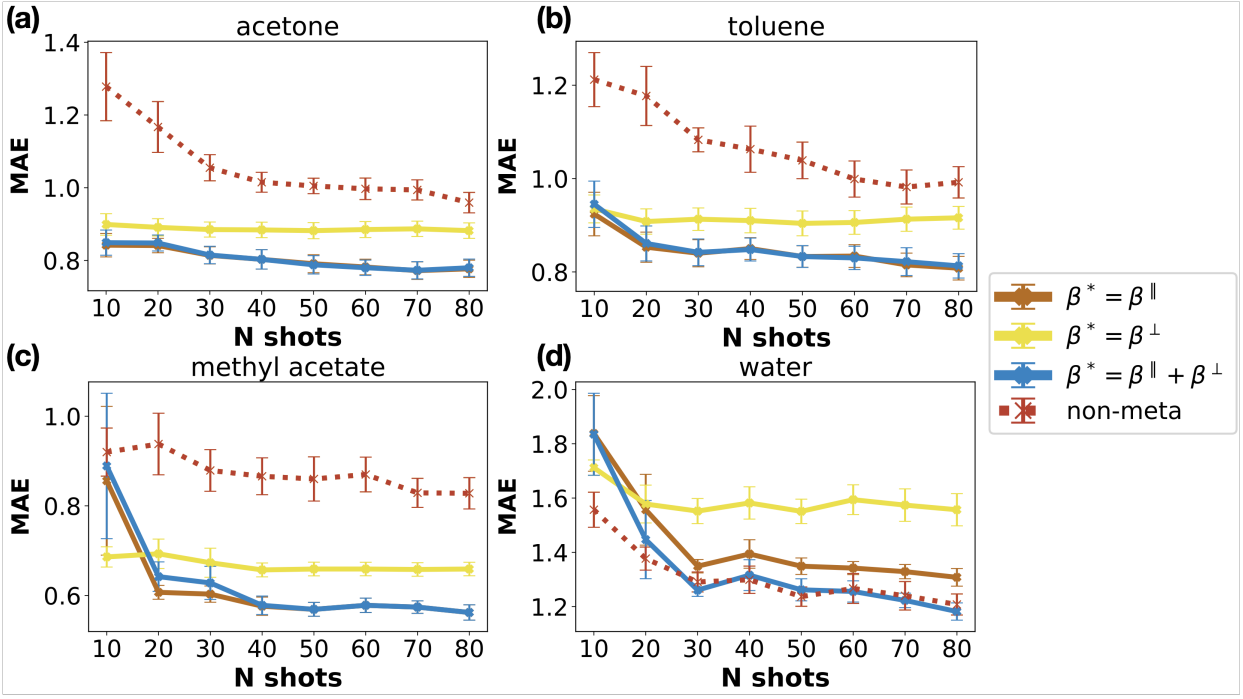

Figure 12: Performance of LAMeL algorithm decomposed into parallel and perpendicular components for representative target tasks: (a) acetone, (b)toluene, (c)methyl acetate, (d) water. Error bars are collected over 10 individual random runs.

6.2 Similarity-Based Support Task Selection

We further tested whether selecting more similar support tasks improves meta-performance. For each target task, we ranked candidate support tasks by a similarity score and selected support subsets corresponding to high similarity support(top-4 most similar tasks), medium similarity support (4 tasks with medium similarity), and low similarity support (4 least similar tasks). Consistent with main text we considered molecular fingerprint and regression vector task similarity. Details of the similarity computation are provided in main text, Section 3.3.

Figure 14 shows that restricting the support set to higher-similarity tasks can improve performance, often achieving comparable or better results with fewer support tasks. The observation is inline with main-text interpretation that support-task relevance is more important than sheer volume of support task set. At the same time, we note that different

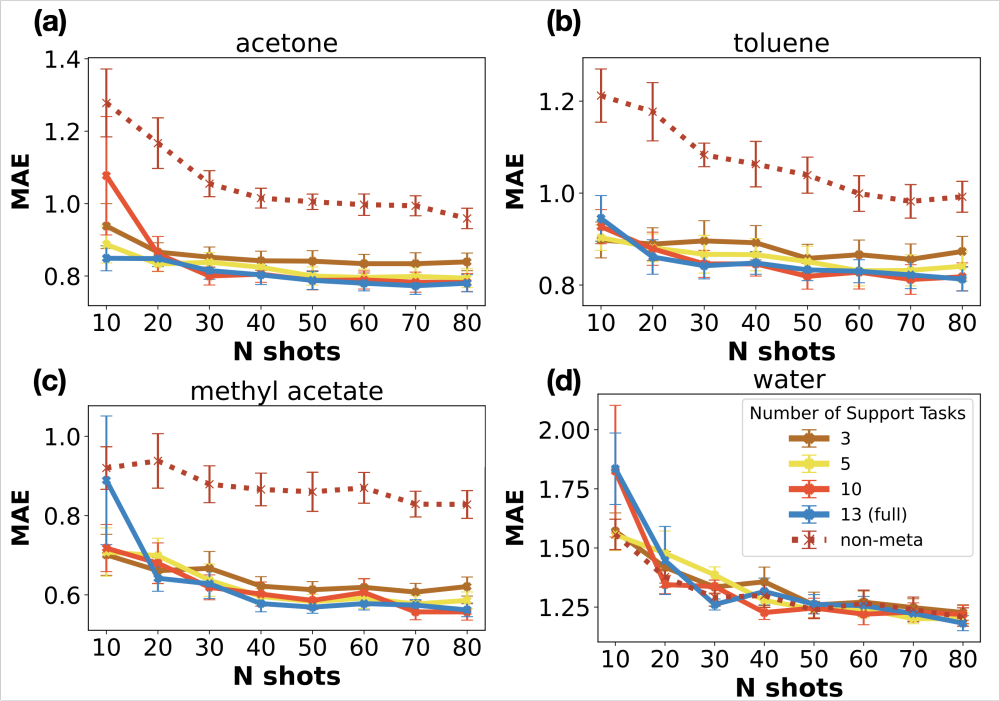

Figure 13: LAMeL performance for randomly sampled support tasks sets of size $T \in \{3, 5, 10, T_{\text{full}}\}$. Error bars are collected over 10 individual random runs.

similarity definitions (left and right columns in Fig. 14) can produce different task rankings for the same target, indicating that robust similarity-based task selection is non-trivial.

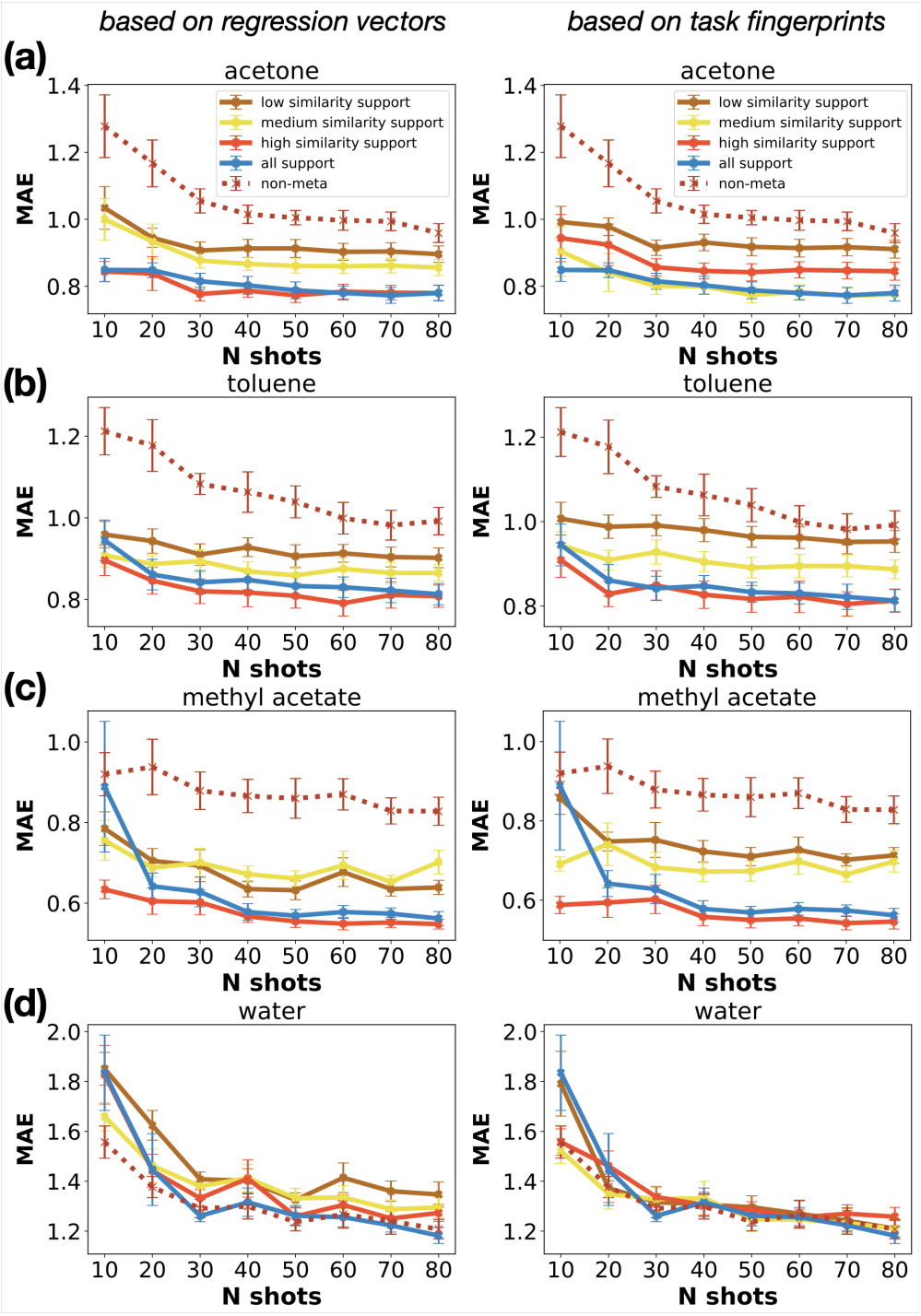

Figure 14: Comparison of *LAMeL* using all support tasks versus subsets selected by similarity for representative target tasks: (a) acetone, (b) toluene, (c) methyl acetate, (d) water. Similarity computed using regression-based (left column) or fingerprint-based (right column) metric. Error bars are collected across 10 independent data splits.

References

(1) Bell, C.; Cortes-Pena, Y. R.; Contributors Chemicals: Chemical properties component of Chemical Engineering Design Library (ChEDL). 2016-2025; `https://github.com/CalebBell/chemicals`.

(2) Wohlfarth, C. Dielectric constants: Datasheet from Landolt-Börnstein - Group IV Physical Chemistry · Volume 17: "Supplement to IV/6" in SpringerMaterials (https://doi.org/10.1007/978-3-540-75506-7_212). `https://materials.springer.com/lb/docs/sm_lbs_978-3-540-75506-7_212`, Copyright 2008 Springer-Verlag Berlin Heidelberg.

(3) Tynes, M.; Taylor, M. G.; Janssen, J.; Burrill, D. J.; Perez, D.; Yang, P.; Lubbers, N. Linear graphlet models for accurate and interpretable cheminformatics. *Digital Discovery* **2024**, *3*, 1980–1996.

(4) Ke, G.; Meng, Q.; Finley, T.; Wang, T.; Chen, W.; Ma, W.; Ye, Q.; Liu, T.-Y. LightGBM: A Highly Efficient Gradient Boosting Decision Tree. Proceedings of the 31st International Conference on Neural Information Processing Systems. pp 3149–3157.

(5) Wang, C.; Wu, Q.; Weimer, M.; Zhu, E. FLAML: A Fast and Lightweight AutoML Library. `http://arxiv.org/abs/1911.04706`.