Opening the black-box: Towards more Interactive and Interpretable Machine Learning

Fabian C. Peña*
Systems and Computing Engineering Department,
Universidad de los Andes, Colombia

John A. Guerra-Gomez†
Systems and Computing Engineering Department,
Universidad de los Andes, Colombia
School of Information,
UC Berkeley

ABSTRACT

Machine Learning (ML) has positioned itself as a hot topic and almost as a synonym to data analysis. However, most of the current real world ML systems use models as black-boxes. Given this, the big question that arises is: what about if model performance is not the unique requirement to be fulfilled? Many real world users will not trust in ML models they cannot understand and therefore will not use them. To address this, new sub-fields of research have been proposed to help users to interact and understand ML models, and by this opening the black-box. In this paper we present a selection of some of the most noteworthy papers of these sub-fields: Interactive ML and Interpretable ML. We contribute a summary of their main characteristics and contrast them to the more classical ML. In addition, we describe some of the guidelines from the state of the art, which can help design better and more user-centric ML models and systems.

Index Terms: Machine Learning—Human-Computer Interaction—Interactive Machine Learning—Interpretable Machine Learning

1 INTRODUCTION

Black-box Machine Learning (ML) is a term commonly applied to the use of ML models that aren't completely understood by users entailing lack of trust and, therefore, hindering decision making. This disadvantage increases with the use of newfangled technologies such as Deep Learning where more parameters need to be adjusted requiring large amounts of data. In black-box ML models, users do not have anything to say about the produced results and the system does not take into account the users expert feedback that could greatly improve the overall performance of the system. As Lipton [13] says: *"While the ML objective might be to reduce error, the real-world purpose is to provide useful information*.

To address this, the research community has proposed sub-fields that tend to open the black-box models by allowing users to interact and interpret them better. In this paper we summarize some of the most significant papers of two of those sub-fields: Interactive ML and Interpretable ML. We contrast their suggestions with the classic ML paradigm, and finally describe some of the guidelines proposed on the current literature, which can guide on developing more user-centric ML models and systems.

The Human Computer Interaction (HCI) field presents a great amount of resources that can lead the way towards better and more user-centric ML systems. For example, Holzinger [11] argues that HCI can be of great benefit for supporting Knowledge Discovery (KDD), suggesting that Interactive ML systems can improve their outcomes by incorporating user feedback. In the same way, Dudley et al. [6] summarize the interaction elements that should be considered when designing Interactive ML systems, namely: **sample review**, **feedback assignment**, **model inspection** and **task overview**. Known strategies for implementing the first two elements are briefly described in Sections 2.1 and 2.2. To the best of our knowledge, although the **model inspection** and **task overview** interaction elements do not have the same level of development, some related works are presented in Section 2.3.

We argue that the **model inspection** element can be complemented by producing different kinds of interpretations about the model continuously refined thanks to **feedback assignment**. Model interpretation indicates the ability to provide explanations about their inner working which could help users understand their results. In addition, these additional components could support the **task overview** interface element because, as Doshi-Velez and Kim [5] and Lipton [13] explain, model interpretation is used to achieve other important requirements in user-centric ML systems such as trust, unbiasedness, privacy, reliability, robustness, causality, transferability, informativeness and usability. More details are discussed in Section 3.

The rest of the paper explains the general characteristics and differences of classic ML versus more user-centric approaches. Then we proceed to summarize some of the main findings of the Interactive and Interpretable ML sub-fields, to conclude by describing some guidelines on how to design such user-centric ML systems.

2 INTERACTIVE MACHINE LEARNING

Involving the user in the learning process is not a trivial task, and for a long time it has not been the concern of both KDD and ML fields. The Human-Computer Interaction (HCI) field, according to Holzinger [10], cares about human perception, cognition, intelligence, decision-making and interactive techniques of visualization. By merging these fields, many research opportunities emerge including the paradigm where algorithms can interact with agents (humans) and optimize their learning behavior through these interactions, as established by Holzinger [11]. A more formal definition of Interactive ML is given by Dudley and Kristensson [6]: *"... is a co-adaptive process, driven by the user, but inherently dynamic in nature as the model and user evolve together during training"*.

As described by Fails and Olsen [8], classic ML generally has some assumptions which can be addressed through the use of Interactive ML, such as:

- The introduction of many features in the model can become noise and therefore affect its performance. An Interactive ML system should provide to user the ability to perform feature selection through a friendly interface and evaluate the change produced immediately.
- There is not enough training data. Similar to feature selection, the user also must be able to perform labeling under consideration or by a systematic way and evaluate the change produced in the model performance.

*e-mail: fc.pena@uniandes.edu.co
†e-mail: ja.guerrag@uniandes.edu.co

- It is desirable that the system can adapt quickly to new training data. Avoid overfitting, addressed by strategies such as cross-validation, is one of the main concerns when designing ML systems. In the context of Interactive ML, the user can take decisions (e.g. labeling correction) in areas where decision frontier is more fuzzy. Active Learning and Visual Interactive Labeling, described in Section 2.2, support this user activity.

According to Fails and Olsen [8], an interface for Interactive ML must meet the following requirements: train very quickly, accommodate hundreds to thousands of features, perform feature selection and support tens to hundreds of thousands of training examples. From a more generalized perspective, Dudley and Kristensson [6] provide 4 interface elements to be considered while designing an Interactive ML system: **sample review**, **feedback assignment**, **model inspection** and **task overview**. A synthesis of Interactive ML scenario is shown in Figure 1, where the first three elements previously mentioned enable the direct user interaction with the data or the model. The **task overview** element is avoided because we considered is transversal to whole system and it is closely related to user and task context.

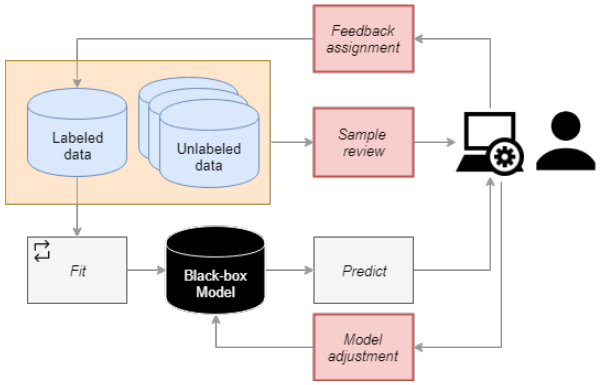

Figure 1: Interactive Machine Learning scenario. Three of four elements presented in Section 2 are highlighted, describing the direction of the interaction.

Next subsections describe some related concepts to tackle the Interactive ML design process including some implementation scenarios. A more extensive revision about families of algorithms and application domains in the context of integrating ML and Visual Analytics can be found in Endert et al. [7].

For **task overview**, no one relevant work was found in the context of Interactive ML because this interface element is closely related to the specific knowledge domain. Nevertheless, Dudley et al. [6] argue model accuracy and related metrics are not enough to evaluate the task fulfillment and this interface element should provide visibility of global objectives but also contextualize about other information such as availability of training data.

2.1 Dimensionality Reduction

In general terms, according to Tenenbaum et al. [20] and Roweis and Saul [18], Dimensionality Reduction (DR) deals with the problem of finding meaningful low-dimensional structures (compact representation) from high-dimensional data. For **sample review**, it is a valuable technique for representing high-dimensional data in principally two dimensions to validate the data distribution. A good DR representation can be one that allows evidence class separability, in the context of classification. Some algorithms for DR are PCA, t-SNE, proposed by Van der Maaten and Hinton [23], and UMAP, developed by McInnes and Healy [14].

In the context of clustering, Wenskovitch et al. [24] discuss about the combination of both techniques, contributing with a series of design challenges and questions from an extensive literature review. In the work of Wenskovitch and North [25], data is projected by DR and a **feedback assignment** mechanism is provided to improve the cluster computation in an iterative way.

2.2 Active Learning and Visual Interactive Labeling

One way of achieving **feedback assignment** is by leveraging users for labeling data. Active Learning (AL), described in Figure 2, consists of a series of analytical methods to select unlabeled data and present it to users in the form of queries for label assignment [11]. Independently of the querying method used, the success of incorporating AL into an Interactive ML system lies on keeping the user labeling effort to a minimum. This can be achieved by only asking for feedback when the hope for a performance improvement given a specific query is high, as specified by Olsson [16] and Tong and Koller [22].

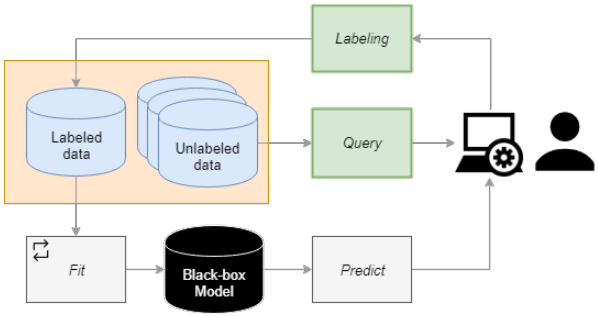

Figure 2: Active Learning scenario. An *oracle*, typically a human, is asked for labels according to criteria based on maximization of performance improvement.

Visual Interactive Labeling (VIL), in contrast to AL, delivers to user the selection of data candidates to be labeled. Nevertheless, user cognitive load can be reduced by incorporating visual techniques such as 2D Colormap, Class Coloring, Convex Hulls and Butterfly Plot, as described by Bernard et al. [1]. AL and VIL are used in scenarios where there is small amount of labeled data, and where the domain knowledge from users can be useful to improve unreliable predictions.

Complementing the work of Bernard et al. [1], they present a comparison among multiple VIL and AL strategies. The decision regarding to use either of the two methods depends highly on the user task complexity, the **sample review** technique and the class separability. In general terms, both techniques can compete to produce better and faster models.

2.3 Parameter Tuning and Error Discovery

Complementary to concepts previously described, we highlight two useful strategies to be included into an Interactive ML system: parameter tuning and error discovery. For the first strategy, we clarify that adjustment of model parameters does not imply necessarily to provide the ability to modify the number of hidden layers or link weights, talking about neural networks. The goal is to design interaction mechanisms usable for users even when these are not experts in ML. In other words, putting model parameters in terms of task domain, being this a non-trivial design decision. Some examples are described below.

The work of Self et al. [19] concludes with a list of user actions affecting implicitly the model parameters, consisting of: dragging points to form one or more new clusters, dragging an outlier into existing cluster, maximize one dimension weight and drag multiple sliders to equally large weights. In a similar way, Kapoor et al. [12] focus on provide users the ability to refine the parameters of the

confusion matrix according to their preferences and thus re-train the model in an iterative way.

An example of Interactive ML system for error discovery is presented by Chen et al. [4]. From a website classification problem, domain knowledge about the target class is introduced facilitating error discovery through semantic data exploration. While elements such as **sample review** and **model inspection** are introduced, an eventual desire of Interactive ML systems is missing: the ability to perform labeling correction and re-train the model to evaluate if it improves its performance.

It is also important to highlight some works focused on implement systems that learn explicitly from user knowledge and not refining an specific algorithm proposed in literature. In Brown et al. [2], users are able to build the distance function for two dimensions data projection according to their own sense of distance. Chang et al. [3] propose a tool for clustering steered by user, having significantly higher quality than those from a pure algorithmic process.

3 Interpretable Machine Learning

In some scenarios, mechanisms to **sample review** and **feedback assignment** (e.g. user labeling or parameter tuning) may not be enough to successfully fulfill the user task. We argue **model inspection** implies opening the black-box model and presenting it to user in an interpretable or explainable way. In other words, **an Interactive ML system is not necessarily interpretable in terms of what model is learning and, in the opposite case, an Interpretable ML system could not involve all enough elements of interaction to perform the task in an usable and efficient way**.

Nevertheless, the decision behind delivering interpretability to users must be careful studied, because as Doshi-Velez and Kim [5] explain, interpretability is necessary and appropriate when there is an incompleteness in the problem formalization and can be avoided in these two scenarios: *"(1) there are no significant consequences for unacceptable results or (2) the problem is sufficiently well-studied and validated in real applications that we trust the systems decision, even if the system is not perfect"*.

Figure 3, based on Hohman [9], Doshi-Velez and Kim [5] and Lipton [13], presents a condensed synthesis regarding to aspects to consider when thinking in produce interpretability. These aspects are grouped in six questions to be asked during the design process:

- Why do you need to produce interpretability? Should the system be in the capacity to generate trust for predictions? Does it protect sensitive information in the data?.
- Who is the user of the system? Is the user a ML theoretical or a domain expert?
- What are the most important elements of the data or the model to visualize?
- How do you plan to represent those most important elements? Is it important for user to interact with those elements?
- When generate interpretations? Is the model under continuous refinement thanks **feedback assignment** or was the model previously trained and validated?
- Where will the system be deployed? Is the system intended to support a real-world problem in a company? Does the system contribute to produce scientific advances in a specific ML subfield or any other knowledge domain?

Interpretable ML is not an isolated concept from Interactive ML, meaning that explanations could be produced in an interactive way to user considering the four interaction elements described in previous sections. Complementary, designers must not forget the system is constrained by an user task and users are those who are in the power to determine if the task was effectively fulfilled. User evaluations for interactivity and interpretability components, as in any HCI study, must be considered.

3.1 Frameworks for Interpretability

Ribeiro et al. [17] develop LIME. Based on local interpretability, this framework is able to explain predictions by learning an interpretable model locally around the prediction. The authors show the potential of the framework in applications based on tabular, image and text data. Complementary, LIME is extended by Teso and Kersting [21] to explain the query produced in an AL scenario. Local explanations are the base of works focused on produce global interpretation as the case of the proposal by Yang et al. [26]. From contribution matrix representing the feature importance for every single data sample, product of frameworks such as LIME, a binary tree is learned to explicitly represent the most important decision rules that are implicitly contained in the black-box model.

In Section 1, we mention that Deep Learning models have the disadvantage of being the least interpretable due to their large number of parameters. Contributions in this field have been achieved by producing interpretation of the features learned at each layer of a Neural Network, as demonstrated by Yosinski et al. [27] and Olah et al. [15]. A more extensive literature review about interpretability in Deep Learning can be found in Zhang and Zhu [28].

4 Future Work

The contribution of this work involves the definition of both Interactive ML and Interpretable ML as well as the description of some related work in the context of natural language processing, image recognition and tabular data. Section 2 presents the four interface elements to consider when designing Interactive ML systems and Section 3 constraints the design process of Interpretable ML systems to six questions that allow to establish the purpose, target user and some alternatives to produce explanations. The results of this effort are the inputs of some future projects based on tabular data provided by tax administration and town planning organizations where involving user knowledge to feed the ML system and facilitate decision making has been determined as one of the main requirements.

The tax administration project implies to provide tax officers with an enriched application to automate some of their tasks in terms of taxpayer supervision and tax base estimation, among other functions of the secretary of finance. Currently, data owners do not dispose of labeled data, reason why training a traditional supervised learning model is not feasible. As second aspect, users must be able to visualize the data in real time to evidence the historic and current situations producing insights from a descriptive perspective but also supported by backstage models of classification and regression. No less important, producing interpretations about, for instance, the probability of avoidance is an important part of the system, qualifying to tax officers to intervene taxpayers opportunely.

Similar to previous project, we develop a system able to provide functionalities for town planning based on patterns found in cities around Latin America and the Caribbean. From clustering algorithms, users want to discover relationships among cities exhibiting similar features. Some important system requirements are to explicitly demonstrate why a particular city is belonging to a certain cluster and which ones are the most nearby cities according to user-defined criteria and other ones defined by Dimensionality Reduction proximity.

Acknowledgments

The authors thank the Urban Health Network for Latin America and the Caribbean and the CAOBA Alliance for supporting our research efforts. Future work will be developed using real problems provided by these organizations.

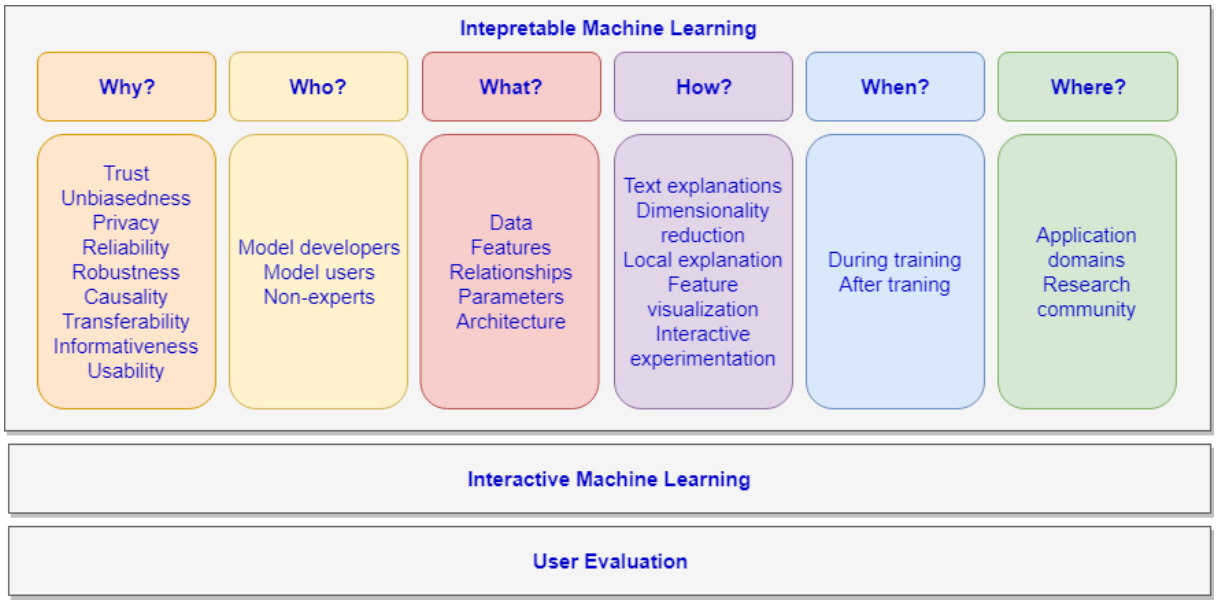

Figure 3: Aspects to consider when designing Interpretable ML systems. Based on Hohman [9], Doshi-Velez and Kim [5] and Lipton [13].

REFERENCES

[1] J. Bernard, M. Hutter, M. Zeppelzauer, D. Fellner, and M. Sedlmair. Comparing Visual-Interactive Labeling with Active Learning: An Experimental Study. *IEEE Transactions on Visualization and Computer Graphics*, 24(1):298–308, 1 2018. doi: 10.1109/TVCG.2017.2744818

[2] E. T. Brown, J. Liu, C. E. Brodley, and R. Chang. Dis-function: Learning distance functions interactively. In *IEEE Conference on Visual Analytics Science and Technology 2012, VAST 2012 - Proceedings*, pp. 83–92, 2012. doi: 10.1109/VAST.2012.6400486

[3] S. Chang, P. Dai, L. Hong, C. Sheng, T. Zhang, and E. H. Chi. AppGrouper: Knowledge-graph-based Interactive Clustering Tool for Mobile App Search Results. In *Proceedings of the 21st International Conference on Intelligent User Interfaces - IUI '16*, pp. 348–358, 2016. doi: 10.1145/2856767.2856783

[4] N.-C. Chen, J. Suh, J. Verwey, G. Ramos, S. Drucker, and P. Simard. AnchorViz: Facilitating Classifier Error Discovery through Interactive Semantic Data Exploration. In *23rd International Conference on Intelligent User Interfaces*, pp. 269–280. ACM, Tokyo, Japan, 2018. doi: 10.1145/3172944.3172950

[5] F. Doshi-Velez and B. Kim. Towards A Rigorous Science of Interpretable Machine Learning. *arXiv preprint*, 2017.

[6] J. J. Dudley and P. Ola Kristensson. A Review of User Interface Design for Interactive Machine Learning. *ACM Trans. Interact. Intell. Syst*, 8(8), 2018. doi: 10.1145/3185517

[7] A. Endert, W. Ribarsky, C. Turkay, B. L. W. Wong, I. Nabney, I. D. Blanco, and F. Rossi. The State of the Art in Integrating Machine Learning into Visual Analytics. *Computer Graphics Forum*, 00(00):1–28, 2017. doi: 10.1111/cgf.13092

[8] J. A. Fails and D. R. Olsen. Interactive machine learning. In *Proceedings of the 8th international conference on Intelligent user interfaces - IUI '03*, p. 39. ACM Press, New York, New York, USA, 2003. doi: 10.1145/604045.604056

[9] F. M. Hohman, M. Kahng, R. Pienta, and D. H. Chau. Visual Analytics in Deep Learning: An Interrogative Survey for the Next Frontiers. *IEEE Transactions on Visualization and Computer Graphics*, 25(1):1–1, 2018. doi: 10.1109/TVCG.2018.2843369

[10] A. Holzinger. Human-Computer Interaction and Knowledge Discovery (HCI-KDD): What is the benefit of bringing those two fields to work together? In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 8127 LNCS, pp. 319–328. 2013. doi: 10.1007/978-3-642-40511-2

[11] A. Holzinger. Interactive machine learning for health informatics: when do we need the human-in-the-loop? *Brain Informatics*, 3(2):119–131, 2016. doi: 10.1007/s40708-016-0042-6

[12] A. Kapoor, B. Lee, D. Tan, and E. Horvitz. Interactive optimization for steering machine classification. In *Proceedings of the 28th international conference on Human factors in computing systems - CHI '10*, p. 1343. ACM Press, New York, New York, USA, 2010. doi: 10.1145/1753326.1753529

[13] Z. C. Lipton. The Mythos of Model Interpretability. *arXiv preprint*, 2017.

[14] L. McInnes and J. Healy. UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction. *arXiv preprint*, 2018.

[15] C. Olah, A. Satyanarayan, I. Johnson, S. Carter, L. Schubert, K. Ye, and A. Mordvintsev. The Building Blocks of Interpretability. *Distill*, 2018. doi: 10.23915/distill.00010

[16] F. Olsson. A literature survey of active machine learning in the context of natural language processing. Technical report, 2009.

[17] M. T. Ribeiro, S. Singh, and C. Guestrin. "Why Should I Trust You?" Explaining the Predictions of Any Classifier. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 1135–1144, 2016. doi: 10.1145/2939672.2939778

[18] S. T. Roweis, L. K. Saul, and S. T. Roweis. Nonlinear Dimensionality Reduction by Locally Linear Embedding. *SCIENCE*, 290(December), 2000.

[19] J. Z. Self, R. K. Vinayagam, J. T. Fry, and C. North. Bridging the gap between user intention and model parameters for human-in-the-loop data analytics. In *Proceedings of the Workshop on Human-In-the-Loop Data Analytics - HILDA '16*, pp. 1–6, 2016. doi: 10.1145/2939502.2939505

[20] J. B. Tenenbaum, V. D. Silva, and J. C. Langford. A Global Geometric Framework for Nonlinear Dimensionality Reduction. *SCIENCE*, 290(December), 2000.

[21] S. Teso and K. Kersting. "Why Should I Trust Interactive Learners?" Explaining Interactive Queries of Classifiers to Users. *arXiv preprint*, 5 2018.

[22] S. Tong and D. Koller. Support Vector Machine Active Learning with Applications to Text Classification. *Journal of Machine Learning Research*, 2:45–66, 2001. doi: 10.1162/153244302760185243

[23] L. J. P. Van Der Maaten and G. E. Hinton. Visualizing high-dimensional data using t-sne. *Journal of Machine Learning Research*, 9:2579–2605, 2008. doi: 10.1007/s10479-011-0841-3

[24] J. Wenskovitch, I. Crandell, N. Ramakrishnan, L. House, S. Leman, and C. North. Towards a Systematic Combination of Dimension Reduction and Clustering in Visual Analytics. *IEEE Transactions on Visualization and Computer Graphics*, 24(1):131–141, 2018. doi: 10.1109/TVCG.2017.2745258

[25] J. Wenskovitch and C. North. Observation-Level Interaction with Clustering and Dimension Reduction Algorithms. In *Proceedings of the 2nd Workshop on Human-In-the-Loop Data Analytics - HILDA'17*, pp. 1–6, 2017. doi: 10.1145/3077257.3077259

[26] C. Yang, A. Rangarajan, and S. Ranka. Global Model Interpretation via Recursive Partitioning. *arXiv preprint*, 2018.

[27] J. Yosinski, J. Clune, A. Nguyen, T. Fuchs, and H. Lipson. Understanding Neural Networks Through Deep Visualization. *arXiv preprint*, 2015.

[28] Q. Zhang and S.-C. Zhu. Visual Interpretability for Deep Learning: a Survey. *arXiv preprint*, 2018. doi: 10.1631/FITEE.1700808