CLOSER: CHARACTERISTIC LOCATIONS FOR EFFICIENT GEOSPATIAL MODEL EVALUATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Geospatial prediction tasks estimate outcomes such as house prices or disease incidence for specific locations or regions. Candidate pipelines for such tasks may select different options for representing locations, additional input features, and prediction models. Comparing these pipelines can require repeated training and evaluation across thousands of locations, often for multiple outcomes or time periods. This paper addresses the challenge of efficient candidate pipeline selection: Can we identify a small weighted subset of characteristic locations to approximate the performance of a candidate pipeline over all locations? We introduce CLOSER, which uses knowledge about task-aware location representations and solutions to the facility location problem to select complementary representative locations, where a weight associates the count of locations that are assigned to each representative location. The subset size is determined using predictive scores on the candidate subsets, without requiring full-location pipeline scores. The selected subset and weights is then applied across all candidate pipelines for that task. Based on evaluations over outcomes including house prices, health statistics, COVID-19 incidence, and point-of-interest spending, we demonstrate that the selected subsets achieve higher Spearman rank correlation with all location rankings, in comparison to matched-size uniform sampling, while retaining less than 2% of locations on most tasks. Paired timing measurements show $6.3\times$–$84.2\times$ speedups for the subset, when compared to evaluation over all locations, over the suite of outcomes and candidate pipelines.

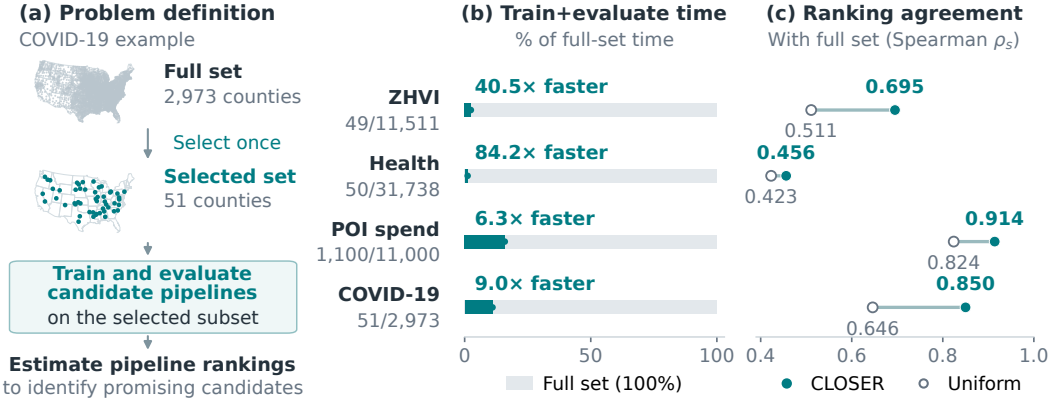


Figure 1: Efficient geospatial pipeline evaluation using small location subsets. (a) Problem definition: can training and evaluating candidate pipelines on a small, reusable subset approximate their full-location ranking? Training and evaluation use separate splits of the selected location pool. Maps show the full COVID-19 pool and a selected subset. (b) Recurring training and evaluation time for complete pipeline suites using CLOSER, relative to full-pool evaluation (gray). Labels show speedups and selected/full location counts. (c) CLOSER's task-level Spearman agreement with full-pool rankings, compared with matched-size uniform sampling; segments connect paired point estimates.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

1 INTRODUCTION

Comparing candidate machine learning pipelines is a fundamental step in geospatial prediction tasks such as estimating house prices or COVID-19 infection counts. A candidate geospatial prediction *pipeline* will select options for how locations are represented, which input features are used, and the choice of prediction model. Identifying the best pipelines requires repeated training and evaluation across thousands of locations, often for multiple outcomes or time periods. We seek to reduce the computational cost of these comparisons by evaluating pipelines over a small, reusable subset of locations. This paper addresses the following question: Can we identify a small weighted subset of characteristic locations to approximate the performance of a candidate pipeline over all locations (Figure 1(a))?

Prior work has demonstrated that small subsets can approximate full-set model scores or rankings in language-model benchmarking (Maia Polo et al., 2024; Vivek et al., 2024; Perlitz et al., 2024; Saranathan et al., 2025) and functional-connectivity evaluation (Zhan et al., 2026). Training coresets similarly reduce computational cost by selecting subsets that approximate full-data optimization or maintain predictive performance (Mirzasoleiman et al., 2020; Killamsetty et al., 2021). Our objective differs from both settings: we seek to preserve the relative performance of multiple candidate pipelines using a single, reusable subset for both training and held-out evaluation.

Achieving this objective in geospatial prediction presents two challenges. First, **location similarity is task-dependent**: locations exhibiting similar housing-market patterns need not exhibit similar health outcomes. Spatially balanced sampling emphasizes geographic coverage (Stevens & Olsen, 2004), which need not capture the task-specific variation relevant to distinguishing pipeline performance. Second, unlike benchmark reduction methods that evaluate already-trained models on smaller test sets, our setting requires **both training and held-out evaluation** within the selected subset, using separate training and test locations. Subset selection therefore affects not only evaluation cost but also the fitted predictors, making the preservation of full-location pipeline rankings more challenging.

To address these challenges, we propose **CLOSER** (**C**haracteristic **Lo**cations for **S**patial **E**valuation and **R**anking), which selects a small, weighted set of *characteristic locations* tailored to the prediction task. We first train a probe to predict observed outcomes from pretrained regional embeddings and use its hidden representations to measure task-specific location similarity. Following the facility-location formulation used in benchmark reduction (Zhao et al., 2025; Yao et al., 2026), we select locations that jointly cover the pool in this learned representation space. Coverage weights reflect how much of the pool each selected location represents. An automatic budget rule balances held-out predictive performance against subset size. The selected locations and weights are then fixed and reused for training and held-out evaluation of candidate pipelines for the same task and location pool.

Our main contributions are as follows:

- We introduce CLOSER, which combines task-aware location similarities, coverage weights, and automatic budget selection to construct a reusable characteristic set.
- We demonstrate that CLOSER improves agreement with full-location rankings over uniform sampling at matched subset sizes across house prices, health statistics, COVID-19 incidence, and point-of-interest spending prediction tasks.
- After one-time setup costs, CLOSER achieves $6.3\times$–$84.2\times$ speedups for the subset, when compared to evaluation over all locations, over the suite of outcomes and candidate pipelines (Figure 1(b,c)).

2 PROBLEM FORMULATION

We define a *geospatial prediction task* as predicting one or more outcomes associated with a geographic unit. These outcomes may be observed across dates or span several regional variables. We call each geographic unit a *location*; for example, it may be the area covered by a ZIP code or a U.S. county. In this paper, we consider house values, public-health outcomes, point-of-interest spending and transaction counts, and COVID-19 cases and deaths.

To solve each prediction task, we may choose among multiple *candidate pipelines*. These pipelines may differ in their input features, geographic representations, or prediction models. Evaluating every candidate pipeline across a large set of locations can be expensive. We therefore seek a much smaller weighted subset that approximates the performance of a candidate pipeline over all locations.

Let $\mathcal{S} = \{s_1, \ldots, s_N\}$ be the full set of N locations. A characteristic set is a subset $P \subset \mathcal{S}$, optionally equipped with nonnegative weights $w = (w_i)_{i \in P}$ satisfying $\sum_{i \in P} w_i = 1$. We write $K = |P|$ and seek K much smaller than N.

Let $\mathcal{M} = \{m_1, \ldots, m_L\}$ denote any finite collection of candidate pipelines that we wish to compare. Full evaluation gives pipeline m a score $q^{\text{full}}(m)$, while evaluation on the characteristic set gives it a score $q^{\text{sub}}(m; P, w)$. The two score vectors are

$$\mathbf{v}^{\text{full}}(\mathcal{M}) = [q^{\text{full}}(m_1), \ldots, q^{\text{full}}(m_L)], \tag{1}$$

and

$$\mathbf{v}^{\text{sub}}(\mathcal{M}; P, w) = [q^{\text{sub}}(m_1; P, w), \ldots, q^{\text{sub}}(m_L; P, w)]. \tag{2}$$

Assume that scores are oriented so that larger values are preferred. Ideally, a characteristic set preserves every pairwise preference made by full evaluation: for any $m_i, m_j \in \mathcal{M}$,

$$q^{\text{full}}(m_i) > q^{\text{full}}(m_j) \quad \Longrightarrow \quad q^{\text{sub}}(m_i; P, w) > q^{\text{sub}}(m_j; P, w). \tag{3}$$

We call this the *exact rank-preservation condition*. It states that every pairwise choice made by full evaluation is unchanged on the characteristic set. A small subset may not preserve every comparison, so we measure how closely the two complete rankings agree.

The score vectors in Equation (1) and Equation (2) induce two rank orders over the same candidate pipelines. We quantify their agreement using

$$\rho_s\big(\mathbf{v}^{\text{full}}(\mathcal{M}), \mathbf{v}^{\text{sub}}(\mathcal{M}; P, w)\big)\,, \tag{4}$$

where ρ_s is Spearman rank correlation, computed from the ranks of the vector entries. Larger values indicate closer ranking agreement; $\rho_s = 1$ corresponds to exact rank preservation. We aim to find a single (P, w) that achieves high ranking agreement for different choices of $\mathcal{M}$.

3 Approach

CLOSER constructs a task-specific weighted characteristic set in three stages, summarized in Figure 2. First, location representations define a pairwise similarity matrix. Second, greedy facility-location selection produces characteristic sets at candidate budgets, and coverage weights reflect how much of the pool each selected location represents. Third, automatic budget selection balances held-out predictive performance against subset size to choose an operating budget K^*. The returned set (P_{K^*}, w_{K^*}) is then frozen and reused to compare candidate pipelines.

3.1 Computing pairwise similarities

Let z_i denote a representation of location i. In our main experiments, z_i is the hidden state of a task-aware probe trained to predict outcome values at predeclared dates or for predeclared variables from a pretrained regional embedding. This makes the representation sensitive to task-relevant similarities between locations. The pretrained embedding model is specified in Section 4.2; Appendix B.1 gives the probe architecture and training settings.

We compute the similarity between locations i and j from their representations:

$$\mathbf{S}_{ij} = s(z_i, z_j), \tag{5}$$

where s is a similarity function. Our main configuration uses cosine similarity; symmetric and asymmetric alternatives are studied in Section 6.2.

Let $\mathcal{A}$ be the locations eligible for selection and $\mathcal{B}$ the locations to be represented. In the standard setting, both equal $\mathcal{S}$, but they may differ in the experiments. The resulting similarity matrix $\mathbf{S}$ has candidate representatives as rows and targets as columns. Larger entries indicate a closer match under the chosen similarity function.

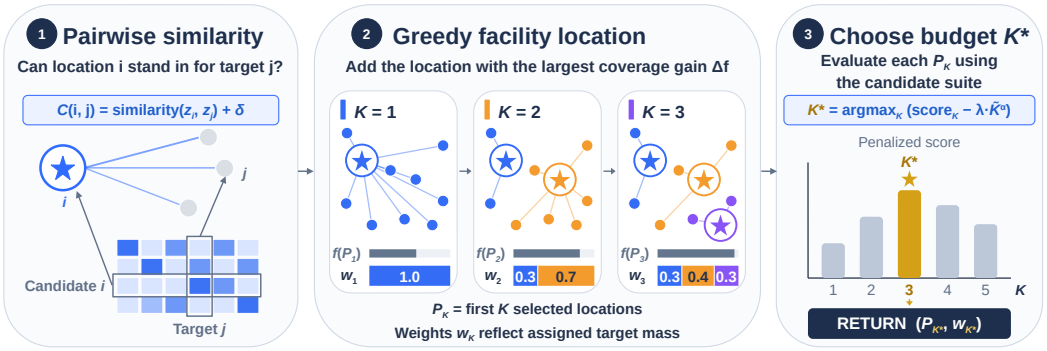


Figure 2: **CLOSER: construction of a weighted characteristic-location set. (1)** Pairwise similarities between location representations define coverage scores $C(i, j)$; a uniform shift δ ensures nonnegativity. **(2)** Greedy facility-location selection adds the representative with the largest marginal coverage gain, producing nested sets P_K. Stars mark selected representatives, colors indicate target assignments, and the bars below each set summarize total coverage $f(P_K)$ and assignment-derived weights w_K. **(3)** Predictive scores on the candidate weighted sets guide budget selection. After scouting and confirmation, the operating budget K^* is chosen among the shortlisted candidates by balancing predictive performance against a subset-size penalty. Here, score_K and $\widetilde{K}$ denote the confirmed predictive score and subset size, respectively, both normalized over the confirmed shortlist. The returned set (P_{K^*}, w_{K^*}) can be reused to screen candidate pipelines, with the aim of preserving their full-location ranking. Locations, budgets, and numerical values are illustrative.

To use these similarities as nonnegative coverage scores, we define

$$\delta = \max\left\{0, -\min_{i\in\mathcal{A},\, j\in\mathcal{B}} \mathbf{S}_{ij}\right\}, \qquad C(i,j) = \mathbf{S}_{ij} + \delta. \tag{6}$$

The uniform additive shift is zero when the similarities are already nonnegative. For every nonempty selected set, it increases the sum of best-representative scores by the same constant $|\mathcal{B}|\delta$. It therefore preserves the ordering of nonempty subsets, the greedy selection order, and each target's best representative. Nonnegativity lets the coverage objective below start at zero for the empty set while remaining monotone. Sparse entries are restored to their recorded fill value before this conversion, as detailed in Appendix B.2.

3.2 Greedy selection and coverage weights

For a candidate budget K, we want K locations that collectively represent all targets well. If two locations represent largely the same targets, selecting both adds little coverage. We therefore seek complementary representatives, so that each new location improves coverage of targets that are not already well represented by the current set.

We encode this complementarity by letting each target contribute only the score of its best selected representative. A new location contributes a marginal gain only for targets whose current best coverage it improves. Summing these best-representative scores gives the objective

$$f(P) = \sum_{j\in\mathcal{B}} \max_{i\in P} C(i,j), \qquad f(\emptyset) = 0, \tag{7}$$

which we maximize subject to $P \subseteq \mathcal{A}$ and $|P| \le K$. This nonnegative facility-location objective is monotone submodular. Greedy selection therefore has the standard $(1 - 1/e)$ approximation guarantee for coverage under a cardinality constraint (Nemhauser et al., 1978).

Starting from $P_0 = \emptyset$, greedy selection repeatedly adds

$$i_t \in \arg\max_{i\in\mathcal{A}\setminus P_{t-1}} \big[f(P_{t-1}\cup\{i\}) - f(P_{t-1})\big], \qquad P_t = P_{t-1}\cup\{i_t\},$$

breaking ties by a stable ordering of location identifiers. With representations, coverage matrix, pools, and tie-breaking fixed, each budget corresponds to the prefix $P_K = \{i_1, \ldots, i_K\}$. Run-

ning greedy selection separately to each budget gives the same nested sets. The random zero-gain completion used in one ablation is a separate exception described in Appendix B.2.

The selected representatives can stand in for different numbers of target locations. At each budget K, we assign every target to its best representative,

$$a_K(j) \in \arg\max_{i \in P_K} C(i, j),$$

using a stable order to break ties. Assuming that each target location contributes equally, the raw coverage weight is

$$\widetilde{w}_{K,i} = \frac{1}{|\mathcal{B}|} \sum_{j \in \mathcal{B}} \mathbf{1}\{a_K(j) = i\}. \tag{8}$$

Thus a representative receives weight in proportion to the fraction of targets assigned to it.

We apply a small floor to these coverage weights before renormalizing:

$$w_{K,i} = \frac{\max\{\widetilde{w}_{K,i}, \epsilon_K\}}{\sum_{\ell \in P_K} \max\{\widetilde{w}_{K,\ell}, \epsilon_K\}}, \qquad 0 \le \epsilon_K < 1/K. \tag{9}$$

The resulting vector is $w_K = (w_{K,i})_{i \in P_K}$. Assignments and weights are recomputed for each K: adding a representative can reassign targets and change the weights of previously selected locations. The floor ϵ_K is a smoothing hyperparameter: smaller values preserve the assignment-derived weights more closely, while larger values reduce the influence of very small clusters. The floor setting and its budget-dependent cap are specified in Appendix B.2.

The coverage objective determines which locations are selected at a fixed budget; it does not choose the operating budget. We next use predictive scores on these subsets to choose the operating budget.

3.3 Automatic budget selection

We choose K by balancing predictive performance on the selected locations against subset size, our proxy for evaluation cost. For each budget in a fixed grid $\mathcal{K}$, we train and score pipelines on repeated $80/20$ location-level splits of (P_K, w_K). Our experiments use the full candidate suite $\mathcal{M}$ for this step. Each pipeline is fitted on the selected training locations and scored on the selected held-out locations, with the same split-seed sequence across pipelines at a given budget. The predictive metric for each task is summarized in Section 4.2; Appendix B.3 specifies how predictions are scored and how coverage weights are applied.

The grid, pipeline suite, and repeat protocol are fixed before scoring. This step compares budgets while keeping the selected locations and weights at each K fixed. Let $s_{K,m,r}$ be the subset predictive score of pipeline $m \in \mathcal{M}$ at budget K in repeat r, oriented so that larger values are preferred. We summarize each budget over R repeats by

$$m_K^{(R)} = \operatorname*{median}_{m \in \mathcal{M}} \operatorname*{median}_{1 \le r \le R} s_{K,m,r}. \tag{10}$$

The inner median reduces sensitivity to unusual splits; the outer median summarizes performance across pipelines.

To limit search cost, we use two stages. *Scouting* evaluates every $K \in \mathcal{K}$ with R_{s} repeats and retains the J largest values of $m_K^{(R_{\mathrm{s}})}$, breaking cutoff ties toward smaller budgets. This gives a shortlist $\mathcal{K}_{\mathrm{short}}$. *Confirmation* evaluates only these shortlisted budgets with $R_{\mathrm{c}} > R_{\mathrm{s}}$ total repeats, providing more stable estimates for the final choice.

We then trade confirmed performance against subset size. Let $m_{\min}$ and $m_{\max}$ be the smallest and largest confirmed scores on $\mathcal{K}_{\mathrm{short}}$, and let $K_{\min}$ and $K_{\max}$ be its smallest and largest budgets. When the corresponding ranges are nonzero, define

$$\mathrm{score}_K \equiv \overline{m}_K^{(R_{\mathrm{c}})} = \frac{m_K^{(R_{\mathrm{c}})} - m_{\min}}{m_{\max} - m_{\min}}, \qquad \widetilde{K} = \frac{K - K_{\min}}{K_{\max} - K_{\min}}.$$

Both terms lie in $[0, 1]$ over the confirmed shortlist. Given penalty strength $\lambda \ge 0$ and shape parameter $\alpha > 0$, we choose

$$K^* = \operatorname*{arg\,max}_{K \in \mathcal{K}_{\mathrm{short}}} \left[\mathrm{score}_K - \lambda \widetilde{K}^{\,\alpha} \right], \tag{11}$$

with ties favoring smaller budgets. Increasing λ strengthens the preference for smaller sets, while $\lambda = 0$ removes the explicit size penalty. The parameter α controls its curvature; our reported $\alpha = 1/2$ penalizes the first increases above $K_{\min}$ more sharply and then flattens.

The method returns $(P, w) = (P_{K^*}, w_{K^*})$. Exact grids, policy constants, and degenerate normalization cases are given in Appendix B.5.

3.4 Reusable pipeline screening

With $(P, w) = (P_{K^*}, w_{K^*})$ fixed, we evaluate each pipeline in the candidate suite $\mathcal{M}$ using the subset evaluator defined in Section 3.3. We average each pipeline's predictive scores across repeated splits using held-out test mass. Collecting these per-pipeline averages gives $\mathbf{v}^{\mathrm{sub}}(\mathcal{M}; P, w)$ from Equation (2); sorting this vector yields a subset ranking. For the same task and location pool, the selected locations and weights can be reused when additional candidate pipelines are introduced, without repeating subset selection.

4 Experimental Setup

4.1 Tasks and candidate pipelines

We study four geospatial prediction tasks: ZIP-level house values from the Zillow Home Value Index (ZHVI) (Zillow Research, 2026), regional health outcomes from CONUS27 (Agarwal et al., 2024), monthly point-of-interest spending (SafeGraph, 2026), and county-level 7-day-mean COVID-19 cases (The New York Times, 2021). Depending on the task, each location contributes observations across dates or several outcome variables.

All four tasks use the same eleven feature combinations, combining mobility embeddings, census variables, and mobility-flow totals characterizing regions and their mobility patterns together with Google's Population Dynamics Foundation Model regional embeddings (PDFM) that encode geographic regions as feature vectors using search trends, weather data and point-of-interest categories (Agarwal et al., 2024). Details of the task data and feature combinations are provided in Appendix B.4, and prediction-model settings in Appendix B.3.

4.2 Main experimental protocol

Geospatial representation models, including global location embeddings (Klemmer et al., 2025) and graph-based regional embeddings (Liang et al., 2022), provide a basis for measuring similarity between locations. Our selector uses pretrained PDFM embeddings. We train a task-aware probe on these embeddings and use its hidden states for subset selection (Section 3.1). The probe architecture and training settings are specified in Appendix B.1.

For each of the four main tasks, candidate pipelines are trained and scored using repeated location-level train/test splits of the selected set. The held-out predictive score $s_{K,m,r}$ used for budget selection (Section 3.3) is coverage-weighted R^2 for POI and COVID-19, and predictive Spearman correlation for ZHVI and Health. We use Spearman correlation for the latter two tasks because R^2 can be unstable on small held-out sets. Appendix B.3 specifies score aggregation across prediction targets; Table 9 gives the evaluation repeat counts. Candidate-budget grids and the scouting, confirmation, and size-penalty settings are specified in Appendix B.5. Alternative representations, similarity functions, and weighting rules are studied in Section 6.

4.3 Metrics and baselines

Full-location evaluation defines the reference pipeline ordering, and characteristic-set scores define the corresponding subset ordering. Our primary metric is Spearman rank correlation between these orderings. Task-level aggregation of regime-specific correlations is defined in Appendix B.3.

We compare CLOSER with uniformly sampled locations at roughly 10% budgets and at its automatically selected task-specific budgets. Each uniform baseline averages five independently sampled sets, while geographic and representation-space k-medoids each use one selected set. Task-specific repeat counts and uncertainty details are reported in Table 9.

5 MAIN RESULTS

Table 1 compares agreement with full-location pipeline rankings on the four main tasks, at both automatically selected and roughly 10% budgets. For each task, CLOSER (Section 3) returns a weighted evaluation set and its operating budget using the configurations in Section 4.

Table 1: Agreement with the full-location ranking of eleven candidate pipelines per task. Each entry reports the task-level Spearman aggregate; parentheses give the number of evaluated locations. Aggregation rules are given in Appendix B.3, and repeat counts in Table 9.

Method	ZHVI $N = 11{,}511$	Health $N = 31{,}738$	POI spend $N = 11{,}000$	COVID-19 $N = 2{,}973$
Uniform random, $\sim$10% budget	0.339 (1100)	0.538 (3000)	0.824 (1100)	0.576 (297)
CLOSER, $\sim$10% budget	**0.586** (1100)	**0.541** (3000)	**0.914** (1100)	**0.664** (297)
Uniform random, matched budget	0.511 (49)	0.423 (50)	0.824 (1100)	0.646 (51)
Geographic k-medoids, matched budget	0.132 (49)	0.336 (50)	0.732 (1100)	0.205 (51)
Representation k-medoids, matched budget	**0.709** (49)	0.330 (50)	0.868 (1100)	0.718 (51)
Embedding-based facility location	0.700 (77)	0.380 (125)	0.864 (1050)	0.805 (700)
CLOSER	0.695 (49)	**0.456** (50)	**0.914** (1100)	**0.850** (51)

Task-specific budgets. CLOSER returns substantially different operating points across tasks, illustrating why the evaluation budget should adapt to the task rather than be fixed globally.

The matched-budget comparisons separate the benefit of location selection from the benefit of choosing K. CLOSER outperforms uniform random and geographic k-medoids on all four tasks; relative to representation k-medoids, it is stronger on Health, POI spend, and COVID-19 and slightly lower on ZHVI. Compared with embedding-based facility location, it improves ranking agreement on Health, POI spend, and COVID-19, while using fewer locations with slightly lower agreement on ZHVI. Together, these results indicate that representation geometry provides a useful starting point, while similarities from task-aware representations can produce more compact evaluation sets and stronger agreement on tasks where that geometry alone is insufficient.

At the selected budgets, CLOSER provides $6.3\times$–$84.2\times$ faster recurring evaluation of complete pipeline suites than full-pool evaluation (Figure 1(b)). These timings exclude one-time construction and budget selection; Appendix C reports the measurement protocol and setup costs.

Comparable 10% budgets. Holding the evaluation fraction approximately fixed helps assess selector quality independently of the task-specific budget returned by the automatic rule. CLOSER achieves higher ranking agreement than uniform sampling on all four tasks, showing that its advantage persists under comparable evaluation budgets.

Beyond rank agreement, CLOSER's subset and full-location pipeline scores exhibit strong linear association comparing against random baselines, with Pearson correlations of ~ 0.9 (Appendix A.1). This suggest that estimating full-location scores is a promising direction for future work.

We also evaluate CLOSER under a evaluation-only forecasting setting; the complete protocol and curve are reported in Appendix A.3. We demonstrate a consistent improvement over the random baseline with different budgets.

6 ABLATION STUDIES

6.1 AUTOMATIC BUDGET SELECTION

We evaluate CLOSER's automatic budget selection in two stages. First, we ask whether scouting retains useful budgets. Figure 3(a) compares all candidate budgets with the ten-budget shortlist at

the same one-repeat scouting precision. The shortlist has higher median ranking agreement on all four tasks, with the largest increases on POI spend and COVID-19.

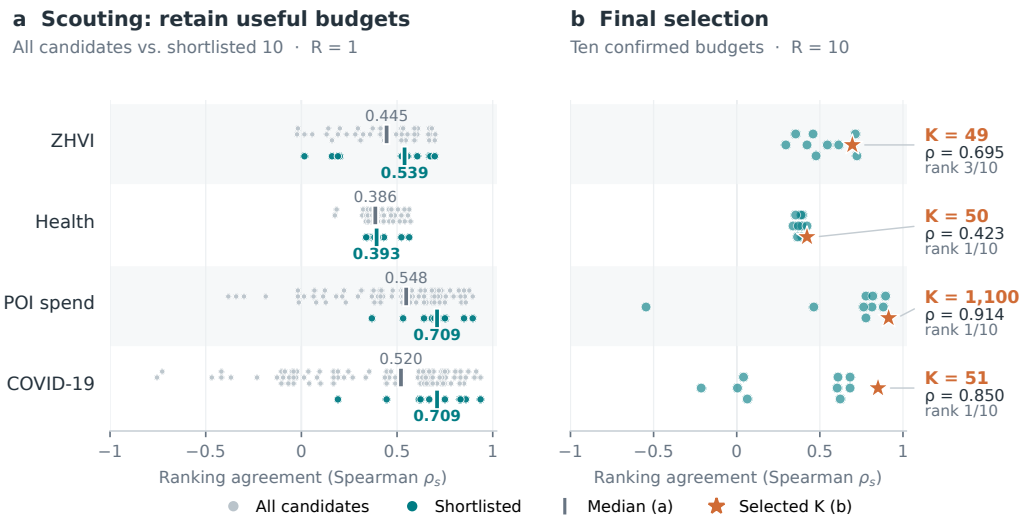


Figure 3: CLOSER's two-stage automatic budget selection. (a) Subset–full ranking agreement for all candidate budgets (gray) and the ten-budget shortlist (teal), both at one-repeat scouting precision. Ticks and labels show medians. (b) Ten-repeat confirmation results for the shortlisted budgets. Orange stars mark the returned budgets; annotations report K^*, agreement, and rank among the ten confirmed candidates. Each dot represents one budget; within each group, vertical offsets only separate overlapping dots. Scores use each task's reported rank-agreement aggregate.

Second, we ask whether the final rule identifies a strong budget within the confirmed shortlist. Figure 3(b) shows the ten-repeat confirmation results and marks the returned budget. By task-level rank agreement, the returned point ranks first on POI spend and COVID-19, joint first on Health, and third on ZHVI. Thus, the budget rule selects a point near the top of the confirmed candidates without using their full-location rankings. These diagnostics use the same tasks and pipeline suites as budget selection. The search grids and policy hyperparameters are specified in Appendix B.5.

6.2 Representation and similarity

BenTo uses in-context transferability to derive similarities for benchmark-task selection (Zhao et al., 2025). Our symmetric location similarities can likewise be viewed as undirected proxies for transferability. This motivates two questions: do similarities that align better with transferability preserve pipeline rankings better, and does introducing directionality improve selection?

We compare PDFM embeddings and probe hidden states using cosine, negative Euclidean distance, and asymmetric cosine (Table 7). The first two are symmetric; the third can score $i \to j$ differently from $j \to i$. We assess alignment against the label-based transferability diagnostic T^{gt}, which measures how well a predictor derived from one location reproduces another's observed outcomes (Appendix B.1). Table 2 reports all six combinations.

Across the six configurations, transferability alignment and downstream ranking agreement are strongly correlated (Spearman $\rho_s = 0.89$). Hidden-state cosine is strongest on both measures, while Asymmetric cosine has lower alignment and ranking agreement than cosine for both representations. These results support CLOSER's use of cosine similarity in task-aware hidden representations to preserve pipeline rankings.

6.3 Coverage weighting

Coverage weights encode how much of the target pool is assigned to each selected representative. To isolate their effect, we hold the selected locations, operating budget, candidate pipelines, and

Table 2: Transferability alignment and ranking preservation on ZHVI. Pairwise ρ_s measures agreement between representation similarities and the label-based profile-R^2 matrix T^{gt}; Selected Int-S measures interpolation ranking agreement at $K = 30$ using coverage weights.

Representation	Similarity	Pairwise ρ_s with T^{gt}	Selected Int-S
PDFM embedding	asymmetric cosine	-0.07	0.417
PDFM embedding	cosine	0.13	0.588
PDFM embedding	negative Euclidean	0.11	0.412
Probe hidden state	asymmetric cosine	-0.21	0.281
Probe hidden state	cosine	**0.45**	**0.728**
Probe hidden state	negative Euclidean	0.14	0.564

evaluation splits fixed, and replace the assignment-derived weights with $w_i = 1/K$. As shown in Table 3, coverage weighting improves ranking agreement on most datasets.

Table 3: Effect of coverage weighting at task-specific operating budgets. Selected locations, pipelines, and evaluation splits are held fixed. Entries use each task's reported rank-agreement aggregate, with repeat counts and aggregation specified in Appendix B.3.

Weighting	ZHVI $K = 49$	Health $K = 50$	POI spend $K = 1100$	COVID-19 $K = 51$
Uniform	0.605	**0.456**	0.900	0.809
Coverage	**0.695**	**0.456**	**0.914**	**0.850**

6.4 Properties of the selected locations

We analyze the composition and spatial coverage of CLOSER's selected locations across all four tasks. We observe task-specific differences from uniform random subsets, and the results suggest a complementary role for coverage weighting. For example, on POI, weighting brings the shares of low-density, lower-spending, and more temporally variable locations closer to full-pool proportions. Appendix A.2 provides the detailed comparisons and a visualization of the POI example.

7 Limitations

Our evaluation focuses on preserving pipeline rankings rather than estimating absolute full-set scores. The encouraging Pearson associations between subset and full-set scores suggest that score estimation may be a useful future direction, but CLOSER should not be interpreted as providing calibrated full-population performance. The evaluated pipeline suites also cover a limited range of method families; future work should include more diverse architectures and training procedures. Finally, CLOSER is intended for rapid iteration and screening rather than as a complete substitute for the full set: after developing or shortlisting pipelines on the subset, the selected method should still be evaluated on the full location set before drawing final conclusions.

8 Conclusion

We introduced CLOSER, which combines task-aware representations, facility-location selection, and automatic budget selection to construct compact, reusable sets of characteristic locations. Across four geospatial prediction tasks, CLOSER achieves higher agreement with full-location pipeline rankings than matched-size uniform sampling, while retaining less than 2% of locations on most tasks. After one-time setup, recurring pipeline evaluation is $6.3\times$–$84.2\times$ faster than full-pool evaluation, demonstrating CLOSER's effectiveness for efficient geospatial model evaluation.

AI USE DISCLOSURE

In this work, we used generative AI tools for designing or providing feedback on research methodology and experiments, implementing methods, and interpreting results. This assistance included clarifying the methodological framework and checking its mathematical formulation. We have not used generative AI tools to propose or refine research hypotheses, assist with translation, or clean and reformat datasets, and generating synthetic datasets, developing new theoretical models, providing critical ingredients for formal mathematical proofs, writing such proofs, and conducting qualitative or thematic data analysis are not applicable to this work. Additionally, we used generative AI tools to create or modify scientific figures, summarize and analyze existing literature, search for information and identify relevant literature, draft and edit manuscript text to improve readability, and propose titles or keywords. We have reviewed all AI-assisted work. Specifically, we checked references against the original sources, tested AI-assisted code, and verified reported results against experimental outputs. We take responsibility for the final content of this work, including text, claims, or artifacts produced with the aid of generative AI.

REFERENCES

Mohit Agarwal, Mimi Sun, Chaitanya Kamath, Arbaaz Muslim, Prithul Sarker, Joydeep Paul, Hector Yee, Marcin Sieniek, Kim Jablonski, Swapnil Vispute, Atul Kumar, Yael Mayer, David Fork, Sheila de Guia, Jamie McPike, Adam Boulanger, Tomer Shekel, David Schottlander, Yao Xiao, Manjit Chakravarthy Manukonda, Yun Liu, Neslihan Bulut, Sami Abu-el-haija, Bryan Perozzi, Monica Bharel, Von Nguyen, Luke Barrington, Niv Efron, Yossi Matias, Greg Corrado, Krish Eswaran, Shruthi Prabhakara, Shravya Shetty, and Gautam Prasad. General geospatial inference with a population dynamics foundation model. *arXiv preprint arXiv:2411.07207*, 2024. URL https://arxiv.org/abs/2411.07207.

Abhimanyu Das, Weihao Kong, Rajat Sen, and Yichen Zhou. A decoder-only foundation model for time-series forecasting. In *Proceedings of the 41st International Conference on Machine Learning (ICML)*, volume 235 of *Proceedings of Machine Learning Research*, pp. 10148–10167, 2024. URL https://proceedings.mlr.press/v235/das24c.html. arXiv:2310.10688.

Aditya Grover and Jure Leskovec. node2vec: Scalable feature learning for networks. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 855–864, 2016. doi: 10.1145/2939672.2939754. URL https://doi.org/10.1145/2939672.2939754.

William L. Hamilton, Rex Ying, and Jure Leskovec. Inductive representation learning on large graphs. In *Advances in Neural Information Processing Systems*, volume 30, pp. 1024–1034, 2017. URL https://proceedings.neurips.cc/paper_files/paper/2017/hash/5dd9db5e033da9c6fb5ba83c7a7ebea9-Abstract.html.

Yuhao Kang, Song Gao, Yunlei Liang, Mingxiao Li, Jinmeng Rao, and Jake Kruse. Multiscale dynamic human mobility flow dataset in the U.S. during the COVID-19 epidemic. *Scientific Data*, 7(1):390, 2020. doi: 10.1038/s41597-020-00734-5. URL https://www.nature.com/articles/s41597-020-00734-5.

Krishnateja Killamsetty, Durga Sivasubramanian, Ganesh Ramakrishnan, and Rishabh Iyer. GLISTER: Generalization based data subset selection for efficient and robust learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pp. 8110–8118, 2021. doi: 10.1609/aaai.v35i9.16988. URL https://ojs.aaai.org/index.php/AAAI/article/view/16988.

Konstantin Klemmer, Esther Rolf, Caleb Robinson, Lester Mackey, and Marc Rußwurm. SatCLIP: Global, general-purpose location embeddings with satellite imagery. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 4347–4355, 2025. doi: 10.1609/aaai.v39i4.32457. URL https://ojs.aaai.org/index.php/AAAI/article/view/32457. arXiv:2311.17179.

Yunlei Liang, Jiawei Zhu, Wen Ye, and Song Gao. Region2Vec: Community detection on spatial networks using graph embedding with node attributes and spatial interactions. In *Proceedings of the 30th International Conference on Advances in Geographic Information Systems (SIGSPATIAL)*, pp. 1–4, 2022. URL https://doi.org/10.1145/3557915.3560974. arXiv:2210.08041.

Felipe Maia Polo, Lucas Weber, Leshem Choshen, Yuekai Sun, Gongjun Xu, and Mikhail Yurochkin. tinyBenchmarks: Evaluating LLMs with fewer examples. In *Proceedings of the 41st International Conference on Machine Learning (ICML)*, volume 235 of *Proceedings of Machine Learning Research*, pp. 34303–34326, 2024. URL https://proceedings.mlr.press/v235/maia-polo24a.html. arXiv:2402.14992.

Baharan Mirzasoleiman, Jeff Bilmes, and Jure Leskovec. Coresets for data-efficient training of machine learning models. In *Proceedings of the 37th International Conference on Machine Learning (ICML)*, volume 119 of *Proceedings of Machine Learning Research*, pp. 6950–6960, 2020. URL https://proceedings.mlr.press/v119/mirzasoleiman20a.html.

George L. Nemhauser, Laurence A. Wolsey, and Marshall L. Fisher. An analysis of approximations for maximizing submodular set functions—I. *Mathematical Programming*, 14(1):265–294, 1978. URL https://doi.org/10.1007/BF01588971.

Yotam Perlitz, Elron Bandel, Ariel Gera, Ofir Arviv, Liat Ein-Dor, Eyal Shnarch, Noam Slonim, Michal Shmueli-Scheuer, and Leshem Choshen. Efficient benchmarking (of language models). In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pp. 2519–2536, 2024. doi: 10.18653/v1/2024.naacl-long.139. URL https://aclanthology.org/2024.naacl-long.139/. arXiv:2308.11696.

SafeGraph. Spend patterns. Dewey Data, 2022. URL https://www.deweydata.io/doi/10-82551-nsf5-r186. Dataset.

SafeGraph. Spend, 2026. URL https://docs.safegraph.com/docs/spend. Accessed: June 16, 2026.

Gayathri Saranathan, Cong Xu, Mahammad Parwez Alam, Tarun Kumar, Martin Foltin, Soon Yee Wong, and Suparna Bhattacharya. SubLIME: Subset selection via rank correlation prediction for data-efficient LLM evaluation. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 30572–30593. Association for Computational Linguistics, 2025. doi: 10.18653/v1/2025.acl-long.1477. URL https://aclanthology.org/2025.acl-long.1477/.

Don L. Stevens, Jr. and Anthony R. Olsen. Spatially balanced sampling of natural resources. *Journal of the American Statistical Association*, 99(465):262–278, 2004. URL https://doi.org/10.1198/016214504000000250.

The New York Times. Coronavirus (COVID-19) data in the United States, 2021. URL https://github.com/nytimes/covid-19-data. Accessed: June 16, 2026.

U.S. Census Bureau. American community survey 5-year data (2009–2024), 2026. URL https://www.census.gov/data/developers/data-sets/acs-5year.html. Accessed: September 24, 2026.

Rajan Vivek, Kawin Ethayarajh, Diyi Yang, and Douwe Kiela. Anchor points: Benchmarking models with much fewer examples. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1576–1601, 2024. doi: 10.18653/v1/2024.eacl-long.95. URL https://aclanthology.org/2024.eacl-long.95/.

Jihan Yao, Gantavya Bhatt, Arnav Das, Peter Jin, Ke Bao, Qiaolin Yu, Khushi Bhardwaj, Chang Su, Jialei Wang, Yikai Zhu, Sugam Devare, Damon Mosk-Aoyama, Zhen Dong, Venkat Krishna Srinivasan, Yineng Zhang, Oleksii Kuchaiev, Jiantao Jiao, Banghua Zhu, and Jeff Bilmes. Coresets before score sets: Evaluation-unsupervised prompt subset selection for LLM benchmarks. *arXiv preprint arXiv:2607.09739*, 2026. URL https://arxiv.org/abs/2607.09739.

Ling Zhan, Zhen Li, Junjie Huang, and Tao Jia. Accelerating benchmarking of functional connectivity modeling via structure-aware core-set selection. In *International Conference on Learning Representations (ICLR)*, 2026. URL https://arxiv.org/abs/2602.05667.

Hongyu Zhao, Ming Li, Lichao Sun, and Tianyi Zhou. BenTo: Benchmark task reduction with in-context transferability. In *International Conference on Learning Representations (ICLR)*, 2025. URL https://arxiv.org/abs/2410.13804.

Zillow Research. Housing data, 2026. URL https://www.zillow.com/research/data/. Accessed: June 16, 2026.

A Additional Results

A.1 Linear association between subset and full-location scores

Table 4 complements the ranking results in Table 1 with Pearson correlations between the eleven pipelines' subset and full-location score vectors. We use the same selected sets, evaluation repeats, and full-reference scores. Within each ten-repeat block, we average each pipeline's predictive scores using held-out test mass and compute Pearson correlation separately against each full-reference regime; regime, block, and random-set averaging follow Appendix B.3. At its automatically selected budgets, CLOSER achieves correlations of 0.968, 0.960, 0.935, and 0.880 on ZHVI, Health, POI spend, and COVID-19, respectively. Pearson measures linear association rather than agreement in absolute score values, so these results motivate future score calibration rather than establish it.

Table 4: Pearson association with full-location pipeline scores for the methods and budgets in Table 1. Each entry reports the task-level Pearson aggregate; parentheses give the number of evaluated locations. Repeat counts and aggregation follow Table 9 and Appendix B.3. Bold marks the highest value in each budget block and task.

Method	ZHVI	Health	POI spend	COVID-19
Uniform random, ∼10% budget	0.881 (1100)	0.968 (3000)	**0.971** (1100)	0.560 (297)
CLOSER, ∼10% budget	**0.922** (1100)	**0.970** (3000)	0.935 (1100)	**0.633** (297)
Uniform random, matched budget	0.856 (49)	0.943 (50)	0.971 (1100)	0.675 (51)
Geographic k-medoids, matched budget	0.207 (49)	**0.974** (50)	0.919 (1100)	0.056 (51)
Representation k-medoids, matched budget	0.957 (49)	0.927 (50)	**0.979** (1100)	0.563 (51)
Embedding-based facility location	**0.984** (77)	0.962 (125)	0.954 (1050)	0.793 (700)
CLOSER	0.968 (49)	0.960 (50)	0.935 (1100)	**0.880** (51)

A.2 Selected-location composition and spatial coverage

We examine two properties of the sets selected by CLOSER at the four reported operating budgets: the share of low-density locations and their geographic coverage. Table 5 compares each set with uniform random subsets of the same size.

Panel (a) separates *which locations are selected* from *how much evaluation weight they receive*. The Eligible column describes the locations available for selection ($\mathcal{A}$), while Coverage describes the pool that the selection represents ($\mathcal{B}$). Both count each location equally. Selected is the fraction of selected locations with fewer than 100 residents per square mile; Weighted is the fraction of total coverage weight assigned to those locations. The latter uses the reported weights, including their floor. Panel (b) measures geographic coverage: the fraction of the coverage pool within 50 great-circle miles of its nearest selected location.

For POI, low-density ZIPs account for 61.7% of selected locations but 53.6% of their coverage weight, bringing the weighted share closer to the pool's 41.4%. Weighting also narrows this gap for Health and COVID-19, whereas it widens the gap for ZHVI relative to its coverage pool. Geographic coverage follows a different pattern: ZHVI and POI fall below the random range, Health above it, and COVID-19 within it. Together, these comparisons characterize the task-specific composition and spatial reach of CLOSER's compact evaluation sets.

Figure 4 illustrates the POI set ($K^* = 1100$): it contains larger shares of low-density, lower-spending, and more temporally variable ZIPs than the full pool, and coverage weighting brings all three proportions closer to their pool values.

Calculation details. For each task, we draw 1,000 uniform size-K subsets without replacement from $\mathcal{A}$ (seed 20260913) and use the same subsets in both panels. The ranges are empirical 2.5th–97.5th percentiles across those draws, rather than confidence intervals for the characteristic set. Density is resident population divided by reported area in the CDC/ATSDR 2022 SVI data. Loca-

Table 5: Composition and geographic coverage at the reported operating points. All values are percentages. (a) Low-density shares by location count (Eligible, Coverage, Selected) or coverage weight (Weighted). The random range applies to the Selected column. (b) Coverage within 50 miles. Random ranges show the central 95% of 1,000 uniform subsets of size K.

(a) Low-density composition (%)

Task (K)	Eligible	Coverage	Selected	Weighted	Random 95% range
ZHVI (49)	25.3	25.3	18.4	12.0	[12.2, 38.8]
Health (50)	25.3	53.3	44.0	56.7	[14.0, 38.0]
POI (1100)	41.4	41.4	61.7	53.6	[38.5, 44.1]
COVID-19 (51)	71.1	71.1	76.5	72.2	[58.8, 84.3]

(b) Geographic coverage within 50 miles (%)

Task (K)	Selected set	Random mean	Random 95% range
ZHVI (49)	28.2	43.8	[37.7, 49.2]
Health (50)	28.0	24.6	[21.7, 27.4]
POI (1100)	89.0	93.9	[92.9, 94.7]
COVID-19 (51)	16.9	18.3	[15.8, 20.5]

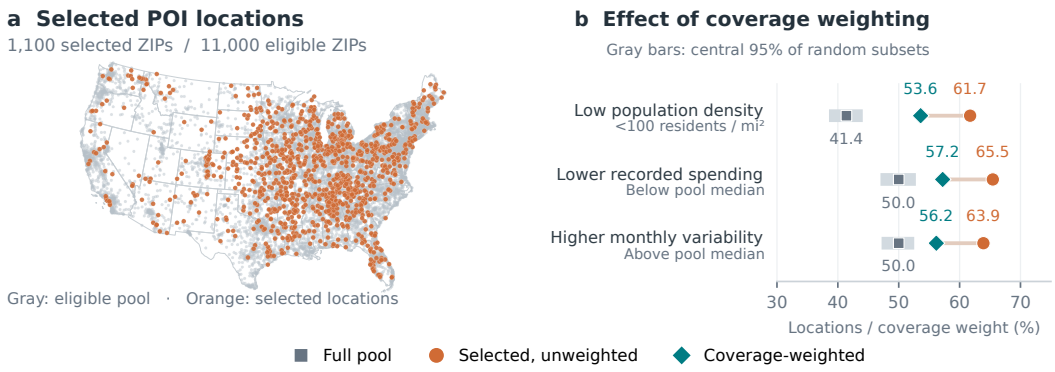


Figure 4: Composition of the POI set selected by CLOSER ($K^* = 1100$). Left: eligible and selected ZIP centroids. Right: unweighted pool and subset proportions versus coverage-weighted proportions. Spending and monthly coefficient-of-variation thresholds are pool medians. Gray bars show the central 95% of 1,000 same-size uniform random subsets.

tions with missing density are excluded from density calculations: all selected locations have density values, and only eight COVID-19 coverage-pool counties lack them.

For the POI analysis in Figure 4, total spending covers January–October 2019, the ten months available for every eligible ZIP. Monthly variability is the population standard deviation of those ten monthly totals divided by their mean. The spending and variability categories use the corresponding eligible-pool medians as thresholds. The map shows ZIP centroids for all 11,000 eligible and 1,100 selected ZIPs in the contiguous US, with Natural Earth state boundaries.

A.3 Forecasting extension

We compare eighteen forecasting pipelines on $N = 11{,}511$ ZIPs over July 2024–June 2025. At the July 2024 origin, CLOSER's probe uses only March–June 2024 ZHVI supervision. Its hidden states, cosine similarity, and the facility-location objective in Section 3.2 produce one greedy sequence. Each budget uses its corresponding prefix, with coverage weights recomputed using floor 0.01:

$$\mathcal{K} = \{4, 6, 9, 13, 20, 30, 33, 50, 75, 100, 150, 230, 345, 458\}.$$

The budget grid is fixed before evaluation to measure how forecasting ranking agreement changes with subset size. This post-selection sensitivity analysis complements the automatic-budget experiments in Section 6.1.

Pipeline suite. We use the 200-million-parameter TimesFM 1.0 model (Das et al., 2024) and July 2014–June 2024 history for all pipelines. The suite contains TimesFM alone, two PDFM-based LightGBM corrections trained on the final three history months (predicting either the label or the log label-to-forecast ratio), and the fifteen sequence refinements in Table 6. The correction models combine the TimesFM forecast with 330 PDFM features and use 400 estimators, learning rate 0.05, and feature fraction 0.8.

Table 6: The fifteen sequence refinements cross each of these five configurations with LSTM, tanh RNN, and Transformer encoders. ACS alignment uses forward filling from calendar-year or midyear anchors. PDFM-enabled variants use 330 features.

Configuration	ACS treatment	PDFM	ACS alignment
S1	Next-step auxiliary prediction	Yes	Calendar year
S2	Next-step auxiliary prediction	No	Calendar year
S3	TimesFM ACS forecast	Yes	Calendar year
S4	Next-step auxiliary prediction	Yes	Midyear
S5	TimesFM ACS forecast	Yes	Midyear

Sequence refinements predict log label-to-forecast ratios using a fixed 12-month training block, two encoder layers of width 128, dropout 0.1, and Adam with learning rate 10^{-3}, batch size 256, 20 epochs, gradient-norm clipping at 1, and seed 42. Transformer encoders use four attention heads. No validation split or randomized training-block length is used. ACS inputs use vintages 2011–2022 and per-table PCA with a 95% variance threshold capped at eight components; auxiliary-loss weight is 1 where applicable. Labels and absolute TimesFM forecasts are floored at 10^{-6} before taking logarithms.

Scoring and results. For pipeline m, prediction $\widehat{y}_{i,t}^{(m)}$, and target $y_{i,t}$, the per-ZIP WMAPE over the $H = 12$ holdout months and the full/subset scores are

$$\ell_i(m) = \frac{\sum_{t=1}^{H} |\widehat{y}_{i,t}^{(m)} - y_{i,t}|}{\sum_{t=1}^{H} |y_{i,t}|}.$$

$$q^{\text{full}}(m) = -\frac{1}{N} \sum_{i \in \mathcal{S}} \ell_i(m), \qquad q^{\text{sub}}(m; P, w) = -\sum_{i \in P} w_i \ell_i(m).$$

The negative sign preserves the larger-is-better convention. Spearman correlation between the eighteen-element score vectors measures ranking agreement. At each budget, the reference comprises $10{,}000$ uniformly sampled sets of the same size, using master seed zero and uniform aggregation weights.

Across all fourteen budgets, CLOSER achieves higher Spearman agreement than the matched-uniform mean, with the clearest separation in the low-cost regime (Figure 5). For example, $K = 50$ evaluates less than 0.5% of the $11{,}511$ ZIPs and reaches $\rho_s = 0.969$, compared with a random mean of 0.888. From $K = 33$ onward, each CLOSER subset also identifies the pipeline ranked first by full-location evaluation.

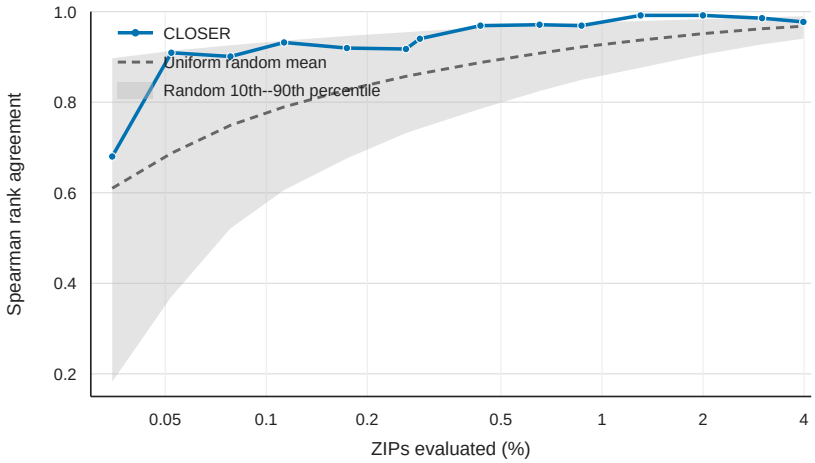


Figure 5: Post-selection forecasting sensitivity on the July 2024–June 2025 temporal holdout. The CLOSER curve reports the fourteen-point budget grid; the random curve and band show the mean and 10th–90th percentiles over $10{,}000$ uniformly sampled sets at each budget.

B Reproducibility Details

B.1 Probe representations and similarity

The selector starts from PDFM embeddings e_i and the probe targets in Table 10. For Health, each of the 21 outcomes is independently rank-Gaussian transformed: an observed value with average rank r_i among n values becomes $\Phi^{-1}((r_i - 1/2)/n)$. We fit a one-hidden-layer probe,

$$h_i = \mathrm{GELU}(W_1 e_i + b_1), \qquad q_\phi(e_i) = W_2 h_i + b_2,$$

by mean squared error over joined locations with complete probe targets, and use $z_i = h_i$ for selection. The hidden layer has 256 units. Training uses 50 epochs of Adam, learning rate 10^{-3}, batch size 256, no dropout or weight decay, and seed 0. Probe supervision uses the recorded outcomes, but not pipeline identities, pipeline scores, the full ranking, or the diagnostic T^{gt}.

Table 7 specifies the similarity functions. ZHVI matrices are dense; the large-pool cosine graphs use the neighbor counts and unstored-entry fill values in Table 8.

Table 7: Similarities used to construct $\mathbf{S}$. Asymmetric cosine normalizes by the target z_j and need not be symmetric.

Similarity	Definition
Cosine	$s_{\cos}(z_i, z_j) = z_i^\top z_j / (\|z_i\|_2 \|z_j\|_2)$
Negative Euclidean	$s_{\mathrm{euc}}(z_i, z_j) = -\|z_i - z_j\|_2$
Asymmetric cosine	$s_{\mathrm{acos}}(z_i, z_j) = z_i^\top z_j / \|z_j\|_2^2$

Transferability diagnostic. For the ZHVI diagnostic in Table 2, a ridge model fitted to each source ZIP's monthly house-value profile predicts the target ZIP's profile; $T^{\mathrm{gt}}_{i \to j}$ is its target-profile R^2. We use complete profiles over the 306 months from January 2000 through June 2025. The ridge model has penalty 1 and uses an unpenalized intercept, a standardized month index, and sine/cosine pairs with periods 12 and 6 months. Pairs with zero target variance are omitted. This label-based

Table 8: Sparse cosine settings for location selection. Unstored entries take the fill value before coverage conversion; the main Health matrix retains the largest 128 entries of each top-256 row, breaking ties by column index.

Task	Representation	Matrix size	Neighbors	Fill
Health	Probe (main)	$31{,}806^2$	128	-0.1881
Health	Raw-target probe ($\sim 10\%$)	$31{,}806^2$	256	0.6646
POI spend	Embedding	$11{,}000^2$	256	-0.0500
POI spend	Probe	$11{,}000^2$	256	0.3215
COVID-19	Embedding	$2{,}973^2$	256	0.0140
COVID-19	Probe	$2{,}973^2$	256	0.6502

score is used only to assess similarity alignment; its full monthly profile is distinct from the four probe targets. The corresponding $K = 30$ subset evaluations use predictive R^2 over eleven pipelines and 100 location splits with seeds 0–99, with rank agreement averaged across repeats.

B.2 Selection, weighting, and baselines

Selection uses the eligible and coverage pools in Appendix B.4. Before applying the coverage conversion in Equation (6), sparse unstored entries take their recorded fill value. Greedy selection breaks ties by stable location-identifier order; best-representative assignment breaks ties by the selected-location order. With this configuration fixed, all budgets use prefixes of the same greedy order, except for the zero-gain ablation below.

For Health's main rank-Gaussian configuration, greedy selection uses the sparse top-128 similarity matrix, while coverage weights use dense selected-to-pool similarities. For each selected set P_K, we compute all $K \times 31{,}806$ cosine similarities and assign each coverage location to its most similar representative. This avoids weight concentration caused by ties at the sparse fill value while retaining the selected locations. After flooring and renormalization, the $K = 50$ set has maximum weight 4.46% and effective size $1/\sum_{i \in P_K} w_i^2 = 39.87$.

Assignments and weights are recomputed at each budget. Coverage-weighted runs request a floor $\epsilon = 0.01$ and apply Equation (9) with

$$\epsilon_K = \min\{\epsilon, \mathrm{nextafter}(1/K, 0)\},$$

where $\mathrm{nextafter}(1/K, 0)$ is the largest representable floating-point value below $1/K$. Uniform-weight ablations use $w_i = 1/K$.

Zero-gain completion. For ZHVI probe hidden states with asymmetric cosine, ZIP code 90401 is the best representative for every target, leaving all remaining marginal gains zero. The $K = 30$ ablation retains this anchor and samples 29 of the other $11{,}510$ ZIPs without replacement, using selection seeds 0–29. The anchor's raw coverage mass is one and the other masses are zero; flooring gives normalized weights $1/1.29$ and $0.01/1.29$, respectively. Every completion uses the same 100 evaluation repeats over eleven pipelines. Averaging interpolation rank agreement over repeats and then completions gives 0.2815, with sample SD 0.2244 and SE 0.0410 across completion seeds.

Matched-budget baselines. Matched-budget baselines use the method's returned budget and the same repeat-seed sequence, with repeat counts specified in Table 9. Uniform sampling uses five sets with seeds 0–4 and weights $1/K$. Geographic and representation k-medoids use squared Euclidean distance on, respectively, the three-dimensional unit-sphere transform of latitude/longitude and raw regional representations. Bisecting k-means with one initialization and seed 0 supplies initial clusters; each center is replaced by its nearest selectable location. Nearest-medoid assignments and within-cluster medoid updates then alternate to a fixed point. Medoids are restricted to the eligible pool, while all coverage locations are assigned; cluster masses are floored using the same ϵ_K.

B.3 Prediction models and evaluation

Prediction models. Each pipeline is a feature bundle with a task-specific fitting procedure, rather than a distinct architecture. Subset evaluation compares ridge regression with penalty 1 against an

MLP with hidden widths 512, 256, and 128, ReLU activations, and dropout 0.2. For each response coordinate, mean validation R^2 in three-fold cross-validation selects the predictor, which is refitted on all selected training locations. Standardization is fitted only on the corresponding training data. The MLP uses 40 epochs of Adam, initial learning rate 0.005, cosine learning-rate decay, and batch size $\min(256, n_{\text{train}})$.

The full-location reference uses Ridge/MLP selection for ZHVI and COVID-19, an MLP for POI, and Ridge/MLP/LightGBM selection for Health. Health LightGBM candidates use 400, 1000, 2000, or 3000 trees, 31 leaves, minimum leaf size 40, and learning rate 0.02. The Health subset evaluator does not include LightGBM. Superresolution selects a county-trained predictor by within-county correlation on a separate ZIP validation set.

Full-reference splits and scores. ZHVI, POI, and COVID-19 reference scores use R^2, with county-heldout interpolation and state-heldout extrapolation. COVID-19 holds out 20% of usable counties or states, respectively, and averages reference scores over seeds 42, 43, and 44; three-fold predictor selection uses seed 42.

Health uses predefined interpolation and extrapolation partitions, with five and four training-validation folds, respectively. After common-support filtering and restricting to locations with population above 500, the training/validation and test sizes are $20{,}398/5{,}773$ for interpolation and $20{,}497/5{,}674$ for extrapolation. Superresolution trains on 3,081 counties, selects the predictor on a separate set of 4,080 ZIPs drawn from the interpolation-training pool, and evaluates on 5,773 test ZIPs. All eleven feature combinations use the same partitions.

Subset scoring. Each pipeline is trained and tested on repeated $80/20$ location splits of the selected set, sharing the repeat-seed sequence across feature bundles. Pipeline-specific feature joins can reduce support. Within each repeat, scores are averaged over response coordinates with valid predictions. POI and COVID-19 use coverage-weighted test R^2; ZHVI and Health use unweighted predictive Spearman correlation within each test fold, averaged across valid outcomes. For these latter two tasks, coverage weights enter the across-repeat score average through the held-out test mass, rather than weighting the within-fold Spearman calculation. The full Health interpolation/extrapolation references use R^2, while superresolution uses mean within-county Pearson correlation. Repeat counts and random-set protocols appear in Table 9.

Table 9: Subset evaluation repeats and random-location controls. Every repeat uses an 80/20 location split. The columns below describe the fixed-budget comparisons. At automatically selected budgets, CLOSER, matched baselines, and weighting ablations use ten repeats, except that Health CLOSER, its five random controls, and its same-selection weighting ablation use 100 repeats, aggregated as ten blocks of ten (A10×10). Every random control uses five independently sampled sets (selection seeds 0–4).

Task	Main predictive subset score	Fixed K	Fixed-budget repeats CLOSER / random
ZHVI	Spearman over held-out locations	1100	10 / 10
Health	Spearman, averaged over 21 outcomes	3000	10 / 10
POI spend	Coverage-weighted R^2	1100	10 / 10
COVID-19	Coverage-weighted R^2	297	100 / 100

At the main matched budgets, sample standard deviations across the five random sets are 0.058 (ZHVI), 0.033 (Health), 0.067 (POI), and 0.147 (COVID-19), using the aggregation defined below. The POI fixed-budget and automatic-budget rows reuse the same $K = 1100$ sets and ten-repeat evaluations.

Ranking metrics and aggregation. Spearman and Kendall correlations compare the pipeline-score vectors $\mathbf{v}^{\text{sub}}$ and $\mathbf{v}^{\text{full}}$; predictive Spearman inside a subset evaluator is a separate quantity. Pearson measures linear score association. Pipelines with invalid full or subset scores are excluded from both vectors.

For each fixed location set, we divide the evaluation repeats into consecutive, non-overlapping blocks of ten. Within each block, we first average each pipeline's predictive scores using held-out test mass, then compute rank agreement with each full-reference vector. A ten-repeat evaluation

contributes one block (A10). The Health $K = 50$ CLOSER result, its five matched random sets, both Health coverage-weighting ablation entries, and the COVID-19 fixed-budget comparison use 100 repeats: we average the ten block-level correlations (A10×10), rather than pooling all 100 predictive scores before ranking. The remaining main comparisons and weighting ablations use A10. All ten repeats within every block are retained.

Interpolation/imputation (Int/Imp) and extrapolation (Ext) form the predictive-generalization family. We macro-average these two regime correlations within each block. Health additionally has an independent scale-transfer/superresolution (SR) task, so we macro-average the predictive-generalization family with SR. For block b, this gives

$$\rho_b^{\text{Health}} = \tfrac{1}{4}\rho_{b,\text{Int}} + \tfrac{1}{4}\rho_{b,\text{Ext}} + \tfrac{1}{2}\rho_{b,\text{SR}}. \tag{12}$$

For ZHVI, POI spend, and COVID-19, the block aggregate is the equal mean of Int and Ext. We then average block aggregates. Each random set follows this procedure independently before computing the mean and sample SD across its five selection seeds; predictive scores from different selected sets are not pooled. Missing regimes are never replaced by zero.

Budget-selection diagnostics retain one split per scouting budget and ten splits per confirmed budget in Figure 3; Health confirmation agreement uses dense coverage weights. The representation/similarity comparison in Table 2 computes interpolation rank agreement separately in each of 100 repeats and then averages those correlations. It retains its repeat-level aggregation rather than pooling predictive scores.

B.4 Data, location pools, and candidate pipelines

Location pools. We distinguish the locations eligible for selection, the locations represented by coverage weights, and the locations used for full-reference evaluation. These pools coincide for ZHVI (11,511 ZIPs; house values), POI spend (11,000 ZIPs / POI regions; monthly total spend), and COVID-19 (2,973 counties; 7-day-mean cases). Reference counts are measured before task-specific population filtering and train/test partitioning. Feature joins may further reduce the usable locations for an individual pipeline.

POI targets are monthly sums of spending at establishments within each ZIP, aggregated from SafeGraph Spend Patterns obtained through Dewey (SafeGraph, 2022).

Health evaluates 21 outcomes at ZIP / region level. Its main selector chooses from 11,508 eligible locations drawn from the ZHVI pool and assigns coverage weights over 31,806 locations using the dense selected-to-pool similarities described in Appendix B.2. The full-reference pool contains 31,738 locations after excluding 68 without flow features. The separate fixed-budget comparison uses this 31,738-location pool for both selection and evaluation. It constructs top-256 cosine similarities from probe hidden states trained on untransformed targets; the main selector instead uses rank-Gaussian targets and top-128 cosine similarities for greedy selection. Probe supervision and downstream evaluation coordinates are specified in Table 10.

Feature sources and construction. We use five feature blocks. The mobility-derived blocks use the GeoDS COVID19USFlows dataset, which provides origin–destination flows derived from aggregated SafeGraph mobile-device visits (Kang et al., 2020). We aggregate flows over time at county level and convert tract-level flows to ZIP-level flows using a ZIP–tract crosswalk: each tract pair's flow is allocated using the product of its origin and destination allocation ratios, normalized within each tract.

- **PDFM.** We use the published 330-dimensional regional embeddings at ZIP/ZCTA or county resolution (Agarwal et al., 2024). These encode aggregated human-activity and environmental information. Geographic identifiers and metadata are used for joining locations, not as embedding coordinates.
- **ACS demographics.** American Community Survey five-year estimates (U.S. Census Bureau, 2026) supply nine educational-attainment features from table B15003 and 120 income-related features from table S1903. Available numeric estimates are averaged across releases to form static regional features; margins of error are excluded. The two tables are always included together, giving 129 demographic features.

Table 10: Probe supervision and downstream response coordinates for the main tasks. Probe labels define selector representations; downstream coordinates define predictive evaluation.

Task	Probe coordinates	Downstream coordinates
ZHVI	March–June 2025 month ends	40 quarter ends: March 31, June 30, September 30, and December 31 in each year 2015–2024
Health	All 21 outcomes below, rank-Gaussian transformed	The same 21 outcomes, scored separately
POI spend	July–October 2019	Ten months, January–October 2019
COVID-19	2021-09-30, 2021-12-31, 2022-03-31, 2022-05-13	Nine dates: 2020-06-30, 2020-09-30, 2020-12-31, 2021-03-31, 2021-06-30, 2021-09-30, 2021-12-31, 2022-03-31, 2022-05-13

The Health outcomes are percentages for high cholesterol, poor physical health, stroke, binge drinking, physical inactivity, annual checkups, cancer excluding skin cancer, diabetes, poor mental health, coronary heart disease, high blood pressure, cholesterol screening, dental visits, asthma, chronic kidney disease, arthritis, chronic obstructive pulmonary disease, blood-pressure medication among adults with high blood pressure, obesity, sleep shorter than seven hours, and smoking. The ZHVI label-based transferability diagnostic uses the full aligned 306-month profile.

- **Flow totals.** We sum population-adjusted origin–destination flows separately over incoming and outgoing trips. Each of these two features is divided by its mean across locations. These totals use the dataset's population-flow estimates, whereas the graph embeddings below use visitor flows.
- **GraphSAGE.** We train a heterogeneous GraphSAGE encoder (Hamilton et al., 2017) on county and ZIP nodes, with directed within-scale mobility edges and bidirectional county–ZIP membership edges. PDFM provides node features; an edge decoder predicts $\log(1 + \text{visitor flow})$ using squared-error loss. Both configurations use two mean-aggregation layers, dropout 0.3, and 1,000 epochs of Adam with learning rate 10^{-3}. ZHVI, Health, and POI use hidden width 256, output dimension 128, and flow edges with at least ten visitors. COVID-19 uses hidden width 660 and output dimension 330, retaining the top 2% of visitor-flow edges separately at each geographic scale; its node inputs also include log-transformed, normalized incoming and outgoing visitor-flow totals from the retained graph.
- **Node2Vec.** Node2Vec (Grover & Leskovec, 2016) learns 128-dimensional embeddings from random-walk neighborhoods on the combined county–ZIP graph, without node attributes. For ZHVI, Health, and POI, we retain flow edges with at least one visitor and use walk length 20, ten walks per node, context size ten, $p = q = 1$, and one negative sample. For these three tasks, training uses SparseAdam with learning rate 0.01, batch size 128, and 100 epochs. The COVID-19 embeddings use walk length 30 and 30 walks per node.

These feature blocks are fixed before fitting the downstream predictors; predictor-side standardization is specified in Appendix B.3.

Candidate pipelines. Each pipeline concatenates the blocks indicated in Table 11 and uses the prediction procedure in Appendix B.3. All four tasks use the same eleven combinations, each with valid full-reference and subset scores in the reported ranking comparisons.

Table 11: The eleven candidate feature combinations used on all four tasks. A checkmark denotes a concatenated feature block. ACS includes both education and income; flow totals include both incoming and outgoing population flows.

Pipeline	PDFM	GraphSAGE	Node2Vec	ACS	Flow totals
1	✓				
2		✓			
3			✓		
4	✓				✓
5	✓		✓		
6	✓			✓	
7	✓			✓	✓
8	✓		✓	✓	
9	✓		✓	✓	✓
10	✓	✓		✓	
11	✓	✓		✓	✓

B.5 Budget grids and selection policy

Budget selection uses all eleven candidate pipelines on each task, with that task's predictive metric. Before scouting, the fixed candidate-budget grid is capped by $Kd_t \leq B = 11{,}000$, where d_t counts the scored responses per location.

Table 12: Response-budget cap $Kd_t \leq B = 11{,}000$, where d_t counts the downstream coordinates in Table 10. Selection considers the task's fixed candidate-budget grid.

Task	d_t	$\lfloor B/d_t \rfloor$	Largest grid K	Selected K^*
ZHVI	40	275	270	49
Health	21	523	500	50
POI spend	10	1100	1100	1100
COVID-19	9	1222	1000	51

The policy uses $R_{\mathrm{s}} = 1$, $J = 10$, $R_{\mathrm{c}} = 10$, $\lambda = 0.3$, and $\alpha = 1/2$, with median aggregation as in Equation (10). Confirmation starts at seed zero, so its ten repeats include the scouting repeat. Equal confirmed scores receive the same normalized performance term; a singleton shortlist returns its sole budget. Objective ties favor smaller budgets.

λ in the predictive aggregation rule was selected from $\{0, 0.1, 0.2, 0.3, 0.4, 0.5\}$, using part of the results on ZHVI. Once this policy is fixed, selecting K^* uses only subset predictive scores. In Figure 3, panel (a) compares the full grid and shortlist at one-repeat precision, while panel (b) uses ten-repeat confirmation.

C Computational Cost

Table 13 reports recurring evaluation costs for complete pipeline suites using a common Ridge/MLP evaluator on the full uniform pool and the frozen coverage-weighted CLOSER subset. This controlled timing study isolates the effect of location reduction; the task-specific full-reference fitting protocols are described separately in Appendix B.3.

Every pipeline runs in a fresh process with seed 0 and ten $80/20$ splits, with eleven pipelines per task. This common timing workload is independent of the repeat counts used to estimate ranking agreement. Each workload is timed once; the splits are part of the workload, not independent timing repetitions. Up to four workers run concurrently, each with one RTX A6000 GPU and four disjoint CPU cores. Reported costs sum process elapsed times, including imports, CUDA initialization, loading, fitting, and writing outputs; they are not parallel-job wall times. Queue time and embedding pretraining are excluded, and filesystem caches are uncontrolled.

Original feature joins are preserved. ZHVI full-pool support ranges from 10,060 to 11,511 across pipelines, with 49 locations in every subset evaluation. POI supports range from 10,035 to 11,000 and from 1,052 to 1,100, respectively. Health and COVID-19 retain the declared counts.

Table 13: Measured recurring evaluation cost for the complete candidate suites. Full and CLOSER columns are summed process times in minutes, including loading and initialization. N is the declared full pool before pipeline-specific feature joins; K is the frozen subset size. Each pipeline is timed once with ten splits. Setup is excluded here and reported in Table 14.

Task	N	K	Pipelines	Full (min)	CLOSER (min)	Speedup
ZHVI	11,511	49	11	888.2	21.9	40.5×
Health	31,738	50	11	1356.7	16.1	84.2×
POI	11,000	1100	11	224.6	35.8	6.3×
COVID-19	2,973	51	11	63.2	7.0	9.0×

Table 14 separates the one-time costs. Construction trains the probe, builds the hidden-cosine geometry, and computes greedy prefixes and weights. Scouting evaluates every admissible budget with one split; confirmation evaluates the ten shortlisted budgets with ten splits before the fixed predictive-only budget rule.

Table 14: Measured one-time setup costs in minutes of summed worker-process time. Build includes probe training, geometry construction, greedy prefixes, and coverage weights. Scouting and confirmation evaluate the complete candidate pipeline suite. Total also includes automatic-K post-processing (less than three seconds per task).

Task	Build (min)	Scout (min)	Confirm (min)	Total (min)
ZHVI	4.7	109.9	173.8	288.4
Health	0.8	121.5	132.0	254.3
POI	0.6	142.9	212.6	356.1
COVID-19	0.4	105.6	93.4	199.5

For setup cost C, q complete suite passes cost $C + qT_{\text{subset}}$, compared with qT_{full}. Charging setup yields first-pass speedups of $2.9\times$ for ZHVI and $5.0\times$ for Health. COVID-19 is slower on the first pass and breaks even on the fourth. These reuse projections follow from the measured components; they are not additional repeated timings.