**2018-2019**
**Coursework submission**
**Cover sheet**

**Student name……**

**College……**

**CRSID……**

**Submit to Graduate Education**
*Date of submission*

---

**Module name**

**Module Code No.**

---

**Assignment No.**

**Assignment mark/grade**

---

**Assessor comments**

GE Records……..

---

*GE stamp date and initials*

**CRSID……**

**Module No……**

**Assignment No……**

# Interaction with Machine Learning
# Diary 1

Zhuoni Jie
zj245@cam.ac.uk

January 27, 2019

*Word Count: 820*

## 1 Summary of the contents of the lecture

This course is a practical Human-Computer Interaction (HCI) course, which uses intelligent tools including visualisation, programming, labelling, and explanation. According to Hassabis's comments on four waves of AI, the fourth wave is about intelligent tools, and the course objective is to deliver it. Students are expected to achieve research competence in this field. The assignment standards, lecture topics, work plan, assessment, and reading list were introduced.

This lecture introduced theories of interaction. HCI has gone through three waves since 1980s. The first wave of HCI is about engineering human factors, and is relatively closely aligned with AI. The user interface is designed independently of the main system and focuses on efficiency, using methods from cognitive science to model users. The second wave of HCI is as social system taking account of other socio-technical information. The third wave of HCI is as culture and experience, and outside workplace efficiency is no longer a priority. Then engineering models of human I/O, memory, CPU, the problem of learning, the sticky problem of viscosity, and wicked problems were discussed.

The lecture then introduced intelligent interaction. Established paradigms of interacting with ML including recommender systems, dialogue models, and programming by example were introduced. Major intelligent interaction topics at HCI conferences and paper selections were shown.

Research methods in HCI were introduced. The lecture first stressed ethical issues in research. Controlled experimental methods were explained, with manipulation and measurement concerns discussed. Pros and cons of self-report and think-aloud were discussed. Then the qualitative data methods were introduced, discussing protocol analysis methods, hypothesis- or theory-driven,

and grounded theory. Then experiment design, analysis, and evaluation as well as field study methods were introduced, discussing their usage and concerns in different situations.

Lastly, the lecture introduced how to plan the study. Ideas about candidate interactive systems/intelligent tools, representative tasks and measures, review of study design options, and theoretical goal were discussed.

## 2 Relations with some of the other literature

The first wave of HCI that have formed the field oriented from industrial engineering and human factors with its focus on optimizing human-machine fit. The second waive stemmed from cognitive science, emphasizing on theory and what is happening in both computer and human mind. To appropriate Flyvbjerg's characterization of the state of social sciences, it raises "rationality and rational analysis to the most important mode of operation for human activity" (1). These two waves can clash at some aspects. For example, Gray and Salzman had the 'Damaged Merchandise' controversy in the mid 90s (2). These years, with the development of technologies such as ubiquitous computing, workplace study, visualization, affective and educational technology, Harrison et al. in CHI'07 puts forward the third paradigm in HCI addressing new issues that are bad fit to prior diagrams. The third paradigm treats interaction not as a form of information processing, but as a form of meaning making where the artifact and its context at all levels are mutually defining and subject to multiple interactions (3).

Tullio et al. conducted a field study to undertand how users perceive and interact with intelligent systems. These systems demand trust from users so that users are willing to delegate important decisions or personal information (4), and this trust comes from an ability to predict the systems' behavior through observation (5). Designers should design intelligent systems that enable formulation of mental models that are predictable enough for users to gain their trust (6). In this six-week field study, they interviewed eight office workers regarding the operation of a system that predicted their managers' interruptibility, comparing their mental model to the actual system model. They suggested that users may need additional, high-level feedback to adopt more correct structures, and provided some challenges including simple feedback in the interface on user inputs may not be enough to accelerate trust and adoption improvement (7).

# 3 Opportunities for further research and interesting questions/conflicts in this topic

The measure and evaluation of design success have possibilities for exploration. Many acceptable measures of system success focus on measuring the comparative effectiveness and efficiency of information transfer, sometimes combined with qualitative data analysis. User self-reported data, and some criteria such as satisfaction or delight, are considered to measure success. However, more criteria suggesting implicit long-term benefits, and maybe providing provoking ideas, should also be considered as beneficial factors of the system, but are not so closely related to effectiveness or efficiency. More long-term study in the wild, which means long-term behavior study outside experimental settings, are desired to understand sufficient measures of system success. Furthermore, balancing the concerns of different stakeholders should also be done in a clever way. When interpreting experimental results, how much emphasis we should put on effect size and significance also remain to be explored. Additionally, we should also be careful about the reliability of data (especially qualitative ones such as interview and think-aloud, since users' behavior may be different in experimental situations, and may not reveal their real thoughts). Users and systems may also co-evolve during the use of intelligent systems, such as learning effects, causing complex interactive effects for research and analysis.

# References


[1] Flyvbjerg, Bent. Making social science matter: Why social inquiry fails and how it can succeed again. Cambridge university press, 2001.

[2] Gray, Wayne D., and Marilyn C. Salzman. "Damaged merchandise? A review of experiments that compare usability evaluation methods." Human-computer interaction 13.3 (1998): 203-261.

[3] Harrison, Steve, Deborah Tatar, and Phoebe Sengers. "The three paradigms of HCI." Alt. Chi. Session at the SIGCHI Conference on Human Factors in Computing Systems San Jose, California, USA. 2007.

[4] Dzindolet, Mary T., et al. "The role of trust in automation reliance." International journal of human-computer studies 58.6 (2003): 697-718.

[5] Muir, Bonnie M. "Trust in automation: Part I. Theoretical issues in the study of trust and human intervention in automated systems." Ergonomics 37.11 (1994): 1905-1922.

[6] Birnbaum, Larry, et al. "Compelling intelligent user interfaces—how much AI?." Proceedings of the 2nd international conference on Intelligent user interfaces. ACM, 1997.

[7] Tullio, Joe, et al. "How it works: a field study of non-technical users interacting with an intelligent system." Proceedings of the SIGCHI Conference on Human Factors in Computing Systems. ACM, 2007.

**2018-2019**
**Coursework submission**
**Cover sheet**

**Student name……**

**College……**

**CRSID……**

**Submit to Graduate Education**
*Date of submission*

---

**Module name**

**Module Code No.**

---

**Assignment No.**

**Assignment mark/grade**

---

**Assessor comments**

GE Records……..

---

*GE stamp date and initials*

**CRSID……**

**Module No……**

**Assignment No……**

# Interaction with Machine Learning
# Diary 2

Zhuoni Jie
zj245@cam.ac.uk

February 5, 2019

*Word Count: 905*

## 1 Summary of the contents of the lecture

This week's lecture introduced Program Synthesis. First, the desired user experience and system response in program synthesis led to the discussion of several usability issues: how to specify applicability; how to control generalization; how to understand what was inferred; how to modify the synthesized program. A classic programming by example - using keyboard in Emacs - was introduced. A keyboard macro records a sequence of user actions that carry out Emacs commands. However, these macro recorders cannot generalize the user commands, and are unable to adjust for different cases. Interactive dialogue is essential. For the machine learning aided intelligent tools, the core problem is to create a model of user intent.

Different interaction techniques that can be used for programming by example (PBE) were compared: classic mixed-initiative PBE (Allen Cypher's "Eager"); PBE in the graphics domain (David Kurlander's "Chimera"); user control of generalization in PBE (Ken Kahn's "ToonTalk"). ToonTalk is intended to be programmed by children. The system's presentation is in the form of animated characters, including robots that can be trained by example. In ToonTalk, users can generalize a constraint with Dusty.

The generalization from examples is fundamental to mental abstraction, which is required by automated action such as programming. This brings discussion possibilities. The Attention Investment model of abstraction was introduced. The standard rules of usability (e.g., immediate feedback, everything you do in UI should have incremental effects) do not apply to programming, which is not like direct manipulation. Formulating and refining abstractions is time-costing

and cognitively hard, and different ways for users to approach their tasks were introduced.

The aims and communication with examples of structured text editing as an machine learning application was discussed. SWYN (See What You Need) is a system inferring regexps to generalize text macros. Some example applications act as the programmer's assistant, inferring cognitive plan elements and patterns from IDEs using models and heuristics. FlashFill for Excel and Data Noodles were introduced as two examples. Letting users trust the generalizations is important in these systems.

## 2 Relations with some of the other literature

Program synthesis is the task of automatically finding programs that satisfy user intent expressed in some form of constraints (e.g., input-output examples, demonstrations, natural language). Programming by demonstration (PBD) extends the functionality of existing applications to accomplish new tasks. Generality is important because PBD technology is meant to allow users to automate their unique repetitive tasks that could not have been predicted by application designers. PBD systems must be general to handle the unexpected.

In 1932 in the early work on constructive mathematics, the first automated theorem provers were developed (1). After this, a lot of pioneering work on deductive synthesis approaches was done (2)(3). Their main idea was to construct a proof of the user-provided specification, and then use the proof to extract the corresponding logical program. Transformation-based synthesis (4) later became popular, where a high-level complete specification was transformed repeatedly until achieving the desired low-level program. In the 1970s, Patrick Winston worked on machine learning concepts from human teachers and studied learning structure descriptions from examples (5). Assuming a complete specification of the desired user intent was complicated, and this led to inductive synthesis approaches such as input-output examples. Pygmalion (6) is the first system for PBD, which introduces icons and mixed initiative computing. There is also pioneering work on using genetic programming approaches to evolve programs that are consistent with a specification (7). Many more recent program synthesis approaches allow a user to provide a grammar in addition to the specification, such as the SKETCH system (8).

## 3 Opportunities for further research and interesting questions/conflicts in this topic

The inherent challenges of program synthesis mainly lie in the intractability of the program space and diversity of user intent (9). Even equipped with efficient

search methods, accurately expressing and interpreting user intent, which is the specification on the desired program, is still one major challenge in program synthesis. Letting users changing preferences is not general enough to satisfy the users' needs for flexibility.

Different methods for expressing user intent range from formal logical specifications to informal natural language descriptions or input-output examples. One of the problems is, specifying specifications may be difficult for users. For example, when giving an input-output example, the program space of the PBE system may contain many algorithms which are consistent with it. These algorithms may also simply overfit the example and are not consistent with user intents. Many current systems including Flash Fill features lack ways to discover this without additional communication with the user (9). And human-computer communication cannot be directly established using implications generated from human-human communication, because human-human communication sometimes is ambiguous, lack of cohesion, coherence, and precision, and sometimes can fail (10). Too many special cases, unable to make perfect specifications beforehand, make users hard to give specifications without actually interacting with the PBE systems. Designing intelligent programming systems according to different user groups and tasks is needed and worth more research.

Human-computer communication specific to domains and tasks is desired in the design of these PBE systems. People should also be allowed to customize the interaction experience. For example, people can use formal languages if they are task-specific, and can use grammars or specifications differently according to contexts. The contributions of users, the PBE system, the applications, and the communication protocols required between the PBE system and the applications should be designed and studied. How PBE can be trusted and add credibility to the analysis can also be studied in different user groups. "Just-in-time" programming, programming "during task-time", and end-user programming in collaborative settings are also interesting topics for more research.

# References


[1] A. N. Kolmogorov. Zur deutung der intuitionistischen logik. Math. Zeitschr, 35:58–365, 1932.

[2] C. Cordell Green. Application of theorem proving to problem solving. In IJCAI, pages 219–240, 1969.

[3] Zohar Manna and Richard J. Waldinger. Toward automatic program synthesis. Commun. ACM, 14(3):151–165, 1971.

[4] Zohar Manna and Richard J. Waldinger. Knowledge and reasoning in program synthesis. Artif. Intell., 6(2):175–208, 1975.

[5] Winston P H. Learning structural descriptions from examples[J]. 1970.

[6] David Canfield Smith. Pygmalion: A Creative Programming Environment. PhD thesis, Stanford University, Stanford, CA, USA, 1975.

[7] John R Koza. Genetic programming as a means for programming computers by natural selection. Statistics and Computing, 4(2):87–112, 1994.

[8] Armando Solar-Lezama. Program synthesis by sketching. ProQuest, 2008.

[9] Gulwani S, Polozov O, Singh R. Program synthesis[J]. Foundations and Trends in Programming Languages, 2017, 4(1-2): 1-119.

[10] Watch what I do: programming by demonstration[M]. MIT press, 1993.

**2018-2019**
**Coursework submission**
**Cover sheet**

**Student name……**

**College……**

**CRSID……**

**Submit to Graduate Education**
*Date of submission*

---

**Module name**

**Module Code No.**

---

**Assignment No.**

**Assignment mark/grade**

---

**Assessor comments**

GE Records……..

---

*GE stamp date and initials*

**CRSID……**

**Module No……**

**Assignment No……**

# Interaction with Machine Learning
# Diary 3

Zhuoni Jie
zj245@cam.ac.uk

February 12, 2019

## 1 Summary of the contents of the lecture

This week's lecture was about Mixed Initiative Interaction.

First, a demo of a classic example of mixed initiative, Dasher, was introduced. Mixed-Initiative User Interfaces advocate elegant coupling of automated services with direct manipulation. According to their principles, autonomous actions should be taken only when an agent believes that they will have greater expected value than inaction for the user. To add value with automation when designing such user interfaces, we should: consider uncertainty about user's goals; consider status of user's attention in timing services; infer ideal action in light of costs, benefits, and uncertainties; employ dialog to resolve key uncertainties; allow efficient direct invocation and termination; minimize cost of poor guesses about action and timing.

Then a probabilistic view to infer user's goal based on observed evidence was introduced. Expected utility of automated action and expected utility threshold for action were explained to give an idea of how the systems react to user behavior based on expected utility. Bayesian inference of user intention were then introduced to provide a probablistic view of interaction. Another bad classic example of mixed initiative Clippy was then discussed, which did not set proper expected utility threshold for action, making it obtrusive and unwelcome. A kind of unobtrusive direct manipulation strategy is semantic pointing, which can be implemented in alert dialogues. Another strategy is gesture keyboard, demonstrated with another example.

Mixed-Initiative Interaction is a negotiation of my ideas with system's ideas, and concerns with two things - automated services and direct manipulation. Unlike traditional view in HCI that action values will always have greater values than inactions, we should bear in mind that automating has a cost of distracting the user. Even a dialogue can be costly to a user.

Then information flow in mixed initiatives was then discussed. In an autonomous vehicle case, to define system boundaries, where information enters the system was discussed. We should notice that even if the system includes "autonomous" closed loop control algorithms, information is acquired through more or less costly interactive processes outside the system boundary. All closed loop control systems do machine learning, but as interaction with such systems becomes routine, these cybernetic components are no longer considered intelligent. Three ways of how human interact with information flow and the system were discussed: conventional system design, hybrid system design, and human-centric system design.

Agency and control were discussed. The experience of agency is defined as the experience of controlling one's own actions and, through this control, affecting the external world. Passivity phenomena in schizophrenia shows a contradiction of fact with the experience of agency. As stated by a golden rule in HCI field, users strongly desire the sense that they are in charge of the system and that the system responds to their actions. To develop a research agenda, an implicit metric to measure peoples' experience of agency was developed. Two experiments that apply this metric in HCI contexts were introduced, including Intentional binding and The Libet clock method. Interval estimation was also discussed with strengths and weaknesses. An experimental manipulation, Skinput (1), was introduced. Its two experimental observations found changes in the input modality and in levels of assistance can have a significant impact on users' experience of personal agency. Intentional binding can provide an implicit metric for probing and mapping experiences of agency, and this metric can be applied is a wide range of design contexts. Conda (2), which is an example of design for control, was introduced. It is a Mixed initiative interface being created for Africa's Voices Foundation

UI should be more accurate to react to people's intentions, and make users less stressed. People as agency must be in control. One thing to notice in experimental study is that people may tell what you want to hear, and may view you as a teacher. We should test mental model not explicitly - not simply ask them "what do you think is going on?"

## 2 Relations with some of the other literature

There are traditionally two groups of views in HCI field. One group of researchers center on building machinery for sensing a user's activity and taking automated actions (3)(4)(5), while another group of researchers explore new kinds of metaphors and conventions that enhance a user's ability to directly manipulate interfaces to access information and invoke services (6)(7). There is great opportunity for designing innovative user interfaces, and new HCI modalities by synthesizing the two ideas and considering designs that take advantage of the power of direct manipulation and potentially valuable automated reasoning

(8). Mixed-initiative interaction is different from both of the views, and is a key aspect of effective human-computer interaction and has great potential to affect work on multiagent systems. Mixed-initiative refers to a flexible interaction strategy, where each agent can contribute to the task what it does best, having the initiative to control or alternatively assist the tasks according to required interactions (9).

In interactive machine learning domain, researchers have investigated mixed-initiative interfaces (10) that let humans provide proper feedback and domain knowledge to machine learning algorithms (11)(12)(13). For example, CueFlik (14) allows end-users to locate images on the web through a combination of keyword search and iterative example-based concept refinement activities. MindMiner (15) is a mixed-initiative interface for collecting and learning subjective similarity measurements from users via a combination of new interaction techniques and machine learning algorithms. Apolo (16) is an interactive sense making system intended to recommend new nodes in a large network by letting users specify exemplars for intended groups.

There are also researches studying how mixed-initiative interfaces can be used in user modeling. One research investigates how AutoTutor (17), which is a frequent conversation patterns from a mixed-initiative dialogue with an intelligent tutoring system, can significantly predict users' affective states (e.g. confusion, eureka, frustration).

# 3 Opportunities for further research and interesting questions/conflicts in this topic

Mixed-initiative interaction can be implemented in a broad spectrum of collaborative problem solving marked by an interleaving of contributions by different participants, which brings possible challenges and opportunities. When people are working together to solve a mutual goal, they need to converge on some common understanding of beliefs about the setting, activity, goals, and the nature and timing of their individual contributions. Group activities can be complicated with diverse group structures, people from different technical and cultural backgrounds, power distances in groups, the focus of attention and comprehension of the participants, the nature of the problem to be solved, and about abilities and intentions to contribute to the solution in different ways. In these collaborative settings, designing mechanisms of interactions, inferring expected utility, and determining threshold for agent action can be complicated and challenging. Designing and studying computing systems to assist effective collaboration via sensing, reasoning, and dialog about context and intentions, is a challenge for fluid, general mixed-initiative interaction.

When users try problem solving with the assistance of mixed-initiative systems, another challenge is to provide systems with the abilities to recognize problem-

solving opportunities, including opportunities outside the scope of someone's current focus of attention, and to understand where automated capabilities might complement human skills in solving the problems in a useful and desirable manner. In addition, considering long-term multiple interaction stages, mixed-initiative systems may also benefit from skills that enable them to decompose problems into sets of sub-problems and to consider how people and machines might each contribute in symphony or sequentially to solving the sub-problems, and how to synthesize the results of problem solving into larger solutions.

# References


[1] Harrison, Chris, Desney Tan, and Dan Morris. "Skinput: appropriating the body as an input surface." Proceedings of the SIGCHI conference on human factors in computing systems. ACM, 2010.

[2] http://www.africasvoices.org/ideas/newsblog/introducing-our-latest-analysis-tool-coda/

[3] Heckerman, David, and Eric Horvitz. "Inferring informational goals from free-text queries: A Bayesian approach." Proceedings of the Fourteenth conference on Uncertainty in artificial intelligence. Morgan Kaufmann Publishers Inc., 1998.

[4] Horvitz, Eric, and Matthew Barry. "Display of information for time-critical decision making." Proceedings of the Eleventh conference on Uncertainty in artificial intelligence. Morgan Kaufmann Publishers Inc., 1995.

[5] Horvitz, Eric, et al. "The Lumiere project: Bayesian user modeling for inferring the goals and needs of software users." Proceedings of the Fourteenth conference on Uncertainty in artificial intelligence. Morgan Kaufmann Publishers Inc., 1998.

[6] Lieberman, Henry. "Letizia: An agent that assists web browsing." IJCAI (1) 1995 (1995): 924-929.

[7] Maes, Pattie. "Agents that reduce work and information overload." Readings in Human–Computer Interaction. 1995. 811-821.

[8] Birnbaum, Larry, et al. "Compelling intelligent user interfaces—how much AI?." Proceedings of the 2nd international conference on Intelligent user interfaces. ACM, 1997.

[9] Allen, J. E., Curry I. Guinn, and E. Horvtz. "Mixed-initiative interaction." IEEE Intelligent Systems and their Applications 14.5 (1999): 14-23.

[10] Horvitz, Eric. "Principles of mixed-initiative user interfaces." Proceedings of the SIGCHI conference on Human Factors in Computing Systems. ACM, 1999.

[11] Amershi, Saleema, et al. "Effective End-User Interaction with Machine Learning." AAAI. 2011.

[12] Amershi, Saleema, et al. "Overview based example selection in end user interactive concept learning." Proceedings of the 22nd annual ACM symposium on User interface software and technology. ACM, 2009.

[13] Chau, Duen Horng, et al. "Apolo: making sense of large network data by combining rich user interaction and machine learning." Proceedings of the SIGCHI Conference on Human Factors in Computing Systems. ACM, 2011.

[14] Fogarty, James, et al. "CueFlik: interactive concept learning in image search." Proceedings of the sigchi conference on human factors in computing systems. ACM, 2008.

[15] Fan, Xiangmin, et al. "Mindminer: A mixed-initiative interface for interactive distance metric learning." Human-Computer Interaction. Springer, Cham, 2015.

[16] Chau, Duen Horng, et al. "Apolo: making sense of large network data by combining rich user interaction and machine learning." Proceedings of the SIGCHI Conference on Human Factors in Computing Systems. ACM, 2011.

[17] Graesser, Arthur C., et al. "AutoTutor: An intelligent tutoring system with mixed-initiative dialogue." IEEE Transactions on Education 48.4 (2005): 612-618.

**2018-2019**
**Coursework submission**
**Cover sheet**

**Student name……**

**College……**

**CRSID……**

**Submit to Graduate Education**
*Date of submission*

---

**Module name**

**Module Code No.**

---

**Assignment No.**

**Assignment mark/grade**

---

**Assessor comments**

GE Records……..

---

*GE stamp date and initials*

**CRSID……**

**Module No……**

**Assignment No……**

# Interaction with Machine Learning
# Diary 4

Zhuoni Jie
zj245@cam.ac.uk

February 17, 2019

## 1 Summary of the contents of the lecture

This week's lecture was about interpretability in machine learning.

Apart from a traditional quantifiable goal - accuracy, there are other goals in machine learning such as safety, fairness, and interpretability. There are often trade-offs among these goals. In the ML context, interpretability means the ability to explain or to present in understandable terms to humans. We are interested in investigating when and how to automatically generate good interpretation, and how to evaluate interpretability.

ML interpretability can be categorized. From a task-oriented perspective, it can be categorized into global and local interpretability. From a method-related perspective, it can be categorized into raw features, derived features that have some semantic meaning to the expert (e.g. pixels in faces), and prototypes. The type of the explanation should be matched with the nature of data/tasks.

Two related papers were introduced. For the first research ([1]), the main idea is to visualize the response of a deep neural network to a specific input using a powerful conditional, multivariate model. For an individual classifier prediction, each feature is assigned with a defined relevance value reflecting its contribution towards or against the predicted class, and this relevance is visualized. This can help to explain why a decision was made, to understand strengths and limitations of a model for further improvement, and to lead to more understanding in less understood areas. Then the approach to calculating the relevance value and experimental evaluations were explained and discussed.

The second research ([2]) discusses generative models and interpretability. The main idea is they define interpretability as a simple relationship to something humans can understand, so a latent space is more interpretable if it manages to explain the relationship to salient attributes more simply. They propose an interpretability framework as a lens on an existing model using fully invertible

transformations, an active learning methodology basing the acquisition function on mutual information with interpretable data attributes, a quantitative metric which defines interpretability as a simple relationship to something humans can understand, and a second interpretability framework jointly optimized for reconstruction and interpretability, and did both qualitative and quantitative evaluations. This is a strategy to bring human subjectivity into interpretability to yield interactive 'human-in-the-loop' interpretability.

## 2 Relations with some of the other literature

Some ML models are inherently interpretable such as sparse linear classifiers used by LIME (3). To develop interpretation models for more complex ML models with more parameters and hidden relationshios such as deep neural models, researchers focus on these methods: conveying and visualizing the uncertainty of a prediction, feature-based interpretations, and example-based interpretations.

*Conveying Uncertainty methods* augment the prediction from a neural network classifier with a confidence score conveying the uncertainty of the model. In a cooperative setting, the uncertainty helps humans decide to trust the model or not (4)(5). To make it more informative, we can also display the confidence scores for the classes other than the top one (6). Estimating uncertainty for a deep neural model can be challenging: due to overfitting, its confidence can be overly high and requires calibration (7). Sometimes uncertainty estimate needs to be combined with anomaly detection to be robust (8).

*Feature-based Interpretations* methods work with the idea that model predictions can be explained by highlighting the most salient features in the input, typically visualized by a saliency map. For a linear classifier, the most salient features are the ones with the largest corresponding weights. For non-linear classifiers, the saliency map can be calculated by the gradient of the loss function with respect to each input feature (9). Alternatively, one can locally approximate a non-linear classifier with a simpler linear model, then use the weights to explain the predictions from the non-linear model (10). Simonyan et al. propose a class saliency visualization method (11). They measure how sensitive the classification score is to small changes in pixel values, by computing the partial derivative of the class score with respect to the input features using standard backpropagation. Shrikumar et al. compare the activation of a unit when a specific input is fed forward through the net to a reference activation for that unit (12).

Inspired by explaining by example, the core idea for *Example-based Interpretations* methods is to find the most influential training examples for the prediction on a test example. The influential examples can be found by nearest neighbor search in the representation space, which is natural to clustering algorithms and their deep variation (13). Examples can also be found according to other

definitions of importance, for example, influence functions to find examples for neural models (14).

# 3 Opportunities for further research and interesting questions/conflicts in this topic

Task-related latent dimensions of interpretability can be an interesting area to explore, which can help studying what might make different tasks similar in interpretation needs and how to accordingly design for their interpretation needs. Some seemingly different applications may share some common categories. For example, an application involving preventing medical error at the bedside and an application involving support for identifying inappropriate language on social media might be similar in that they involve making a decision about a specific case in a relatively short period of time (15). Here with time constrants, the interpretation needs in these scenarios might be different from an application which requires thorough understanding of a large dataset to generate scientific insights. Apart from time constraints, other factors may include the needs of global/local interpretability, severity of incompleteness (if the interpretation is incomplete, what outcomes it might bring), cognitive burdens, and nature of user expertise. Human-grounded experiments in tasks need to be conducted to determine which factors are most useful in helping defining interpretation needs.

For human-in-the-loop systems, investigating how users are affected by complexity, unpredictability or mis-prediction, and lack of control for the system is important. Balancing the trade-off of these factors and implementing disparate interpretation methods need to be studied. Gajos et al. (16) showed that increasing predictability and accuracy lead to improved satisfaction, while Kangasraasio et al. (17) showed that allowing users to see the predicted effects of an action before committing to it can improve task performance and acceptance. Users of PeerFinder — a tool that recommends similar students based on academic profiles — were more confident and engaged when given more control even with the negative effect of added complexity (18). More exploration in respect to simulated/real tasks, simplified/full tasks, and tasks with different interpretation needs, need to be investigated.

# References


[1] Zintgraf, Luisa M., et al. "Visualizing deep neural network decisions: Prediction difference analysis." arXiv preprint arXiv:1702.04595 (2017).

[2] Adel, Tameem, Zoubin Ghahramani, and Adrian Weller. "Discovering interpretable representations for both deep generative and discriminative models." International Conference on Machine Learning. 2018.

[3] Antifakos, Stavros, et al. "Towards improving trust in context-aware systems by displaying system confidence." Proceedings of the 7th international conference on Human computer interaction with mobile devices & services. ACM, 2005.

[4] Rukzio, Enrico, et al. "Visualization of uncertainty in context aware mobile applications." Proceedings of the 8th conference on Human-computer interaction with mobile devices and services. ACM, 2006.

[5] Liu, Shixia, et al. "Towards better analysis of machine learning models: A visual analytics perspective." Visual Informatics 1.1 (2017): 48-56.

[6] Guo, Chuan, et al. "On calibration of modern neural networks." Proceedings of the 34th International Conference on Machine Learning-Volume 70. JMLR. org, 2017.

[7] Hendrycks, Dan, and Kevin Gimpel. "A baseline for detecting misclassified and out-of-distribution examples in neural networks." arXiv preprint arXiv:1610.02136 (2016).

[8] Birnbaum, Larry, et al. "Compelling intelligent user interfaces—how much AI?." Proceedings of the 2nd international conference on Intelligent user interfaces. ACM, 1997.

[9] Simonyan, Karen, Andrea Vedaldi, and Andrew Zisserman. "Deep inside convolutional networks: Visualising image classification models and saliency maps." arXiv preprint arXiv:1312.6034 (2013).

[10] Ribeiro, Marco Tulio, Sameer Singh, and Carlos Guestrin. "Why should i trust you?: Explaining the predictions of any classifier." Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining. ACM, 2016.

[11] Simonyan, Karen, Andrea Vedaldi, and Andrew Zisserman. "Deep inside convolutional networks: Visualising image classification models and saliency maps." arXiv preprint arXiv:1312.6034 (2013).

[12] Shrikumar, Avanti, et al. "Not just a black box: Learning important features through propagating activation differences." arXiv preprint arXiv:1605.01713 (2016).

[13] Papernot, Nicolas, and Patrick McDaniel. "Deep k-nearest neighbors: Towards confident, interpretable and robust deep learning." arXiv preprint arXiv:1803.04765 (2018).

[14] Koh, Pang Wei, and Percy Liang. "Understanding black-box predictions via influence functions." Proceedings of the 34th International Conference on Machine Learning-Volume 70. JMLR. org, 2017.

[15] Doshi-Velez, Finale, and Been Kim. "Towards a rigorous science of interpretable machine learning." arXiv preprint arXiv:1702.08608 (2017).

[16] Gajos, Krzysztof Z., et al. "Predictability and accuracy in adaptive user interfaces." Proceedings of the SIGCHI Conference on Human Factors in Computing Systems. ACM, 2008.

[17] Kangasrääsiö, Antti, Dorota Glowacka, and Samuel Kaski. "Improving controllability and predictability of interactive recommendation interfaces for exploratory search." Proceedings of the 20th international conference on intelligent user interfaces. ACM, 2015.

[18] Du, Fan, et al. "Finding similar people to guide life choices: Challenge, design, and evaluation." Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems. ACM, 2017.

**2018-2019**
**Coursework submission**
**Cover sheet**

**Student name……**

**College……**

**CRSID……**

**Submit to Graduate Education**
*Date of submission*

---

**Module name**

**Module Code No.**

---

**Assignment No.**

**Assignment mark/grade**

---

**Assessor comments**

GE Records……..

---

*GE stamp date and initials*

**CRSID……**

**Module No……**

**Assignment No……**

# Interaction with Machine Learning
# Diary 5

Zhuoni Jie
zj245@cam.ac.uk

February 24, 2019

## 1 Summary of the contents of the lecture

This week's lecture was about labelling.

Labels from traditional statistical philosophy capture mathematical and objective characteristics of the data. However, when faced with the irregular, noisy, and subjective application domains of human-centric systems, this assumption unfortunately produces numerous challenges which can result in both a poor user experience as well as poorer resultant models. To address this problem, end-user activity of labelling needs to be studied and designed. This raises our attention to the human-centric approach to labelling.

The Cranyons ([1]) enables end-users to build image segmentation classifiers using a brush tool to train the model, and allows further annotation on misclassifies areas. Another example of an end-user controlled interactive machine learning tool is Elucidebug ([2]). Both systems involve a training loop, where the user provides annotations either in the form of training examples or potentially by manually adjusting model parameters. Next, a model is trained, and the model output will suggest which further annotation or adjustment actions would be useful. By providing labelled training instances that exemplify how the system should behave, users are forced to abide by statistical assumptions of supervised ML models embedded in IML systems.

Labelling could be viewed as programming or model construction. From the model construction perspective, labelling tries to uncover some underlying truth for the users and build models based on the users' intentions. From a techno-pragmatist view, the interaction is designed around satisfying the technical needs of statistical models. The purpose of the user, within the overall system design, is to satisfy the requirement for an 'objective' function, encoding the underlying 'law', in which the labels provided by the user define the 'ground truth' of that law. But IML is often different from objective automation, and cannot

be determined by direct measurement. It usually involves humanist interpretations, and is inherently subjective. The programming perspective suggests that the user wants the system to behave in a certain way, and is training it to do so. A label is an instruction to the system, and users' intentional creative acts are statistically encoded into the system.

Human judgements can be categorized as: perceptual judgements, judgements that reflect domain expertise, judgements of patterns in human experience (universal aspects of human experience. E.g., affect judgements. The challenges are peoples' age, gender, culture), and jusgement of patterns in individual intent (inherently subjective unlike the former three). The human origins of data can bring problems. There are ethical challenges of data collection, and label quality usually depends a lot on the labeller. Individual intents usually involve self-reporting, which brings issues including motivation and consistency. Some applications require fast convergence ([3]) ([4]). And distinction between unclear labels and unclear label boundaries has a very big difference between ML perspective and users'.Outliers and 'unrateables', and incorrect framing of regression as classification are also discussed. Accommodating flexibility is illustrated with two examples. Humans are fallible, and sometimes can lack consistency and stamina. Some labelling methods embrace error to improve speed. Measuring label reliability uses inter- and intra-rater reliability.

Two case studies were discussed. Case study 1 was about structured labelling for concept evolution. Label concepts evolve and drift over time. Users' interests are changing. Sometimes methods such as discarding information or using moving windows are needed. The system should help users to refine the model.Case study 2 was about sorting movement assessments. The problem is to create consistent labels, and numeric scoring has poor labeller agreement. A partial solution is to use preference judgements. However, it is not scalable. A better solution is to use setwise comparison with TrueSkill inference. Then the interface, SorTable, was introduced and discussed, which infers the labels and eases the burden of labelling.

## 2 Relations with some of the other literature

Automatically comparing images based on visual properties is inherently costly. Liang and Grauman proposed a setwise active learning method for training relative attribute ranking functions, with the goal of requesting human comparisons only where they are most informative ([5]). They introduce a novel criterion (diverse setwise low margin criterion) that requests a partial ordering for a set of examples that minimizes the total rank margin in attribute space, subject to a visual diversity constraint. Unlike previous work such as the "crowd kernel" method ([6]), they actively select comparisons on a describable property, so as to efficiently learn a predictive function that can estimate attribute strength in any new image. The developed setwise criterion helps amortize effort by identify-

ing mutually informative comparisons, and the diversity requirement safeguards against requests a human viewer will find ambiguous. They develop an efficient strategy to search for sets that meet this criterion.

Existing crowdsourcing schemes are too expensive to scale up with the expanding volume of data. Therefore, providing rapid judgements for classification labels is important. Krishna et al. present a technique that produces extremely rapid judgments for binary and categorical labels using crowdsourcing by embracing acceptable and even expected errors ([7]). Most previous work speeds labelling by punishing errors hashly ([8])([9]), but the traditional way to punish all errors actually causes workers to proceed slowly and deliberately. This work also demonstrates that it is possible to rectify these errors by randomizing task order and modeling response latency.

Improving label quality is of great importance, especially for crowdsourcing approaches where creating comprehensive guidelines is often prohibitive. Chang et al. propose Revolt, a collaborative approach that brings ideas from expert annotation workflows to crowd-based labeling ([10]). Revolt eliminates the burden of creating detailed label guidelines by harnessing crowd disagreements to identify ambiguous concepts and create groups of semantically related items for post-hoc label decisions.

Existing crowdsourced clustering approaches have difficulties supporting the global context needed for workers to generate meaningful categories. To reduce the needs of human judgments, Chang et al. introduce Alloy, a hybrid approach that combines the richness of human judgments with the power of machine algorithms ([11]). Alloy supports greater global context through a new "sample and search" crowd pattern which changes the crowd's task from classifying a fixed subset of items to actively sampling and querying the entire dataset. It also improves efficiency through a two phase process in which crowds provide examples to help a machine cluster the head of the distribution, then crowd workers classify the more difficult low-confidence examples in the tail. To accomplish this, Alloy introduces a modular "cast and gather" approach which leverages a machine learning backbone to stitch together different types of judgment tasks.

# 3 Opportunities for further research and interesting questions/conflicts in this topic

Ensuring the quality of obtained data is one of the primary problems in crowdsourcing. In crowdsourced labelling task, delivering better results from individual workers is one of the approaches to ensure label quality. Shah and Zhou proposed a two-stage setting for crowdsourced tasks, named self-correction, which shows other workers' task results to each worker after the results are submitted, allowing the worker to update his/her results ([12]). Kobayashi et al. studied the effectiveness of self-correction in a real crowdsourcing setting, and observed

involuntary short- and long-term perceptual learning effects in self-correction microtasks (13). They found workers notice mistakes they made in the first stage (in the same task), and there were quality improvements in a successive sequence of similar but different tasks.

In this work, only visual categorization tasks were explored. The effectiveness of self-correction in other tasks and with different question difficulties should also be studied. There are also other questions in studying human factors in crowdsourcing. Studying the effects of the varied quality of data shown to the workers in the second stage, studying the different effects of allowing varied number of times a question can be self-corrected, and designing better incentives for workers to treat self-correction seriously are of future research interests. Providing references can inherently lead to possible bias to the label task, studying the bias in this kind of group-decision making is also complex. Moreover, we can also investigate why workers sometimes change their answers to emulate the self-correction reference answer. Clarifying the factors that lead workers to revise their answers will contribute to many applications.

# References


[1] Fails, Jerry Alan, and Dan R. Olsen Jr. "Interactive machine learning." Proceedings of the 8th international conference on Intelligent user interfaces. ACM, 2003.

[2] Kulesza, Todd, et al. "Principles of explanatory debugging to personalize interactive machine learning." Proceedings of the 20th international conference on intelligent user interfaces. ACM, 2015.

[3] Herbrich, Ralf, Tom Minka, and Thore Graepel. "TrueSkill$^{TM}$: a Bayesian skill rating system." Advances in neural information processing systems. 2007.

[4] Fei-Fei, Li, Rob Fergus, and Pietro Perona. "One-shot learning of object categories." IEEE transactions on pattern analysis and machine intelligence 28.4 (2006): 594-611.

[5] Liang, Lucy, and Kristen Grauman. "Beyond comparing image pairs: Setwise active learning for relative attributes." Proceedings of the IEEE conference on Computer Vision and Pattern Recognition. 2014.

[6] Tamuz, Omer, et al. "Adaptively learning the crowd kernel." arXiv preprint arXiv:1105.1033 (2011).

[7] Krishna, Ranjay A., et al. "Embracing error to enable rapid crowdsourcing." Proceedings of the 2016 CHI conference on human factors in computing systems. ACM, 2016.

[8] Sheng, Victor S., Foster Provost, and Panagiotis G. Ipeirotis. "Get another

label? improving data quality and data mining using multiple, noisy labelers." Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, 2008.

[9] Smyth P, Burl M C, Fayyad U M, et al. Knowledge Discovery in Large Image Databases: Dealing with Uncertainties in Ground Truth[C]//KDD workshop. 1994: 109-120. Smyth, Padhraic, et al. "Knowledge Discovery in Large Image Databases: Dealing with Uncertainties in Ground Truth." KDD workshop. 1994.

[10] Chang, Joseph Chee, Saleema Amershi, and Ece Kamar. "Revolt: Collaborative crowdsourcing for labeling machine learning datasets." Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems. ACM, 2017.

[11] Chang, Joseph Chee, Aniket Kittur, and Nathan Hahn. "Alloy: Clustering with crowds and computation." Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems. ACM, 2016.

[12] Kobayashi, Masaki, et al. "An empirical study on short-and long-term effects of self-correction in crowdsourced microtasks." Sixth AAAI Conference on Human Computation and Crowdsourcing. 2018.

[13] Shah, Nihar, and Dengyong Zhou. "No oops, you won't do it again: Mechanisms for self-correction in crowdsourcing." International conference on machine learning. 2016.

**2018-2019**
**Coursework submission**
**Cover sheet**

**Student name……**

**College……**

**CRSID……**

**Submit to Graduate Education**
*Date of submission*

---

**Module name**

**Module Code No.**

---

**Assignment No.**

**Assignment mark/grade**

---

**Assessor comments**

GE Records……..

---

*GE stamp date and initials*

**CRSID……**

**Module No……**

**Assignment No……**

# Interaction with Machine Learning
# Diary 6

Zhuoni Jie
zj245@cam.ac.uk

March 3, 2019

## 1 Summary of the contents of the lecture

This week's lecture was about risks and bias in machine learning.

In some situations, people are more likely to trust decisions made by machines than by other people, and are less likely to exercise meaningful review of or identify problems with automated systems. These kinds of automation bias can cause problems for engineers, users, and reviewers. However, it is possible to remove bias from ML systems. The opacity which inherently lies in machine learning brings problems for accountability and oversight, introducing problems in design and engineering.

Technology may be non-neutral as a result of its design, reflecting some goal of its creator, and cannot act independently of human action. Technology including ML systems is inherently normative. Users, engineers, and designers should think about in what context and for what purpose could a given ML system be used.

Then errors in ML systems were discussed. ML systems will make mistakes and these mistakes will have consequences, and training is a process to make models have an acceptable margin of error. Engineers need to think about detecting, rectifying, and accommodating errors.

Bias of technology and ML systems have been studied in many cases (1), and people are searching for possible ways to fix it (2). There are a lot of and different kinds of bias in ML systems. Sometimes particular groups are or historically were treated less favourably, and model may repeat this difference in treatment. Sometimes particular groups are or were societally disadvantaged, and model may repeat the disadvantage. Sometimes training data is not sufficiently varied for the system to have been trained to adequately handle all possible inputs, which may result in models which are incapable of dealing with certain inputs equally to others. Issues such as historical problems, prejudices and bias about

gender, disability, LGBT, races, can sometimes be encoded into and cause serious problems in trained ML and artificial intelligent systems (3). ML systems are limited by their training data, and those poorest, most marginalized, and most vulnerable models are most likely to be affected.

Discrimination is another important problem in ML systems. 'Fair' systems can still be discriminatory. Predictive privacy harms include inaccurate predictions and wrong-person disclosures, which can feed into discriminatory actions and other problems. ML's usages in surveillance and in problem solutions were also discussed, also bringing problems to certain fields.

In a word, machine learning problems are human problems with human responsibility, and problems can only be avoided if engineers know about the risks and proactively take steps to avoid them.

## 2 Relations with some of the other literature

With rapid progress in ML and AI technologies, increasing studies have investigated the potential impacts of technologies on society. Amodei et al. (4) studied unintended and harmful behavior that may emerge from poor design of real-world AI systems. They presented five research problems related to accident risk, categorized according to whether the problem originates from having the wrong objective function, an objective function that is too expensive to evaluate frequently, or undesirable behavior during the learning process. They also considered the high-level question of how to think most productively about the safety of forward-looking applications of AI.

Leike et al. (5) focused on AI safety by presenting a suite of reinforcement learning environments called *gridworlds* illustrating different problems: safe interruptibility (6), avoiding side effects (7), absent supervisor (8), reward gaming (9), safe exploration (10), as well as robustness to self-modification, distributional shift (11), and adversaries (12) (13). They evaluated A2C and Rainbow, two recent deep reinforcement learning agents, on the created environments and show that they are not able to solve the AI safety problems satisfactorily.

Among the aforementioned AI safety questions, Orseau and Armstrong focused on safely interruptible agents. Their work (6) explores a way to make sure a learning agent will not learn to prevent or seek being interrupted by the environment or a human operator which may be resulted from reinforcement learning. They provided a formal definition of safe interruptibility and exploited the off-policy learning property to prove that either some agents are already safely interruptible, like Q-learning, or can easily be made so, like Sarsa. They showed that even ideal, uncomputable reinforcement learning agents for deterministic general computable environments can be made safely interruptible.

# 3 Opportunities for further research and interesting questions/conflicts in this topic

Brundage et al. (14) discussed the malicious use of AI technologies. As AI capabilities become more powerful and widespread, the growing use of AI systems may bring new threats, expand and change existing threats, leading to potential new vulnerabilities attacks to the systems. The malicious use of AI will impact how we construct and manage our digital infrastructure as well as how we design and distribute AI systems, and will likely require policy and other institutional responses. Investigating how to forecast, prevent, and mitigate the harmful effects of malicious uses of AI when necessary is of great importance.

Many real-world AI applications such as automotive driving and personal assistants require modeling phenomena. However, it is impossible and unnecessary to model every phenomenon, and an AI system is expected to act without having a complete model of the world. Expanding models, learning causal models, developing a portfolio of models, and developing monitoring of anomalies are general ways to deal with to develop robust AI systems with unmodeled phenomena. When developing these systems, unobtrusively detecting users' characterstics such as gender, age, disability, cultural background, personality, and encoding these attributes into model inputs, are important to make models responsive and adaptive to varied real-world scenes. When constructing knowledge base and learning reactions, AI systems are expected to be designed for varied user groups in complex interaction scenarios, thus introducing potential errors. In this way, the more ideal model may not be the one with lowest error rate.

# References


[1] https://www.propublica.org/article/machine-bias-risk-assessments-in-criminal-sentencing

[2] https://fivethirtyeight.com/features/technology-is-biased-too-how-do-we-fix-it/

[3] https://www.theverge.com/2016/3/24/11297050/tay-microsoft-chatbot-racist

[4] Amodei, Dario, et al. "Concrete problems in AI safety." arXiv preprint arXiv:1606.06565 (2016).

[5] Leike, Jan, et al. "AI safety gridworlds." arXiv preprint arXiv:1711.09883 (2017).

[6] Orseau, Laurent, and M. S. Armstrong. "Safely interruptible agents." (2016).

[7] Amodei, Dario, et al. "Concrete problems in AI safety." arXiv preprint arXiv:1606.06565 (2016).

[8] Stuart Armstrong. AI toy control problem, 2017. https://www.youtube.com/watch?v=sx8JkdbNgdU.

[9] Jack Clark and Dario Amodei. Faulty reward functions in the wild, 2016. https://blog.openai.com/ faulty- rewardfunctions/.

[10] Pecka, Martin, and Tomas Svoboda. "Safe exploration techniques for reinforcement learning–an overview." International Workshop on Modelling and Simulation for Autonomous Systems. Springer, Cham, 2014.

[11] Quionero-Candela, Joaquin, et al. Dataset shift in machine learning. The MIT Press, 2009.

[12] Auer, Peter, et al. "The nonstochastic multiarmed bandit problem." SIAM journal on computing 32.1 (2002): 48-77.

[13] Szegedy, Christian, et al. "Intriguing properties of neural networks." arXiv preprint arXiv:1312.6199 (2013).

[14] Brundage, Miles, et al. "The malicious use of artificial intelligence: forecasting, prevention, and mitigation." arXiv preprint arXiv:1802.07228 (2018).

**2018-2019**
**Coursework submission**
**Cover sheet**

**Student name……**

**College……**

**CRSID……**

**Submit to Graduate Education**
*Date of submission*

---

**Module name**

**Module Code No.**

---

**Assignment No.**

**Assignment mark/grade**

---

**Assessor comments**

GE Records……..

---

*GE stamp date and initials*

**CRSID……**

**Module No……**

**Assignment No……**

# Interaction with Machine Learning
# Diary 7

Zhuoni Jie
zj245@cam.ac.uk

March 11, 2019

## 1 Summary of the contents of the lecture

This week's lecture was about visualization.

Several classic visualization diagrams including Time Series area chart, Lifespan chart, Bar chart, Cholera map, and two diagrams used in army were introduced. Each of the technique has features supporting their specific utilities. Visualization techniques include charts, statistical visualizations, typography and typesetting, diagrams, illustrations and drawings, infographics, symbols, as well as marks. There are also some artistic works that go beyond traditional visualization methods and convey rich information. Several innovative cases such as Designing English and A Humument were introduced and discussed.

Some theories of visualization have been studied and formulated. Principles of visualization can be categorized as structural, perceptual/cognitive, and aesthetic/designerly. Gestalt principles of visual perception introduce a set of principles of account for the observation that humans naturally perceive objects as organized patterns and objects. Among these principles, for example, the law of common fate has been used extensively in user interface design. The graphic resources, correspondence, as well as design uses of marks, symbols, regions, and surfaces, have been studied and discussed in groups. The grammar of graphics is the foundation of R package ggplot, and has been simplified and adapted to suit the mental model and usability of users. The directionality of data to visualisation in the grammar of graphics can also be limiting, and there is also research looking into visualisation to data to clarify hypotheses ([1]). The role of interaction and its costs in visualization has also been studied ([2])([3]).

Different visualization methods can be used in latent semantic analysis. In one example, semantic zooming and graphical interaction histories can be used to present both overview and detail. Heatmaps and expansion tree were also introduced. Braincel explores the effect of lightness scaling. Gatherminer uses visualization methods including color-mapped matrix, bar graphs, decision trees,

thumbnail scroll bar, gathering, for overview, showing details, reordering, annotating, and explaining.

## 2 Relations with some of the other literature

Detecting interesting patterns in time series data and developing model-drive explanation are explored by Sarkar et al. (4). The authors proposed Gatherminer, which directly addresses drawbacks in visual discovery and explaining using a compact visualization scheme, automated rearrangement, and explanations driven by machine learning. There are many other techniques in visualizations for time series. Bertin proposed a visual procedure where pieces of paper representing rows of a matrix were cut and manually reordered on a flat surface to reshuffle data series (5). Elmqvist et al. incorporated a significant reordering step in their "Zoomable Adjacency Matrix Explorer" (6). Mansmann et al. explored the use of correlationbased arrangements of time series for movement analysis in behavioural ecology (7). Unlike Gatherminer, these techniques do not build on the rearranged visualization to present visual analyzing of attributes and patterns in time series.

Interaction and visualization for large high-dimensional data have been studied. The number of possible data representations grows exponentially with the amount of data dimensions. Also, not all views from a possibly large view space are potentially relevant to a given analysis task or user. Some overview approaches generate effective layouts to efficiently spot patterns of interest, such as Value-and-Relation display (8) and using small slyohs to show time series (9). A number of automatic filtering of views for potential structures of interest has also been proposed, such as ScagExplorer(10) and the Scagnostics approach (11). Based on these approaches, Behrisch et al. introduced a new framework for a feedback-driven view exploration which is inspired by relevance feedback approaches used in Information Retrieval (12). Self et al. (13) explored and studied the differences, advantages, and drawbacks among parametric interaction, observation-level interaction, and their combination in high-dimensional data analysis using the tool Andromeda, assessing these techniques' effects on domain-specific high-dimensional data analyses performed by non-experts of statistical algorithms.

Interactive visualization systems to support user exploration of machine learning classifier analysis have been studied. Talbot et al. proposed EnsembleMatrix (14), an interactive visualization system that presents a graphical view of confusion matrices to help users understand relative merits of various classifiers. Unlike previous visualization methods which are tied to specific algorithms such as (15)(16), this system supports comparisons across algorithm types.

# 3 Opportunities for further research and interesting questions/conflicts in this topic

For visual data exploration and analysis, current theories and classification schemes provide some initial insight on which techniques are oriented to certain data types, but users still do not know for sure what makes a visualization technique more suitable than others to explore a particular data set. Selection of a system/techniques seem to be largely dependent on the specific task, data type, and is largely intuitive. Users have to implement past knowledge, experience, and have to compare various techniques to weight their relative strengths and weakness.

There are some works comparing different visualization techniques by rating their capabilities in terms of data characteristics, tasks supported, and visualization characteristics ([17]) ([18]). However, these comparisons exclude user efforts and do not consider personal experience. And there is not much work on empirical evaluation of systems or techniques, either. Hoffman et al. provided an attempt at quantitatively evaluating Table Data visualizations using a Display Utilization Grid ([19]). However, the experimentation was not extensive and formal enough to be conclusive regarding the usefulness of such defined metrics for a priori selection of the potentially more effective techniques for a particular situation. Additional work on both qualitative and quantitative evaluation of different interaction techniques is still desired to allow human-in-the-loop selection of desired techniques in different tasks in the wild.

# References


[1] Mărăşoiu, Mariana, et al. "Clarifying hypotheses by sketching data." Proceedings of the Eurographics/IEEE VGTC Conference on Visualization: Short Papers. Eurographics Association, 2016.

[2] Yi, Ji Soo, Youn ah Kang, and John Stasko. "Toward a deeper understanding of the role of interaction in information visualization." IEEE transactions on visualization and computer graphics 13.6 (2007): 1224-1231.

[3] Lam, Heidi. "A framework of interaction costs in information visualization." IEEE transactions on visualization and computer graphics 14.6 (2008): 1149-1156.

[4] Sarkar, Advait, et al. "Visual discovery and model-driven explanation of time series patterns." 2016 IEEE Symposium on Visual Languages and Human-Centric Computing (VL/HCC). IEEE, 2016.

[5] Bertin, Jacques. Graphics and graphic information processing. Walter de Gruyter, 2011.

[6] Elmqvist, Niklas, et al. "ZAME: Interactive large-scale graph visualization." 2008 IEEE Pacific visualization symposium. IEEE, 2008.

[7] Mansmann, Florian, et al. "Correlation-based Arrangement of Time Series for Movement Analysis in Behavioural Ecology." (2012).

[8] Yang, Jing, et al. "Value and relation display: Interactive visual exploration of large data sets with hundreds of dimensions." IEEE Transactions on Visualization & Computer Graphics 3 (2007): 494-507.

[9] Ward, Matthew O., and Zhenyu Guo. "Visual Exploration of Time-Series Data with Shape Space Projections." Computer Graphics Forum. Vol. 30. No. 3. Oxford, UK: Blackwell Publishing Ltd, 2011.

[10] Dang, Tuan Nhon, and Leland Wilkinson. "Scagexplorer: Exploring scatterplots by their scagnostics." 2014 IEEE Pacific Visualization Symposium. IEEE, 2014.

[11] Wilkinson, Leland, Anushka Anand, and Robert Grossman. "Graph-theoretic scagnostics." IEEE Symposium on Information Visualization, 2005. INFOVIS 2005.. IEEE, 2005.

[12] Behrisch, Michael, et al. "Feedback-driven interactive exploration of large multidimensional data supported by visual classifier." 2014 IEEE Conference on Visual Analytics Science and Technology (VAST). IEEE, 2014.

[13] Self, Jessica Zeitz, et al. "Observation-level and parametric interaction for high-dimensional data analysis." ACM Transactions on Interactive Intelligent Systems (TiiS) 8.2 (2018): 15.

[14] Talbot, Justin, et al. "EnsembleMatrix: interactive visualization to support machine learning with multiple classifiers." Proceedings of the SIGCHI Conference on Human Factors in Computing Systems. ACM, 2009.

[15] Ankerst, Mihael, et al. "Visual classification: an interactive approach to decision tree construction." KDD. Vol. 99. 1999.

[16] Becker, Barry, Ron Kohavi, and Dan Sommerfield. "Visualizing the simple Bayesian classifier." Information visualization in data mining and knowledge discovery 18 (2001): 237-249.

[17] Keim, Daniel A., and H-P. Kriegel. "Visualization techniques for mining large databases: A comparison." IEEE Transactions on knowledge and data engineering 8.6 (1996): 923-938.

[18] Keim, Daniel A. "Visual exploration of large data sets." Communications of the ACM 44.8 (2001): 38-44.

[19] Hoffman, Patrick Edward. Table visualizations: a formal model and its applications. University of Massachusetts Lowell, 2000.