Optimizing Data Collection for Machine Learning

Rafid Mahmood1  James Lucas1  Jose M. Alvarez1  Sanja Fidler1,2,3  Marc T. Law1
1NVIDIA  2University of Toronto  3Vector Institute
{rmahmood, jalucas, josea, sfidler, marcl}@nvidia.com

Abstract

Modern deep learning systems require huge data sets to achieve impressive performance, but there is little guidance on how much or what kind of data to collect. Over-collecting data incurs unnecessary present costs, while under-collecting may incur future costs and delay workflows. We propose a new paradigm for modeling the data collection workflow as a formal optimal data collection problem that allows designers to specify performance targets, collection costs, a time horizon, and penalties for failing to meet the targets. Additionally, this formulation generalizes to tasks requiring multiple data sources, such as labeled and unlabeled data used in semi-supervised learning. To solve our problem, we develop Learn-Optimize-Collect (LOC), which minimizes expected future collection costs. Finally, we numerically compare our framework to the conventional baseline of estimating data requirements by extrapolating from neural scaling laws. We significantly reduce the risks of failing to meet desired performance targets on several classification, segmentation, and detection tasks, while maintaining low total collection costs.

1 Introduction

When deploying a deep learning model in an industrial application, designers often mandate that the model must meet a pre-determined baseline performance, such as a target metric over a validation data set. For example, an object detector may require a certain minimum mean average precision before being deployed in a safety-critical setting. One of the most effective ways of meeting target performances is by collecting more training data for a given model.

Determining how much data is needed to meet performance targets can impact costs and development delays. Overestimating the data requirement incurs excess costs from collection, cleaning, and annotation. For instance, annotating segmentation masks for a driving data set takes between 15 to 40 seconds per object. For 100,000 images the annotation could require between 170 and 460 days-equivalent of time [1,2]. On the other hand, collecting too little data may incur future costs and workflow delays from having to collect more later. For example, in medical imaging applications, this means further clinical data acquisition rounds that require expensive clinician time. In the worst case, designers may even realize that a project is infeasible only after collecting insufficient data.

The growing literature on sample complexity in machine learning has identified neural scaling laws that scale model performance with data set sizes according to power laws [3,10]. For instance, Rosenfeld et al. [6] fit power law functions on the performance statistics of small data sets to extrapolate the learning curve with more data. In contrast, Mahmood et al. [2] consider estimating data requirements and show that even small errors in a power law model of the learning curve can translate to massively over- or underestimating how much data is needed. Beyond this, different data sources have different costs and scale differently with performance [11,12]. For example, although unlabeled data may be easier to collect than labeled data, some semi-supervised learning tasks may need an order of magnitude more unlabeled data to match the performance of a small labeled set. Thus, collecting more data based only on estimation will fail to capture uncertainty and collection costs.

Figure 1: In the optimal data collection problem, we iteratively determine the amount of data that we should have, pay to collect the additional data, and then re-evaluate our model. Our approach, Learn-Optimize-Collect, optimizes for the minimum amount of data \( q_t^* \) to collect.

In this paper, we propose a new paradigm for modeling the data collection workflow as an *optimal data collection problem*. Here, a designer must minimize the cost of collecting enough data to obtain a model capable of a desired performance score. They have multiple collection rounds, where after each round, they re-evaluate the model and decide how much more data to order. The data has per-sample costs and moreover, the designer pays a penalty if they fail to meet the target score within a finite horizon. Using this formal framework, we develop an optimization approach for minimizing the expected future collection costs and show that this problem can be optimized in each collection round via gradient descent. Furthermore, our optimization problem immediately generalizes to decisions over multiple data sources (e.g., unlabeled, long-tail, cross-domain, synthetic) that have different costs and impacts on performance. Finally, we demonstrate the value of optimization over naïvely estimating data set requirements (e.g., \([2]\)) for several machine learning tasks and data sets.

Our contributions are as follows. (1) We propose the optimal data collection problem in machine learning, which formalizes data collection workflows. (2) We introduce Learn-Optimize-Collect (LOC), a learning-and-optimizing framework that minimizes future collection costs, can be solved via gradient descent, and has analytic solutions in some settings. (3) We generalize the data collection problem and LOC to a multi-variate setting where different types of data have different costs. To the best of our knowledge, this is the first exploration of data collection with general multiple data sets in machine learning, covering for example, semi-supervised and long-tail learning. (4) We perform experiments over classification, segmentation, and detection tasks to show, on average, approximately a \(2 \times \) reduction in the chances of failing to meet performance targets, versus estimation baselines.

2 Related work

**Neural Scaling Laws.** According to the neural scaling law literature, the performance of a model on a validation set scales with the size of the training data set \( q \) via a power law \( V \propto q^{\theta} \). \( [5, 6, 8–10, 15–19] \). Hestness et al. \([5]\) observe this property over vision, language, and audio tasks, Bahri et al. \([9]\) develop a theoretical relationship under assumptions on over-parametrization and the Lipschitz continuity of the loss, model, and data, and Rosenfeld et al. \([6]\) estimate power laws using smaller data sets and models to extrapolate future performance. Multi-variate scaling laws have also been considered for some specific tasks, for example in transfer learning from synthetic to real data sets \([11]\). Finally, Mahmood et al. \([2]\) explore data collection by estimating the minimum amount of data needed to meet a given target performance over multiple rounds. Our paper extends these prior studies by developing an optimization problem to minimize the expected total cost of data collected. Specifically, we incorporate the uncertainty in any regression estimate of data requirements and further generalize to multiple data sources with different costs.

**Active Learning.** In active learning, a model sequentially collects data by selecting new subsets of an unlabeled data pool to label under a pre-determined labeling budget that replenishes after each round \([20–24]\). In contrast, our work focuses on systematically determining an optimal collection budget. After determining how much data to collect, we can use active learning techniques to collect the desired amount of data.

**Statistical Learning Theory.** Theoretical analysis of the sample complexity of machine learning models is typically only tight asymptotically, but some recent work have empirically analyzed these
relationships [25, 26]. Particularly, Bisla et al. [10] study generalization bounds for deep neural networks, provide empirical validation, and suggest using them to estimate data requirements. In contrast, our paper formally explores the consequences of collection costs on data requirements.

**Optimal Experiment Design.** The topic of how to collect data, select samples, and design scientific experiments or controlled trials is well-studied in econometrics [27, 29]. For example, Bertsimas et al. [30] optimize the assignment of samples into control and trial groups to minimize inter-group variances. Most recently, Carneiro et al. [31] optimize how many samples and covariates to collect in a statistical experiment by minimizing a treatment effect estimation error or maximizing t-test power. However, our focus on industrial machine learning applications differs from experiment design by having target performance metrics and continual rounds of collection and modeling.

### 3 Main Problem

In this section, we give a motivating example before introducing the formal data collection problem. We include a table of notation in Appendix A.

**Motivating Example.** A startup is developing an object detector for use in autonomous vehicles within the next \(T = 5\) years. Their model must achieve a mean Average Precision greater than \(V^* = 95\%\) on a pre-determined validation set or else they will lose an expected profit of \(P = 1,000,000\). Collecting training data requires employing drivers to record videos and annotators to label the data, where the marginal cost of obtaining each image is approximately \(c = \$1\). In order to manage annual finances, the startup must plan how much data to collect at the beginning of each year.

Let \(z \sim p(z)\) be data drawn from a distribution \(p\). For instance, \(z := (x, y)\) may correspond to images \(x\) and labels \(y\). Consider a prediction problem for which we train a model with a data set \(D\) of points sampled from \(p(z)\). Let \(V(D)\) be a score function evaluating the model trained on \(D\).

**Optimal Data Collection.** We possess an initial data set \(D_{q_0} := \{z_i\}_{i=1}^{q_0}\) of \(q_0\) points; we omit the subscript on \(D\) referring to its size when it is obvious. Our problem is defined by a target score \(V^* > V(D_{q_0})\), a cost-per-sample \(c\) of collection, a horizon of \(T\) rounds, and a penalty \(P\) for failure. At the end of each round \(t \in \{1, \ldots, T\}\), let \(q_t\) be the current amount of data collected. Our goal is to minimize the total collection cost while building a model that can achieve the target score:

\[
\min_{q_1, \ldots, q_T} \ c(q_T - q_0) + P \mathbb{I}\{V(D_{q_T}) < V^*\} \quad \text{s.t.} \quad q_0 \leq q_1 \leq \cdots \leq q_T
\]

\[
= \min_{q_1, \ldots, q_T} \ c \sum_{t=1}^{T} (q_t - q_{t-1}) + P \mathbb{I}\{V(D_{q_T}) < V^*\} \quad \text{s.t.} \quad q_0 \leq q_1 \leq \cdots \leq q_T \tag{1}
\]

The collection cost is measured by the difference in data set size between the final and the 0-th round \(c(q_T - q_0) = c \sum_{t=1}^{T} (q_t - q_{t-1})\). Because we collect data iteratively over multiple rounds (see Figure 1), we break (1) into the sum of differences per round. Specifically in each round, we

1. Decide to grow the data set to \(q_t \geq q_{t-1}\) points by sampling \(\hat{D} := \{\tilde{z}_i\}_{i=1}^{q_t-q_{t-1}} \sim p(z)\). Pay a cost \(c(q_t - q_{t-1})\) and update \(D \leftarrow D \cup \hat{D}\).
2. Train the model and evaluate the score. If \(V(D) \geq V^*\), then terminate.
3. If \(t = T\), then pay the penalty \(P\) and terminate. Otherwise, repeat for the next round.

The model score typically increases monotonically with data set size [5, 6]. This means that the minimum cost strategy for (1) is to collect just enough data such that \(V(D_{q_T}) = V^*\). We can estimate this minimum data requirement by modeling the score function as a stochastic process. Let \(V_q := V(D_q)\) and let \(\{V_q\}_{q \in \mathbb{Z}_+}\) be a stochastic process whose indices represent training set sizes in different rounds. Then, collecting data in each round yields a sequence of subsampled data sets \(D_{q_t} \subset D_{q_{t-1}}\) and their performances \(V(D_{q_t})\). The minimum data requirement is the stopping time

\[
D^* := \arg \min_q \{ q \mid V_q \geq V^* \} \tag{2}
\]

which is a random variable giving the first time that we pass the target. Note that \(q^*_1 = \cdots = q^*_T = D^*\) is a minimum cost solution to the optimal data collection problem, incurring a total cost \(c(D^* - q_0)\).

---

\[\text{We assume that } c(D^* - q_0) < P, \text{ since otherwise the optimal strategy would be to collect no data.}\]
Estimating $D^*$ using past observations of the learning curve is difficult since we have only $T$ rounds. Further, Mahmood et al. [2] empirically show that small errors in fitting the learning curve can cause massive over- or under-collection. Thus, robust policies must capture the uncertainty of estimation.

## 4 Learn-Optimize-Collect (LOC)

Our solution approach, which we refer to as Learn-Optimize-Collect (LOC), minimizes the total collection cost while incorporating the uncertainty of estimating $D^*$. Although $D^*$ is a discrete random variable, it is realized typically on the order of thousands or greater. To simplify our problem and ensure differentiability, we assume that $D^*$ is continuous and has a well-defined density.

**Assumption 1.** The random variable $D^*$ is absolutely continuous and has a cumulative density function (CDF) $F(q)$ and probability density function (PDF) $f(q) := dF(q)/dq$.

In Section 4.1 we first develop an optimization model when given access to the CDF $F(q)$ and PDF $f(q)$. In Section 4.2 we estimate these distributions and combine them with the optimization model. Finally in Section 4.3 we delineate our optimization approach from prior regression methods.

### 4.1 Optimization Model

We propose an optimization problem that for any $t$, can simultaneously solve for the optimal amounts of data to collect $q_t, \ldots, q_T$ in all future rounds. Consider $t = 1$ and to develop intuition, suppose we know a priori the exact stopping time $D^*$. Then, problem (1) can be re-written as

$$
\min_{q_1, \ldots, q_T} L(q_1, \ldots, q_T; D^*) \quad \text{s.t. } q_0 \leq q_1 \leq \cdots \leq q_T
$$

where the objective function is defined recursively as follows

$$
L(q_1, \ldots, q_T; D^*) := c(q_1 - q_0) + \mathbb{I}\{q_1 < D^*\} \left( c(q_2 - q_1) + \mathbb{I}\{q_2 < D^*\} \left( c(q_3 - q_2) \cdots + \mathbb{I}\{q_{T-1} < D^*\} \left( c(q_T - q_{T-1}) + P \mathbb{I}\{q_T < D^*\} \right) \right) \right) \\
= c \sum_{t=1}^{T} (q_t - q_{t-1}) \mathbb{I}\{q_s < D^*\} + P \prod_{t=1}^{T} \mathbb{I}\{q_s < D^*\} \\
= c \sum_{t=1}^{T} (q_t - q_{t-1}) \mathbb{I}\{q_s < D^*\} + P \mathbb{I}\{q_T < D^*\}.
$$

The objective differs slightly from (1) due to the indicator terms, which ensure that once we collect enough data, we terminate the problem. The second line follows from gathering the terms. The third line follows from observing that $q_1 \leq q_2 \leq \cdots \leq q_T$ are constrained.

In practice, we do not know $D^*$ a priori since it is an unobserved random variable. Instead, suppose we have access to the CDF $F(q)$. Then, we take the expectation over the objective $\mathbb{E}[L(q_1, \ldots, q_T; D^*)]$ to formulate a stochastic optimization problem for determining how much data to collect:

$$
\min_{q_1, \ldots, q_T} c \sum_{t=1}^{T} (q_t - q_{t-1}) (1 - F(q_{t-1})) + P (1 - F(q_T)) \quad \text{s.t. } q_0 \leq q_1 \leq \cdots \leq q_T.
$$

Note that the collection variables should be discrete $q_1, \ldots, q_T \in \mathbb{Z}^+$, but similar to the modeling of $D^*$, we relax the integrality requirement, optimize over continuous variables, and round the final solutions. Furthermore, although problem (4) is constrained, we can re-formulate it with variables $d_t := q_t - q_{t-1}$; this consequently replaces the current constraints with only non-negativity constraints $d_t \geq 0$. Finally due to Assumption 1, problem (4) can be optimized via gradient descent.

### 4.2 Learning and Optimizing the Data Requirement

Solving problem (4) requires access to the true distribution $F(q)$, which we do not have in reality. In each round, given a current training data set $D_{q_t}$ of $q_t$ points, we must estimate these distribution functions $F(q)$ and $f(q)$ and then incorporate them into our optimization problem.
Given a current data set $D_{q_t}$, we may sample an increasing sequence of $R$ subsets $D_{q_t}/R \subset D_{q_t}/R \subset \cdots \subset D_{q_t}$, fit our model to each subset, and compute the scores to obtain a data set of the learning curve $R := \{(q_t/R, V(D_{q_t}/R))\}_{r=1}^{R}$. In order to model the distribution of $D^*$, we can take $B$ bootstrap resamples of $R$ to fit a series of learning functions and obtain corresponding estimates $\{\hat{D}_{r}\}_{r=1}^{B}$. Given a set of estimates of the data requirement, we estimate the PDF via Kernel Density Estimation (KDE). Finally, to fit the CDF, we numerically integrate the PDF.

In our complete framework, LOC, we first estimate $F(q)$ and $f(q)$. We then use these models to solve problem (4). Note that in the $t$-th round of collection, we fix the prior decision variables $q_1, \ldots, q_{t-1}$ constant. Finally, we collect data as determined by the optimal solution $q^*_t$ to problem (4). Full details of the learning and optimization steps, including the complete Algorithm, are in Appendix C.

4.3 Comparison to Mahmood et al. [2]

Our prediction model extends the previous approach of Mahmood et al. [2], who consider only point estimation of $D^*$. They (i) build the set $R$, (ii) fit a parametric function $\hat{v}(q; \theta)$ to $R$ via least-squares minimization, and (iii) solve for $\hat{D} = \min_{q} \{q | \hat{v}(q; \theta) \geq V^*\}$. They use several parametric functions from the neural scaling law literature, including the power law function (i.e., $\hat{v}(q; \theta) := \theta_0 q^{\theta_1} + \theta_2$) where $\theta := \{\theta_0, \theta_1, \theta_2\}$, and use an ad hoc correction factor obtained by trial and error on past tasks to help decrease the failure rate. Instead, we take bootstrap samples of $R$ to fit multiple regression functions, estimate a distribution for $\hat{D}$, and incorporate them into our novel optimization model. Finally, we show in the next two sections that our optimization problem has analytic solutions and extends to multiple sources.

5 Analytic Solutions for the $T = 1$ Setting

In this section, we explore analytic solutions for problem (4). The unobservable $D^*$ and sequential decision-making nature suggest this problem can be formulated as a Partially Observable Markov Decision Process (POMDP) with an infinite state and action space (see Appendix C), but such problems rarely permit exact solution methods [32]. Nonetheless, we can derive exact solutions for the simple case of a single $T = 1$ round, re-stated below

$$\min_{q_1} \epsilon(q_1 - q_0) + P(1 - F(q_1)) \quad \text{s. t. } q_0 \leq q_1$$

(5)

**Theorem 1.** Assume $F(q)$ is strictly increasing and continuous. If there exists $q_1 \geq q_0, \hat{\epsilon} \geq 0$ where

$$\frac{c}{P} \leq \frac{F(q_1) - F(q_0)}{q_1 - q_0}, \quad \hat{\epsilon} \leq 1 - F(q_0), \quad P = c/f(F^{-1}(1 - \hat{\epsilon}))$$

(6)

then there exists an $\epsilon \leq 1 - F(q_0)$ that satisfies $P = c/f(F^{-1}(1 - \epsilon))$ and an optimal solution to the corresponding problem [5] is $q^*_1 := F^{-1}(1 - \epsilon)$. Otherwise, the optimal solution is $q^*_1 := q_0$.

When the penalty $P$ is specified via a failure risk $\epsilon$, the optimal solution to problem (5) is equal to a quantile of the distribution of $D^*$. We defer the proof and some auxiliary results to Appendix C.2.

Theorem 1 further provides guidelines on choosing values for the cost and penalty parameters. While $c$ is the dollar-value cost per-sample, which includes acquisition, cleaning, and annotation, $P$ can reflect their inherent regret or opportunity cost of failing to meet their target score. A designer can accept a risk $\epsilon$ of failing to collect enough data $Pr\{q^* < D^*\} = \epsilon$. From Theorem 1, their optimal strategy should be to collect $F^{-1}(1 - \epsilon)$ points, which is also the optimal solution to problem (5).

6 The Multi-variate LOC: Collecting Data from Multiple Sources

So far, we have assumed that a designer only chooses how much data to collect and must pay a fixed per-sample collection cost. We now explore the multi-variate extension of the data collection problem where there are different types of data with different costs. For example, consider long-tail learning where samples for some rare classes are harder to obtain and thus, more expensive [33], semi-supervised learning where labeling data may cost more than collecting unlabeled data [34], or domain adaptation where a source data set is easier to obtain than a target set [35]. In this section, we highlight our main formulation and defer the complete multi-variate LOC to Appendix D.
We use a simulation model of the deep learning workflow following the procedure of Mahmood et al. [2], to approximate the true problem while simplifying the experiments (see Appendix E for full details). To avoid repeatedly sampling data, re-training a model, and evaluating the score, each experiment is conducted only once. We follow the same steps shown in Section 4 for this problem. First, the learning curve is now a multi-variate function that can handle an arbitrary number of different types of data. In Appendix D, we propose an additive family of power law regression functions that can handle an arbitrary number of data sources (e.g., labeled and unlabeled) and for each \( k \in \{1, \ldots, K\} \), let \( z^k \sim p_k(z^k) \) be data drawn from their distribution. We train a model with a data set \( D := \bigcup_{k=1}^K D^k \) where each \( D^k \) contains points of the \( k \)-th source. The performance or score function of our model is \( V(D^1, \ldots, D^K) \). For each \( k \), we initialize with \( q_0^k \) points. We let \( q_0 = (q_0^1, \ldots, q_0^K)^T \) denote the vector of data set sizes and let \( c = (c^1, \ldots, c^K)^T \) denote costs (i.e., \( c^k \) is the cost of collecting data from \( p_k(z^k) \)). Given a target \( V^* \), penalty \( P \), and \( T \) rounds, we want to minimize the total cost of collection

\[
\min_{q_1, \ldots, q_T} c^T \sum_{t=1}^T (q_t - q_{t-1}) + P \mathbb{1}\{V(D_{q_1^T}, \ldots, D_{q_T^T}) < V^*\} \quad \text{s.t.} \quad q_0 \leq q_1 \leq q_2 \leq \cdots \leq q_T
\]

We follow the same steps shown in Section 4 for this problem. First, the learning curve is now a stochastic process \( \{V_q\}_{q \in \mathbb{R}^K_+} \) indexed in \( K \) dimensions. Next, the multi-variate analogue of the minimum data requirement in [2] is the minimum cost amount of data needed to meet the target:

\[
D^* := \arg \min_q \{c^T q \mid V_q \geq V^*\}
\]

We randomly pick a unique solution to break ties. From Assumption 1, \( D^* \) is a random vector with a PDF \( f(q) \) and a CDF \( F(q) := \int_{\mathbb{R}} f(q) dq \). Finally, the multi-variate analogue of problem (4) is

\[
\min_{q_1, \ldots, q_T} c^T \sum_{t=1}^T (q_t - q_{t-1}) (1 - F(q_{t-1})) + P (1 - F(q_T)) \quad \text{s.t.} \quad q_0 \leq q_1 \leq q_2 \leq \cdots \leq q_T
\]

The Multi-variate LOC requires multi-variate PDFs, which we can fit in the same way as discussed in Section 4.2. However, we now need multi-variate regression functions that can accommodate different types of data. In Appendix D, we propose an additive family of power law regression functions that can handle an arbitrary number of \( K \) sources. In our experiments, we also generalize the estimation approach of Mahmood et al. [2] to the multi-source setting for comparison.

### 7 Empirical Results

We explore the data collection problem over two sets of experiments covering single-variate \( K = 1 \) (Section 4) and multi-variate \( K = 2 \) (Section 6) problems. We consider image classification, segmentation, and object detection tasks. For every data set and task, LOC significantly reduces the number of instances where we fail to meet a data requirement \( V^* \), while incurring a competitive cost with respect to the conventional baseline of naively estimating the data requirement [2].

In this section, we summarize the main results. We detail our data collection and experiment setup in Appendix E. We expand our full results and experiments with additional baselines in Appendix F.
We consider \( T = 1, 3, 5 \) rounds and \( V^* \in [V(D_0) + 1, V(D)] \) targets, where \( D \) is the entire data set. We evaluate all methods on (i) the failure rate, which is how often the method fails to achieve the given \( V^* \) within \( T \) rounds, and (ii) the cost ratio, which is the suboptimality of an algorithm for solving problem \( \mathcal{P} \), i.e., \( c^T(q^*_T - q_0)/c^T(D^* - q_0) - 1 \). Note that the suboptimality does not count the penalty for failure since this would distort the average metrics. For \( K = 1 \), we also measure the ratio of points collected \( q^*_T/D^* \). Although there is a natural trade-off between low cost ratio (under-collecting) and failure rate (over-collecting), we emphasize that our goal is to have low cost but with zero chance of failure.

**The Value of Optimization over Estimation** when \( K = 1 \). Figure 2 compares LOC versus the corresponding power law regression baseline when \( c = 1 \) and \( P = 10^7 \) (\( P = 10^6 \) for VOC and \( P = 10^9 \) for ImageNet). If a curve is below the black line, then it failed to collect enough data to meet the target. LOC consistently remains above this black line for most settings. In contrast, even with up to \( T = 5 \) rounds, collecting data based only on regression estimates leads to failure.

Table 1 aggregates the failure rates and cost ratios for each setting. To summarize, LOC fails at less than 10% of instances for 12/18 settings, whereas regression fails over 30% for 15/18 settings. In particular, regression nearly always under-collects data when given a single \( T = 1 \) round. Here, LOC reduces the risk of under-collecting by 40% to 90% over the baseline. While this leads to a marginal increase in costs, our cost ratios are consistently less than 0.5 for 12/18 settings, meaning that we spend at most 50% more than the true minimum cost.

We remark that previously, Mahmood et al. [2] observed that incorrect regression estimates necessitated real machine learning workflows to collect data over multiple rounds. Instead, with LOC, we can make significantly improved data collection decisions even with a single round.

**Robustness to Cost and Penalty Parameters** (see Appendix F.2 for details). Figure 3 evaluates the ratio of points collected for \( T = 5 \) when the cost and the penalty of the optimization problem are...
Figure 3: Mean ± standard deviation of 5 seeds of the ratio of data collected $q_T^*/D^*$ for different $V^*$ and fixed $T = 5$. Rows 1 & 3: We sweep the cost parameter from 0.001 to 1 and fix $P = 10^7$. Rows 2 & 4: We sweep the penalty parameter from $10^6$ to $10^9$ and fix $c = 1$. The dashed black line corresponds to collecting exactly the minimum data requirement. See Appendix F for all $T$.

Table 1: Average cost ratio $c^T(q_T^* - q_0)/c^T(D^* - q_0) - 1$ and failure rate measured over a range of $V^*$ for each $T$ and data set. We fix $c = 1$ and $P = 10^7$ ($P = 10^9$ for VOC and $P = 10^8$ for ImageNet). The best performing failure rate for each setting is bolded. The cost ratio is measured only for instances that achieve $V^*$. LOC consistently reduces the average failure rate, often down to 0%, while keeping the average cost ratio almost always below 1 (i.e., spending at most $2 \times$ the optimal amount).
varied. Our algorithm is robust to variations in these parameters, as LOC retains the same shape and scale for almost every parameter setting and data set. Further, LOC consistently remains above the horizontal 1 line, showing that even after varying $c$ and $P$, we do not fail as frequently as the baseline. Finally, validating Theorem 1, the penalty parameter $P$ provides natural control over the amount of data collected. As we increase $P$, the ratio of data collected increases consistently.

The Value of Optimization over Estimation when $K = 2$ (Appendix F.3). Figure 4 compares LOC versus regression at $T = 5$ with different costs, showing that we maintain a similar cost ratio to the regression alternative, but with lower failure rates. Table 2 aggregates failure rates and cost ratios for all settings, showing LOC consistently achieves lower failure rates for nearly all settings of $T$. When $T = 5$, LOC also achieves lower cost ratios versus regression on CIFAR-100, meaning that with multiple rounds of collection, we can ensure meeting performance requirements while paying nearly the optimal amount of data. However, solving the optimization problem is generally more difficult as $K$ increases, and we sometimes over-collect data by large margins. In practice, these outliers can be identified from common sense (e.g., if a policy suggests collecting more data than we can reasonably afford, then we would not use the policy suggestion). Consequently, we report these results after removing the 99-th percentile outliers with respect to total cost for both methods. Nonetheless, this challenge remains when $T = 1$, particularly for CIFAR-100.

8 Discussion

We develop a rigorous framework for optimizing data collection workflows in machine learning applications, by introducing an optimal data collection problem that captures the uncertainty in estimating data requirements. We generalize this problem to more realistic settings where multiple data sources incur different collection costs. We validate our solution algorithm, LOC, on six data sets covering classification, segmentation, and detection tasks to show that we consistently meet pre-determined performance metrics regardless of costs and time horizons.

Our approach relies on estimating the CDF and PDF of the minimum data requirement, which is a challenging problem, especially with multiple data sources. Nonetheless, LOC can be deployed on top of future advances in estimating neural scaling laws. Further, we allow practitioners to input problem-specific costs and penalties, but these quantities may not always be readily available. We provide some theoretical insight into parameter selection and show that LOC is robust to these parameters. Finally, our empirical analysis focuses on computer vision, but we expect our approach to be viable in other domains governed by scaling laws.
We thank Jonathan Lorraine, Daiqing Li, Jonah Philion, David Acuna, and Zhiding Yu, as well as the anonymous reviewers and meta-reviewers for valuable feedback on earlier versions of this paper.

Acknowledgments and Disclosure of Funding

We thank Jonathan Lorraine, Daiqing Li, Jonah Philion, David Acuna, and Zhiding Yu, as well as the anonymous reviewers and meta-reviewers for valuable feedback on earlier versions of this paper.

References


Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper’s contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [Yes] See the Conclusion.
   (c) Did you discuss any potential negative societal impacts of your work? [Yes] See the Conclusion.
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]

2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [Yes] See Assumption [1] and the theorem statements.
   (b) Did you include complete proofs of all theoretical results? [Yes] See Appendix [C]

3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [No] The code is proprietary. The data is publicly available. See Algorithm [1] and Appendix [E] for implementation details.
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes] See Appendix [E] and [F] for details.
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [Yes] All experiments were over five seeds. See the for standard deviation ranges.
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes] See Appendix [E]

4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes] We cite all data sets and model architectures in Appendix [E]
   (b) Did you mention the license of the assets? [N/A] All data sets and models are publicly available.
   (c) Did you include any new assets either in the supplemental material or as a URL? [No]
(d) Did you discuss whether and how consent was obtained from people whose data you’re using/curating? [N/A]
(e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [N/A]

5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]