Data-Driven Subgroup Identification for Linear Regression

Zachary Izzo [1] **Ruishan Liu** [2] **James Zou** [2]

Abstract

Medical studies frequently require to extract the relationship between each covariate and the outcome with statistical confidence measures. To do this, simple parametric models are frequently used (e.g. coefficients of linear regression) but usually fitted on the whole dataset. However, it is common that the covariates may not have a uniform effect over the whole population and thus a unified simple model can miss the heterogeneous signal. For example, a linear model may be able to explain a subset of the data but fail on the rest due to the nonlinearity and heterogeneity in the data. In this paper, we propose DDGroup (data-driven group discovery), a data-driven method to effectively identify subgroups in the data with a uniform linear relationship between the features and the label. DDGroup outputs an interpretable region in which the linear model is expected to hold. It is simple to implement and computationally tractable for use. We show theoretically that, given a large enough sample, DDGroup recovers a region where a single linear model with low variance is well-specified (if one exists), and experiments on real-world medical datasets confirm that it can discover regions where a local linear model has improved performance. Our experiments also show that DDGroup can uncover subgroups with qualitatively different relationships which are missed by simply applying parametric approaches to the whole dataset.

1. Introduction

In scientific and medical analyses, simple parameteric models are frequently fit to data to draw qualitative or quantitative conclusions about the relationships between different variables of interest. Typically, a single interpretable model is fit on the entire dataset, implicitly assuming that there are uniform relationships between the covariates and target variable across the whole population. In practice, the data may instead come from a heterogeneous population, where different *subgroups* of the population may obey qualitatively different trends.

For example, suppose we fit a linear model with features including several patient biomarkers, as well as blood concentration of a particular drug, to predict blood pressure. After fitting the model to the whole dataset, we find that there is a statistically signficant negative coefficient on the drug concentration. We may be tempted to conclude that this drug should be administered to a general patient in order to reduce blood pressure. However, there may be a small subgroup in the data (say, patients over the age of 80) for whom the drug actually *increases* blood pressure. In this case, naively fitting a single model to the entire dataset not only reduces our predictive accuracy, it also leads to adverse outcomes for this subgroup of the population.

Modern high-capacity models such as neural networks can help to avoid this problem as they represent a much richer function class. However, these models are often inherently difficult to interpret, making them unsuitable if the primary goal is to draw scientific or clinical conclusions about the data rather than simply having good predictive performance. This motivates our desire to find interpretable regions in the data where interpretable models (such as linear regression) perform well. We call this the *subgroup selection* problem.

1.1. Our Contributions

In this work, we consider a flexible formalization of the subgroup selection problem. We propose an general algorithmic framework and a specific instantiation, DDGroup (data-driven group discovery), for data-driven subgroup selection. We prove that DDGroup has desirable theoretical properties, and results on synthetic and real data show the effectiveness of DDGroup in practice.

1.2. Related Work

Subgroup identification is an important topic in biostatistics (Lipkovich et al., 2017). Here, the main focus is on identifying subsets of the population with a significant beneficial

[1] Department of Mathematics, Stanford University, USA [2] Department of Biomedical Data Science, Stanford University, USA. Correspondence to: Zachary Izzo <zizzo@stanford.edu>.

Proceedings of the 40^{th} International Conference on Machine Learning, Honolulu, Hawaii, USA. PMLR 202, 2023. Copyright 2023 by the author(s).

treatment effect from a new drug or procedure. Common approaches include *global outcome modeling*, in which the user models the patient response with and without treatment separately, then reconstructs the treatment effect from these models; *global treatment modeling*, in which the user models the treatment effect directly; and *local modeling*, where the user tries to identify a region with a strong positive treatment effect. Of these approaches, our method is most closely related to the local modeling approach. However, existing local modeling methods typically use tree-based greedy approaches to region selection which do not come with any guarantees (Lipkovich et al., 2017).

In the knowledge discovery in databases (KDD) community, the problem of subgroup discovery has been studied more extensively; see (Atzmueller, 2015) and (Song et al., 2016) for surveys. In its most general form, subgroup discovery in this community refers to finding regions of the data with "interesting" properties, typically quantified by the use of a score function. For instance, a basic subgroup discovery method may try to find regions of the data where the mean or distribution of some target features are markedly different from the rest of the data. Later work has addressed more complicated tasks such as finding regions with exceptional regression models (Duivesteijn et al., 2012) or regions in which some pre-specified ML model works well (Sutton et al., 2020). Subgroups in this context are often specified by a *pattern*, which in the KDD literature refers to (usually pre-defined) selector variables. For instance, these could be some pre-defined thresholds on the features. Selection of the best subgroup with respect to the chosen score function then typically proceeds via either an exhaustive or greedy search over the valid patterns. The existing literature does not provide theoretical guarantees on the correctness of the selected subgroup. In contrast, we provide an efficient algorithm (not requiring exhaustive search) with provable guarantees and with data-driven (rather than pre-defined) selection criteria.

Our problem framework also has connections to list-decodable learning (Charikar et al., 2017), specifically list-decodable linear regression (Karmalkar et al., 2019; Raghavendra and Yau, 2020). In the list-decodable setting, we assume that an α fraction of the data come from a "trusted" source which we are trying to model; this would correspond to the subset of our data belonging to the region we are trying to detect. The goal is to output a small list (polynomial in α^{-1}) which contains a model that will perform well on the trusted data. While an algorithm for the list-decodable linear regression problem will return a model that performs well for the "good" region, it does not directly solve the problem of actually finding this region itself.

Piecewise linear regression is another method for adding flexibility to linear models while preserving interpretability. Here, the assumption is that the response is a piecewise linear function of the covariates. Early works focused on the one-dimensional covariate case (Vieth, 1989), and recently methods have been proposed for piecewise linear regression in higher dimensions (Siahkamari et al., 2020; Diakonikolas et al., 2020). Unlike the piecewise linear setting, we make no assumptions on the regression function outside of the "good" region which we are trying to detect.

Our work is also similar in spirit to previous works on conditional linear regression (Juba, 2017; Calderon et al., 2020). In this setting, the goal is also to find the largest possible subset of the data for which there is an accurate linear model. However, similar to many methods from the KDD literature, the subgroup identification in this case is made in terms of *pre-defined* binary features, which are assumed to be provided with the data in addition to the regressor variables. While one could instantiate our problem by defining the binary inclusion variables as indicators of whether or not each regressor is above or below a certain threshold, doing so would result in exponentially many possible selection rules and will therefore be computationally intractable for our setting. One can also view our work as finding data-driven binary inclusion labels for the conditional linear regression problem.

A core element of our problem setting is in selecting a region which avoids certain "bad" points. Related problems have been extensively studied in the computational geometry community (Dobkin et al., 1988; Backer and Keil, 2010; Dumitrescu and Jiang, 2013), but even approximate algorithms for solving related problems are not practical for high dimensions, and indeed even some seemingly simple region selection problems can be shown to be NP hard (Backurs et al., 2016). We propose tractable alternatives and show that they have desirable properties both theoretically and empirically.

As we seek to learn a subset of the data on which we are willing to make predictions, our work is connected to the literature on learning with rejection (Cortes et al., 2016) or learning to defer (Madras et al., 2018; Mozannar and Sontag, 2020; Keswani et al., 2021), in which a model is given the option not to make a prediction. These works focus primarily on classification and decide whether or not to make a prediction on individual data point via thresholding model confidence. While this implicitly defines a subgroup on which we expect the model to perform well—namely, the points for which the model does not defer—, this subgroup will typically be uninterpretable (if the model is a neural network). If logistic regression is used, the subgroup will be the complement of a slab between two parallel hyperplanes, which may be considered interpretable but is fairly inflexible in terms of the region selected. (Wiener and El-Yaniv, 2012) also considered a similar model in the regression setting, where the learner must simultaneously learn a regression function f and a selection function g which specifies the group of points on which to make predictions. Similar to learning with rejection, in this case, the subgroup is defined implicitly via g and in general will not be interpretable. In our setting, we focus on the regression problem and on explicitly defining an interpretable region in which we will not defer.

2. Problem Setup

The general subgroup selection problem can be formulated as follows. Let $\mathcal{Z} = \mathcal{X} \times \mathcal{Y}$ denote the sample space, $\mathcal{F} \subseteq \mathcal{Y}^{\mathcal{X}}$ denote a class of functions (e.g. linear regression models), and let $\ell : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}$ be a loss function measuring the performance of our model. We will always have $\mathcal{X} \subseteq \mathbb{R}^d$ and $\mathcal{Y} \subseteq \mathbb{R}$. Our goal is to find a (interpretable, large as possible) region $R \subseteq \mathcal{X}$ of the *feature space* where the loss

$$\operatorname*{argmin}_{f \in \mathcal{F}} \mathbb{E}[\ell(y, f(x)) \mid x \in R]$$

is small. In order to satisfy the interpretability criterion, we will consider regions R which are axis-aligned boxes. This corresponds to a subgroup where each feature lies within a specified range (corresponding to the sides of the axis-aligned box). The algorithm we develop also easily allows the user to control the tradeoff between the size of the region and the loss of the selected model within the region.

For this paper, we will specify the function class $\mathcal{F}$ to be linear models and the loss $\ell(y, \hat{y}) = (y - \hat{y})^2$ to be the squared loss. For our theoretical results, we will assume that there exists a "good" region $R^* \subseteq \mathcal{X}$ where the linear model is well-specified with low conditional variance of $y|x$. In particular, we will assume that when $x \in R^*$, we have $y|x \sim \mathcal{N}(x^\top \beta, \sigma^2)$ for some coefficients β. In this case, the goal will be to recover R^*.

3. Algorithmic Framework

We introduce an algorithmic framework with three distinct phases.

- **Phase 1:** Compute a rough approximation to the regression function in the good region.
- **Phase 2:** Using the approximate fit, define labels ℓ_i for each point in the training data, where $\ell_i \approx \mathbb{1}\{x_i \text{ could not reasonably belong to } R^*\}$.
- **Phase 3:** Find a large region which contains no rejected points.

In this work, we give specific implementations of each phase, but we note that this general framework is modular and can likely be modified to work in other settings (e.g. classification or survival analysis).

In Phase 1, we find a "core set" of points which should belong to the good region, then fit a model to these points. For Phase 2, we reject points by thresholding the residuals from the model found in Phase 1. For Phase 3, we remark that even if it is known which points should be included or excluded from the region, actually computing the largest region consistent with these points is NP hard, even if we restrict ourselves to axis-aligned boxes (Backurs et al., 2016).

Phase 1: We denote a dataset $D = (X, Y)$ to be a collection of n feature vectors (collected in $X \in \mathbb{R}^{n \times d}$) and corresponding labels (collected in $Y \in \mathbb{R}^n$). Here $\text{KNN}(x, k, D)$ denotes the k nearest neighbors of x (and their corresponding labels) in the dataset D, $\text{OLS}(D)$ denotes the output of ordinary least squares on feature matrix X and response vector Y, and $\text{MSE}(\hat{\beta}, D)$ denotes the mean squared error of linear model $\hat{\beta}$ on the data X, Y.

The pseudocode for selecting the core group is provided in Algorithm 1. Given a choice of core group size k, for each datapoint, we fit a local model to that point's k nearest neighbors. We then select the group of points with the lowest training error of its local model as the core group.

Algorithm 1 CoreGroup(k, D)

```
input Core group size $k$, dataset $D$
  $\text{MSE}^* \leftarrow \infty$
  for $(x, y) \in D$ do
    $D_{\text{nbhd}} = (X_{\text{nbhd}}, Y_{\text{nbhd}}) \leftarrow \text{KNN}(x, k, D)$
    $\hat{\beta} \leftarrow \text{OLS}(X_{\text{nbhd}}, Y_{\text{nbhd}})$
    if $\text{MSE}(\hat{\beta}, D_{\text{nbhd}}) < \text{MSE}^*$ then
      $D_{\text{core}} \leftarrow D_{\text{nbhd}}$
      $\text{MSE}^* \leftarrow \text{MSE}(\hat{\beta}, D_{\text{nbhd}})$
    end if
  end for
output $D_{\text{core}}$
```

Phase 2: For our theoretical results, we use the threshold

$$\rho^{\text{grow}}_{\sigma,n} = 2.1\sigma\sqrt{\log n}. \tag{1}$$

Here n is the size of the training set. The inclusion labels ℓ_i are then computed as $\ell_i = \mathbb{1}\{|y_i - \hat{\beta}^\top x_i| \geq \rho^{\text{grow}}_{\sigma,n}\}$. We define the set of *rejected points* $X_{\text{rej}} = \{x_i \in X \mid \ell_i = 1\}$. For our empirical results, the threshold will be considered as a hyperparameter and chosen using a validation set. For more detail, refer to Section 5.

Phase 3: Let $U \subseteq \mathbb{R}^d$. We define the *directed infinity norm* $\|x\|_{U,\infty}$ by

$$\|x\|_{U,\infty} = \max_{u \in U} x^\top u.$$

We note that for many sets U, $\|\cdot\|_{U,\infty}$ may not be a norm, nor even a seminorm. In what follows, U will initially be defined as $U = \{\pm e_i\}_{i=1}^d$, in which case $\|\cdot\|_{U,\infty} = \|\cdot\|_\infty$ coincides with the usual infinity norm on $\mathbb{R}^d$. We will then gradually remove directions which are no longer relevant to consider.

The region will be described in terms of linear constraints. We will overload notation and use a set $R = \{(u_i, a_i)\}_{i=1}^m$ of constraint directions and values to denote the region $R = \{x \in B \,:\, x^\top u_i \le a_i\}$.

The pseudocode for the growing box is provided in Algorithm 2. When $U = \{\pm e_i\}_{i=1}^d$, Algorithm 2 begins expanding an ℓ_∞ ball centered at $\bar{x}$ with each side growing at an equal rate. Whenever one of the sides runs into a rejected point, we add the corresponding linear constraint and continue growing the other sides of the box. (The directed infinity norm is what we use to measure which point will collide with the box next. For a discussion on the geometric intuition for this step, see Appendix A.) This continues until all sides of the box have a support point, or there are no points left to constrain the box.

Note that the set U simply specifies the normal vectors to the sides of the constraint polytope. The lengths of these vectors effectively determine the speed at which the constraint region will grow in that direction. By changing U, this method can select polytopes of any desired shape. Since axis-aligned boxes provide easily interpretable inclusion criteria, we use such regions for all of our experiments.

Algorithm 2 GROWBOX($\bar{x}, X_{\text{rej}}, U$)

```
input Starting point (center) $\bar{x}$, rejected points $X_{\text{rej}}$, normal
  vectors defining the shape of the selected region $U$
  $X_{\text{rej}} \leftarrow X_{\text{rej}} + \{-\bar{x}\}$ {Center the points at $\bar{x}$. + denotes
  Minkowski sum.}
  $\hat{R} \leftarrow \emptyset$
  while $X_{\text{rej}} \neq \emptyset$ do
    $x^* \leftarrow \text{argmin}_{x \in X_{\text{rej}}}\{\|x\|_{U,\infty}\}$
    $a^* \leftarrow \|x^*\|_{U,\infty}$
    $u^* \leftarrow \text{argmax}_{u \in U}\{u^\top x^*\}$ {$u^*$ is the next support
    direction for the polytope}
    Add $(u^*, a^*)$ to $\hat{R}$
    Remove $u^*$ from $U$
    $X_{\text{rej}} \leftarrow \{x \in X_{\text{rej}} \mid x^\top u^* < a^*\}$
  end while
output $\hat{R} + \{\bar{x}\}$ {Undo the centering procedure from the
  first part of the algorithm.}
```

Combining Phases 1-3 gives an algorithm for automatic subgroup selection. We summarize the entire pipeline in Algorithm 3. Note that if the variance σ^2 is not known, we can replace it with a standard unbiased estimate computed on the core group.

We also remark that after $\hat{R}$ has been selected, rather than using the coefficients $\hat{\beta}$ fit just to the core group, we can also choose to re-fit $\hat{\beta}$ on all of the training points contained in $\hat{R}$. We use this additional step in our experiments, but it does not affect our theoretical results.

Algorithm 3 DDSUBGROUP(k, U, D)

```
input Core group size $k$, normal vectors defining the shape
  of the selected region $U$, dataset $D$

  Phase 1: Find a core group and fit a coarse model.
  $D_{\text{core}} \leftarrow \text{CoreGroup}(k, D)$
  $\hat{\beta} \leftarrow \text{OLS}(D_{\text{core}})$

  Phase 2: Label which points should be excluded.
  for $i = 1, \ldots, n$ do
    $\ell_i \leftarrow \mathbb{1}\{|y_i - \hat{\beta}^\top x_i| \ge \rho_{\sigma,n}^{\text{grow}}\}$
  end for
  $X_{\text{rej}} \leftarrow \{x_i \in X \mid \ell_i = 1\}$

  Phase 3: Approximate $R^*$.
  $\bar{x} \leftarrow \text{Mean}(X_{\text{core}})$
  $\hat{R} \leftarrow \text{GrowBox}(\bar{x}, X_{\text{rej}}, U)$
output $\hat{R}$
```

Runtime The runtime of DDGroup as described by Algorithm 3 is $O(kn \log n)$. We treat the dimension as a constant. After constructing a K-D tree in $O(n \log n)$ time, the k-nearest neighbors of a point can be found in time $O(k \log n)$. Computing the OLS fit on k points in constant dimension takes $O(k)$ time, making the runtime for each step of the core group search $O(k \log n)$ since we do not need to recompute the K-D tree for each of these steps. This step is repeated n times, once for each candidate core group. The box expansion requires only $O(n)$ work once the core group has been determined, thus the overall runtime for the algorithm is $O(n \log n) + O(kn \log n) + O(n) = O(kn \log n)$. While this is only the cost of a single run of DDGroup, these runs can easily be parallelized, making DDGroup highly efficient even for large datasets and large hypeparameter searches.

4. Theoretical Guarantees

In this section, we examine some of the theoretical properties of DDGroup. All proofs are deferred to Appendix D. In what follows, “with high probability” means with probability approaching 1 as $n, k \to \infty$. We make the following assumptions.

1. The samples $(x_i, y_i) \overset{\text{iid}}{\sim} \mathcal{P}$ for a probability distribution $\mathcal{P}$ on $\mathcal{Z}$. We let $S = \text{supp}(x)$ denote the support of the marginal distribution of the features.

2. The features x are bounded: $\|x\| \leq B$ deterministically.
3. The marginal distribution of x has a density f with respect to the Lebesgue measure. Furthermore, the density is bounded from above and below on the support of x: $0 < c_f \leq f(x) \leq C_f < \infty$ for all $x \in S$.
4. There is a region $R^* \subseteq S$ in which the linear model holds. That is, conditional on $x \in R^*$, y is generated according to the linear model: $x \in R^* \Rightarrow y|x \sim \mathcal{N}(x^\top \beta^*, \sigma^2)$ for some fixed β^*.
5. The region R^* is an axis-aligned box with nonempty interior, i.e. $R^* = \prod_{i=1}^{d}[a_i, b_i]$ for some $a_i < b_i$.
6. Conditional on $x \notin R^*$, y is Gaussian with variance at least σ_0^2, where $\sigma_0 \geq C\sigma$ for some absolute constant C.

The lower bound in Assumption 3 ensures that the samples will cover the sample space (so that we can detect R^*). The upper bound prevents degeneracies, e.g., if the feature distribution contains atoms with large enough mass, the KNN of certain points may contain many copies of a single point. Assumption 4 ensures that our model is well-specified on R^*. Assumption 6 ensures that R^* is in fact the "best" region for us to select, namely, there is no other region where we can have better predictive power. This condition also ensures that the random fluctuations in y_i are large enough to be detected by the test. We remark that the absolute constant C is no greater than 50, but we have made no effort to optimize this constant in our analysis and it can certainly be reduced.

Our first result shows that almost all of the group selected by Algorithm 1 lies in R^*.

Lemma 4.1. *The core group selected by Algorithm 1 has $X_{\text{core}} \setminus R^* = o(k)$ with high probability.*

The next result states that we will not erroneously reject any points that actually belong to R^*.

Lemma 4.2. *Let X_{core} be the core group selected by Algorithm 1 and let $\hat{\beta}$ be the OLS estimator fit to X_{core}. Let X_{rej} be the set of rejected points defined by the thresholding procedure in Phase 2. With high probability, none of the points in X_{rej} belong to R^*.*

Combining Lemmas 4.1 and 4.2, we show that DDGroup precisely recovers R^* given sufficient data.

Theorem 4.3. *As $n \to \infty$, there exist positive scalars $\{s_j^{\pm}\}_{j=1}^d$ and a constant $c > 0$ such that if $U = \{s_j^+ e_j, -s_j^- e_j\}_{j=1}^d$ and $k = \Omega(n)$ with $k \leq cn$, Algorithm 3 returns $\hat{R}$ with $R^* \subseteq \hat{R}$ with high probability. Furthermore,* $\text{vol}(\hat{R} \setminus R^*) \to 0$.

We remark that the scalars $s_j^{\pm}$ can depend on the dataset and the constant c may depend on R^* and the other parameters in Assumptions 1-6.

As an immediate corollary to Theorem 4.3, we see that under slightly modified assumptions, DDGroup can be used to find multiple subgroups in the data by iteratively applying Algorithm 3.

Corollary 4.4. *Suppose that Assumptions 1-6 hold, but instead of a single region R^*, there are multiple disjoint regions R_g, $g = 1, \ldots, G$ where for $x \in R_g$, $y|x \sim \mathcal{N}(\beta_g^\top x, \sigma_g^2)$. Furthermore, assume that Assumption 6 holds with $\sigma_0 > C\sigma_g$ whenever the $x_i \notin \bigcup_{g=1}^G R_g$. Let $\hat{R}_g$, $g = 1, \ldots, G$ be the outputs after running Algorithm 3 G times, removing the training points which are contained in $\hat{R}_g$ after the g-th run. Then under the same conditions as in Theorem 4.3, we have $R_g \subseteq \hat{R}_g$ and* $\text{vol}(\hat{R}_g \setminus R_g) \to 0$ *for all g.*

5. Experiments

In this section, we evaluate the performance of DDGroup on both synthetic and real-world medical datasets.

Methods for Comparision We compare DDGroup with several other baselines.

1. Standard linear regression, i.e., a linear model fit to the whole dataset. It is equivalent to the situation where the selected region includes all of the data and it is the method employed by the original medical studies on the real-world datasets we consider.
2. An unsupervised clustering method. Here we use k-means clustering and identify the cluster with the smallest MSE as the most coherent subgroup. We use the bounding box defined by the selected subgroup as the interpretable inclusion criteria.
3. Linear model trees. These are decision trees with a linear regression model in each leaf (Wang and Witten, 1996; Potts and Sammut, 2005). Though LMT is not designed for subgroup identification, we can still use its decision path as a way to select cohorts. In order to identify the most coherent subgroup, we pick the leaf of the LMT with the smallest MSE.

Experiment Setup For the real-world datasets, we randomly split them into training, test and validation sets, with ratio 50%, 30% and 20%. In each experiment, we fit the models on the training set with a grid search over hyperparameters and select the region with lowest validation MSE. We then refit the linear model on the training points in the selected region and evaluate its performance on the test set. For DDGroup, we used a more general form of the threshold $\rho_{\gamma_1,\gamma_2}(x_i) = \sigma\gamma_1\|x_i\| + \sigma\gamma_2$ and tuned γ_1 and γ_2 as additional hyperparameters. Specifically, the algorithm works well by simply setting $\gamma_2 = 0$ and tuning $\gamma_1 \in \{2^{-4}, 2^{-3}, \ldots, 2^5\}$. We also set the size k of the core group equal to p times the size of the training set, where p was selected from within $\{0.01, 0.05, 0.1, 0.15, 0.2\}$. We also tried two different "speed" settings for Algorithm 2: the sides of the box either grow all at the same rate, or each side grows at a rate proportional to the length of the bounding box B in that dimension. For k-means clustering, the number of clusters is a critical parameter and is scanned from 2 to twice the dimension of the data for the best performance. For LMT, the tree depth is an important parameter and is scanned from 1 to the dimension of the data on the validation set for the best performance.

5.1. Demonstration on Synthetic Data

To visualize our method and test its performance in a well-specified setting, we construct a synthetic dataset where the desired region to be selected is known. Let $B \subseteq \mathbb{R}^d$ be the feature space, and let $R^* \subseteq B$ be the "true" region that we wish to recover. The data are generated as follows. We first sample the features $x \sim \mathrm{Unif}(B)$. If $x \in R^*$, set $y = \beta^\top x + \varepsilon_{\text{in}}$. Else if $x \notin R^*$, set $y = \varepsilon_{\text{out}}$. Here $\beta \neq 0 \in \mathbb{R}^d$ are the fixed true model weights for the region R^*. The error terms ε_{in} and ε_{out} follow $\varepsilon_{\text{in}} \sim \mathcal{N}(0, \sigma_{\text{in}}^2)$ and $\varepsilon_{\text{out}} \sim \mathcal{N}(0, \sigma_{\text{out}}^2)$ with $\sigma_{\text{in}} < \sigma_{\text{out}}$. We set the dimension $d = 3$ so that the selected region can be easily visualized. (The third dimension just allows us to incorporate a bias term, so we will only visualize two dimensions.) We define the bounding box for the features $B = [-1, 1]^2 \times \{1\}$ and the true region $R = [-1/3, 1/3] \times [-2/3, 2/3] \times \{1\}$, and we generate $n = 1000$ data points.

Figure 1a shows the results of running Algorithm 3 on this synthetic data. The gray shaded region is R^*. The red "x" (resp. blue "o") markers denote points that were rejected (resp. not rejected) by the threshold (1), and the green rectangle shows the boundary of $\hat{R}$ returned by DDGroup. There is a nearly perfect overlap between R^* and $\hat{R}$, meaning DDGroup is able to precisely recover the true region. In contrast, the green rectangles in Figure 1b and Figure 1c shows the region selected by k-means clustering and LMT. The k-means clustering method erroneously excludes points within the correct subgroup, while LMT tends to select points outside of R^*.

Figure 1d shows the robustness of DDGroup to a misspecified core group. We replace the output of Algorithm 1 with a manually supplied set of points. We start by providing a core group whose center coincides with that of R^*. The x-axis of the plot denotes the offset of this initial core group: at position x on the plot, the center of the core group has been shifted by (x, x). Because we grow the sides of $\hat{R}$ at the same speed, it becomes harder to recover the full R^* when the center of the core group is closer to the edge of R^* (larger x value on the plot). We plot three quantitites:

- Precision $= \mathrm{vol}(\hat{R} \cap R^*)/\,\mathrm{vol}(\hat{R})$,
- Recall $= \mathrm{vol}(\hat{R} \cap R^*)/\,\mathrm{vol}(R^*)$, and
- F1 score.

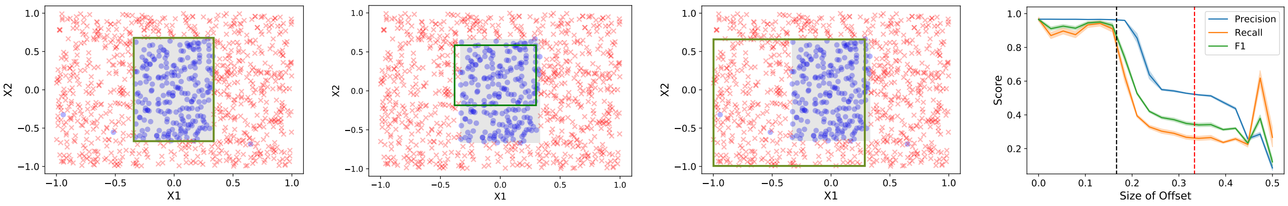
The vertical dashed black line denotes the point at which the core group starts to include points which do not belong to R^*. The vertical dashed red line denotes the point at which the center of the core group (and thus the base point from which we grow $\hat{R}$) lies outside of R^*. We see that DDGroup is quite robust to the location of the core group within R^*. However, once "bad" points are included in the core group, the performance (in particular the recall) begins to drop sharply. The precision is more robust to core group misspecification, remaining well above the baseline of 0.22 (which is equivalent to selecting the whole region) even when the core group is more than 50% misspecified.

Table 1 shows the performance of different methods (measured by F1 score) vs. sample size. DDGroup has a statistically significant performance improvement compared to the other methods for all sample sizes. LMT eventually identifies most of the correct region, but it is much less sample efficient than DDGroup. Increasing the sample size does not appear to help for the clustering method.

5.2. Evaluation on Real-World Datasets

We further evaluate our method on five real-world medical related datasets, where linear coefficients were used for interpretation in their original publications.

1. Brazil Health Dataset (Cavalcante et al., 2018) is from a longitudinal ecological study for 645 municipalities in the state of São Paulo, Brazil. The study uses a linear model to identify key features for hospitalization of heart failure (HF) and strokes.
2. China Glucose Dataset (Wang et al., 2017) consists of 5,726 female (F) and 5,457 male (M) Chinese individuals with normal glucose tolerance. The study uses linear model to describe the relationship between fasting plasma glucose and serum uric acid levels (SUA).
3. China HIV Dataset (Zhang et al., 2016) consists of 2,987 participants living with HIV from Guangxi province, China. The study uses linear regression to study how routes of HIV infection affect the HIV internalized stigma scale, adjusted by patients' characteristics.



(a) Region selected by DDGroup. (b) Region selected by clustering. (c) Region selected by LMT. (d) Robustness of DDGroup.

Figure 1. Demonstration on synthetic dataset. (a-c) The region selected by (a) DDGroup, (b) k-means clustering and (c) linear model tree. The grey shaded area denotes the correct subgroup and the green box corresponds to the learned boundary. For k-means clustering, the number of clusters is searched from 2 to 10, and the bounding box for the cluster with smallest MSE is reported in (b). The depth of LMT is searched from 1 to 10, and the best performance is reported in (c). (d) Robustness of DDGroup to core group misspecification. The shaded region shows standard error of the mean over 50 trials. The black dashed line denotes the point at which "bad" points are included in the core region. The red dashed line denotes the point at which the center of the supplied core set is outside of R. The y-axis records precision, recall, and F1 score (higher is better).

Table 1. Performance on single subgroup identification for k-means clustering, linear model tree, and DDGroup on synthetic datasets of varying sizes. We report the average F1 score plus or minus the standard error of the mean over 20 trials. DDGroup outperforms the comparison methods for all sample sizes and finds accurate results even with few samples.

n	200	400	800	1600	3200	6400	12800
DDGroup	**0.73 ± 0.03**	**0.68 ± 0.07**	**0.93 ± 0.02**	**0.98 ± 0.00**	**0.99 ± 0.00**	**0.99 ± 0.00**	**1.00 ± 0.00**
LMT	0.23 ± 0.09	0.32 ± 0.10	0.19 ± 0.09	0.28 ± 0.10	0.48 ± 0.11	0.92 ± 0.05	0.93 ± 0.05
Clustering	0.07 ± 0.04	0.18 ± 0.07	0.02 ± 0.01	0.12 ± 0.05	0.12 ± 0.06	0.14 ± 0.07	0.11 ± 0.06

4. Dutch Drinking dataset (Boelema et al., 2015) consists of the individual life survey data of alcohol use among 2,230 Dutch adolescents. The study uses linear regression to analyze how drinking affects adolescents' inhibition (inh), working memory (wm) and shift attention (sha).

5. Korea Grip Dataset (Wen et al., 2017) is for the Donggu study of 2,251 Korean adults with osteoarthritis (OA). The study uses linear regression to explore the associations between grip strength and individual radiographic feature scores of OA.

Performance Evaluation for a Single Group We first examine the case in which we try to find a single subgroup of the data. Table 2 shows the mean test MSE and fraction of test points included in the selected region (both ± the standard error of the mean) averaged over 10 random train/validation/test splits. DDGroup correctly identifies a subgroup on which the linear model has low test error and consistently outperforms the baseline methods on all five real-world medical datasets. Across all of the datasets, it most frequently has the lowest test MSE, and *never* has a test MSE which was statistically significantly worse than any other method. We demonstrate that there exist subgroups within the real-world population where a linear model is a good proxy and should be used to enhance interpretability. Our current method focuses on finding the most coherent region within the dataset, thus it always identifies small subgroups with the strongest signal. If a larger subgroup is desired, one may enforce this by selecting the best region which includes e.g. at least a certain fraction of the validation set. In our case, we required that at least 5% of validation was selected. We also remark that DDGroup is computationally efficient in practice. The average runtime for Algorithm 3 across one run of each dataset was 1.98 seconds on an AMD 7502 CPU, and no individual dataset took longer than 10 seconds.

Performance Evaluation for Multiple Groups Next, we examine the performance of the competing methods when selecting multiple subgroups in the data; in this case, we select three subgroups. We modified the DDGroup procedure according to Corollary 4.4. For the other methods, we performed a similar iterative procedure, repeatedly removing the selected subgroups from the training data. Table 3 shows the same statistics as reported in Table 2, but averaged across the three selected subgroups (as well as the random train/validation/test splits). When selecting multiple subgroups, DDGroup maintains its advantage over the other methods. As in the single group case, it usually has the lowest test MSE and is never statistically significantly worse than any other method.

Case Study Here we use the China HIV Dataset to illustrate how DDGroup can enhance our understanding of the data. The original study analyzes how different HIV infection routes affect the internalized stigma by fitting a multivariate linear regression model with confounders (Zhang et al., 2016). In their main results, the blood transfusion route is found to have positive effect on internalized stigma (coefficient β larger than zero), but in low confidence with a large p value. In our analysis, we observed similar behavior: after data standardization, the linear model fit to the whole dataset predicts blood transfusion route to have a positive effect on internalized stigma with β of 0.12, but low confidence level with a p value of 0.67. On the other hand, DDGroup identifies a subgroup of 21% of the participants where blood transfusion route has the opposite effect on stigma ($\beta = -1.71$) with a strong signal ($p = 0.006$). The selected subgroup consists of younger participants with lower self-esteem, lower anxiety level, and less social support. The result indicates that while blood transfusion route seems not to associate with internalized stigma in the general population living with HIV, it is coherently associated with lower stigma in a certain subpopulation. This seems plausible, as the other infection routes include sex with stable partners, sex with casual partners, sex with commercial partners, and injecting drug use. Younger participants may have stronger feelings of shame associated with these activities than older participants. In general, interpretation of the learned selection rules could be of great interest in real applications.

Before concluding the section, we remark that DDGroup offers flexibility in choosing between the size of the selected subgroups and the MSE of the linear model on these subgroups. In these experiments, we required that the selected subgroups contain at least 5% of the validation set in order to be considered. This threshold can easily be modified, and in general a higher threshold will encourage the selection of larger regions at the expense of a higher MSE. See Appendix B for more details and experiments regarding this tradeoff.

6. Discussion

In this paper, we considered a flexible formalization of the cohort selection problem. We proposed a general algorithmic framework and a specific instantiation, DDGroup, for solving the problem, and we proved that DDGroup recovers the correct subgroup given sufficient data. Experiments on both synthetic and real data verify our theory and show the practical usefulness of DDGroup.

6.1. Limitations & Future Work

While the assumption that there is a region in which the linear model holds exactly may seem strong at first glance, if the true regression function for the data is differentiable, then a linear model will always hold locally. Thus, if it is acceptable to select a small group, DDGroup can still succeed in nonlinear cases. However, if the true regression function is highly oscillatory, these locally linear regions may be very small, and a large amount of data will be required to find them. Another limitation may arise in situations where there is not a unique "best" region (or collection of best regions), i.e., when $\mathbb{V}(y|x)$ is roughly the same across the whole data space. In such cases, the regions discovered by DDGroup may be unstable across different random splits of the data, as there is not a strong reason for DDGroup to prefer one region of the data over another.

There are a number of important open questions which remain to be addressed. If a hyperparameter search is used with DDGroup to train a linear model (as we did with our real data experiments), further analysis is needed to give meaningful (but valid) p-values for the resulting model coefficients. For any extensive hyperparameter search, a naive Bonferroni correction is likely to be too conservative. Another important question is how to extend our framework to classification and survival analysis data.

6.2. Societal Impact

In particular if this method is used for medical applications, safety concerns must always be paramount. Even if we control some notion of the false discovery rate, it is conceivable that the method will discover a region with a favorable relationship between the covariates and labels that holds only by chance, and if such a region is used to make clinical decisions, it could lead to adverse outcomes for patients. Thus biological plausibility and medical best practices must always be kept in mind when applying DDGroup.

Acknowledgements

We thank all of the anonymous reviewers for their helpful comments and feedback. J.Z. is supported by NSF CAREER 1942926 and grants from the Silicon Valley Foundation and the Chan–Zuckerberg Initiative. Z.I. is supported by a Stanford Interdisciplinary Graduate Fellowship.

Table 2. Performance on single subgroup identification for baseline (linear regression model on the whole data), k-means clustering, linear model tree, and DDGroup on the real-world datasets. Here d denotes the dimension of the features, and subgroup size denotes the fraction of the data included in the selected subgroup. We report the average results (± the standard error of the mean) for 10 runs of different random splits.

Dataset	Task	d	Test MSE				Subgroup Size		
			Baseline	Clustering	LMT	DDGroup	Clustering	LMT	DDGroup
Brazil Health	HF	6	0.80 ± 0.06	0.33 ± 0.03	0.21 ± 0.02	**0.04 ± 0.00**	18% ± 2%	13% ± 1%	6% ± 0%
	stroke	6	1.14 ± 0.22	0.21 ± 0.01	0.16 ± 0.01	**0.06 ± 0.00**	20% ± 2%	14% ± 1%	6% ± 0%
China Glucose	SUA-F	11	0.83 ± 0.02	0.69 ± 0.03	0.73 ± 0.04	0.69 ± 0.06	27% ± 2%	24% ± 6%	21% ± 3%
	SUA-M	11	0.94 ± 0.01	0.89 ± 0.04	**0.80 ± 0.02**	0.81 ± 0.04	21% ± 5%	15% ± 1%	8% ± 1%
China HIV	stigma	27	0.84 ± 0.01	0.86 ± 0.08	0.83 ± 0.08	**0.69 ± 0.04**	6% ± 1%	18% ± 4%	21% ± 3%
Dutch Drinking	inh	16	0.64 ± 0.01	0.56 ± 0.02	0.51 ± 0.03	**0.50 ± 0.02**	11% ± 1%	24% ± 5%	11% ± 2%
	wm	16	0.71 ± 0.01	0.61 ± 0.02	**0.56 ± 0.02**	0.57 ± 0.02	11% ± 1%	18% ± 3%	9% ± 1%
	sha	16	0.64 ± 0.01	0.49 ± 0.02	0.47 ± 0.02	**0.42 ± 0.02**	14% ± 2%	18% ± 4%	10% ± 1%
Korea Grip	strength	11	0.71 ± 0.02	0.84 ± 0.13	0.92 ± 0.10	**0.69 ± 0.04**	7% ± 1%	33% ± 6%	20% ± 3%

Table 3. Performance on multiple subgroups identification for baseline (linear regression model on the whole data), k-means clustering, linear model tree, and DDGroup on the real-world datasets. Here we select **three** subgroups (rather than a single subgroup as in Table 2) and report the average results for the selected groups. Here d denotes the dimension of the features, and subgroup size denotes the fraction of the data included in the selected subgroups. We report the average results for 10 runs of different random splits (± the standard error of the mean).

Dataset	Task	d	Test MSE				Subgroup Size		
			Baseline	Clustering	LMT	DDGroup	Clustering	LMT	DDGroup
Brazil Health	HF	6	0.80 ± 0.06	0.42 ± 0.02	0.35 ± 0.01	**0.21 ± 0.03**	19% ± 1%	14% ± 0%	7% ± 1%
	stroke	6	1.14 ± 0.22	0.27 ± 0.00	0.26 ± 0.01	**0.15 ± 0.01**	23% ± 2%	15% ± 1%	6% ± 1%
China Glucose	SUA-F	11	0.83 ± 0.02	0.75 ± 0.06	0.82 ± 0.03	**0.72 ± 0.03**	29% ± 3%	16% ± 1%	14% ± 2%
	SUA-M	11	0.94 ± 0.01	0.92 ± 0.02	0.88 ± 0.02	0.88 ± 0.03	15% ± 1%	16% ± 1%	14% ± 4%
China HIV	stigma	27	0.84 ± 0.01	0.96 ± 0.04	0.91 ± 0.04	**0.80 ± 0.04**	38% ± 6%	16% ± 2%	20% ± 2%
Dutch Drinking	inh	16	0.64 ± 0.01	0.56 ± 0.02	0.55 ± 0.01	**0.49 ± 0.02**	12% ± 1%	14% ± 1%	10% ± 1%
	wm	16	0.71 ± 0.01	0.64 ± 0.01	**0.58 ± 0.01**	0.59 ± 0.01	13% ± 1%	13% ± 0%	13% ± 3%
	sha	16	0.64 ± 0.01	0.52 ± 0.01	0.51 ± 0.01	**0.47 ± 0.01**	12% ± 1%	14% ± 1%	11% ± 1%
Korea Grip	strength	11	0.71 ± 0.02	0.99 ± 0.17	0.86 ± 0.05	**0.70 ± 0.07**	10% ± 3%	23% ± 2%	23% ± 4%

References

Martin Atzmueller. Subgroup discovery. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 5(1):35–49, 2015.

Jonathan Backer and J Mark Keil. The mono-and bichromatic empty rectangle and square problems in all dimensions. In *Latin American Symposium on Theoretical Informatics*, pages 14–25. Springer, 2010.

Arturs Backurs, Nishanth Dikkala, and Christos Tzamos. Tight hardness results for maximum weight rectangles. *arXiv preprint arXiv:1602.05837*, 2016.

Sarai R Boelema, Zeena Harakeh, Martine JE Van Zandvoort, Sijmen A Reijneveld, Frank C Verhulst, Johan Ormel, and Wilma AM Vollebergh. Adolescent heavy drinking does not affect maturation of basic executive functioning: longitudinal findings from the trails study. *PloS one*, 10(10):e0139186, 2015.

Diego Calderon, Brendan Juba, Sirui Li, Zongyi Li, and Lisa Ruan. Conditional linear regression. In *International Conference on Artificial Intelligence and Statistics*, pages 2164–2173. PMLR, 2020.

Denise de Fátima Barros Cavalcante, Valéria Silva Cândido Brizon, Livia Fernandes Probst, Marcelo de Castro Meneghim, Antonio Carlos Pereira, and Gláucia Maria Bovi Ambrosano. Did the family health strategy have an impact on indicators of hospitalizations for stroke and heart failure? longitudinal study in brazil: 1998-2013. *PLoS One*, 13(6):e0198428, 2018.

Moses Charikar, Jacob Steinhardt, and Gregory Valiant. Learning from untrusted data. In *Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing*, pages 47–60, 2017.

Sourav Chatterjee. *Superconcentration and related topics*, volume 15. Springer, 2014.

Corinna Cortes, Giulia DeSalvo, and Mehryar Mohri. Learning with rejection. In *International Conference on Algorithmic Learning Theory*, pages 67–82. Springer, 2016.

Ilias Diakonikolas, Jerry Li, and Anastasia Voloshinov. Efficient algorithms for multidimensional segmented regression. *arXiv preprint arXiv:2003.11086*, 2020.

David P Dobkin, Herbert Edelsbrunner, and Mark H Overmars. Searching for empty convex polygons. In *Proceedings of the fourth annual symposium on Computational geometry*, pages 224–228, 1988.

Wouter Duivesteijn, Ad Feelders, and Arno Knobbe. Different slopes for different folks: mining for exceptional regression models with cook's distance. In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 868–876, 2012.

Adrian Dumitrescu and Minghui Jiang. On the largest empty axis-parallel box amidst n points. *Algorithmica*, 66(2): 225–248, 2013.

Zachary Izzo, James Zou, and Lexing Ying. How to learn when data gradually reacts to your model. In *International Conference on Artificial Intelligence and Statistics*, pages 3998–4035. PMLR, 2022.

Brendan Juba. Conditional Sparse Linear Regression. In Christos H. Papadimitriou, editor, *8th Innovations in Theoretical Computer Science Conference (ITCS 2017)*, volume 67 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 45:1–45:14, Dagstuhl, Germany, 2017. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik. ISBN 978-3-95977-029-3. doi: 10.4230/LIPIcs.ITCS.2017.45. URL http://drops.dagstuhl.de/opus/volltexte/2017/8151.

Sushrut Karmalkar, Adam Klivans, and Pravesh Kothari. List-decodable linear regression. *Advances in neural information processing systems*, 32, 2019.

Vijay Keswani, Matthew Lease, and Krishnaram Kenthapadi. Towards unbiased and accurate deferral to multiple experts. In *Proceedings of the 2021 AAAI/ACM Conference on AI, Ethics, and Society*, pages 154–165, 2021.

Ilya Lipkovich, Alex Dmitrienko, and Ralph B D'Agostino Sr. Tutorial in biostatistics: data-driven subgroup identification and analysis in clinical trials. *Statistics in medicine*, 36(1):136–196, 2017.

David Madras, Toni Pitassi, and Richard Zemel. Predict responsibly: improving fairness and accuracy by learning to defer. *Advances in Neural Information Processing Systems*, 31, 2018.

Hussein Mozannar and David Sontag. Consistent estimators for learning to defer to an expert. In *International Conference on Machine Learning*, pages 7076–7087. PMLR, 2020.

Francesco Orabona and Dávid Pál. Optimal non-asymptotic lower bound on the minimax regret of learning with expert advice. *arXiv preprint arXiv:1511.02176*, 2015.

Duncan Potts and Claude Sammut. Incremental learning of linear model trees. *Machine Learning*, 61(1):5–48, 2005.

Prasad Raghavendra and Morris Yau. List decodable learning via sum of squares. In *Proceedings of the Fourteenth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 161–180. SIAM, 2020.

Ali Siahkamari, Aditya Gangrade, Brian Kulis, and Venkatesh Saligrama. Piecewise linear regression via a difference of convex functions. In *International Conference on Machine Learning*, pages 8895–8904. PMLR, 2020.

Hao Song, Meelis Kull, Peter Flach, and Georgios Kalogridis. Subgroup discovery with proper scoring rules. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 492–510. Springer, 2016.

Christopher Sutton, Mario Boley, Luca M Ghiringhelli, Matthias Rupp, Jilles Vreeken, and Matthias Scheffler. Identifying domains of applicability of machine learning models for materials science. *Nature communications*, 11 (1):1–9, 2020.

Roman Vershynin. *High-dimensional probability: An introduction with applications in data science*, volume 47. Cambridge university press, 2018.

Elisabeth Vieth. Fitting piecewise linear regression functions to biological responses. *Journal of applied physiology*, 67(1):390–396, 1989.

Yong Wang and Ian H Witten. Induction of model trees for predicting continuous classes. 1996.

Yunyang Wang, Jingwei Chi, Kui Che, Ying Chen, Xiaolin Sun, Yangang Wang, and Zhongchao Wang. Fasting plasma glucose and serum uric acid levels in a general chinese population with normal glucose tolerance: A u-shaped curve. *PLoS One*, 12(6):e0180111, 2017.

Lihui Wen, Min-Ho Shin, Ji-Hyoun Kang, Yi-Rang Yim, Ji-Eun Kim, Jeong-Won Lee, Kyung-Eun Lee, Dong-Jin Park, Tae-Jong Kim, Sun-Seog Kweon, et al. Association between grip strength and hand and knee radiographic osteoarthritis in korean adults: Data from the dong-gu study. *PLoS One*, 12(11):e0185343, 2017.

Yair Wiener and Ran El-Yaniv. Pointwise tracking the optimal regression function. *Advances in Neural Information Processing Systems*, 25, 2012.

Chen Zhang, Xiaoming Li, Yu Liu, Shan Qiao, Liying Zhang, Yuejiao Zhou, Zhenzhu Tang, Zhiyong Shen, and Yi Chen. Stigma against people living with hiv/aids in china: does the route of infection matter? *PloS one*, 11 (3):e0151078, 2016.

A. Geometric Intuition for the Directed Infinity Norm

Here we provide a concrete example of the use of the directed infinity norm in Algorithm 2. For simplicity, take $U = \{\pm e_i\}_{i=1}^d$, where e_i are the standard basis vectors for $\mathbb{R}^d$. For any vector x, we have $\|x\|_{U,\infty} = \max\{x[i], -x[i]\}_{i=1}^d$, where $x[i]$ is the i-th component of x. Thus we see that $\|x\|_{U,\infty} = \|x\|_\infty$ coincides with the standard ℓ_∞ norm in this case. We therefore select the point x^* with the smallest ℓ_∞ norm first. If we think of expanding an ℓ_∞ ball starting from the origin, x^* is the first point the surface of the ball will come into contact with as it expands.

Suppose WLOG that $u^* = e_1$. This means that the side of the expanding ℓ_∞ ball with outward normal in the e_1 direction is the side which "contacted" x^*. Thus any points with $e_1^\top x > e_1^\top x^*$ lie past this face of the expanding box, and therefore cannot possibly constrain any of the other sides of the box. Thus, we no longer need to consider such points.

The side of the box which was expanding in the e_1 direction is no longer moving outwards. Of the remaining directions of expansion, our goal is to find the next point that a side will come into contact with. By the same logic as before, with $U = \{-e_1\} \cup \{\pm e_i\}_{i=2}^d$, the $\operatorname{argmax}_{u \in U} u^\top x$ tells us which of the *remaining* faces x lies above. Therefore, $\operatorname{argmin} \|x\|_{U,\infty}$ is the point which supports one of the remaining growing faces closest to the origin, i.e., the next point of contact for the expanding box.

B. MSE vs. Subgroup Size

As mentioned in Section 2, DDGroup offers a flexible tradeoff between subgroup size and MSE. To implement this tradeoff, we can simply require that the selected region contains at least a proportion p of the validation set. We then select the region with the lowest validation MSE among those regions satisfying this requirement. By varying p between 0 and 1, we can smoothly trade off between the size of the selected subgroup (larger p) and the MSE on the selected subgroup (smaller p).

Figure 2 shows the results of this procedure. The x-axis shows the fraction of test points included in the selected region, and the y-axis shows the test MSE of the model in that region (normalized by the MSE of the baseline model fit to the entire dataset; lower is better). We generated these plots by choosing $p \in \{0.05,\ 0.1,\ 0.2,\ 0.3,\ \dots,\ 0.9, 1.0\}$ and repeating the experiment across 10 random train/validation/test splits for each dataset. As expected, there is a general positive correlation between the size of the selected group (x-axis) and the MSE.

C. More Experiment Details

For the experiment in Table 1, we set $R^* = [-1/3, 1/3]^2$ and the bounding region $B = [-1, 1]^2$. We also set $\sigma_{\text{in}} = 0.3$ and $\sigma_{\text{out}} = 5.0$. For DDGroup, we did a hyperparameter search over constant rejection thresholds $\rho \in \{2, 4, 8, 16, 32, 64\}$. The core group size was always chosen to be $k = n/20$. Rather than using the variable box growing speeds, we instead added a "shrinkage" hyperparameter δ: in Algorithm 2, we only consider points $x \in X_{\text{rej}}$ where $x^\top u^* < a^* - \delta$. Geometrically, after the growing box "collides" with a rejected point, we "shrink" that side back by δ opposite to its normal vector. We did a hyperparameter search over $\delta \in \{0.1, 0.05, 0.025, 0.01\}$.

For each experiment, we used 20% of the n points as a validation set to select the hyperparameters. For DDGroup, to select the hyperparameters using the validation set, we used the following procedure. Let $\hat{R}$ be the region selected by a particular setting of the hyperparameters. Let $\hat{\sigma}$ be an estimate of σ from the core group:

$$\hat{\sigma}^2 = \frac{1}{k-d} \sum_{\substack{(x,y) \\ x \in X_{\text{core}}}} (\hat{\beta}^\top x - y)^2.$$

Let $\hat{q}$ be the 0.9-quantile of the absolute residuals on the validation set:

$$\hat{q} = \inf \left\{ q \ : \ \frac{1}{k} \sum_{\substack{(x,y) \\ x \in X_{\text{core}}}} \mathbb{1}\{|\hat{\beta}^\top x - y| \le q\} \ge 0.9 \right\}.$$

We selected the hyperparameters which produced the largest region $\hat{R}$ (measured in terms of volume) for which $\hat{q} \le 3\hat{\sigma}$.

Lastly, for the LMT method on this experiment, we tuned the tree depth from 1 to twice the dimension.

Code for the experiments can be found at `https://github.com/zleizzo/DDGroup`.

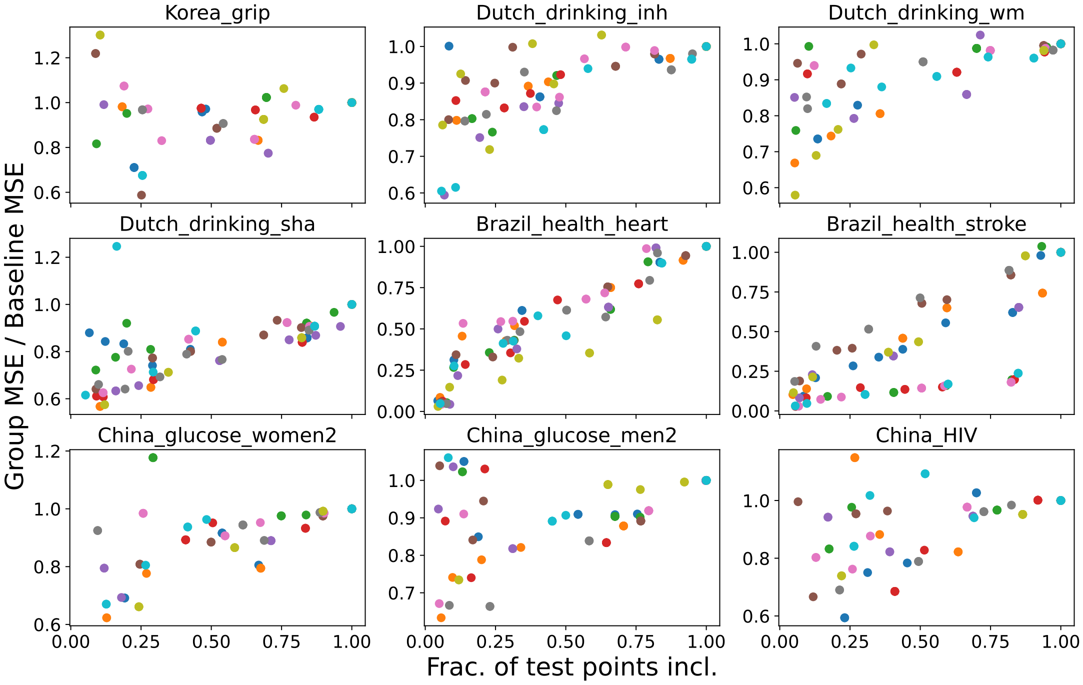


Figure 2. MSE vs. subgroup size selected by DDGroup. The x-axis shows the fraction of test points included in the selected region. The y-axis shows the MSE of the model on the test points in the selected region, normalized by the test MSE of the base model on the whole dataset. (Lower is better.) Different colored points correspond to different random training/validation/test splits on the same dataset. There are 10 random splits in total for each dataset.

D. Omitted Proofs

Let n be the total number of points, k the size of the core group. In what follows, $\|\cdot\|$ denotes the ℓ_2 norm. We will also sometimes use $x[i]$ to refer to the i-th component of the vector x.

X will be used to denote the design matrix of a particular group of points (usually a group of k nearest neighbors as considered by the first phase of the algorithm), and Y will denote the vector of labels corresponding to X. We also use the notation $X \cap R^*$; this refers to the matrix of rows of X which are contained in R^*.

We use the notation $\mathcal{B}(x, r)$ to denote the ℓ_2 ball of radius r centered at x. Finally, "with high probability" means with probability approaching 1 as $n \to \infty$ or $k \to \infty$.

We remark briefly that we select the KNN neighborhood of a point based on the Euclidean norm. This may seem to be a geometric mismatch with the region R^*; since R^* is an axis-aligned box, an ℓ_∞ neighborhood (which is also an axis-aligned box) might seem more appropriate. Since the ℓ_2 and ℓ_∞ norms are equivalent, this distinction will not make any difference for our theoretical results. Empirically, we also do not notice much change in performance. Thus, we will use the more familiar ℓ_2 norm for selecting the core group.

Lemma D.1. *Let Z_i be independent random variables with uniformly bounded fourth moments. Then*

$$\mathbb{P}\left(\left|\frac{1}{n}\sum_{i=1}^{n} Z_i - \mathbb{E}Z_i\right| \geq n^{-1/8}\right) = O(n^{-3/2}).$$

Proof. This is just a generalization of the standard Chebyshev inequality, and the proof proceeds in the same way. By Markov's inequality, we have

$$\mathbb{P}\left(\left|\frac{1}{n}\sum_{i=1}^{n} Z_i - \mathbb{E}Z_i\right| \geq t\right) = \mathbb{P}\left(\left(\sum_{i=1}^{n} Z_i - \mathbb{E}Z_i\right)^4 \geq n^4 t^4\right) \leq \frac{\mathbb{E}[(\sum_{i=1}^{n} Z_i - \mathbb{E}Z_i)^4]}{n^4 t^4}.$$

Expanding $(\sum_{i=1}^{n} Z_i - \mathbb{E}Z_i)^4$ and taking expectation, by linearity of expectation and independence of the Z_i, the only terms which do not vanish are of the form $(Z_i - \mathbb{E}Z_i)^4$ and $(Z_i - \mathbb{E}Z_i)^2(Z_j - \mathbb{E}Z_j)^2$. There are $O(n^2)$ of all of these terms, each with expectation bounded by $O(1)$, so we obtain

$$\mathbb{P}\left(\left|\frac{1}{n}\sum_{i=1}^{n} Z_i - \mathbb{E}Z_i\right| \geq t\right) = O\left(\frac{1}{n^2 t^4}\right).$$

Substituting $t = n^{-1/8}$ completes the proof. □

Lemma D.2. *There exists a constant $c_1 > 0$ (which can depend on R^*, c, C, R) such that if $k \leq c_1 n$, there exists a set of k nearest neighbors of some point in X which is contained in R^* with high probability.*

Proof. By Assumption 5, R^* has nonempty interior. Thus there exists a point $\bar{x} \in R^*$ and a radius $r > 0$ such that $\mathcal{B}(\bar{x}, r) \subseteq R^*$. Consider $\mathcal{B}(\bar{x}, r/4) \subseteq R^*$. By Assumption 3, $\mathbb{P}(x \in \mathcal{B}(\bar{x}, r/4)) \geq c_f \cdot \mathrm{vol}(\mathcal{B}(\bar{x}, r/4)) \equiv p_1 = \Omega(1)$. By Hoeffding's inequality, we have

$$\mathbb{P}\left(\sum_{i=1}^{n} \mathbb{1}\{x_i \in \mathcal{B}(\bar{x}, r/4)\} \leq p_1 n - t\right) \leq e^{-2t^2/n} \quad \implies \quad \sum_{i=1}^{n} \mathbb{1}\{x_i \in \mathcal{B}(\bar{x}, r/4)\} \geq p_1 n - \sqrt{\frac{n \log n}{2}}$$

with probability at least $1 - 1/n$. In particular, since $p_1 = \Omega(1)$, for n large enough we have that $\mathcal{B}(\bar{x}, r/4)$ contains at least *one* point x_{i^*} with high probability.

Next, consider $\mathcal{B}(\bar{x}, r/2)$. Setting $p_2 = c_f \cdot \mathrm{vol}(\mathcal{B}(\bar{x}, r/2))$, the same argument as above shows that

$$\sum_{i=1}^{n} \mathbb{1}\{x_i \in \mathcal{B}(\bar{x}, r/2) \geq p_2 n - \sqrt{\frac{n \log n}{2}} \qquad \text{w.p.} \geq 1 - 1/n.$$

In particular, if we take $c_1 = p_2/2 = \Omega(1)$, then $k \le c_1 n$ implies that there are at least k points contained in $\mathcal{B}(\bar{x}, r/2)$ with high probability for n large enough. We claim that in this event, the KNN of x_{i^*} is contained in $\mathcal{B}(\bar{x}, r) \subseteq R^*$. This is because for $x' \notin \mathcal{B}(\bar{x}, r)$, we have

$$\|x' - x_{i^*}\| \ge \|x' - \bar{x}\| - \|\bar{x} - x_{i^*}\| > r - r/4 = 3r/4.$$

However, for a point $x' \in \mathcal{B}(\bar{x}, r/2)$, we have

$$\|x' - x_{i^*}\| \le \|x' - \bar{x}\| + \|\bar{x} - x_{i^*}\| \le r/2 + r/4 = 3r/4.$$

Thus with probability approaching 1 as $n \to \infty$, there exists a set of k nearest neighbors contained in R^* provided that $k \le c_1 n$. □

Henceforth, we will assume that $k \le c_1 n$.

Lemma D.3. *Let u be any unit vector and define $S_{u,r} = \{x \in X \ : \ |x^\top u| \le r\}$. With probability at least $1 - 1/n$, we have $|S_{u,r}| \le c_2 r n + \sqrt{\frac{n \log n}{2}}$ for some constant c_2.*

Proof. Since S is bounded, we have that $\mathrm{vol}(S_{u,r} \cap S) \le c_2' r$ for some constant c_2' (which depends on the size of S). By Assumption 3, since the density of x is bounded, this means that $\mathbb{P}(x \in S_{u,r}) \le c_2 r$ for some constant c_2. Thus by Hoeffding's inequality, we have that

$$|\{x \ : \ |u^\top x| < r\}| \le c_2 n r + \sqrt{\frac{n \log n}{2}} \qquad \text{w.p.} \ge 1 - 1/n$$

as desired. □

Lemma D.4. *If $k = \Omega(n)$, then with high probability,* every *group of k points X has $\sigma_{\min}(\frac{1}{k} X^\top X) = \Omega(1)$.*

Proof. First, consider a fixed $\|u\| = 1$. For any group of k points, let X be the associated data matrix and define $A = \frac{1}{k} X^\top X$. By Lemma D.3, we have

$$\begin{aligned} u^\top A u &= \frac{1}{k} \sum_{i=1}^{k} (x_i^\top u)^2 \\ &\ge \frac{1}{k} \sum_{i \,:\, |x_i^\top u| \ge r} (x_i^\top u)^2 \\ &\ge \frac{1}{k} \left(k - c_2 r n - \sqrt{\frac{n \log n}{2}} \right) r^2. \end{aligned}$$

Let $c_3 > 0$ be a constant such that $k \ge c_3 n$ and define $r = c_3 / 2c_2$. We obtain the lower bound

$$u^\top A u \ge \left(1 - \frac{c_2 r n}{c_3 n} - \frac{\sqrt{\frac{n \log n}{2}}}{c_3 n} \right) \frac{c_3^2}{4 c_2^2} = \Omega(1)$$

for n large enough. Let $c_4 = \Omega(1)$ be a lower bound on this quantity for large n.

Observe that by the fact that the x_i are bounded, we trivially have $\|A\| \le B^2$:

$$u^\top A u = \frac{1}{k} \sum_{i=1}^{k} (u_i^\top x)^2 \le \frac{1}{k} \sum_{i=1}^{k} \|x_i\|^2 \le B^2.$$

Next, let $E = \{u_i\}_{i=1}^N$ be an ε-net for the unit sphere, and note that we can take $N \leq (3/\varepsilon)^d$. By applying a union bound over E, we have that $u_i^\top(\frac{1}{k}X^\top X)u_i \geq c_4$ for all i with probability at least $1 - N/n$. Let $\|u\| = 1$ be arbitrary and choose u_i such that $\|u - u_i\| \leq \varepsilon$. Then we have

$$\begin{aligned} u^\top A u &= u_i^\top A u_i + (u - u_i)^\top A u + u_i^\top A(u - u_i) \\ &\geq c_4 - \|A\|\|u - u_i\|\|u\| - \|A\|\|u_i\|\|u - u_i\| \\ &\geq c_4 - 2B^2\varepsilon. \end{aligned}$$

Thus if we take $\varepsilon = c_4/4B^2$, we have that $u^\top A u \geq c_4/2 = \Omega(1)$ for all $\|u\| = 1$. This occurs with probability at least $1 - (3/\varepsilon)^d/n = 1 - O(1/n)$, i.e. with high probability. (Note: We only need that $|S_{u_i,r}|$ is bounded according to Lemma D.3 for each u_i, then these inequalities hold simultaneously for *all* groups of k points. In particular, we do not need to take a union bound over the groups of neighboring points.) □

Lemma D.5. *Suppose that $k \geq d$ and $X \in \mathbb{R}^{k\times d}$ has full rank. Then there exist unit vectors $u_i \in \mathbb{R}^n$, $i = 1, \ldots, d$ such that $H \equiv X(X^\top X)^{-1}X^\top = \sum_{i=1}^d u_i u_i^\top$.*

Proof. Let $X = U\Sigma V^\top$ be the SVD of X, where $\Sigma \in \mathbb{R}^{k\times d}$ has diagonal entries σ_i. Since X has full rank, $\sigma_i > 0$ for all $i = 1, \ldots, d$. Define $\Sigma^{-2} = \text{diag}(\sigma_i^{-2}) \in \mathbb{R}^{d\times d}$. We have

$$\begin{aligned} H &= (U\Sigma V^\top)(V\Sigma^{-2}V^\top)V\Sigma^\top U^\top \\ &= U\Sigma(\Sigma^{-2})\Sigma^\top U^\top \\ &= \sum_{i=1}^d u_i u_i^\top, \end{aligned}$$

where $u_i \in \mathbb{R}^k$ are the columns of U (i.e., the left singular vectors of X). □

Lemma D.6. *Let $Y \sim \mathcal{N}(0, \Sigma)$, where $\Sigma \in \mathbb{R}^{m\times m}$ has singular values $\sigma_1 \geq \cdots \geq \sigma_m$. Let $\mu \in \mathbb{R}^m$ be independent of Y. If $\sigma_m \geq \sigma$, then*

$$\mathbb{P}\left(\|\mu + Y\| \leq \sigma\sqrt{m} - t\right) \leq \mathbb{P}\left(\|Y\| \leq \sigma\sqrt{m} - t\right) \leq 2\exp\left(-Ct^2/\sigma^2\right)$$

for some universal constant C.

Proof. This follows directly from Lemmas 9 and 10 in (Izzo et al., 2022). □

Lemma 4.1. *The core group selected by Algorithm 1 has $X_{\text{core}} \setminus R^* = o(k)$ with high probability.*

Proof. By Lemma D.2, there exists a group of k points contained in R^* with high probability. For these points, we have

$$\begin{aligned} \min_\beta \frac{1}{k}\sum_{i=1}^k (x_i^\top\beta - y_i)^2 &\leq \frac{1}{k}\sum_{i=1}^k (x_i^\top\beta^* - y_i)^2 && (2) \\ &= \frac{1}{k}\sum_{i=1}^k (y_i - \mathbb{E}[y_i|x_i])^2 \\ &\leq \sigma^2 + k^{-1/8} \quad \text{w.p.} \geq 1 - O(k^{-3/2}). && (3) \end{aligned}$$

On the other hand, let $\delta > 0$ be fixed and consider a group of k points at least $m \geq \delta k$ of which are *not* in R^*. WLOG assume that the first $i = 1, \ldots, m$ points lie outside R^* and the remaining $k - m$ points are in R^*. Let $\mu_i = \mathbb{E}[y_i|x_i]$ and

$z_i = y_i - \mu_i$. Then we have

$$\begin{aligned}
\min_\beta \frac{1}{k} \sum_{i=1}^{k} (x_i^\top \beta - y_i)^2 &= \min_\beta \frac{1}{k} \sum_{i=1}^{k} (x_i^\top \beta - (\mu_i + z_i))^2 \\
&\geq \frac{1}{k} \min_\beta \sum_{i=1}^{m} (x_i^\top \beta - (\mu_i + z_i))^2 + \frac{1}{k} \min_\beta \sum_{i=m+1}^{k} (x_i^\top \beta - (\mu_i + z_i))^2 \\
&= \frac{1}{k} \left\| (I - H_1)(\boldsymbol{\mu}_1 + Z_1) \right\|^2 + \frac{1}{k} \left\| (I - H_2)(\boldsymbol{\mu}_2 + Z_2) \right\|^2 , \qquad (4)
\end{aligned}$$

where we define X_1 as the data matrix for the m points not in R^* and $H_1 = X_1(X_1^\top X_1)^{-1} X_1^\top$. We define $\boldsymbol{\mu}_1 = (\mu_1, \dots, \mu_m)^\top$ and $Z_1 = (z_1, \dots, z_m)^\top$. $X_2, H_2, \boldsymbol{\mu}_2$, and Z_2 are defined similarly for the $k - m$ points in R^*.

By Lemma D.5, we can write $H_1 = \sum_{i=1}^{d} u_i u_i^\top$ for some orthonormal $u_i \in \mathbb{R}^m$. Extend to an orthonormal basis $u_1, \dots, u_m$ for $\mathbb{R}^m$, and note that $I = \sum_{i=1}^{m} u_i u_i^\top$. It follows that $I - H_1 = \sum_{i=d+1}^{m} u_i u_i^\top$, and therefore

$$(I - H_1) Z_1 = (u_{d+1}^\top Z_1) u_{d+1} + \cdots + (u_m^\top Z_1) u_m.$$

In particular, $(I - H_1) Z_1 \sim \mathcal{N}(0, I_{m-d})$. A similar calculation shows that $(I - H_2) Z_2 \sim \mathcal{N}(0, I_{n-m-d})$. (Here we assume that $m \leq n - d$; if not, we can just use the fact that the second term in (4) is nonnegative.) By applying Lemma D.6, we obtain

$$\begin{aligned}
&\mathbb{P}\left[\min_\beta \sum_{i=1}^{k} (x_i^\top \beta - y_i)^2 \leq \sigma_0^2 \left(\sqrt{m-d} - \sqrt{\frac{\log k}{C}} \right)^2 + \sigma^2 \left(\sqrt{k-m-d} - \sqrt{\frac{\log k}{C}} \right)^2 \right] \\
&\leq \mathbb{P}\left[\| (I - H_1)(\boldsymbol{\mu}_1 + Z_1) \| \leq \sigma_0 \left(\sqrt{m-d} - \sqrt{\frac{\log k}{C}} \right) \right] + \mathbb{P}\left[\| (I - H_2)(\boldsymbol{\mu}_2 + Z_2) \| \leq \sigma \left(\sqrt{k-m-d} - \sqrt{\frac{\log k}{C}} \right) \right] \\
&\leq 4/k.
\end{aligned}$$

The final inequality is obtained by applying Lemma D.6 to each of the two preceding terms. It follows that with high probability (at least $1 - 4/k$), we have that

$$\begin{aligned}
\min_\beta \frac{1}{k} \sum_{i=1}^{k} (x_i^\top \beta - y_i)^2 &\geq \frac{\sigma_0^2 m}{k} + \frac{\sigma^2 (k - m)}{k} - o(1) \\
&\geq \sigma^2 + (\sigma_0^2 - \sigma^2)\delta - o(1). \qquad (5)
\end{aligned}$$

For any constant $\delta > 0$ and k large enough, (5) will be strictly greater than the upper bound in (3). It follows that with high probability, all but $o(k)$ points in the selected core group will belong to R^*. □

Lemma D.7. *Let X be the group of k points selected by Algorithm 1, and define $\widetilde{X} = X \cap R^*$. Then we have that $\|(\frac{1}{k} X^\top X)^{-1} - (\frac{1}{k} \widetilde{X}^\top \widetilde{X})^{-1}\| = o(1)$ with high probability.*

Proof. Let $A = \frac{1}{k} X^\top X$ and $B = \frac{1}{k} \widetilde{X}^\top \widetilde{X}$. By Lemma D.4, $\sigma_{\min}(A) \geq c_4 = \Omega(1)$ with high probability. By Lemma 4.1, $\widetilde{X}$ contains all but $o(k)$ of the selected points. Also recall that by our assumptions, all of the x are bounded. By Weyl's inequality, we have

$$\sigma_{\min}(B) \geq \sigma_{\min}(A) - \sigma_{\max}\left(\frac{1}{k} \sum_{x \in X \setminus R^*} x x^\top \right) = \Omega(1) - o(1) = \Omega(1).$$

In particular, this means that $\|B^{-1}\| = \sigma_{\min}(B)^{-1} = O(1)$. Finally, since $A^{-1} - B^{-1} = A^{-1}(B - A)B^{-1}$, we have

$$\|A^{-1} - B^{-1}\| \leq \|A^{-1}\| \|B - A\| \|B^{-1}\| = O(1) \cdot o(1) \cdot O(1) = o(1).$$

□

Lemma D.8. $\|\hat{\beta}\| = O(1)$ *with high probability.*

Proof. We know from the proof of Lemma 4.1 that there exists a group of k nearest neighbors contained in R^*, and that for such a group, the training MSE is at most $\sigma^2 + o(1)$ with high probability. Thus, since the core group has the minimum training MSE, we must have

$$\begin{aligned}\sigma^2 + o(1) &\geq \frac{1}{k}\|X\hat{\beta} - Y\|^2 \\ &\geq \frac{1}{k}\|\widetilde{X}\hat{\beta} - \widetilde{Y}\|^2 \\ &\geq \frac{1}{k}\|\widetilde{X}\beta^* - \widetilde{Y}\|^2 - \frac{2}{k}\|\widetilde{X}\beta^* - \widetilde{Y}\|\|\widetilde{X}(\hat{\beta} - \beta^*)\| + \frac{1}{k}\|\widetilde{X}(\hat{\beta} - \beta^*)\|^2 \\ &\geq \sigma^2 - o(1) + \frac{1}{k}\left(\|\widetilde{X}(\hat{\beta} - \beta^*)\|^2 - 2\|\widetilde{X}\beta^* - \widetilde{Y}\|\|\widetilde{X}(\hat{\beta} - \beta^*)\|\right). \qquad (6)\end{aligned}$$

Inequality (6) holds because $\widetilde{X}, \widetilde{Y}$ contain $k - o(k)$ points and by roughly the same logic used to obtain the upper bound in (3). It follows that

$$\frac{1}{k}\left(\|\widetilde{X}(\hat{\beta} - \beta^*)\|^2 - 2\|\widetilde{X}\beta^* - \widetilde{Y}\|\|\widetilde{X}(\hat{\beta} - \beta^*)\|\right) = o(1). \qquad (7)$$

Again using the logic from (3), we have that $\|\widetilde{X}\beta^* - \widetilde{Y}\| = \sigma\sqrt{k} + o(\sqrt{k})$. Combining this with equation (7) therefore shows that $\|\widetilde{X}(\hat{\beta} - \beta^*)\| = O(\sqrt{k})$ (this follows from a simple application of the quadratic formula), or equivalently $\|\widetilde{X}(\hat{\beta} - \beta^*)\|^2 = O(k)$.

Finally, observe that

$$\|\widetilde{X}(\hat{\beta} - \beta^*)\|^2 = (\hat{\beta} - \beta^*)\widetilde{X}^\top\widetilde{X}(\hat{\beta} - \beta^*) \geq \sigma_{\min}(\widetilde{X}^\top\widetilde{X})\|\hat{\beta} - \beta^*\|^2.$$

By the proof of Lemma D.7, we know that $\sigma_{\min}(\frac{1}{k}\widetilde{X}^\top\widetilde{X}) = \Omega(1)$, so $\sigma_{\min}(\widetilde{X}^\top\widetilde{X}) = \Omega(k)$. Thus we have

$$O(k) = \|\widetilde{X}(\hat{\beta} - \beta^*)\|^2 \geq \Omega(k)\|\hat{\beta} - \beta^*\|^2 \quad \Longrightarrow \quad \|\hat{\beta} - \beta^*\| = O(1).$$

Since $\|\beta^*\|$ is a constant, we conclude that $\|\hat{\beta}\| = O(1)$ by the triangle inequality. □

Lemma D.9. $\left\|\frac{1}{k}X^\top Y - \frac{1}{k}\widetilde{X}^\top\widetilde{Y}\right\| = o(1)$ *with high probability.*

Proof. We begin with the same observation used to prove Lemma D.8, namely that the training MSE for the core group must be upper bounded by $\sigma^2 + o(1)$ with high probability. By Assumption 3, $\|x\| = O(1)$, and by Lemma D.8, $\|\hat{\beta}\| = O(1)$ with high probability. Let C be a constant such that $\|x\|\|\hat{\beta}\| \leq C$. We then have

$$\begin{aligned}\frac{1}{k}\|X\hat{\beta} - Y\|^2 &\geq \frac{1}{k}\|\widetilde{X}\hat{\beta} - \widetilde{Y}\|^2 + \frac{1}{k}\sum_{(x,y)\,:\,x\notin R^*}(x^\top\hat{\beta} - y)^2 \\ &\geq \sigma^2 - o(1) + \frac{1}{k}\sum_{\substack{(x,y):x\notin R^* \\ |y|\geq 2C+1}}|y|(|y| - 2\|x\|\|\hat{\beta}\|) \\ &\geq \sigma^2 - o(1) + \frac{1}{k}\sum_{\substack{(x,y):x\notin R^* \\ |y|\geq 2C+1}}|y|.\end{aligned}$$

It therefore follows that

$$\frac{1}{k}\sum_{\substack{(x,y):x\notin R^* \\ |y|\geq 2C+1}}|y| = o(1). \qquad (8)$$

Since $\|x\| \leq B$, we now have

$$\begin{aligned}\left\|\frac{1}{k}X^\top Y - \frac{1}{k}\widetilde{X}^\top \widetilde{Y}\right\| &\leq \frac{1}{k}\sum_{(x,y):x\notin R^*} \|x\||y| \\ &\leq \frac{1}{k}\sum_{\substack{(x,y):x\notin R^* \\ |y|<2C+1}} B(2C+1) + \frac{1}{k}\sum_{\substack{(x,y):x\notin R^* \\ |y|\geq 2C+1}} B|y| \\ &= o(1).\end{aligned}$$

The final conclusion holds by applying Lemma 4.1 and equation (8) to the two terms in the previous line. □

Lemma 4.2. *Let X_{core} be the core group selected by Algorithm 1 and let $\hat{\beta}$ be the OLS estimator fit to X_{core}. Let X_{rej} be the set of rejected points defined by the thresholding procedure in Phase 2. With high probability, none of the points in X_{rej} belong to R^*.*

Proof. First, we will show that $\|\beta^* - \hat{\beta}\| = o(1)$ with high probability. Let $\widetilde{X} = X \cap R^*$ denote the data matrix for the core points which belong to R^*, and let $\widetilde{Y}$ denote the response vector corresponding to these points. We use the identity

$$\begin{aligned}\hat{\beta} - \beta^* = &\underbrace{\left[\left(\frac{1}{k}X^\top X\right)^{-1} - \left(\frac{1}{k}\widetilde{X}^\top \widetilde{X}\right)^{-1}\right]\left(\frac{1}{k}X^\top Y\right)}_{\text{(I)}} + \underbrace{\left(\frac{1}{k}\widetilde{X}^\top \widetilde{X}\right)^{-1}\left[\frac{1}{k}X^\top Y - \frac{1}{k}\widetilde{X}^\top \widetilde{Y}\right]}_{\text{(II)}} \\ &+ \underbrace{\left(\frac{1}{k}\widetilde{X}^\top \widetilde{X}\right)^{-1}\left(\frac{1}{k}\widetilde{X}^\top \widetilde{Y}\right) - \beta^*}_{\text{(III)}}.\end{aligned}$$

Term (I) By Lemma D.7, $\|(\frac{1}{k}X^\top X)^{-1} - (\frac{1}{k}\widetilde{X}^\top \widetilde{X})^{-1}\| = o(1)$. By Lemma D.9 and an application of the triangle inequality, $\|\frac{1}{k}X^\top Y\| = O(1)$, so term (I) is $o(1)$.

Term (II) By the proof of Lemma D.7, $\|(\frac{1}{k}\widetilde{X}^\top \widetilde{X})^{-1}\| = O(1)$. By Lemma D.9, $\|\frac{1}{k}X^\top Y - \frac{1}{k}\widetilde{X}^\top \widetilde{Y}\| = o(1)$, so term (II) is $o(1)$.

Term (III) Let k' be the number of points in $\widetilde{X}$ (so $k' = k - o(k)$) and WLOG assume that the points in $\widetilde{X}$ are the first k' points $x_1, \ldots, x_{k'}$. We have $\widetilde{Y} = \widetilde{X}\beta^* + E$, where $E = (\varepsilon_i)_{i=1}^{k'}$ is the vector of error terms and $\varepsilon_i \sim \mathcal{N}(0, \sigma^2)$. Define $\tilde{\beta} = (\frac{1}{k'}\widetilde{X}^\top \widetilde{X})^{-1}(\frac{1}{k'}\widetilde{X}^\top \widetilde{Y})$ and note that this is still equal to the first term in (III). It follows that

$$\tilde{\beta} = \beta^* + \left(\frac{1}{k'}\widetilde{X}^\top \widetilde{X}\right)^{-1}\sum_{i=1}^{k'}\varepsilon_i x_i,$$

This implies that

$$\|\hat{\beta} - \beta^*\| \leq \left\|\left(\frac{1}{k'}\widetilde{X}^\top \widetilde{X}\right)^{-1}\right\|\left\|\frac{1}{k'}\sum_{i=1}^{k}\varepsilon_i x_i\right\| = \sigma\sigma_{\min}^{-1}\left\|\frac{1}{k'}\sum_{i=1}^{k'} g_i x_i\right\|,$$

where $g_i \overset{\text{iid}}{\sim} \mathcal{N}(0,1)$ and $\sigma_{\min} = \sigma_{\min}(\frac{1}{k}\widetilde{X}^\top \widetilde{X})$. It remains to bound $\left\|\frac{1}{k'}\sum_{i=1}^{k'} g_i x_i\right\|$ with high probability. Observe that

$$\mathbb{P}\left(\left\|\frac{1}{k'}\sum_{i=1}^{k'} g_i x_i\right\| \geq t\right) \leq \sum_{j=1}^{d}\mathbb{P}\left(\left|\frac{1}{k'}\sum_{i=1}^{k'} g_i x_{ij}\right| \geq \frac{t}{\sqrt{d}}\right).$$

Standard Gaussian concentration results (see e.g. (Vershynin, 2018)) show that the RHS is bounded by $2d\exp\left(\frac{-ck't^2}{d}\right)$ for some universal constant c. Setting this bound equal to $1/k'$ and solving for t, we see that $\|\hat{\beta}-\beta^*\| \le C\sigma\sigma_{\min}^{-1}\sqrt{\frac{d\log(2dk')}{k'}}$ with probability at least $1-1/k'$, i.e. with high probability. (Here C is another universal constant.) Since $\sigma_{\min}=\Omega(1)$, we have $\|\hat{\beta}-\beta^*\|=o(1)$.

Next, we look at $|y_i - x_i^\top\hat{\beta}|$ for a point $x_i \in R^*$. In this case, applying the triangle inequality and Cauchy-Schwarz, we have

$$|y_i - x_i^\top\hat{\beta}| = |x_i^\top\beta^* - x_i^\top\hat{\beta} + \varepsilon_i| \le \|\beta^* - \hat{\beta}\|\|x_i\| + |\varepsilon_i|.$$

Since our dataset contains n points, there are at most n points in R^*. Again by standard Gaussian concentration results and a union bound, we have that $|\varepsilon_i| \le \sigma\sqrt{2\log\frac{2n}{\alpha}}$ for all $x_i \in R^*$ simultaneously with probability at least $1-\alpha$. Thus we have that

$$|y_i - x_i^\top\hat{\beta}| \le \sigma\sqrt{2\log\frac{2n}{\alpha}} + o(1)$$

for all $x_i \in R^*$ with probability at least $1-\alpha-o(1)$. Setting $\alpha = 1/n$ and adjusting the constants slightly to account for the $o(1)$ term, we see that

$$|y_i - x_i^\top\hat{\beta}| \le 2.1\sigma\sqrt{\log n}$$

with high probability and for large enough n, as desired. □

Lemma D.10. *With high probability, the average of the core point feature vectors belongs to R^*.*

Proof. In this proof, we will use the fact that R^* is an axis-aligned box, but we note that this is just for ease of exposition and the results extend readily to the case when R^* is a general convex body.

Let $R^* = \prod_{i=1}^d [a_i, b_i]$ with $a_i < b_i$ for each i and define $\partial R^*_\varepsilon = \prod_{i=1}^d [a_i+\varepsilon, b_i-\varepsilon]$ for $\varepsilon < \min_i(b_i-a_i)/2$. (That is, ∂R^*_ε consists of those points in R^* which are at most ε away from the boundary of R^*.) A direct calculation shows that $\mathrm{vol}(\partial R^*_\varepsilon) = O(\varepsilon)$. By the same logic as in Lemma D.3, it follows that there exists an $\varepsilon = \Omega(1)$ such that ∂R^*_ε contains at most $m \le c_3 n/2$ points with high probability, where here c_3 is a constant such that $k \ge c_3 n$.

Let $\bar{x}$ be the average of the core group feature vectors. We will show that $\bar{x}[i] \le b_i$. A nearly identical argument will show that all of the components of $\bar{x}$ satisfy the constraints required to belong to R^*. Observe that

$$\begin{aligned}\bar{x}[i] &= \frac{1}{k}\left(\sum_{x\in R^*\setminus\partial R^*_\varepsilon} x[i] + \sum_{x\in\partial R^*_\varepsilon} x[i] + \sum_{x\notin R^*} x[i]\right) \\ &\le \frac{1}{k}((k-m-o(k))(b_i-\varepsilon) + mb_i + o(k)) && (9)\\ &= b_i + (\frac{m}{k}-1)\varepsilon + o(1) \\ &\le b_i - \varepsilon/2 + o(1) && (10)\\ &\le b_i.\end{aligned}$$

Inequality (9) follows from the fact that the features x (and hence each component $x[i]$) are bounded, and the fact that at most $o(k)$ points in the core group are not in R^* by Lemma 4.1. Inequality (10) holds because $m \le c_3 n/2$ and $k \ge c_3 n$. This completes the proof. □

Lemma D.11. *Suppose that $Y_i \sim \mathcal{N}(0,\sigma_i^2)$ are independent. Then $\mathbb{P}(\max|Y_i - a_i| \le t) = \mathbb{P}(\max|Y_i| \le t)$ and consequently $\mathbb{E}\max|Y_i - a_i| \ge \mathbb{E}\max|Y_i|$ for any constants a_i.*

Proof. Lemma D.6 implies that $\mathbb{P}(|Y_i - a_i| \le t) \le \mathbb{P}(|Y_i| \le t)$ for all i, t. Furthermore, we have that

$$\mathbb{P}(\max|Y_i - a_i| \le t) = \prod_i \mathbb{P}(|Y_i - a_i| \le t) \le \prod_i \mathbb{P}(|Y_i| \le t) = \mathbb{P}(\max|Y_i| \le t).$$

Integrating by parts shows that

$$\mathbb{E}\max|Y_i - a_i| = \int_0^\infty \mathbb{P}(\max|Y_i - a_i| \geq t)\,dt \geq \int_0^\infty \mathbb{P}(\max|Y_i| \geq t)\,dt = \mathbb{E}\max|Y_i|.$$

□

Lemma D.12. *Let $Y_i \sim \mathcal{N}(0, \sigma_i^2)$ with $\sigma_i^2 \geq \sigma^2$ for all i. Then for any constants a_i and m large enough, we have the following inequality:*

$$\mathbb{E}\left[\max_{i=1}^m |Y_i - a_i|\right] \geq 0.12\sigma\sqrt{\log m}.$$

Furthermore, we have that $\max_{i=1}^m |Y_i - a_i| = \Omega(\sigma\sqrt{\log m})$ with high probability as $m \to \infty$.

Proof. Note that by Lemma D.11, it suffices to show the result for $a_i = 0$. Next, observe that since $\sigma^2/\sigma_i^2 \leq 1$ for all i, we have

$$\mathbb{E}\left[\max_{i=1}^m |Y_i|\right] \geq \mathbb{E}\left[\max_{i=1}^m \frac{\sigma}{\sigma_i}|Y_i|\right] \geq \mathbb{E}\left[\max_{i=1}^m Z_i\right],$$

where $Z_i \overset{\text{iid}}{\sim} \mathcal{N}(0, \sigma^2)$. By Theorem 3 of (Orabona and Pál, 2015), $\mathbb{E}\left[\max_{i=1}^m Z_i\right] \geq 0.13\sigma\sqrt{\log m} - 0.7\sigma \geq 0.12\sigma\sqrt{\log m}$ for large enough m.

Next, we will show that $\mathbb{P}(\max|Y_i| \leq t)$ is decreasing in σ_i. We have

$$\begin{aligned}\mathbb{P}(\max|Y_i| \leq t) &= \prod_{i=1}^m \mathbb{P}(|\sigma_i Z_i| \leq t), \quad Z_i \overset{\text{iid}}{\sim} \mathcal{N}(0,1) \\ &= \prod_{i=1}^m \mathbb{P}(|Z_i| \leq t/\sigma_i). \end{aligned} \tag{11}$$

Since $\{|Z_i| \leq t/\sigma_i\} \subseteq \{|Z_i| \leq t/\sigma_i'\}$ when $\sigma_i \geq \sigma_i'$, the terms in (11) are decreasing in σ_i.

Next, suppose that $\sigma_i = \sigma$ for all i. We will show that $\mathbb{V}\left(\max_{i=1}^m |Y_i - a_i|\right) \leq \sigma^2$. By homogeneity, it suffices to show this inequality for $\sigma = 1$. Define $f(Y_1, \ldots, Y_m) = \max_{i=1}^m |Y_i|$. By the Gaussian Poincaré inequality (see e.g. (Chatterjee, 2014), pg. 47), we have that

$$\mathbb{V}\left(\max_{i=1}^m |Y_i|\right) \leq \sum_{i=1}^n \mathbb{E}|\partial_i f(Y)|^2.$$

We have $\partial_i f(Y) = \text{sign}(Y_i)\mathbb{1}\{|Y_i| = \max_j |Y_j|\}$ almost everywhere, so

$$\mathbb{E}|\partial_i f(Y)|^2 = \mathbb{P}(|Y_i| = \max_j |Y_j|\}) = 1/m$$

for all i. It follows that $\mathbb{V}(\max_{i=1}^m |Y_i|) \leq 1$. By Chebyshev's inequality, we therefore have that

$$\mathbb{P}\left(\max_{i=1}^m |Y_i| \leq 0.12\sigma\sqrt{\log m} - \sigma t\right) \leq 1/t^2.$$

In particular, we can take $t = 0.06\sqrt{\log m}$, then $\max_{i=1}^m |Y_i| = \Omega(\sigma\sqrt{\log m})$ with probability at least $1 - O(1/\log m) = 1 - o(1)$, i.e. with high probability. □

Theorem 4.3. *As $n \to \infty$, there exist positive scalars $\{s_j^\pm\}_{j=1}^d$ and a constant $c > 0$ such that if $U = \{s_j^+ e_j, -s_j^- e_j\}_{j=1}^d$ and $k = \Omega(n)$ with $k \leq cn$, Algorithm 3 returns $\hat{R}$ with $R^* \subseteq \hat{R}$ with high probability. Furthermore,* $\text{vol}(\hat{R} \setminus R^*) \to 0$.

Proof. By Lemma D.10, the average of the core group points $\bar{x}$ (and therefore the point from which we begin growing the box in Algorithm 2) lies in the interior of R^*. Let ∂R^* denote the boundary of R^*. For each $j = 1, \ldots, d$, denote by $\partial R^*_{j,+}$ the face of ∂R^* which upper bounds the j-th dimension, and let $\partial R^*_{j,-}$ be the opposite face which lower bounds the j-th dimension. Let $s_j^\pm = d(\bar{x}, \partial R^*_{j,\pm})$ be the distance from the center to the appropriate face of R^*. Note that Algorithm 2 with these speeds and this center is equivalent to running the algorithm from the origin and with uniform speeds, after shifting the data so that $\bar{x}$ lies at the origin and then rescaling each axis by $s_j^{\pm}$. In this case, R^* is transformed into a ℓ_∞ ball of radius 1 centered at the origin.

By Lemma 4.2, R^* contains no rejected points with high probability. (Note that the transformations we performed above preserve this fact.) Since the region returned by Algorithm 2 returns a region which contains the largest centered ℓ_∞ ball with no rejected points in it, and R^* is a centered ℓ_∞ ball with no rejected points, we must have $R^* \subseteq \hat{R}$ as desired.

Since we have assumed that R^* is an axis-aligned box, we can write $R^* = \{x \mid a_j < x_j < b_j\}$. Fix $\varepsilon > 0$ and let

$$\partial R^*_{\varepsilon,j,+} = \{x \mid b_j \leq x_j \leq b_j + \varepsilon,\ \ell_m < x_m < u_m, m \neq j\}$$

$$\partial R^*_{\varepsilon,j,-} = \{x \mid a_j - \varepsilon \leq x_j \leq a_j,\ \ell_m < x_m < u_m, m \neq j\}.$$

(These are just the sets of points which are at most ε “above” the upper dimension j face of R^* and “below” the lower dimension j face of R^*, respectively.)

By the same logic as in the proof of Lemma D.3, there is some constant $c_6 > 0$ (which can depend on R^*) such that at least $c_6 \varepsilon n$ points lie in ∂R^*_ε with high probability. Take $\varepsilon = n^{-1/2}$ and apply Lemma D.12 to the $c_6 \varepsilon n$ points in $\partial R^*_{\varepsilon,j,\pm}$. We see that

$$\max_{x_i \in \partial R^*_{\varepsilon,j,\pm}} |x_i^\top \hat{\beta} - y_i| \geq 0.06 \sigma_0 \sqrt{\frac{1}{2} \log c_6 n}$$

with high probability. Since $\sigma_0 > 50\sigma$, for n large enough we have

$$\max_{x_i \in \partial R^*_{\varepsilon,j,\pm}} |x_i^\top \hat{\beta} - y_i| \geq 0.06 \sigma_0 \sqrt{\frac{1}{2} \log c_6 n} > 2.1\sigma \sqrt{\log \varepsilon n}.$$

The means that Algorithm 2 will stop growing the $(j, \pm)$ side of $\hat{R}$ at some point in $\partial R^*_{\varepsilon,j,\pm}$. It follows that $\hat{R} \subseteq R^*_\varepsilon$ with $\varepsilon = n^{-1/2}$. This completes the proof. □