Applications of data valuation in machine learning

MIGUEL DE BENITO DELGADO, appliedAI Institute gGmbH

November 20th, 2023

At TransferLab we have extensively covered existing and developing methods for data valuation, the task of attributing value to samples in a dataset. This family of techniques can help in data collection, active learning, model development and debugging, or fair compensation to data providers among other use cases. This blog reviews the most important of these applications.

In this blog post, we work within the frameworks introduced in our series on data valuation, primarily focusing on the model-dependent context. Here, methods compute the worth of individual training samples based on "how much they contribute" to the final performance of a model over a valuation set. For details on the methods themselves, we refer to the introductory section of our series and to our paper pills.

Contents

1 Data engineering

Perhaps the most relevant uses of data value are in the tasks of improving data and data collection processes, as they impact almost every application in science and industry.

1.1 Repairing and pruning corrupt data

Data can be corrupted in many ways, be it adversarially or not: labels or features can be noisy, and training samples can be tampered with to globally reduce performance or to enable targeted misclassification at test time.

With the valuation function for the training set in our hands, we can try to clean the data to improve performance. By ranking all samples according to their data value and discarding a portion of the lowest-valued ones, followed by retraining the model, we can potentially enhance the model's performance. The intuition is that in-distribution points should have higher values than contaminated or extraneous ones. And indeed, empirically, this procedure tends to improve test error to a certain degree, depending on the quality of the initial data, the robustness of the model, and the accuracy of the computation of the value function. [GZ] illustrates this in several experiments, and we see the same behaviour in our own tests with multiple approaches, see [Tra] and Figure 1.

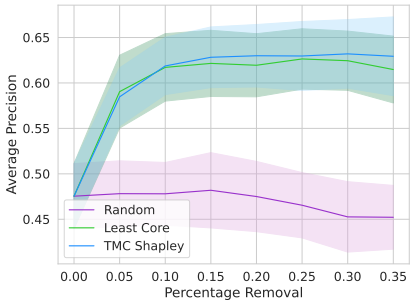

Figure 1. Removing points of low value in succession for 2 game-theoretic valuation methods and a random baseline.

The first issue is that there is no automated way of determining the threshold at which to stop removing low-value samples, and by iteratively removing them and retraining one overfits to the test set. For this reason, instead of trying to automate the process, one can involve domain experts to examine the data, both low- and high-value,

to identify significant patterns (cf. Section 1.4). One successful use case is the construction of scientific or benchmark datasets [TGY+]. Generally speaking data valuation can be a useful tool when used carefully, not only for the potential gain in performance, but also because of the insights gained into what makes data good or bad.

A more fundamental difficulty derives from an intrinsic weakness of metric evaluation over a fixed valuation set. The ability to distinguish harmful data will strongly depend on whether that set is clean of outliers itself or not, and on the robustness of the model to outliers in the training data. These drawbacks are common to all supervised anomaly detection methods, and using (negative) data value as an anomaly score is fraught with the same problems. Techniques like DATA-OOB [KZ] and CGS [NCC] circumvent this issue either through bagging, or by avoiding training the model altogether.

In a very similar vein, when data is scarce, instead of discarding data that impairs performance, we can try to identify what needs fixing beforehand in order to reduce time spent in discussion with customers, domain experts or data providers.

Since influence functions can be computed for each individual test sample, they provide a method to decide which labels to fix first, namely those of training samples that are highly influential for erroneous predictions.

Finally, data value is sometimes used to explain the actions of black-box data repair tools (commercial tools to impute missing data, whose actions are usually opaque to the user). For more on this, see [DFGS] and the references therein.

1.2 Pruning superfluous data

In contrast to the previous setting, where we aimed at identifying harmful data, pruning superfluous data aims at removing redundant or uninformative data. Before, we could focus on high- or low-value points, but uninformative points will tend to have values concentrated around zero for many methods.

In deep learning, longstanding observations have shown power scaling laws that describe error reduction as a function of increasing training set or model size, which drive the increasingly high computational and energy costs of training large models. However, recent research shows that one can improve the situation and possibly achieve exponential scaling by choosing a good metric to dictate the order in which to discard training examples for any pruned dataset size. [SGS+] run extensive benchmarks and perform an exploration of optimal scaling laws. Alas, they demonstrate that there is no silver bullet, and show how the situations in which pruning is doable, desirable or counterproductive depend on model capacity, the amount of data available and its quality.

One way to prune the data effectively is to do the following: First, train a simple model to define the metric. Then, use it to throw some of the data away and train the costly model on the remaining data.

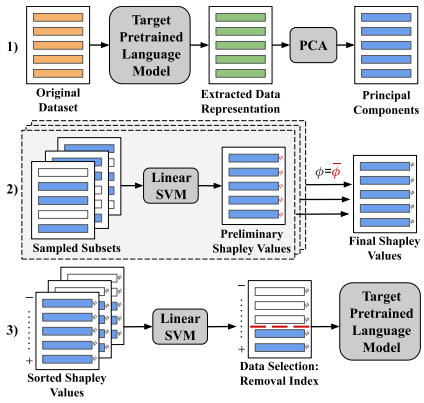

Figure 2. [SGS+] An overview of TS-DSHAPLEY: 1) Process the data using the target LM; 2) Compute sampling chains using a subset of the training set and aggregate the resulting Shapley values; and 3) Transfer the estimated data value information for use with the target LM by estimating the optimal low value data removal index.

A related application is fine-tuning, where a pre-trained model is fine-tuned on a pruned new data set, guided by a metric that uses the initial model. [SMJ] proposes this workflow as TS-DSHAPLEY (Figure 2), the first application of data valuation in the context of LLMs. The initial model is used to compute embeddings for the fine-tuning dataset, and a simple proxy model is trained on these. Values are then estimated and subsequently used to prune this data by removing the lowest-valued points.

In a more adversarial setting, data offered by a provider can be trivially augmented, duplicated or simply unrelated in order to inflate size in an attempt to increase the price. A data purchaser is therefore interested in identifying irrelevant data.[1] Alas, data markets typically only offer previews of the datasets, making an effective pruning strategy impossible, except if offered by the market platform itself.

[1] For more on augmented and irrelevant data, see [JKW+]

Influence functions can be used for the pruning metric, and optimal transport valuation has been proposed in [JKW+], but more research and testing are required, especially in the application to data markets.

Although not strictly a valuation method, CRAIG [MBL] is worth mentioning as a data-centric technique to select an interesting data pruning technique. The idea is to pick a batch over which the gradient most closely approximates the full gradient in order to train only over it. This is performed with a greedy search using marginal utility as the objective. Theoretical guarantees (for the exact solution) hold in certain settings, in particular assuming a Lipschitz condition on the gradient of the model.

1.3 Batch active learning

Another application of data valuation is the labeling of new data, a task whose cost often necessitates carefully selecting what to label next. Batch active learning is a method to enhance this efficiency by selecting groups, or "batches", of new samples to optimize learning performance. The general idea is as follows:[2]

[2] For a very good review of most techniques in batch active learning, we recommend the excellent blog post by Lilian Weng [Wen].

1. The model is trained on an initial set of labeled data.
2. A score is computed for each unlabeled data point in the dataset, reflecting the potential value of labeling that point. The scoring can be based on information gain, diversity of data or expected influence on the model.
3. Based on these scores, a batch of data points is selected for labeling. The goal is to choose a diverse set of high-scoring points that will collectively add significant new information to the model.
4. These new labeled points are added to the training set, and the model is retrained on the updated set.

As an alternative to commonly used scoring functions in the field, such as information gain, [GZE] proposes to use Shapley values. First these are computed for the training set, and then a regression model is trained on them.[3] This valuation model is used to estimate the value of

[3] Actually, because computing Shapley values is so expensive, a surrogate KNN model replaces the last layers a base DNN, effectively computing the Shapley value for a KNN classifier over the embeddings learnt by the network for each sample, which can be done exponentially faster than in the general case, thanks to the local structure of KNN classifiers. This method was introduced in [JDW+].

new samples, which is then used to select the next batch, see Figure 3. The authors report their method to work well for noisy or heterogeneous data, and even under domain shift.

Yet another option is the influence function: [LDZ+] use it as an estimate of the change in the loss when adding an unseen sample to the training set. Because the influence requires the gradient of the loss w.r.t. model parameters, and consequently requires the label, they instead use the output of the model on the unlabeled sample as a pseudo-label.

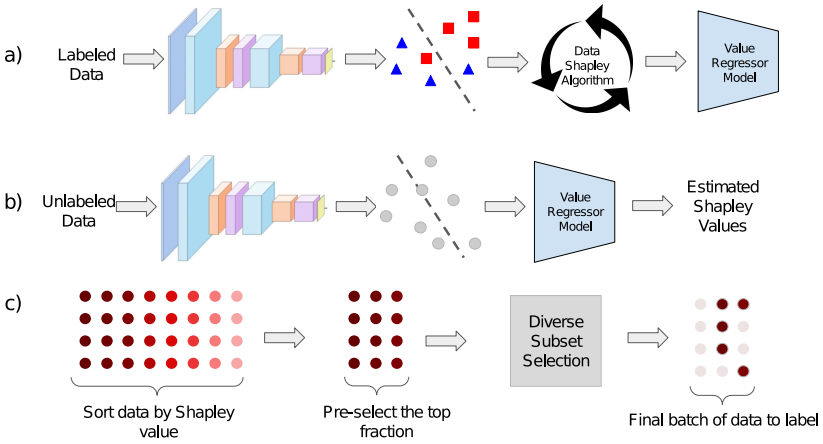

Figure 3. [GZE] Active Data Shapley Enhancing a Diversity Method. (a) Given a trained model, labeled data is featurized, exact Shapley values are computed, and a regression model is trained to predict Shapley values from features. (b) Unlabeled data is featurized, and Shapley values are estimated with the regressor. (c) Unlabeled data is ranked by the estimated Shapley value. The top fraction is pre-selected and fed into any given diversity method to obtain the final batch of points to label

Reported reductions in the number of required annotations w.r.t. random sampling for a fixed test-time accuracy range from 10% to upwards of 30% for both methods. These numbers vary greatly with the domain (always computer vision in the cited papers) and measured accuracy, but can still represent substantial economic savings. Regrettably, to date, no benchmark of game-theoretic vs. influence-based methods has been performed within this context.

1.4 Data collection

Similarly to data labeling, the cost of acquisition often makes targeted choices necessary. If the values computed for the training samples are representative of the true population rather than being an artifact of the sampling, or, roughly put, if the value of a point is stable under changes in the dataset, then it makes sense to try and identify the most valuable points in order to gather more like them. This can also help in hypothesizing relations between data features and predictions: Imagine an accurate pricing model to assist human sales representatives. If high-value data points are identified and the most influential features in those are extracted in a subsequent step, the human operator can use this information when setting or negotiating prices.

Conversely, the least valuable points might help improve data collection, e.g., by detecting patterns of mislabeling or problems with specific data sources. This can save much effort, in particular in the first stages of a project.

2 Model development

2.1 Interpretation and debugging

Numerous techniques exist to examine the behaviour of supervised models considered as black boxes. While a complete taxonomy is beyond the scope of this text (for a thorough review, see, e.g., [BGG+]), most methods revolve around test-time predictions. Some approaches seek global explanations, perhaps in terms of how input features affect the overall outcome, while others are local and focus on individual test samples.[4]

The approaches we consider instead look at the effect that individual *training* samples have on the result. By exploring the most or least valuable or influential of the samples, it is possible to explore the limitations of the model.

As an illustration, consider a K-class classification problem where accuracy is currently low for some class k_0. Construct a valuation set restricted to samples in k_0, and compute the value of the training samples (i.e., their "contribution" towards achieving a good model performance measured on the restricted valuation set). Their values should then reflect their usefulness in predicting the class k_0.

Suppose now that the highest-ranking points are of a different true class $k_i \neq k_0$. Because the predictions are wrong, we might suspect that the model is looking for irrelevant common features, or, in causal terms, *confounders* for the class posterior. Perhaps the data was gathered in different environments, and the model is sensitive to unsuspected features present across these (e.g., backgrounds or lighting conditions). In this case, we might want to try to mitigate the effect, either by improving the data collection process, by identifying and transforming the relevant features, or by changing the model.[5]

As a final example, consider a single misclassified data point z. The influence function allows computing the most influential training points for z. Upon locating them, these points can be explored with feature attribution methods to understand the cause of their influence and potentially improve the model.

2.2 Sensitivity / robustness analysis

Conversely to the previous application, it is possible to study the effect of the *removal* of highly valuable or influential samples. [BGM] show how the removal of very few points can completely reverse the conclusions of a linear regression analysis, even when abundant data are available.

More precisely, they show on real datasets how *removing less than 1% of the data can flip the sign of, and confidence interval around, the parameters* of a regression model.[6] Using a first-order approximation to the influence function, they define a new notion of robustness and estimate a lower bound on the number of samples that one must remove in order to achieve any desired changes to the conclusions of an analysis, e.g., effect sizes and their signs, or arbitrary scalar functions of parametric estimators.[7]

[4] So-called (black-box) *eXplainable AI* comes with lots of caveats and pitfalls, like unstable explanations and conflicting outcomes from different methods [GB]. Therefore, it is always advisable to prefer simpler, interpretable models or to use these techniques during development and debugging. The risk of bogus explanations negatively affecting human decisions is a serious concern. For an insightful review of the many dangers, see [Rud]. For an example (out of many) in clinical practice, see [JPM+].

[5] Of course, it could happen that the points with the highest value are of the same class, in which case we would use different tools to look for commonalities possibly causing the bad performance.

[6] For applications of this *most influential subset* method to moderately-sized LLMs we refer to [FLP+].

[7] In this context, **effect** refers to the magnitude of the parameters in a linear regression model.

Interestingly, the sample bound they obtain is unrelated to classical notions of robustness in statistics, i.e., it is not driven by model misspecification or outliers. Instead, it is roughly a function of the ratio between the uncertainty in the effect that one tries to estimate and the noise in the data. The consequence is that even correctly specified models and non-contaminated observations can yield models highly sensitive to the removal of very few data points if the estimands are small with respect to the noise.

If only a few data are very influential (the model is not robust in their sense), one has to ponder whether the modeling approach taken is sound, whether the data were properly gathered, or whether there is perhaps some intrinsic quality of the problem that requires further analysis.

3 Attacks

Valuation can help in the detection of manipulation, theft and contamination of data. Here, we mention just two applications.

3.1 Watermark removal

Watermarking in the context of ML consists in developing models whose origin can be ascertained, e.g., by testing them against specially crafted samples. This aids developers of proprietary models in finding out whether their licensed architecture and weights are being misused.[8]

[JWS+] suggests an attack against watermarking based on data valuation: points of low value in a training set are likely to be part of the watermarking mechanism since they don't contribute to performance on a correct validation set (i.e., one without watermarks). Note that the usefulness of such an attack is debatable, as an attacker is unlikely to have access to the full training set used by the developers of a proprietary model. Nevertheless, the experimental results in [JWS+] suggest that data values (and in particular surrogate ones that can be computed quickly) can help in identifying samples blatantly out of distribution, given the right conditions, thus supporting the use of valuation as a method for anomaly detection.

[8] For instance, [ABC+] adds a so-called *trigger set* of random samples (e.g., abstract images) with random labels to the training set and trains the model to memorize it. Because of this random nature, if a deployed model correctly labels samples from this trigger set, it must be the one trained on them.

3.2 Poisoning attacks

[KL] proposes employing the influence function to design training points that increase error. For example, in the context of i.i.d. parameter estimation for an r.v. X with range in $\mathbb{R}^d$, this means choosing a perturbation $\delta \in \mathbb{R}^d$ such that for a given influential point x_i, the shifted $x_i + \delta$ induces a large change in the estimator. The same idea applies for regression problems.

The feasibility of such an attack, which requires access to the model, training procedure, and data, is questionable. While one might consider using this method to strengthen a model's robustness via adversarial training, it is unclear whether this particular form of adversarial examples would lead to good robustness and, crucially, whether such robustness is relevant across all applications.[9]

[9] It's worth mentioning the interesting work by [TDS+] which conducts extensive experiments with hundreds of models in computer vision. They come to the conclusion that any ability to generalize to "natural" distribution shifts (e.g., data from the same source but collected differently, as opposed to synthetic modifications) comes at the price of reduced in-distribution performance.

4 Data markets

An escalating demand for data has long been observed across all industries, propelled by augmented data collection operations within organizations and from consumer devices. This surge has motivated the emergence of solutions to connect providers and consumers of data, incorporating mechanisms for economic compensation. Market pricing naturally depends on the value addition for the buyer (e.g., an expected increase in prediction accuracy), but it also considers the seller's perspective differently across scenarios: In business-to-business (B2B), the price will also reflect the costs of the seller's data acquisition. In a business-to-consumer (B2C) context, the price further accounts for the requisite level of privacy of the end-user generating the data.[10]

Two-sided data markets are one of the applications first proposed for data valuation [ADS].[11] The goal is to connect data providers with data consumers, either directly or through a broker. Examples include retail stores gathering customer data, logistics companies optimizing their warehouse planning, or radiology centers sharing data with developers of medical diagnosis software.

An application to marketplaces requires a notion of value that assigns "fair" prices to the data. The game theoretic value functions we discussed above, while providing a certain sense of fairness, have some limitations in this context, for instance, in that they do not intrinsically protect against duplicates and other adversarial behaviors. Nevertheless, there has been progress in this area using Shapley values in certain settings [ADS], and even in the context of federated learning [WRZ+], where order of arrival is important (as opposed to the usual assumption when using Shapley values). These works posit two-sided markets where there is a central data broker and must make certain simplifying assumptions, but there is an extensive literature on the subject.[12]

Valuation approaches like LAVA [JKW+] or CRAIG [MBL] target a scenario where the model is not, and cannot be, given in advance, which can sometimes be a more convenient assumption in this context.[13]

[10] Additionally, the European Data Act requires individuals to have the right to choose data processors for any data harvested from them. Data markets might then include end-users as stakeholders.

[11] Data markets can be classified into **sell-side**, **buy-side**, and **two-sided** markets [ZBL]. In sell-side markets, data's worth is gauged by the information it provides to consumers, e.g., by the expected gain in performance of some model or metric. In buy-side markets, data signifies an owner's cost of acquisition or the value of their privacy, with different privacy concepts determining the measure of privacy loss. In two-sided markets, data holds value from both perspectives.

[12] For a full review of different approaches, we refer to [ZBL].

[13] Although, as mentioned, CRAIG is not strictly for valuation.

BIBLIOGRAPHY

[ABC+] Yossi Adi, Carsten Baum, Moustapha Cisse, Benny Pinkas, and Joseph Keshet. Turning Your Weakness Into a Strength: Watermarking Deep Neural Networks by Backdooring. Pages 1615–1631.

[ADS] Anish Agarwal, Munther Dahleh, and Tuhin Sarkar. A Marketplace for Data: An Algorithmic Solution. EC '19, pages 701–726. Association for Computing Machinery.

[BGG+] Francesco Bodria, Fosca Giannotti, Riccardo Guidotti, Francesca Naretto, Dino Pedreschi, and Salvatore Rinzivillo. Benchmarking and Survey of Explanation Methods for Black Box Models.

[BGM] Tamara Broderick, Ryan Giordano, and Rachael Meager. An Automatic Finite-Sample Robustness Metric: When Can Dropping a Little Data Make a Big Difference?

[DFGS] Daniel Deutch, Nave Frost, Amir Gilad, and Oren Sheffer. Explanations for Data Repair Through Shapley Values. In *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*, CIKM '21, pages 362–371. Association for Computing Machinery.

[FLP+] Jillian Fisher, Lang Liu, Krishna Pillutla, Yejin Choi, and Zaid Harchaoui. Influence Diagnostics under Self-concordance. In *Proceedings of The 26th International Conference on Artificial Intelligence and Statistics*, pages 10028–10076. PMLR.

[GB] Alicja Gosiewska and Przemyslaw Biecek. Do Not Trust Additive Explanations.

[GZ] Amirata Ghorbani and James Zou. Data Shapley: Equitable Valuation of Data for Machine Learning. In *Proceedings of the 36th International Conference on Machine Learning, PMLR*, pages 2242–2251. PMLR.

[GZE] Amirata Ghorbani, James Zou, and Andre Esteva. Data Shapley Valuation for Efficient Batch Active Learning. In *2022 56th Asilomar Conference on Signals, Systems, and Computers*, pages 1456–1462.

[JDW+] Ruoxi Jia, David Dao, Boxin Wang, Frances Ann Hubis, Nezihe Merve Gurel, Bo Li, Ce Zhang, Costas Spanos, and Dawn Song. Efficient task-specific data valuation for nearest neighbor algorithms. 12(11):1610–1623.

[JKW+] Hoang Anh Just, Feiyang Kang, Tianhao Wang, Yi Zeng, Myeongseob Ko, Ming Jin, and Ruoxi Jia. LAVA: Data Valuation without Pre-Specified Learning Algorithms.

[JPM+] Maia Jacobs, Melanie F. Pradier, Thomas H. McCoy, Roy H. Perlis, Finale Doshi-Velez, and Krzysztof Z. Gajos. How machine-learning recommendations influence clinician treatment selections: The example of antidepressant selection. 11(1):1–9.

[JWS+] Ruoxi Jia, Fan Wu, Xuehui Sun, Jiacen Xu, David Dao, Bhavya Kailkhura, Ce Zhang, Bo Li, and Dawn Song. Scalability vs. Utility: Do We Have To Sacrifice One for the Other in Data Importance Quantification? Pages 8239–8247.

[KL] Pang Wei Koh and Percy Liang. Understanding Black-box Predictions via Influence Functions. In *Proceedings of the 34th International Conference on Machine Learning*, pages 1885–1894. PMLR.

[KZ] Yongchan Kwon and James Zou. Data-OOB: Out-of-bag Estimate as a Simple and Efficient Data Value. In *Proceedings of the 40th International Conference on Machine Learning*, pages 18135–18152. PMLR.

[LDZ+] Zhuoming Liu, Hao Ding, Huaping Zhong, Weijia Li, Jifeng Dai, and Conghui He. Influence Selection for Active Learning. Pages 9274–9283.

[MBL] Baharan Mirzasoleiman, Jeff Bilmes, and Jure Leskovec. Coresets for Data-efficient Training of Machine Learning Models. In *Proceedings of the 37th International Conference on Machine Learning*, pages 6950–6960. PMLR.

[NCC] Ki Nohyun, Hoyong Choi, and Hye Won Chung. Data Valuation Without Training of a Model.

[Rud] Cynthia Rudin. Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. 1(5):206–215.

[SGS+] Ben Sorscher, Robert Geirhos, Shashank Shekhar, Surya Ganguli, and Ari S. Morcos. Beyond neural scaling laws: beating power law scaling via data pruning.

[SMJ] Stephanie Schoch, Ritwick Mishra, and Yangfeng Ji. Data Selection for Fine-tuning Large Language Models Using Transferred Shapley Values.

[TDS+] Rohan Taori, Achal Dave, Vaishaal Shankar, Nicholas Carlini, Benjamin Recht, and Ludwig Schmidt. Measuring robustness to natural distribution shifts in image classification. Volume 33, pages 18583–18599. Curran Associates, Inc.

[TGY+] Siyi Tang, Amirata Ghorbani, Rikiya Yamashita, Sameer Rehman, Jared A Dunnmon, James Zou, and Daniel L Rubin. Data valuation for medical imaging using Shapley value and application to a large-scale chest X-ray dataset. 11(1):8366.

[Tra] Team TransferLab. PyDVL: The Python Data Valuation Library.

[Wen] Lilian Weng. Learning with not Enough Data Part 2: Active Learning.

[WRZ+] Tianhao Wang, Johannes Rausch, Ce Zhang, Ruoxi Jia, and Dawn Song. A Principled Approach to Data Valuation for Federated Learning. In Qiang Yang, Lixin Fan, and Han Yu, editors, *Federated Learning: Privacy and Incentive*, Lecture Notes in Computer Science, pages 153–167. Springer International Publishing.

[ZBL] Mengxiao Zhang, Fernando Beltrán, and Jiamou Liu. A Survey of Data Pricing for Data Marketplaces. 9(4):1038–1056.