Data Sculpting: Interpretable Algorithm for End-to-End Cohort Selection

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

Abstract

Many scientific and medical analysis involves fitting a parametric model over a heterogeneous data set. The model is often chosen to be low capacity (e.g. logistic regression) in order to make statistical inference about the association between each feature and the outcome (e.g. odds ratio). However the simple model often cannot capture the heterogeneity in the data. For example, a subset of the data might follow a clean logistic relation, but other data points could follow different relations so that the fitting a logistic regression over the entire set may not find any association. In this paper, we propose a novel algorithm, Data Sculpting, for simultaneously learning to select a subset of the data while fitting the desired parametric model on the selected cohort. Data Sculpting retains the statistical inference convenience of the original model, while leveraging end-to-end differentiable optimization (via the Concrete selector) to learn interpretable rules for selecting the cohort. Extensive experiments demonstrate that Data Sculpting is efficient, robust and substantially improves over the standard approaches.

1. Introduction

Many medical and social science studies follow a two step procedure: a dataset is first curated, and then a parametric model (e.g. linear or logistic regression) is fit over the data (Finlay & Agresti, 1986). From the model, the data scientist can then make inferences about specific features (e.g. what's the odds ratio corresponding to a feature) or predict labels/outcomes for new samples. The data curation step is often decoupled from the statistical analysis. For the latter step, the whole dataset is simply assumed to be given and is used for fitting and inference. As data becomes increasingly heterogeneous, however, such decoupling of data curation and statistical modeling has substantial limitations and can miss the underlying signal.

As an example, suppose we have collected demographics and biomarkers of a large heterogeneous dataset, and we want to quantify how these features contribute to the risk of high blood pressure. The standard procedure is to fit a linear model over the data using these features with blood pressure as the outcome (simple parametric models are often preferred here for their interpretability and ease of inference). It could very well be that the linear relation is a good proxy for a subset of the data, say women older than 30. But on the rest of the data, either due to bias in the sample collection or underlying biological differences, this parametric model is not a good match. If we simply fit the model on the entire dataset, we might not find any signal. Alternatively, we could try to fit a more flexible complex model over the data, but this loses the desired inferences and interpretability. For instance, clinicians often want to know the explicit odds ratio of each feature, which is not easy to infer from nonlinear models. A better solution would be if an algorithm can automatically learn to focus on a particular subset of the cohort—say women older than 30—and focus the parametric modeling just on this subset. Then we would retain interpretability while capturing signal in the features in the selected cohort.

A reasonable question is: is it sufficient to only capture a subset of the data with our model? Capturing an appropriate subset turns out to be perfectly reasonable in many applications. When evaluating treatment efficacy or in designing clinical trials, it's good if we can identify a subset of the large cohort where the statistical model is a good match and the signal is clean (Meinert, 2012). The current approach for identifying the subset relies on mostly manual curation by experts—often in the form of inclusion/exclusion conditions—which is still decoupled from the downstream statistical modeling. The challenge is that manual cohort selection is *ad hoc*, labor intensive, and often fails to capture the most salient sub-population.

In this paper, we propose a novel and intuitive approach to address the aforementioned challenges by automatically learning the subpopulation while simultaneously fitting the desired parametric model. This approach integrates the two

[1]Department of Electrical Engineering, Stanford University, CA, USA [2]Department of Biomedical Data Scienc, Stanford University, CA, USA. Correspondence to: Ruishan Liu <ruishan@stanford.edu>, James Zou <jamesz@stanford.edu>.

Proceedings of the 39^{th} International Conference on Machine Learning, Baltimore, Maryland, USA, PMLR 162, 2022.

steps for cohort selection and model fitting in an-end-to-end optimization that we call Data Sculpting. Data Sculpting learns interpretable rules for selecting the appropriate subset of the data (i.e. cohort selection) while preserving valid statistical inference of the model parameters on the selected cohort.

Our contributions **Conceptual:** We formulate the important problem of interpretable cohort selection, which is widely useful in many data science problems. This is a novel formulation, to the best of our knowledge. **Algorithmic** We then propose an efficient new algorithm, Data Sculpting, which simultaneous selects the cohort and performs downstream model fitting. Data Sculpting leverages Concrete random variables and is both flexible and easy to interpret. **Empirical:** Extensive experiments on real and synthetic data show that Data Sculpting captures much cleaner signal than fitting the model on the entire data.

Related Works Fitting parametric regression models is a bedrock of many data science projects, and it is especially commonly used in medical and social science studies (Friedman et al., 2001; Zou et al., 2014; Finlay & Agresti, 1986). There is a rich literature on robust statistics where the goal is to fit such models (e.g. linear regression) in a way that is less sensitive to outliers (Huber, 2004). Data Sculpting is quite different from outlier removal and robust statistics in several aspects. First, outliers are typically modeled as worst case (adversarial) perturbations and they are assumed to make up a relatively small proportion of the data (certainly less than 50%) (Cousineau & Chartier, 2010). In Data Sculpting, the data that are not in the selected cohort often are not outliers or adversarial perturbations, as our real-data experiments show. They are whatever data where the simple parametric model is not a good fit, which is much more relaxed than outliers. In fact the non-selected data could be the majority of the entire dataset, as is the case in many of our real-world experiments.

The second key difference is that in Data Sculpting, the learned cohort selection rules should be directly interpretable. In order to be clinically useful, for example, the algorithm needs have very simple critera for deciding which individuals are eligible to join the cohort, and the selection criteria should be as few as possible. This motivates our algorithm and the usage of Concrete selectors to identify a sparse set of features on which to select the cohort. Most robustness training or outlier removal works do not consider such interpretability considerations. There is also a rich literature in statistics and machine learning on sparse feature selection (Tibshirani, 1996; Gimenez et al., 2018). However the goal there is to identify a small set of features with which to make the final predictions. This includes the original paper that proposed using concrete selector for supervised learning and for reconstruction (Balın et al., 2019). In Data Sculpting, in contrast, the sparse feature selection is used only for deciding the boundaries of the cohort, and not for the final prediction. On the selected cohort, the full set of features could be used since the goal is often to infer the odds ratio and perform hypothesis testing on every feature.

Cohort selection is tremendously important in applications like clinical trials, where the objective is to identify a sufficiently large slice of the general population in which a treatment is more likely to be useful or there is a high signal (which can be quantified in a regression model) (Meinert, 2012). In all of such applications that we are aware of, the cohort selection is done manually, where experts write down a set of eligibility criteria based on their prior experience and domain knowledge. A major challenge in the field is how to optimize cohort selection in a data-driven manner, which motivates our work. To the best of our knowledge, this paper is the first to formalize cohort selection as an end-to-end optimization problem.

2. Data Sculpting

2.1. Background and Example

We first formulate the problem of data (cohort) selection for supervised learning. Modern machine learning has achieved remarkable accuracy with complex models such as deep neural network (DNN). However it is challenging extract the association between each feature and the output of the DNN while providing statistical quantification for odds ratio and confidence intervals. In many scientific and medical applications, the interpretation of each feature's contribution is often even more important than prediction. This is a key reason why simple parametric models are still widely used especially in science and medicine, where feature interpretation can be even more important than prediction accuracy. For example, odds ratios (OR) — coefficients in logistic regression model — is the standard reporting metric in medical and social science studies (Bland & Altman, 2000).

Due to the limited capacity of simple models, one problem arises: *Fitting a simple model globally over the entire data can fail to capture the heterogeneous relation between features and outcome*. For example, it is common that a medical dataset consists of patients from diverse demographic and genetic background. People with same background generally have similar feature-output associations, but the underlying relationship can be very different across groups. In this case, a simple model fitted globally can be confused by the heterogeneity and can fail to capture any of the true associations.

Consider an illustrative dataset for binary classification in Figure 1a. Here the two subpopulations correspond to the

two Gaussians. If a logistic regression model is fitted on the whole dataset, the reported odds ratios can be misleading. Thus it is desirable to identify a single cohort and fit the linear model only on that subset. In real-world application, the two features x_1, x_2—which form the axes of Figure 1a—could be age and income. Then we might want to select a cohort, i.e. a subset of the whole dataset, by learning simple thresholds based on age and income and infer the odds ratio of these two features on this selected cohort. The shaded region in Fig. 1a is the cohort selected by our algorithm Data Sculpting, which we will explain below.

To retrieve meaningful model coefficients for interpretation, it is necessary to identify coherent cohorts in the data on which the linear models are fitted. In practice, the identification step is mainly done manually or by heuristics. This is not optimal for heterogeneous and high dimensional data. In the light of this, we develop

1. An algorithm automatically identifies a cohort in the data which gives the best performance for a simple and interpretable model.
2. The cohort selection criteria are simple and easy to interpret.

2.2. Cohort Selection

We formulate the cohort selection problem. We are given a dataset $D = \{x^{(i)}\}$, where $x^{(i)} \in \mathbb{R}^d$, and a model $f_\theta(\cdot)$ which we would like to fit. For concreteness, we set $f_\theta(\cdot)$ as logistic regression model and the coefficient $\theta \in \mathbb{R}^d$ gives the widely-used odds ratio. The Data Sculpting framework can work with any differentiable $f_\theta(\cdot)$; we work with logistic regression because it is the most commonly used model in medical and social science studies.

Our goal is to automatically select a cohort, which is a subset of data $S \subseteq D$. We additionally require that the selected S be *interpretable*, so that it is very intuitive to a clinician or other non-technical user which data are selected to be in S. In this paper, we require that $S = D \cap \otimes[s_j^{\min}, s_j^{\max}]$, where $s_j^{\min} < s_j^{\max}$ are scalars that set the upper and lower bounds for feature j. In other words, selecting S simply corresponds to learning two bounding thresholds for each feature. In order to further improve interpretability and to ensure $|S|$ is reasonably large, we will often require that for most of the features j, $s_j^{\min} = -\infty$ and $s_j^{\max} = \infty$, so that selection is done using a sparse subset of the features.

Conventionally, the cohort selection criteria are composed of rules for different features. The datapoint is not selected unless all the rules are satisfied. For example, given a dataset with feature age and income, possible selection rules could be 1) age larger than 18 and less than 45 and 2) income larger than 100k. Then a datapoint $x^{(i)}$ is selected only if $18 < x_1^{(i)} < 45$ and $x_2^{(i)} > 100k$.

We model the probability that rule for feature j is satisfied:

$$p_{s_j}(x_j) = \begin{cases} 1 & s_j^{\min} \leqslant x_j \leqslant s_j^{\min} \\ 0 & \text{otherwise} \end{cases} \tag{1}$$

Here $s_j^{\min}$ ($s_j^{\max}$) is the lower (upper) bound of feature j and Eq. (1) is equivalent to $S = D \cap \otimes[s_j^{\min}, s_j^{\max}]$. When $s_j^{\min} \leqslant \min x_j$, the condition $s_j^{\min} \leqslant x_j \leqslant s_j^{\min}$ is equivalent to $x_j \leqslant s_j^{\min}$. When both $s_j^{\min}$ and $s_j^{\min}$ are out of data distribution, Eq. (1) indicates that there is no rule for feature j. For categorical features, this formulation is also applicable with one-hot encoding of the features. In this case, the selection criteria for S corresponds to requiring that the sample belongs to a certain subset of the categories (e.g. eye color is black or blue, but not brown).

By definition, datapoints are selected only if all the rules are satisfied. Then the probability of data x being selected is

$$p_s(x) = \prod_{j=1}^{d} p_{s_j}(x_j) \tag{2}$$

Soft binning function Most standard cohort selection rules are exactly described by Eq. (1). However the hard binning function is non-differentiable and cannot be directly learned in an end-to-end algorithm.

To derive a differentiable approximation for Eq. (1), we resort to soft binning function (Dougherty et al., 1995), which has proved great success in approximating hard binning with different implementations such as neural networks (Revaud et al., 2019) and differentiable decision trees (Yang et al., 2018).

$$p_{s_j}(x_j) = \mathrm{softmax}((x_j c + b_j)/\tau)_2 \tag{3}$$

$$b_j = [0, -s_j^{\min}, -s_j^{\min} - s_j^{\max}]$$

where $c = [1, 2, 3]$ is a constant vector, the subscript $(\cdot)_2$ denotes the value of second dimension and $\tau > 0$ is the temperature factor. In the original soft binning formulation, $\mathrm{softmax}((x_j c + b_j)/\tau)$ provides the probability that x_j falls into the three bins $(-\infty, s_j^{\min})$, $[s_j^{\min}, s_j^{\max}]$ and $(s_j^{\max}, \infty)$. Here we only care about the probability that x_j lies in the second bin $[s_j^{\min}, s_j^{\max}]$, thus only the second dimension of the soft binning output is used. In the limit $\tau \to 0$, Eq. (3) smoothly approaches Eq. (1).

Together with Eq. (2), we are able to learn the cohort selection criteria by simply optimizing over s, the bounding thresholds of each feature. During the training, we always initialize s by $s_j^{\min} = \min x_j$ and $s_j^{\max} = \max x_j$. That is, we let the algorithm to learn shrinking the selection range for each feature. On the test dataset D_{test}, the subset is

selected using the hard binning function with the learned bounds, i.e., $S_{\text{test}} = D_{\text{test}} \cap \otimes[s_j^{\min}, s_j^{\max}]$, which is highly interpretable.

2.3. Formulation of Joint Objective

Our task is to learn selection criteria such that the given model $f_\theta(\cdot)$ achieves best performance when fitted on the selected cohort.

In other words, when the model $f_\theta(\cdot)$ is fitted on the subset defined by the selection probability $p_s(x)$, the expected loss is minimized

$$L(\theta, s) = \frac{\mathbb{E}[p_s(x) CE(y; f_\theta(x))]}{\mathbb{E}[p_s(x)]} \tag{4}$$

By optimizing θ and s with loss $L(\theta, s)$, we are able to identify a subset where the averaged classification loss is smallest for model $f_\theta(\cdot)$. Here we overload notation and let $s = \{s_j^{\min}, s_j^{\max}\}$ be the set of upper and lower bounds. The algorithm is end-to-end and produces the best cohort selector and fitted coefficients θ at the same time.

Size vs. performance The formulation of loss in Eq. (4) mainly considers the averaged performance on the selected subset, without any preference about the subset size. In real applications, a larger subset is always more desirable. For generality, we propose an option to add a regularization term to favor a larger selected cohort:

$$L^{\text{reg}}(\theta, s) = L(\theta, s) + \text{reg} \cdot \mathbb{E}[||p_s(x) - 1||_2^2],$$

where reg is the weight for the regularization term, which encourages the model to select as many data points as possible (i.e. p_s close to 1).

In this paper, in order to present the most consistent results across all the experiments and datasets, we set reg to be zero in all of our experiments. We note that the results are robust for small reg. In practice, reg can be chosen based on the users' preference over the subset size or averaged classification performance, and it can be determined via cross-validation.

2.4. Concrete Cohort Selector

For interpretation purpose, it is often favorable to have as few selection rules as possible. For example, in clinical trial design, there are efforts to reduce the number of eligibility criteria used for cohort selection (Meinert, 2012). For the previously defined cohort selector (Eqn. 3), all of the features could be used to construct selection rules. While this formulation enables large flexibility, it is also of great interest to identify only a small amount of rules that are sufficient to define a good cohort.

Algorithm 1 Data Sculpting

Input: training dataset $X \in \mathbb{R}^{n \times d}$ and $y \in \mathbb{R}^n$, total training epochs N, selector temperature τ, number of epochs N, learning rate λ, classification function $f_\theta(\cdot)$, whether to use concrete selector concrete, number of concrete dimensions d_c, initial concrete temperature T_0 and finial concrete temperature T_N.

Set $c = [1, 2, 3]$ and define equals $X_{ij} := X_j^{(i)}$.
for $j = 1$ to d **do**
  Initialize the bounding thresholds as $s_j^{\min} = \min_i X_j^{(i)}$ and $s_j^{\max} = \max_i X_j^{(i)}$.
end for
for $t = 1$ to N **do**
  for $j = 1$ to d **do**
    Set $b = [0, -s_j^{\min}, -s_j^{\min} - s_j^{\max}]$
    $\boldsymbol{\pi}_j = \text{softmax}((X_j c + b)/\tau)_2$
  end for
  if concrete **then**
    Adjust concrete temp. $T = T_0(T_N/T_0)^{t/N}$
    for $i = 1$ to d_c **do**
      Sample $\boldsymbol{m}^{(i)} \sim \text{Concrete}(\boldsymbol{\alpha}^{(i)}, T)$
    end for
    Set $\boldsymbol{\pi} = \boldsymbol{\pi} \cdot \boldsymbol{m}^T$
  end if
  Selection probability $p = \prod_j \boldsymbol{\pi}_j$
  Loss $L = \sum_i (p_i \cdot CE(y_i; f_\theta(X^{(i)})))/\sum_i p_i$
  Update $s := s - \lambda \nabla_s L$ and $\theta := \theta - \lambda \nabla_\theta L$
  if concrete **then**
    Update $\boldsymbol{\alpha} := \boldsymbol{\alpha} - \lambda \nabla_{\boldsymbol{\alpha}} L$
  end if
end for
if concrete **then**
  The selected feature index ids $= \arg\max_j \boldsymbol{\alpha}_j$
end if

Output: Fitted coefficient θ, bounds s and (if concrete) selected feature index ids.

In this section, we provide an extra module—the Concrete cohort selector—for Data Sculpting which enables the algorithm to learn to use a sparse number of features for cohort selection choice to model the selection probability. We note that the Concrete selector is a modular and optional part of the overall Data Sculpting algorithm; user can simply not use it and the rest of the algorithm does not change.

We build the concrete cohort selector based on Concrete random variables (Maddison et al., 2016; Jang et al., 2016), which is a powerful tool for feature selection (Balın et al., 2019).

Given the number of selected features $d_c < d$, the concrete random variable $m^{(i)} \in \mathbb{R}^d$, $i = 1, ..., d_c$ follows the Concrete distribution $m^{(i)} \sim \text{Concrete}(\alpha^{(i)}, T)$ defined as

$$m_j^{(i)} = \frac{\exp((\log \alpha_j^{(i)} + g_j)/T)}{\sum_{k=1}^{d} \exp((\log \alpha^k + g_k)/T}$$

where $\alpha \in \mathbb{R}^{d_c \times d}$ is the concrete parameter, T is the concrete temperature and g is sampled from Gumbel distribution (Gumbel & Lieblein, 1954). With this reparametrization trick, the concrete random variable becomes differentiable with respect to the concrete parameter α (Kingma & Welling, 2013). When $T \to 0$, the concrete variable tends to the discrete distribution and $m_j^{(i)} = 1$ appears with probability $\alpha_j^{(i)} / \sum_k \alpha_k^{(i)}$.

We construct d_c concrete features from the original d features in the following way. For the ith concrete feature, we sample the corresponding concrete random variable $m^{(i)} \in \mathbb{R}^d$. The probability that data x satisfies the rules for this concrete feature is $\sum_{j=1}^{d} m_j^{(i)} p_{s_j}(x_j)$. Similarly to Eq. (2), datapoints are selected only if the rules for all the concrete features are satisified. The selection probability is updated as

$$p_s(x) = \prod_{i=1}^{d_c} \left(\sum_{j=1}^{d} m_j^{(i)} p_{s_j}(x_j) \right)$$

During test time, we select d_c features by the operation $\arg\max_j \alpha_j$. The user can specify the number of features d_c she would like to be involved in feature selection.

Annealing schedule To encourage the exploration of different combinations of features, the concrete temperature T is set to be high initially. During the training, we follow an annealing schedule for T as (Balın et al., 2019) for discretized feature selection. More specifically, the concrete temperature T experiences a first-order exponential decay at epoch t as $T(t) = T_0(T_N/T_0)^{t/N}$, where T_0 (T_N) is the initial (final) concrete temperature.

2.5. Experiment Setup.

The pseudocode for Data Sculpting is in Algorithm 1. We implemented it in Pytorch (Paszke et al., 2017) and trained with GPU. For all of the experiments, Adam optimizer is used with learning rate of 10^{-3}. We run each experiment for 1,000 epoches with batch size 1,000. The selector temperature τ is 0.1 and the initial (final) concrete temperature T_0 (T_N) is set to be 10 (0.1).

In the experiments, we randomly split the datasets into training, validation and test set, with ratio 50%, 20% and 30%. We fit the model on training set and report its performance on test set. For each experiment, we run the algorithm for five times with different initialization and use the validation set for early stopping and report the best result.

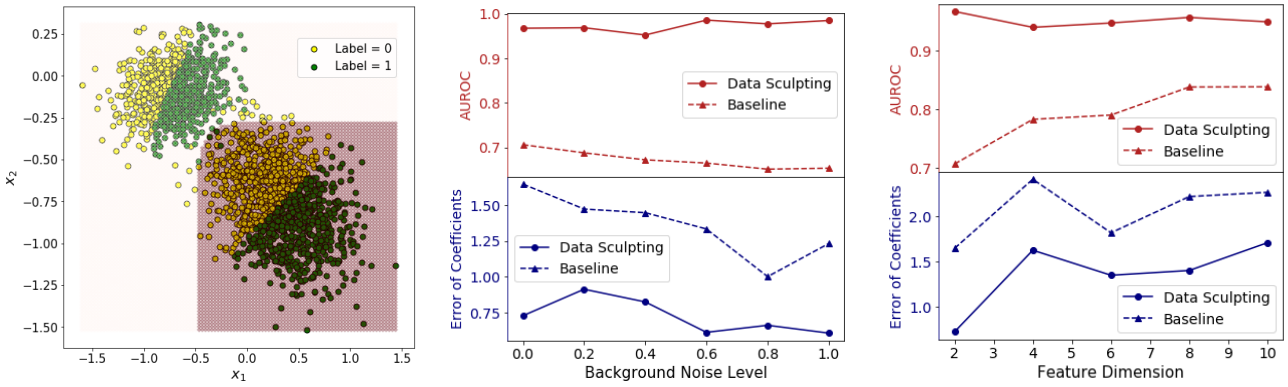
3. Benchmark on Synthetic Data

We first experiment with synthetic data to benchmark the performance of Data Sculpting, before moving onto complex real-world datasets. The benefit of using the synthetic data first is that we know the ground truth parameters and cohort and can evaluate how well different approaches recover the ground truth. In real data, such ground truth is typically not available.

3.1. Cohort Selection

We emulate the situation where we have a heterogeneous dataset and the features contribute to the outcome differently across subsets of the data. Our task is to identify a large subset where the statistical model (logistic regression) is a good fit and the signal is clean. In light of this, we generate two Gaussians with random centers and covariance matrices. The association between features and label follows the logistic regression function with randomly generated coefficients. The true coefficients for the two Gaussians are different and are generated separately to emulate the heterogeneous property. The synthetic dataset is of size 4,000 and further splitted into into training, validation and test set; see the experiment setup in Sec. 2.5. An illustrative example is given in Figure 1a. In this case, Data Sculpting correctly captures the underlying cohort, as indicated by the shade area in Figure 1a. In this section, we implement Data Sculpting without the additional concrete feature selection. The concrete cohort selector will be benchmarked in the next section.

Evaluation metrics We use five metrics to evaluate the performance of the models. 1) We compute the proportional of the data selected by Data Sculpting—this is denoted as the cohort size. 2) We compute the cohort accuracy: how accurate our algorithm identifies the cohort. Suppose the set of selected data index is I and the sets of data index for two correct cohorts are I_0^1 and I_0^2. Then the index set for the whole dataset is $I_0^1 \cup I_0^1$. The cohort accuracy is defined as $\max_i |I \cap I_0^i| / |I \cup I_0^i|$, which describes the overlap between learned cohort with the true cohort (the largest overlap for the two true cohorts). 3) On the selected cohort, we have the learned coefficients θ of the logistic regression fit on this subset, which are the odds ratios. We compute the relative error of θ compared to the ground truth θ_0: $|\theta - \theta_0|/|\theta_0|$. Both θ and θ_0 are vectors in $\mathbb{R}^d$. Since there are two subsets in the data and they each have a different θ_0. In our task, we only need the learned coefficients to capture any of the two θ_0 and the error of coefficients is reported as the smallest error for two θ_0 of the two cohorts. 4) and 5) Finally we compute the accuracy and AUROC of the logistic regression trained on the selected cohort as the last two metrics. For comparison, the coefficients, accuracy and AUROC are also computed by the baseline — logistic



(a) (b) (c)

Figure 1. (a) Illustrative example of a synthetic dataset. The shaded region is the selected cohort by Data Sculpting. (b) Performance of Data Sculpting and baseline (logistic regression on the whole dataset) for different background noise level on the 2-dimensional synthetic dataset. (c) Peformance of Data Sculpting and baseline vs. feature dimension of the synthetic dataset. (b-c) The two Gaussians have equal size and the results are averaged on 20 independent synthetic datasets with randomly generated coefficients. We fit the model on training set and report its performance on test set.

Table 1. Performance of Data Sculpting (DS) and baseline (logistic regression model on the whole data) on test set for synthetic data with different cohort size. Here the cohort size is scaled by the whole dataset size. The feature dimension is 2 and the results are averaged on 20 independent synthetic datasets with randomly generated coeffients.

Cohort Size		DS Cohort	Error of Coefficients		AUROC		Accuracy	
True	DS	Accuracy	Baseline	DS	Baseline	DS	Baseline	DS
0.5	0.62	78.2%	1.65	**0.73**	70.6%	**96.7%**	60.5%	**73.8%**
0.6	0.61	86.6%	1.54	**0.72**	71.8%	**98.0%**	61.8%	**72.2%**
0.7	0.73	85.7%	1.20	**0.72**	76.3%	**96.7%**	65.4%	**76.6%**
0.8	0.79	90.3%	1.59	**0.86**	83.7%	**97.8%**	72.9%	**77.7%**
0.9	0.91	94.8%	1.54	**0.71**	90.4%	**98.1%**	**82.0%**	78.7%

regression model trained on the whole data. The coefficients, accuracy and AUROC are computed using the hold-out validation (test) data set (we pass the test data through the trained cohort selector).

Different cohort size We first evaluate how Data Sculpting and baseline (logistic regression model trained on the whole data) perform with different underlying (true) cohort size, presented by Table 1. As there are two cohorts in the datasets, we report the size of the largest cohort (scaled by the whole dataset size). Data Sculpting correctly captures the underlying cohort in the data with the high cohort accuracy. The learned cohort is always in the similar size as the largest true one. Moreover across all the cohort sizes, Data Sculpting consistently outperforms the baseline in all of the metrics (Table 1). With the increase of true cohort size, the baseline's performance improves as expected, since the true cohort becomes very close to the entire data set, which is what the baseline logistic regression is trained on.

Different noise level Here we emulate the situation where the dataset has a cohort with clean signal and other cohorts are noisy. Here we generate the two Gaussians with the same size and randomly flip the labels of one Gaussian with certain probability p_{flip}. When $p_{\text{flip}} = 0.5$, the labels of that cohort are generated completely at random. Here we view the noisy cohort as the background for the clean one and define the background noise level as $2p_{\text{flip}}$. Data Sculpting demonstrates substantial improvement over the baseline for different noise level, shown by Fig. 1b. In particular, when the background noise level is 1.0, the labels of one Gaussian are completely generated at random. That is, Data Sculpting is also able to identify the subset from completely noisy background and capture the clean signal.

Different feature dimension Finally, we evaluated the performance of Data Sculpting and baseline for varying number of dimensions in the dataset. The improved performance of Data Sculpting over baseline is consistent in all of the settings (Figure 1c). For data with higher dimension, we also have the option the select core features with concrete

Table 2. Peformance of Data Sculpting with concrete cohort selector on test set for synthetic dataset. Here the total feature dimension is 20 and the results are averaged on 20 independent synthetic datasets with randomly generated coeffients.

#Core Features	Accuracy of Core Feature Identification			AUROC		Error of Coefficients	
	Random	Lasso	Data Sculpting	Baseline	Data Sculpting	Baseline	Data Sculpting
2	10%	16.0%	**27.5%**	82.4%	**87.6%**	0.93	**0.90**
5	25%	23.5%	**36.0%**	85.0%	**85.5%**	1.00	**0.96**

cohort selector, which will be addressed in the next section.

3.2. Concrete Data Selector

We further emulate the situation where a dataset contains one cohort where the statistical model is strong and that cohort is defined by only a subset of all the features. To do this, we generate one Gaussian with random center and covariance matrix. By randomly select d_c dimensions and set random thresholds on the selected features, we obtain the underlying cohort. On that cohort, the association between all the features and outcome follows the logistic regression function with randomly generated coefficients. The labels are generated randomly on the rest of the data. The size of the Gaussian is 2,000 data points and the total feature dimension is 20. We fit the model on training set and report its performance on test set following the experiment setup in Sec. 2.5.

Core feature identification Here Data Sculpting with concrete data selector is used to identify core features. We generate synthetic data with 2 and 5 core features and use Data Sculpting with the same concrete dimension, as present in Table 2. We compare Data Sculpting with Lasso-based feature selection method. For the comparison method, the features and label are fit into a Lasso regression model. By tuning the L1 regularizer strength, we obtain d_c selected features.

We first evaluate how accurately each method identifies the core features. The accuracy of Lasso is not too different from simply randomly choosing features—Lasso is slightly more accurate than random when there are two core features and is less accurate with five core features. Data Sculpting is significantly more accurate than either methods. Like before, we can also evaluate the accuracy of the logistic regression classifier when trained on the selected cohort—Data Sculpting also performs better than regression on the whole dataset (baseline). The estimated parameters from Data Sculpting is also closer to the ground truth.

4. Experiments on Real-world Data

In this section, we implement Data Sculpting to learn interpretable subset of data and report cleaner fitted coefficients on real-world dataset.

4.1. Dataset Description

We evaluate our method on five medical and social science datasets, on which odds ratio is an important objective in the original publications.

Congo Fever Dataset consists of the Typhoid fever outbreak data in the Democratic Republic of Congo with 320 cases and 640 controls (Brainard et al., 2018). The study reports the odds ratios predicting the effect of demographic, environmental and exposure characteristics on fever outbreak with logistic regression model.

India HIV Dataset consists of the pornography and HIV-related sexual behaviours data for 11,219 male migrants in India (Mahapatra & Saggurti, 2014). The study uses logistic regression model to analyze the odds ratios for socio-demographic and migration factors in exposure to pornography among male Indian migrant workers.

India Distress Dataset consists of the distress health financing data in India with 42,869 hospitalization cases (Kastor & Mohanty, 2018). The study presents the odds ratio as the result of logistic regression of incurring distress financing hospitalization for different demographic and disease covariates.

Zambia Perception Dataset consists of the perceptions of male circumcision (MC) data from 934 women in Zambia (Haberland et al., 2016). The study reports the odds ratios from logistic regression model to capture the relation between socio-demographic characteristics and the awareness of MC.

USA Obesity Dataset consists of dietary patterns and the obesity data for 13,160 adults in the United States (Cohen et al., 2018). The study estimates the odds ratios of obesity from demographic, physical activity and dietary patterns.

Following the experiment setup in Sec. 2.5, we split the real-world datasets, fit the model on the training set and report the performance on the set set. The categorical variables are one-hot encoded. For interpretation purpose, we we use concrete data selector with concrete dimension of two — Data Sculpting used Concrete selector to select two features and learned thresholds on these two features to select the cohort.

Table 3. Performance of Data Sculpting with concrete data selector and baseline (logistic regression model on whole data) on test set for real-world datasets. Here dim stands for the feature dimension and the cohort size is the subset size identified by Data Sculpting (scaled by the total data size). The difference of OR is the L2 distance between OR from Data Sculpting and OR from baseline. For the selection criteria, Q402 indicates the frequency of vising native place.

Dataset	dim	Accuracy		AUROC		Data Sculpting (DS)		
		Baseline	DS	Baseline	DS	Learned Selection Criteria	Cohort Size	Difference of OR
Congo Fever	11	59.9%	**86.4%**	64.9%	**80.7%**	Plate sharing: Regularly Occupation: Not Labourer	0.12	1.12
India HIV	17	59.9%	**63.2%**	63.7%	**66.9%**	Age: 25-30 Q402: Not Many Times	0.17	1.04
India Distress	26	61.8%	**66.4%**	66.5%	**69.6%**	MPCE Tertile: Rich Disease: Not Injury	0.25	0.43
Zambia Perception	16	55.4%	**62.5%**	59.4%	**63.0%**	Ever married: No Religion: Christian	0.38	1.46
USA Obesity	9	62.7%	**69.9%**	66.7%	**70.7%**	FairPoorHealth: 1 Sex: Male	0.17	0.30

4.2. Performance Evaluation

On all five real-world datasets, Data Sculpting substantially outperforms the standard logistic regression baseline, which was the model used in the original publications (Table 3). On the selected cohort, logistic regression is much more accurate (and have higher AUROC) compared to the logistic regression model trained on the whole dataset. As before, the accuracy and AUROC are evaluated using a hold-out test set.

Odds ratios. We report the relative difference in the odds ratios (OR) in the selected cohort vs. the full data. The ORs can be substantially different, indicating that the statistical model fitted globally can be confused by the heterogeneity in the data. In the original publications of these datasets, the odds ratios are only computed globally and directly used to explain the whole dataset. In order to capture the true associations between features and outcome, it is necessary to identify coherent cohort in these situations. Here the relative error of coefficient is no longer reported, due to the fact that the underlying truth is unknown for real-world data.

Interpretable selection criteria. The cohort selection rules learned by Data Sculpting are highly interpretable, as shown in Table 3. Substantial improvements are achieved with only two simple rules which are easy to use in practice. Moreover, interpreting the criteria by themselves could also be of interest for better understanding of the data.

5. Discussion

In this paper, we formalize the problem of cohort selection and propose a new algorithm, Data Sculpting, that jointly optimizes cohort selection while also fitting the statistical model on the selected cohort. This enables the cohort selection to be data driven and targeted for the specific downstream statistical modeling (e.g. logistic regression). Once the selection rules are learned, we apply it on the hold-out validation data and make the statistical inference on this hold-out set. This assures that there is no issue with selective inference and that the fitted model parameters (e.g. odds ratios) are statistically valid (Russo & Zou, 2019).

An important consideration in cohort selection is that the selection rules be interpretable and sparse. To achieve this goal, Data Sculpting uses simple threshold selection rules combined with sparse feature selection to ensure that only a small number of thresholding is needed to determine the cohort. The thresholds and features are learned in the joint optimization by leveraging the Concrete parametrization. More general selection rules could be implemented in the same optimization framework as outlined in Algorithm 1. We can replace threshold rules with differentiable decision trees, for example, or even a neural network serve as the selector. We experimented with using a neural network to parametrize the selector in both real and synthetic data. The accuracy of final logistic regression is comparable when the cohort is selected using a neural network or the simpler threshold rules in Algorithm 1. And both are much better than fitting on the entire dataset. This suggests that in many real datasets, simple thresholding is sufficient in sculpting the cohort.

References

Balın, M. F., Abid, A., and Zou, J. Concrete autoencoders: Differentiable feature selection and reconstruction. In *International Conference on Machine Learning*, pp. 444–453, 2019.

Bland, J. M. and Altman, D. G. The odds ratio. *Bmj*, 320 (7247):1468, 2000.

Brainard, J., D'hondt, R., Ali, E., Van den Bergh, R., De Weggheleire, A., Baudot, Y., Patigny, F., Lambert, V., Zachariah, R., Maes, P., et al. Typhoid fever outbreak in the democratic republic of congo: Case control and ecological study. *PLoS neglected tropical diseases*, 12 (10):e0006795, 2018.

Cohen, S. A., Greaney, M. L., and Sabik, N. J. Assessment of dietary patterns, physical activity and obesity from a national survey: Rural-urban health disparities in older adults. *PloS one*, 13(12), 2018.

Cousineau, D. and Chartier, S. Outliers detection and treatment: a review. *International Journal of Psychological Research*, 3(1):58–67, 2010.

Dougherty, J., Kohavi, R., and Sahami, M. Supervised and unsupervised discretization of continuous features. In *Machine Learning Proceedings 1995*, pp. 194–202. Elsevier, 1995.

Finlay, B. and Agresti, A. *Statistical methods for the social sciences*. Dellen, 1986.

Friedman, J., Hastie, T., and Tibshirani, R. *The elements of statistical learning*, volume 1. Springer series in statistics New York, 2001.

Gimenez, J. R., Ghorbani, A., and Zou, J. Knockoffs for the mass: new feature importance statistics with false discovery guarantees. *arXiv preprint arXiv:1807.06214*, 2018.

Gumbel, E. J. and Lieblein, J. Some applications of extreme-value methods. *The American Statistician*, 8(5):14–17, 1954.

Haberland, N. A., Kelly, C. A., Mulenga, D. M., Mensch, B. S., and Hewett, P. C. Women's perceptions and misperceptions of male circumcision: A mixed methods study in zambia. *PloS one*, 11(3), 2016.

Huber, P. J. *Robust statistics*, volume 523. John Wiley & Sons, 2004.

Jang, E., Gu, S., and Poole, B. Categorical reparameterization with gumbel-softmax. *arXiv preprint arXiv:1611.01144*, 2016.

Kastor, A. and Mohanty, S. K. Disease-specific out-of-pocket and catastrophic health expenditure on hospitalization in india: Do indian households face distress health financing? *PLoS One*, 13(5), 2018.

Kingma, D. P. and Welling, M. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

Maddison, C. J., Mnih, A., and Teh, Y. W. The concrete distribution: A continuous relaxation of discrete random variables. *arXiv preprint arXiv:1611.00712*, 2016.

Mahapatra, B. and Saggurti, N. Exposure to pornographic videos and its effect on hiv-related sexual risk behaviours among male migrant workers in southern india. *PloS one*, 9(11), 2014.

Meinert, C. L. *ClinicalTrials: design, conduct and analysis*, volume 39. OUP USA, 2012.

Paszke, A., Gross, S., Chintala, S., Chanan, G., Yang, E., DeVito, Z., Lin, Z., Desmaison, A., Antiga, L., and Lerer, A. Automatic differentiation in pytorch. 2017.

Revaud, J., Almazán, J., Rezende, R. S., and Souza, C. R. d. Learning with average precision: Training image retrieval with a listwise loss. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 5107–5116, 2019.

Russo, D. and Zou, J. How much does your data exploration overfit? controlling bias via information usage. *IEEE Transactions on Information Theory*, 66(1):302–323, 2019.

Tibshirani, R. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society: Series B (Methodological)*, 58(1):267–288, 1996.

Yang, Y., Morillo, I. G., and Hospedales, T. M. Deep neural decision trees. *arXiv preprint arXiv:1806.06988*, 2018.

Zou, J., Lippert, C., Heckerman, D., Aryee, M., and Listgarten, J. Epigenome-wide association studies without the need for cell-type composition. *Nature methods*, 11 (3):309–311, 2014.