
Making machine learning useable

Jianlong Zhou* and Fang Chen

National ICT Australia (NICTA),
Level 5, 13 Garden Street,
Eveleigh, NSW 2015, Australia
Email: Jianlong.Zhou@nicta.com.au
Email: Fang.Chen@nicta.com.au
*Corresponding author

Abstract: Despite the recognised value of machine learning (ML) techniques and high expectation of applying ML techniques within various applications, users often find it difficult to effectively apply ML techniques in practise because of complicated interfaces between ML algorithms and users. This paper focuses on investigating making ML useable from the point of view of how human-computer interaction (HCI) techniques benefit ML in order to simplify the interface between users and ML algorithms. We formulate possible research directions in making ML useable based on human factors, decision making and trust in ML. We strongly believe that a trustworthy decision making based on ML results, which is the ultimate goal of ML-based applications, contributes to the overall application performance and makes ML more useable. Two case studies of measurable decision making and revealing internal states of ML process are presented to show how HCI techniques are used to make ML useable.

Keywords: machine learning; making machine learning useable; HCI; human-computer interaction; decision making; trust; visualisation.

Reference to this paper should be made as follows: Zhou, J. and Chen, F. (2015) 'Making machine learning useable', *Int. J. Intelligent Systems Technologies and Applications*, Vol. 14, No. 2, pp.91–109.

Biographical notes: Jianlong Zhou is a Researcher of Machine Learning Research Group in NICTA. He got a PhD in Computer Science from the University of Sydney, Australia. His research interests include transparent machine learning, human-computer interaction, cognitive computing, volume visualisation, spatial augmented reality and related applications.

Fang Chen is a Senior Principal Researcher of Machine Learning Research Group in NICTA. She holds a PhD in Signal and Information Processing, an MSc and BSc in Telecommunications and Electronic Systems, respectively, and an MBA. Her research interests are behaviour analytics, machine learning, and pattern recognition in human and system performance prediction and evaluation. She has done extensive work on human-machine interaction and cognitive load modelling. She pioneered theoretical framework of measuring cognitive load through multimodal human behaviour, and provided much of empirical evidence on using human behaviour signals, and physiological responses to measure and monitor cognitive load.

1 Introduction

Machine learning (ML) refers to a set of topics dealing with the creation and evaluation of algorithms that facilitate pattern recognition, classification, and prediction, based on models derived from existing data. It utilises computational and statistical methods to automatically or semi-automatically extract information from datasets. As a result, computers have the ability of ‘learning’. Much of ML research is inspired by significant problems from various fields such as biology, finance, medicine, society etc. (Wagstaff, 2012). Various ML algorithms offer a large number of useful ways to approach those problems that otherwise require cumbersome manual solution.

Despite the recognised value of ML techniques and high expectation of applying ML techniques within various applications, users often find it difficult to effectively apply ML techniques in practise because of complicated interfaces between ML algorithms and users, such as complex parameter settings and intermediate decisions in ML algorithms. The complex interfaces often require expert-level ML knowledge in order to fully understand them. Because of these complexities, it is very hard to see ML as a general solution for widespread applications. As a result, ML is regarded as a large bag of tricks grasped by ML experts instead of a universal tool for non-experts. It is necessary to get ML techniques out of the bag of ML experts and into various application domains from which ML could benefit.

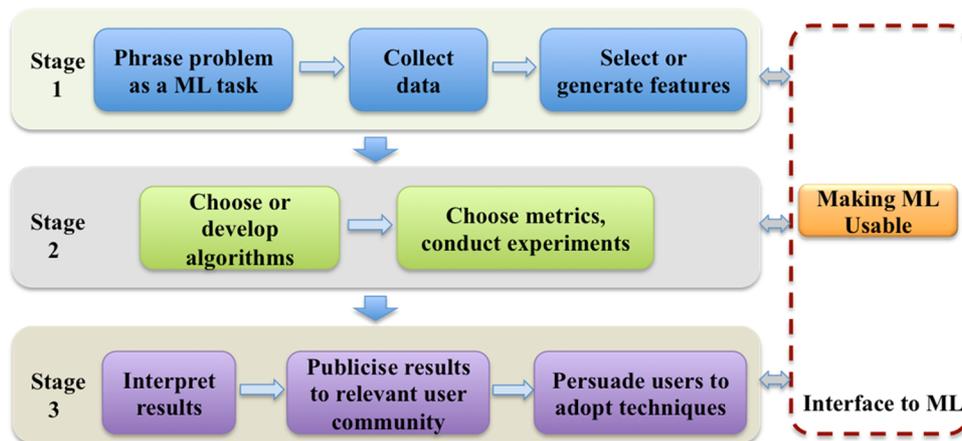
Therefore, besides the development of ML algorithms, research of making ML useable is emerging as one of active research fields recently. Making ML useable aims to make ML easily understandable and accessible by domain professionals without requiring training in complex ML algorithms and mathematical concepts. Wagstaff (2012) motivated this problem by presenting an ML research program with a three-stage model as shown in Figure 1. In this model, the stage 1 is the preparation stage for a ML research program, the stage 2 is the development of solutions for a problem, and the stage 3 aims to improve the impact of ML algorithms on real-world applications. The current research is highly biased towards the stage 2 of the invention of new learning algorithms and does introduce human subjects into ML (Jun et al., 2013; Salazar et al., 2013), but few work contributes to how ML can be used by non-experts more easily and meaningfully to improve the impact of ML approaches. It is the stage 3 that translates ML results into impact on real world. As shown in Figure 1, making ML usable should be added in the three-stage model of an ML research program. The addition of making ML useable allows effective interactions with ML algorithms, thus provides opportunities to improve impact of ML algorithms on real world applications.

Furthermore, it has been widely accepted that the development of an ML system should encompass attention to human-computer interaction (HCI). HCI involves the study, planning, and design of the interaction between human and computers. It represents a necessary condition to effective ML system development and implementation. Such topics have been frequently discussed in the past decades. For example, as early as in ICML’96, a special workshop ‘ML Meets HCI Workshop’ was organised to discuss what HCI can offer to ML, or conversely what ML can offer to HCI. Despite the identification of obstacles to ML impact by Wagstaff (2012), it does not identify HCI as one of solutions to widespread ML impact on the real world and make

ML useable. To this end, most of previous work focuses on how visualisation is used to depict ML process of a specific ML algorithm or represent ML results. More work is highly necessary on how HCI techniques are used to simplify the interface between users and ML technologies to make ML useable.

This paper aims to review factors which affect making ML useable. The limitations of current ML research and challenges of applied ML are firstly discussed. Different visualisation techniques are then investigated to show how visualisation techniques benefit ML to simplify the interface between users and ML algorithms, which includes visualisation in the representation of data space, depiction of ML results, and presentation of ML performance. We also formulate possible research directions in making ML useable based on human factors, decision making as well as trust in ML. We strongly believe that a trustworthy decision making based on ML results, which is the ultimate goal of ML-based applications, contributes to the overall application performance and makes ML more useable. We present two case studies of measure decision making and revealing internal states of ML process to show how HCI techniques are used to make ML useable.

Figure 1 Three stages of an ML research program (see online version for colours)



Source: Adapted from Wagstaff (2012)

2 General issues of applied machine learning

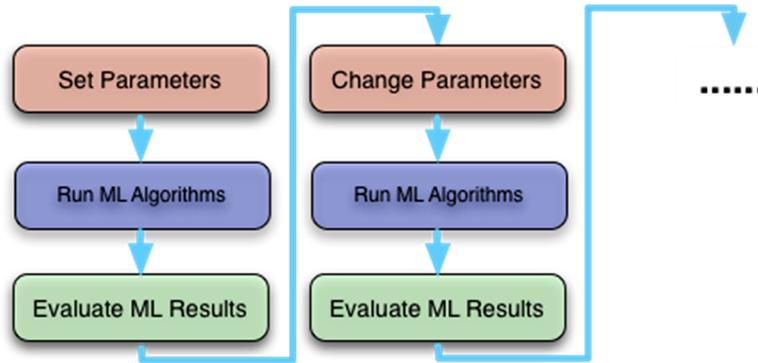
According to Wagstaff (2012), limitations of current ML research work lie in algorithmic advances accompanied by empirical studies that are divorced from true impact on wide application areas. Such limitations can also be observed from conferences and journals available for ML researches. The main ML conferences such as ICML and NIPS focus more on theoretic contributions to ML but not favour studies in ML-based applications so much. The ML journals, such as JMLR and Machine Learning, also have similar favourites as ML conferences.

From the user’s point of view, a typical applied ML workflow can be summarised as follows (Patel et al., 2008; Talbot et al., 2009):

- develop ML models by defining low-level choices such as algorithm selection, parameter sets
- iteratively refine the previous step using manual trial and error through the model space
- compare models using accuracy, performance, or other coarse measurements.

In this workflow as represented in Figure 2, users first set parameters for ML algorithms. After finishing the running of ML algorithms, users evaluate ML results. If ML results do not meet their requirements, users need to change parameters and run the whole process again. This process iteratively continues until users get acceptable ML results.

Figure 2 Conventional ML-based data analysis workflow (see online version for colours)



Patel et al. evaluated the current use of ML by non-experts and identified three difficulties they encounter when using ML techniques (Patel et al., 2008), e.g., difficulty in following an iterative and exploratory process, as well as evaluating performance. Another significant difficulty lies in the definition of complex parameter space. All these difficulties affect the users' trust on ML models.

When interpreting ML results and making decisions, users have to face following challenges:

- difficult to compare various ML results from various parameter settings
- difficult to understand abstract outputs
- difficult to make decisions solely based on numbers.

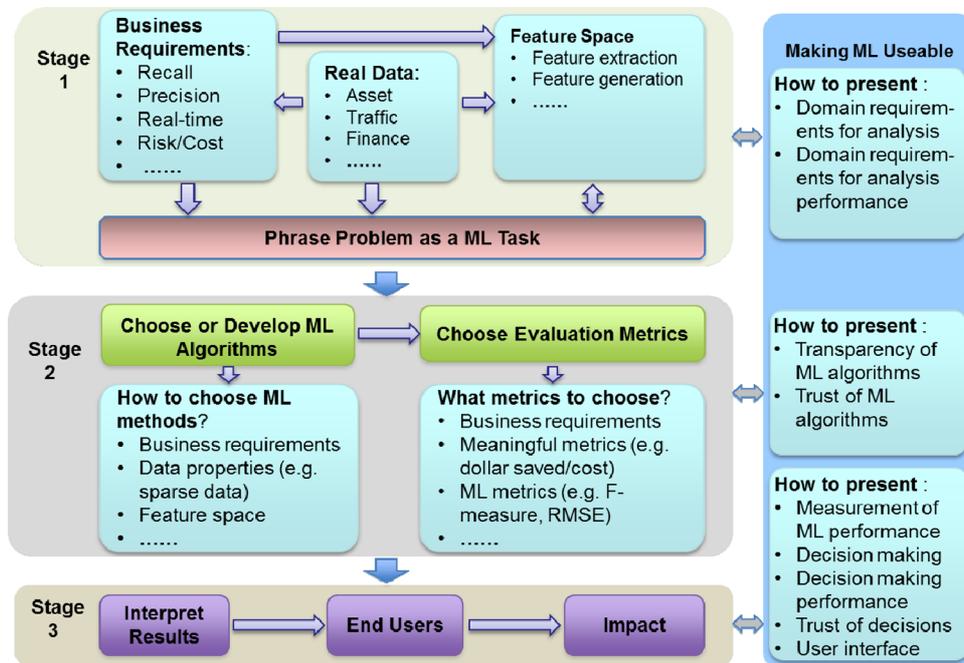
These challenges definitely affect the users' trust on decisions based on ML results.

3 Making machine learning usable

We strongly argue that making ML useable should consider various aspects in all three stages based on Wagstaff's model (see Figure 3). At the Stage 1, users need to understand the domain requirements, such as what the goals of data analyses are. They also need to learn domain requirements for analysis performance, such as recall, risk/cost. At the Stage 2, making ML useable should consider to let users perceive transparency of ML

models, such as what is going on inside the ML models and how to accomplish the learning problem. More importantly, ML models should be trustable by users. At the Stage 3, measurement of ML performance has to be presented meaningfully and users need to understand how to make decisions effectively based on ML results. Furthermore, the decision performance has to be presented to users. Trust on decisions has to be perceived by users to make trustworthy decisions. Besides, the user interface also plays significant roles in making ML useable. This section mainly discusses domain knowledge, the roles of decision making and trust in making ML useable. The investigations in this section formulate possible research directions in making ML useable.

Figure 3 Three stages of an ML research program in Making ML useable (see online version for colours)



3.1 Domain requirements and ML models

As mentioned, much of ML research is inspired from various domain problems. In an ML-based data analysis pipeline, domain requirements drive ML researchers to use different ML models for solving problems (see Figure 3). Therefore, to make ML useable, the domain requirements need to be presented in an appropriate manner to link with ML models. Furthermore, it is also need to set up relations between analysis performance in domain specifications and ML performance. For example, in the water pipe failure history data analysis, users' principal requirement is to create risk maps of pipes to maintain pipes more effectively. The performance of this analysis lies

in the amount of maintaining fund saved after following new risk maps. However, users are not interested in or even do not understand ML terms such as statistical probabilities and ROC area or others. Therefore, making ML useable should set up and present connections between domain requirements and ML models to make users more trust ML models and ML results.

3.2 *Decision making in ML*

In the most cases, to drive or improve decision making is the ultimate goal of an ML-based data analysis, which belongs to the third stage in Wagstaff's model. When an ML approach is used to infer a model from input data, the quality of the model should be judged not from the point of view of how it fits the 'true model', but from the point of view of how good are the decisions one makes based on this model (Bousquet, 2005). The goodness of a decision should be measured by how much return the user gets, compared to the best possible return. Therefore, there is a close connection between ML and decision theory. Despite the wide use of ML results in decision making in previous work (Krishnamurthy et al., 2009), they focus on the direct use of ML results in decision making, such as choosing decision alternatives which have the highest values of ML results as final decisions. Furthermore, previous work does not pay much attention to the evaluation of how ML affects decision making.

From the user's point of view, ML techniques are supposed to support him or her effectively, efficiently, and satisfactorily while doing his or her job. They must be considered as motivational instruments. Therefore, ML-based applications need to allow users perceive the factors of ML results affecting decision making and also allow users improve decision making performance by refining ML results used to make decisions. In such case, users can make decisions adaptively based on ML results in order to make ML useable.

Therefore, the related research questions on decision making based on ML results include:

- what factors of ML results affect decision making
- how to measure decision performance with different ML results
- how decision making contributes to making ML useable.

3.3 *Trust in ML*

Recently, trust has been realised as one of the most important factors in management and organisational behaviour for all personal and business decision making as well as for efficiency and task performance (Donaldson, 2001). Trust has been defined by Mayer et al. (1995) and many others as an individual's (or a party's) willingness to accept the vulnerabilities of actions or behaviour of the other party based on the expectation that the other will perform the actions important to the individual (or the trustor).

Since the ultimate goal of ML-based applications is to make decisions based on ML results in the most cases, a natural consideration before and after making decisions is to evaluate whether the decisions are trusted. Therefore, to make ML useable, users should consider the trust both on ML algorithms themselves and on decisions made based on ML results. The KDnuggets website had a polling on whether ML can be

trusted (<http://www.kdnuggets.com/polls/2008/can-machine-learning-be-trusted.htm>). It was shown that there were 65% of participants who trusted ML algorithms. But there were still more than one third of participants who did not trust ML algorithms at all or did not know. The investigation shows that the trustworthiness of ML algorithms is related to many factors such as:

- *The cost of errors*: This depends on what level of cost users can accept if ML algorithms fail to work.
- *Good models of ML algorithm competence*: Models are necessary to compare various ML algorithms.
- *Inspectability of ML algorithms*: This requires ML algorithms to be more inspectable and transparent.
- *Quality of data used to build ML models*: An ML algorithm is not possible to be better than its data. The data which represent all possible outcomes of an event will give higher trust on ML algorithms.
- *Trusted software processes for certifying ML algorithms*: ML algorithms will be trusted more if they were tested by a trusted process.

Therefore, a trustworthy decision making based on ML results, which is the ultimate goal of ML-based applications, contributes to the overall application performance and makes ML more useable.

4 Visualisation for machine learning

The information visualisation and visual data mining leverage the very high processing capabilities of the human visual system to allow interactive exploration and analysis of massive datasets. ML could benefit from information visualisation and visual data mining in trade-off between completeness of the representation and the process rate of datasets (Rossi, 2006). For example, when users explore ML results, the results are often in a descriptive or an abstracted form, such as summaries of the sub-regions. The examination of instances in the region is also very important for understanding the data point distribution. Thus, without a visualisation of ML results, users cannot build connections between patterns and instances. To assist ML development, much research has been done in visualising ML algorithms, such as Naïve-Bayes (Becker et al., 2002), decision trees (Ankerst et al., 1999), SVMs (Caragea et al., 2001), and HMMs (Dai and Cheng, 2008). This section firstly investigates requirements for visualisation in ML. Visualisation used in ML is then reviewed. Interactive visualisation interfaces are also discussed in this section.

4.1 Requirements for visualisation in ML

The visualisation can help users easily understand the structure of a classifier. It can transform the classifier from incomprehensible ‘black-box’ to a useful tool that converts the data into knowledge. Becker et al. (2002) proposed an approach to visually represent

the simple Bayesian model. The presented approach allows users to interact with the simple Bayesian model, examine specific values, and show probabilities of picked objects visually. The visualisation techniques used primarily include pie chart and bar chart representations. Besides, Becker et al. (2002) also identifies design requirements for the visualisation of the simple Bayesian classifier, such as showing the complete model and easily comparing the relative evidence in visualisations.

4.2 *Visualisation techniques in ML*

Data charts such as bar charts, line charts, pie charts, and scatter plots are arguably among the most common data representations used in ML. They have been used in a variety of disciplines to communicate information to users. For example, data charts are often used to present ML results and other visualisation techniques are frequently used in ML to plot data instances in the projection of feature space and visualise model prediction boundaries (Frank and Hall, 2003).

Different visualisation techniques are used to present an ML process. Erra et al. (2011) introduced a visual clustering approach which utilises collective behavioural model. Huang et al. (2011) used a visual clustering method for assisting the analysis of email attacks. Paiva et al. (2011) presented an approach that employs the similarity tree visualisation to distinguish groups of interest within the dataset. Jakulin et al. (2005) employed nomograms for visualising trained SVMs. Using nomograms, SVMs can be effectively visualised in attribute spaces with many dimensions.

Visualisation also provides interaction interfaces for users in data analysis. Guo et al. (2011) introduced a visual interface named *Nugget Browser* allowing users to interactively submit subgroup mining queries for discovering interesting patterns dynamically. Talbot et al. (2009) allowed users to visually ensemble multiple classifiers together and provides a summary visualisation of results of these multiple classifiers. Fails and Olsen (2003) described the importance of human involvement in an interactive ML.

These visualisation techniques can help users understand and/or interact with ML models effectively in some degree. However, the previous work does not visually set up the connection between parameter settings and ML results explicitly. This connection can allow users make comparisons between different ML results and corresponding parameter settings. The comparison provides guidance information for refining ML processes.

5 **Representation of machine learning performance**

Previous investigations show that various visualisation approaches can be used not only to depict ML processes, but also to represent ML results visually. This section further investigates that visualisation could also be used to represent performance of ML algorithms. The representation can be combined together with ML result representations to help users make informed decisions. The section first shows the problem of representation of performance measures in ML. Visualisation techniques for uncertainty representations are then reviewed to develop visualisation methods for representations of performance measures and ML results in ML applications.

5.1 Representation of performance measures

As investigated previously, performance of ML algorithms is usually measured with abstract evaluation metrics such as classification accuracy, RMSE. These measures do not reveal problem-specific details. Furthermore, to compare errors of different ML algorithms, Bravo et al. (2008) used Kurtosis in ML applications. Kurtosis is a non-parametric measure of how close (in average) the probabilities are from the true results. Evaluation of ML algorithms is essentially domain dependent and any evaluation measures must thus necessarily be also domain dependent (Guyon et al., 2009). In addition to the mentioned traditional measures of performance, Wagstaff (2012) motivated more meaningful evaluation methods. For example, dollars saved, lives preserved, time conserved, effort reduced, quality of living increased, etc. are easy understandable and meaningful metrics in various domains or applications. However, a link needs to be set up between traditional measures of performance and meaningful evaluation methods.

5.2 Uncertainty and visualisation

In practical applications, data such as ML results are rarely absolutely certain. The performance of ML algorithms is usually represented with numbers. In the literature (Pang, 2001), uncertainty is defined as a multi-faceted characterisation about data, whether from measurements and observations of some phenomenon, and predictions made from them. It may include several concepts including error, accuracy, precision, validity, quality, variability, noise, completeness, confidence, and reliability (Pang, 2001).

By developing ways to make the uncertainty associated with data more visible, users can better understand their data and make decisions more easily and confidently. The goal of uncertainty visualisation is to translate numerical data of probabilities and other information into images faithfully, so that users can ‘see’ the uncertainty. The most common visualisation method used for representing uncertainty is error bars (Skeels et al., 2010). In scientific visualisation, the uncertainty information is often incorporated into the data visualisation together (Rhodes et al., 2004). Uncertainty is represented using hue and/or texture together with the data visualisation (Rhodes et al., 2004). Uncertainty may also be associated with the multi-resolution approximations of the original data in visualisation to faithfully represent the datasets (Ma et al., 2012). In weather information visualisation, uncertainty is represented with noodle like glyphs (Sanyal et al., 2010). Pang (2001) discussed two general ways of combining uncertainty into a geospatial data visualisation:

- mapping uncertainty information as an additional piece of data
- creating new visualisation primitives and abstractions that incorporate uncertainty information.

Zuk and Carpendale (2006) did a heuristic evaluation of eight visualisations of uncertainty using principles or heuristics identified for information visualisation in general.

These representations of uncertainty motivate new approaches to visually depict performance measurements of ML algorithms. We can imagine that ML results and

performance of ML algorithms can be represented together visually in order to make decisions faithfully.

6 Toolkits and libraries for making ml useable

Currently, many ML algorithm implementations are available as standalone toolkits or libraries. Active researches are also done to provide ML as online services. This section shows typical ML toolkits and libraries that are popular in current research. Some of online ML services are also reviewed for the completeness overview of ML toolkits and libraries.

6.1 *ML toolkits*

Weka and Orange are two widely used ML toolkits. Weka (Witten et al., 2011) is a comprehensive ML toolkits which aims to build a workbench for ML. It includes a suite of Java class libraries that implement many ML and data mining algorithms. The toolkit provides a common interface which allows users compare different methods and identify those that are most appropriate for the problem at hand (Witten et al., 2011). Different from Weka, Orange (<http://orange.biolab.si/>) is a component-based data mining and ML Open Source software suite implemented in C++ and Python. Its graphical user interface builds upon the cross-platform Qt framework. Orange features a visual programming front-end for explorative data analysis and visualisation, and Python bindings and libraries for scripting. It includes a set of components for data preprocessing, feature scoring and filtering, modelling, model evaluation, and exploration techniques.

Other ML based data analysis tools are also widely used in various applications, such as Pentaho (<http://www.pentaho.com/>), Rapid-I RapidMiner (<http://rapid-i.com/content/view/181/190/lang,en/>).

6.2 *ML libraries*

There are many free Open Source ML libraries available. Some libraries are specific for a single ML algorithm, such as LIBSVM, while other libraries are collections of various algorithm implementations in different programming languages, such as MLC++, PyBrain, Scikit-learn. This section only shows some typical widely used libraries as shown in Table 1.

LIBSVM (Chang and Lin, 2011) is an Open Source library for support vector classification, regression and distribution estimation. It supports multi-class classification. It is widely used in research and also included in some data mining environments such as RapidMiner, PCP, and LIONSolver. MLC++ (<http://www.sgi.com/tech/mlc/>) is a free library of C++ classes for supervised machine learning. It provides general ML algorithms that can be used by end users, analysts, professionals, and researchers. More than just a collection of existing algorithms, MLC++ is an attempt to extract commonalities of machine learning algorithms and decompose them for a unified view that is simple, coherent, and extensible.

PyBrain (Schaul et al., 2010) is an Open Source ML library written in Python. The library includes a variety of predefined environments and benchmarks to test and compare algorithms. Implemented algorithms in this library include long short-term memory (LSTM), policy gradient methods, (multidimensional) recurrent neural networks and deep belief networks. Apache Mahout (<http://mahout.apache.org/>) is a scalable ML library which has implementations of typical ML and data mining algorithms. It mainly supports four use cases such as recommendation mining, clustering, classification, and frequent item set mining. Scikit-learn (<http://scikit-learn.org/stable/>) is an open source ML library in Python. It features various classification, regression and clustering algorithms including support vector machines, logistic regression, Naive Bayes, K-Means and DBSCAN. This library also provides tools for dataset transformation, and dataset loading utilities.

Table 1 Typical open source ML libraries

<i>Library</i>	<i>Language</i>	<i>ML Algorithms</i>	<i>Features</i>
LIBSVM	C++, also available for other language wrappers	SVM	The most widely used SVM library
MLC++	C++	Different supervised ML algorithms	A library of C++ classes for supervised ML
PyBrain	Python	Various ML algorithms especially neural networks and reinforcement learning	Modular, flexible, and easy to use
Scikit-learn	Python	Various classification, regression and clustering algorithms	Contained in Python as a package

6.3 ML online services

Recently, the concept of ML as a service is being researched deeply. This section introduces typical ML online services from the functionality perspective.

- *Upload data and create model based on data by the service.* BigML (<https://www.bigml.com/>) is an online tool which allows users to upload their own data and set up a data source and to format data source for BigML. It then creates a model based on the formatted data source. The model is then used to generate predictions for new data.
- *Upload data and find the best ML algorithm or upload ML algorithm to evaluate.* MLcomp (<http://mlcomp.org>) is focused on users either to upload datasets to find the algorithm that performs the best, or to upload algorithms and have the system automatically evaluate it against all previously uploaded datasets. It is focused on machine learning practitioners.
- *Resource-oriented RESTful web service.* Protocols and structures for inference (PSI) (<http://psi.cecs.anu.edu.au/>) develops protocols and structures of the concept of ML as a Service. It aims to develop an architecture for presenting ML algorithms, their inputs and outputs as resource-oriented RESTful web services to make ML technology accessible to a broader range of people than just ML researchers.

- *API directly used in an application.* Google Prediction API (<https://developers.google.com/prediction/>) is a business model that focuses on providing a scalable ML black box that can be used directly or integrated into an application. The service works by the user first uploading a dataset to the Google storage service, training an opaque model from the data, and deriving predictions from the prepared model.

Other similar online ML service providers include Skytree (<http://www.skytreecorp.com/>), OpenTox (<http://opentox.org/>), Precog (<http://precog.com/>), etc. However, these work still lack the definition of efficient HCI interfaces for non-experts in using ML algorithms. The efficient definition of parameter space is still a problem for these ML service providers.

Besides the concept of ML as a service, Lawrence (2012) proposed a concept of ‘machine learning as engine design’. He argues that ML is very much like the early days of engine design. From steam engines to jet engines the aim of engines is to convert heat into kinetic energy. The aim of ML algorithms is to convert data into actionable knowledge.

In summary, standalone ML toolkits such as Orange usually provides a visual interface for setting up an ML based analysis pipeline and visually presenting ML outputs. ML libraries provide convenient implementations that can be used in users’ applications. A typical online ML service usually asks users to specify parameters online after uploading data or algorithms. After waiting some running time, the tool outputs ML reports on screen. However, one of obvious shortcomings of these tools is that they do not have powerful mechanisms on easing the definition and refinement of ML parameters as well as easing decision making based on ML results.

7 Active researchers/institutions in making ML usable

Many researchers and institutions work on making ML useable. Kiri Wagstaff is a researcher at California Institute of Technology, CA, USA, investigating ways that ML can be used to increase the autonomy of space missions. Her position paper (Wagstaff, 2012) presented on ICML2012 outlines some ways in which much of current ML research has become disconnected from real problems of significance to the world outside of ML. It instigates creative discussions about how to improve impact of ML on wider applications. Other examples of researchers or institutions on making ML useable include:

- *Interactive visualisation for ML.* Justin Talbot (from Stanford University, USA), and Bongshin Lee, Ashish Kapoor, Desney S. Tan (from Microsoft Research at Redmond, WA, USA) use interactive visualisation to support ML with multiple classifiers (Talbot, 2009).
- *Human-guided ML.* Despite Saleema Amershi, Bongshin Lee, Ashish Kapoor, and Desney S. Tan at Microsoft Research focusing on visualisation and HCI research, they also do much work on human-guided ML in typical applications (Amershi et al., 2011; Kapoor et al., 2012).
- *Development environment supporting applying ML.* Kayur Patel, James Fogarty, and James Landay etc. at the DuB Group on research of Human-Computer Interaction and Design in University of Washington, USA do active research to support the

process of applying ML (Patel et al., 2008, 2010). They examined difficulties users meet when applying ML technologies in software development and other applications.

- *End-user interfaces for interactive ML*. Rebecca Fiebrink et al. in Princeton University, USA focus on the design of end-user interfaces for interactive ML in real-time application domains such as music composition and performance (Fiebrink and Trueman, 2012; Fiebrink, 2011). Weka (Witten et al., 2011) developer of Machine Learning Group at University of Waikato, New Zealand does active research in making ML techniques generally available.
- *ML as a service*. In Australia, one of the most relevant researches is the project of Machine Learning as a Service done at Research School of Computer Science, Australian National University and National ICT Australia (NICTA). Williamson et al. (PSI: <http://psi.cecs.anu.edu.au/>) do active researches in ML as a service. As mentioned, they aim to develop an architecture for presenting ML algorithms, their inputs and outputs as resource-oriented RESTful web services to make ML technology accessible to a broader range of people.

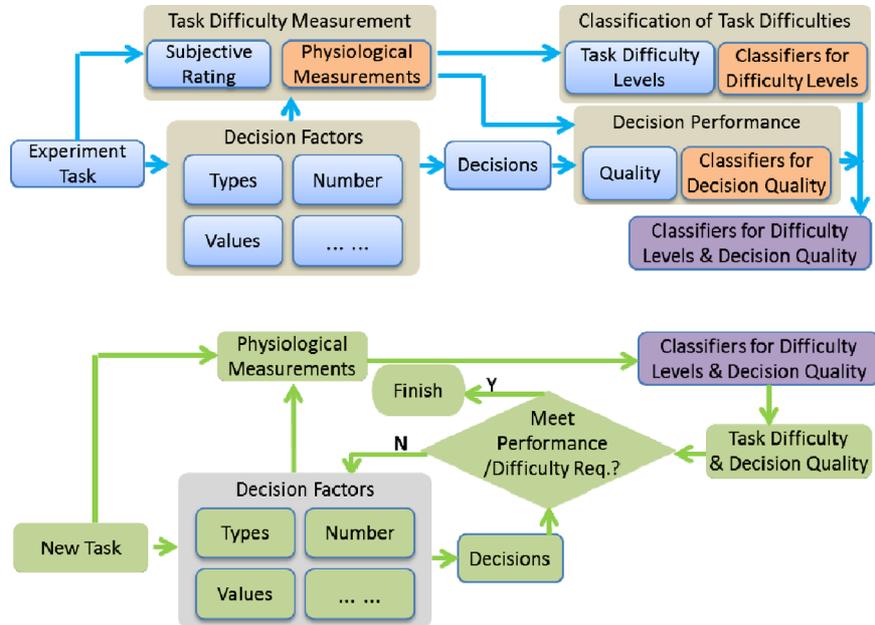
8 Case studies

In this section, we focus on making ML useable from two aspects which fit in the stage three in 3-stage model as shown in Figure 3: measurable decision making and revealing internal status of ML process to make ML transparent (we call transparent ML or TML). Measurable decision making aims to set up connections between physiological signals of humans and ML-based decision making. Such connection helps users choose and refine ML-based decision factors adaptively during decision making and make high quality decisions confidently. Therefore, ML models can be evaluated based on decision qualities, which is more meaningful both for ML researchers and domain users. Transparent ML aims to make a ‘black-box’ ML process transparent by presenting real-time internal status update of the ML process to users to make ML analysis process more easily understandable, the real-time status update more meaningful, and users more convincing on ML results.

8.1 Making ML useable with measurable decision making

As mentioned, the ultimate goal of ML-based applications is to make decisions based on ML results. The quality of an ML model should be judged from the point of view of how good are the decisions one makes based on this model. Based on these observations, we present a framework of ML-based adaptive measurable decision making for making ML useable (see Figure 4) (Zhou et al., 2015a). In this framework, when an experiment task with ML-based decision factors is exposed to users for decision making, task difficulty is measured at the same time with subjective ratings and physiological sensors such as eye-tracker and Galvanic Skin Response (GSR). After the user makes decisions, the decision performance is also evaluated with the user’s choice and physiological measurements. The measured information is then analysed and classifiers for decision quality and task difficulty are derived.

Figure 4 Framework of making ML useable with measurable decision making (see online version for colours)



When a new task is coming, users' workload during decision making is recorded using physiological sensors in real-time. This measurement is sent to classifiers for difficulty levels and decision quality learned in the experiment task stage. The task difficulty level and decision quality from classifiers are exported to users. If users are satisfied with the decision performance and decision itself, then the decision making process is finished. Otherwise, ML-based decision factors are refined (e.g., increase/decrease number of ML-based decision factors) based on the analysis results at the experiment task stage to continue a new decision making session. For example, if the decision difficulty level derived from ML models is low, more ML results may be included in order to get higher quality decisions. This process is iteratively performed until decisions meet performance and difficulty requirements from users.

In summary, it was demonstrated that the effects of ML results on decision making can be measured in real-time to evaluate whether a decision task was in an appropriate difficulty level or whether a decision made by users was in high quality. The connection between physiological signals and ML-based decision making was set up. This connection helps users choose and refine ML-based decision factors adaptively during decision making to make ML useable as shown in Figure 4. For example, users may include more ML-based decision factors in order to make decisions with higher qualities.

8.2 Making ML useable with transparent ML

For a domain professional who may not have expertise in ML or programming, an ML algorithm acts as a 'black-box', where the user defines parameters and input data for the 'black-box' and gets output from its execution. This 'black-box' approach has obvious drawbacks: it is difficult for the user to understand the complicated ML models,

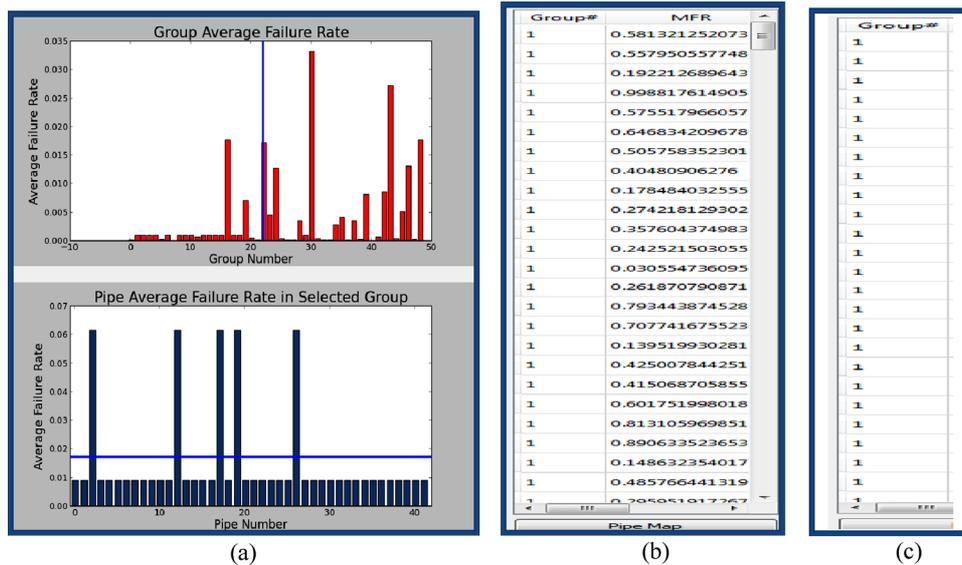
such as what is going on inside the ML models and how to accomplish the learning problem. As a result, the user is uncertain about the usefulness of ML results and this affects the effectiveness of ML methods. Furthermore, in most cases, even if automatic ML algorithms give satisfactory results, visual feedback of ML processes is very important to offer insight into the reasons for learning failure or success. Therefore, interactive ML interfaces must not only supply users with the information on input data and output results, but also enable them to perceive internal real-time status update of ML processes with visual feedback.

We make a ‘black-box’ ML process transparent by presenting real-time internal status update of the ML process to users explicitly with a concept of transparent ML (TML) (Zhou et al., 2013, 2015b). Meaningful internal states of ML algorithms are selected and revealed during the real-time status update. Various visualisation techniques (see Figure 5) are used to allow users interactively view how the final results are obtained in ML. As a result, the ‘black-box’ ML becomes ‘transparent’ to users providing better understanding of the overall process.

The study showed that advanced ML approaches along with graphical presentations can help improve the easiness of understanding the ML data analysis process, meaningfulness of real-time status update, and convincingness of ML results. Even if classical ML approaches were used, users still gave higher ratings when graphical presentations were used than advanced ML approaches without graphical real-time status update. From the study, we concluded that

- it is obvious that the ML approach affects users’ ratings on easiness, meaningfulness, and convincingness,
- the visualisation method also greatly affects users’ overall ratings on ML’s easiness, meaningfulness, and convincingness of an ML approach, regardless of what ML method is being used.

Figure 5 Different presentation methods for real-time status update (see online version for colours)



9 Conclusions

The paper reviewed various factors in making ML useable. It first reviewed general issues of applied ML, such as limitations of current ML research and challenges of applied ML, which affects making ML useable. Specifically, we investigated various visualisation techniques that are used to help interact with ML algorithms and representation of ML performance to make ML useable. We formulated possible research directions in making ML useable based on human factors, decision making as well as trust. Two case studies were presented to show how HCI techniques are used to make ML useable.

Acknowledgements

NICTA is funded by the Australian Government as represented by the Department of Broadband, Communications and the Digital Economy and the Australian Research Council through the ICT Centre of Excellence program.

References

- Amershi, S., Lee, B., Kapoor, A., Mahajan, R. and Christian, B. (2011) 'Human-guided machine learning for fast and accurate network alarm triage', *Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence – Volume 3*, Barcelona, Catalonia, Spain, pp.2564–2569.
- Ankerst, M., Elsen, C., Ester, M. and Kriegel, H-P. (1999) 'Visual classification: an interactive approach to decision tree construction', *Proceedings of the fifth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, New York, NY, USA, pp.392–396.
- Becker, B., Kohavi, R. and Sommerfield, D. (2002) 'Visualizing the simple Bayesian classifier', in Fayyad, U., Grinstein, G.G. and Wierse, A. (Eds.): *Information Visualization in Data Mining and Knowledge Discovery*, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, pp.237–249.
- Bousquet, O. (2005) *Machine Learning Thoughts – Decision Making*, 29 June, 2005 [Online]. Available: http://ml.typepad.com/machine_learning_thoughts/2005/06/decisionmaking.html (Accessed 15 September, 2013).
- Bravo, C., Lobato, J.L., Weber, R. and L'Huillier, G. (2008) 'A hybrid system for probability estimation in multiclass problems combining SVMs and neural networks', *Proceedings of the 2008 8th International Conference on Hybrid Intelligent Systems*, Washington DC, USA, pp.649–654.
- Caragea, D., Cook, D. and Honavar, V.G. (2001) 'Gaining insights into support vector machine pattern classifiers using projection-based tour methods', *Proceedings of the Seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, San Francisco, CA, USA, pp.251–256.
- Chang, C-C. and Lin, C-J. (2011) 'LIBSVM: a library for support vector machines', *ACM Trans. Intell. Syst. Technol.*, Vol. 2, No. 3, Article 27, 27 pages.
- Dai, J. and Cheng, J. (2008) 'HMMEditor: a visual editing tool for profile hidden Markov model', *BMC Genomics*, Vol. 9, Suppl. 1, March, p.S8.
- Donaldson, T. (2001) 'The ethical wealth of nations', *Journal of Business Ethics*, Vol. 31, pp.25–36.

- Erra, U., Frola, B. and Scarano, V. (2011) 'An interactive bio-inspired approach to clustering and visualizing datasets', *Proceedings of the 15th International Conference on Information Visualisation 2011*, London, UK, pp.440–447.
- Fails, J.A. and Olsen Jr., D.R. (2003) 'Interactive machine learning', *Proceedings of IUI2003*, Miami, FL, USA, pp.39–45.
- Fiebrink, R. (2011) *Real-time Human Interaction with Supervised Learning Algorithms for Music Composition and Performance*, PhD Thesis, Princeton University, Princeton, NJ, USA.
- Fiebrink, R. and Trueman, D. (2012) 'End-user machine learning in music composition and performance', *CHI 2012 Workshop on End-User Interactions with Intelligent and Autonomous Systems*, Austin, Texas, USA, pp.1–4.
- Frank, E. and Hall, M. (2003) 'Visualizing class probability estimators', in Lavrač, N., Gamberger, D., Todorovski, L. and Blockeel, H. (Eds.): *Knowledge Discovery in Databases: PKDD 2003*, Springer, Cavtat-Dubrovnik, Croatia, pp.168–179.
- Guo, Z., Ward, M.O. and Rundensteiner, E.A. (2011) 'Nugget browser: visual subgroup mining and statistical significance discovery in multivariate datasets', *Proceedings of the 2011 15th International Conference on Information Visualisation*, Washington DC, USA, pp.267–275.
- Guyon, I., Luxburg, U.V. and Williamson, R.C. (2009) 'Clustering: science or art', *NIPS 2009 Workshop on Clustering Theory*, Vancouver, BC, Canada, pp.1–11.
- Huang, M.L., Zhang, J., Nguyen, Q.V. and Wang, J. (2011) 'Visual clustering of spam emails for DDoS analysis', *Proceedings of the 2011 15th International Conference on Information Visualisation*, Washington DC, USA, pp.65–72.
- Jakulin, A., Možina, M., Demšar, J., Bratko, I. and Zupan, B. (2005) 'Nomograms for visualizing support vector machines', *Proceedings of the Eleventh ACM SIGKDD International Conference on Knowledge Discovery in Data Mining*, New York, NY, USA, pp.108–117.
- Jun, K-S., Zhu, J., Settles, B. and Rogers, T. (2013) 'Learning from human-generated lists', *Proceedings of ICML2013*, Atlanta, USA, pp.181–189.
- Kapoor, A., Lee, B., Tan, D.S. and Horvitz, E. (2012) 'Learning to learn: algorithmic inspirations from human problem solving', *Proceedings of the Twenty-Sixth AAAI Conference on Artificial Intelligence*, Toronto, Canada, pp.1571–1577.
- Krishnamurthy, S., Thamilarasu, G. and Bauckhage, C. (2009) 'MALADY: a machine learning-based autonomous decision-making system for sensor networks', *Proceedings of International Conference on Computational Science and Engineering 2009*, Vol. 2, pp.93–100.
- Lawrence, N. (2012) *Machine Learning as Engine Design* [Online], Available: <http://uspace.shef.ac.uk/blogs/profneil/2012/07/05/machine-learning-as-engine-design> (Accessed 7 November, 2012).
- Ma, J., Murphy, D., O'Mathuna, C., Hayes, M. and Provan, G. (2012) 'Visualizing uncertainty in multi-resolution volumetric data using marching cubes', *Proceedings of the International Working Conference on Advanced Visual Interfaces*, New York, NY, USA, pp.489–496.
- Mayer, R.C., Davis, J.H. and Schoorman, F.D. (1995) 'An integrative model of organizational trust', *Academy of Management Review*, Vol. 20, pp.709–734.
- Paiva, J.G., Florian, L., Pedrini, H., Telles, G. and Minghim, R. (2011) 'Improved similarity trees and their application to visual data classification', *IEEE Transactions on Visualization and Computer Graphics*, Vol. 17, No. 12, December, pp.2459–2468.
- Pang, A. (2001) 'Visualizing uncertainty in geo-spatial data', *Proceedings of the Workshop on the Intersections between Geospatial Information and Information Technology*, Arlington, TX, USA, pp.1–14.
- Patel, K., Bancroft, N., Drucker, S.M., Fogarty, J., Ko, A.J. and Landay, J. (2010) 'Gestalt: integrated support for implementation and analysis in machine learning', *Proceedings of the 23rd Annual ACM Symposium on User Interface Software and Technology*, New York, NY, USA, pp.37–46.

- Patel, K., Fogarty, J., Landay, J.A. and Harrison, B. (2008) 'Examining difficulties software developers encounter in the adoption of statistical machine learning', *Proceedings of the 23rd National Conference on Artificial Intelligence – Volume 3*, Chicago, pp.1563–1566.
- Patel, K., Fogarty, J., Landay, J.A. and Harrison, B. (2008) 'Investigating statistical machine learning as a tool for software development', *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, Florence, Italy, pp.667–676.
- Rhodes, P., Laramée, R., Bergeron, R. and Sparr, T. (2004) 'Uncertainty visualization methods in isosurface rendering', *Eurographics 2003*, Granada, Spain, pp.83–88.
- Rossi, F. (2006) 'Visual data mining and machine learning', *Proceedings of European Symposium on Artificial Neural Networks (ESANN 2006)*, Bruges, Belgium, pp.251–264.
- Salazar, E., Bogdan, R., Gorka, A., Hariri, A. and Carin, L. (2013) 'Exploring the mind: integrating questionnaires and fMRI', *Proceedings of ICML2013*, Atlanta, USA, pp.262–270.
- Sanyal, J., Zhang, S., Dyer, J., Mercer, A., Amburn, P. and Moorhead, R. (2010) 'Noodles: a tool for visualization of numerical weather model ensemble uncertainty', *IEEE Transactions on Visualization and Computer Graphics*, Vol. 16, No. 6, November, pp.1421–1430.
- Schaul, T., Bayer, J., Wierstra, D., Sun, Y., Felder, M., Sehnke, F., Rückstieß, T. and Schmidhuber, J. (2010) 'PyBrain', *J. Mach. Learn. Res.*, Vol. 11, March, pp.743–746.
- Skeels, M., Lee, B., Smith, G. and Robertson, G.G. (2010) 'Revealing uncertainty for information visualization', *Information Visualization*, Vol. 9, No. 1, March, pp.70–81.
- Talbot, J., Lee, B., Kapoor, A. and Tan, D.S. (2009) 'EnsembleMatrix: interactive visualization to support machine learning with multiple classifiers', *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, Boston, USA, pp.1283–1292.
- Wagstaff, K. (2012) 'Machine learning that matters', *Proceedings of ICML2012*, Edinburgh, Scotland, pp.529–536.
- Witten, I.H., Frank, E. and Hall, M.A. (2011) *Data Mining: Practical Machine Learning Tools and Techniques*, Elsevier, Burlington, MA, USA.
- Zhou, J., Khawaja, M.A., Li, Z., Sun, J., Wang, Y. and Chen, F. (2015b) 'Making machine learning useable by revealing internal states update – a transparent approach', *International Journal of Computational Science and Engineering*, in press.
- Zhou, J., Li, Z., Wang, Y. and Chen, F. (2013) 'Transparent machine learning – revealing internal states of machine learning', *Proceedings of IUI2013 Workshop on Interactive Machine Learning*, Santa Monica, CA, USA, pp.1–3.
- Zhou, J., Sun, J., Chen, F., Wang, Y., Taib, R., Khawaji, A. and Li, Z. (2015a) 'Measurable decision making with GSR and pupillary analysis for intelligent user interface', *ACM Transactions on Computer-Human Interaction*, Vol. 21, No. 6, p.33.
- Zuk, T. and Carpendale, S. (2006) 'Theoretical analysis of uncertainty visualizations', *Proceeding of SPIE-IS&T Electronic Imaging*, San Jose, USA, Vol. 6060, p.606007.

Websites

- 'Apache Mahout: Scalable machine learning and data mining', 2012 [Online], Available: <http://mahout.apache.org/> (Accessed 15 October, 2012).
- 'BigML – Machine Learning Made Easy', 2012 [Online], Available: <https://www.bigml.com/> (Accessed 20 October, 2012).
- 'Google', *Google Prediction API*, 2012 [Online], Available: <https://developers.google.com/prediction/> (Accessed 20 October, 2012).
- 'KDnuggets', *Polls: Can Machine Learning be Trusted*, June 2008 [Online], Available: <http://www.kdnuggets.com/polls/2008/can-machine-learning-be-trusted.htm> (Accessed 29 October, 2013).
- 'MLcomp', 2013 [Online], Available: <http://mlcomp.org> (Accessed 15 October, 2013).

- 'OpenTox', 2013 [Online], Available: <http://opentox.org/> (Accessed 10 October, 2013).
- 'Orange', 2012 [Online], Available: <http://orange.biolab.si/> (Accessed 20 October, 2012).
- 'Pentaho', *Business Analytics and Business Intelligence Pentaho*, 2012 [Online], Available: <http://www.pentaho.com/> (Accessed 20 October, 2012).
- 'Precog', 2013 [Online], Available: <http://precog.com/> (Accessed 10 October, 2013).
- 'PSI', *Machine Learning as a Service*, 2013 [Online], Available: <http://psi.cecs.anu.edu.au/> (Accessed 15 October, 2013).
- 'RapidMiner', *Rapid-I RapidMiner*, 2012 [Online], Available: <http://rapid-i.com/content/view/181/190/lang,en/> (Accessed 20 October, 2012).
- 'Scikit-learn: Machine Learning in Python', 2013 [Online], Available: <http://scikit-learn.org/stable/> (Accessed 10 September, 2013).
- 'SGI', *MLC++*, 2012 [Online], Available: <http://www.sgi.com/tech/mlc/> (Accessed 15 October, 2012).
- 'Skytree: The Machine Learning Company', 2012 [Online], Available: <http://www.skytreecorp.com/> (Accessed 20 October, 2012).