Introduction to MAchine Learning & Knowledge Extraction (MAKE)

Article · July 2017
DOI: 10.3390/make1010001

1 author:

Andreas Holzinger
Medical University of Graz
467 PUBLICATIONS **6,114** CITATIONS

SEE PROFILE

machine learning & knowledge extraction

MDPI

Editorial

Introduction to MAchine Learning & Knowledge Extraction (MAKE)

Andreas Holzinger

Holzinger Group, HCI-KDD, Institute for Medical Informatics & Statistics, Medical University Graz, Auenbruggerplatz 2/V, 8036 Graz, Austria; a.holzinger@hci-kdd.org; Tel.: +43-316-385-13883

Received: 8 May 2017; Accepted: 23 June 2017; Published: 3 July 2017

Abstract: The grand goal of Machine Learning is to develop software which can learn from previous experience—similar to how we humans do. Ultimately, to reach a level of usable intelligence, we need (1) to learn from prior data, (2) to extract knowledge, (3) to generalize—i.e., guessing where probability function mass/density concentrates, (4) to fight the curse of dimensionality, and (5) to disentangle underlying explanatory factors of the data—i.e., to make sense of the data in the context of an application domain. To address these challenges and to ensure successful machine learning applications in various domains an integrated machine learning approach is important. This requires a concerted international effort without boundaries, supporting collaborative, cross-domain, interdisciplinary and transdisciplinary work of experts from seven sections, ranging from data pre-processing to data visualization, i.e., to map results found in arbitrarily high dimensional spaces into the lower dimensions to make it accessible, usable and useful to the end user. An integrated machine learning approach needs also to consider issues of privacy, data protection, safety, security, user acceptance and social implications. This paper is the inaugural introduction to the new journal of MAchine Learning & Knowledge Extraction (MAKE). The goal is to provide an incomplete, personally biased, but consistent introduction into the concepts of MAKE and a brief overview of some selected topics to stimulate future research in the international research community.

Keywords: Machine Learning; Knowledge Extraction

Section 1 of this inaugural paper contains a very short executive summary, intentionally without references for the sake of brevity. Section 2 provides an incomplete introduction of and a personal view to the field of MAchine Learning (MA) and Section 3 to Knowledge Extraction (KE). Section 4 provides three selected future research challenges aiming to act as teaser to stimulate further research. Section 5 lists three benefits, differences and added values of the new journal. Finally, Section 6 provides an overview on the *integrative machine learning* approach on which this new journal builds and fosters.

1. Executive Summary: Why MAchine Learning & Knowledge Extraction (MAKE)?

Machine learning deals with *understanding intelligence* for the design and development of algorithms that can learn from data, to gain knowledge from experience and improve their learning behaviour over time. The challenge is to discover relevant structural and/or temporal patterns ("knowledge") in data, which is often hidden in arbitrarily high dimensional spaces, thus not accessible to a human. Today, machine learning is the fastest growing technical field, having many application domains, e.g., smart health, smart factory (Industry 4.0), etc. with many use cases from our daily life, e.g., recommender systems, speech recognition, autonomous driving, etc. The grand challenges are in sensemaking, in context understanding, and in decision making under uncertainty. The real-world is full of uncertainties and probabilistic information—and probabilistic inference enormously influenced artificial intelligence and statistical learning. The inverse probability allows to infer unknowns, to learn

from data and to make predictions to support decision making. Increasingly complex data sets require efficient, useful and usable solutions for knowledge discovery and Knowledge Extraction.

2. Machine Learning

Machine Learning (ML) is a very practical field offering many solutions to problems in our daily life, thus making it so enormously useful today [1]. This visible and convincing success is mostly due to three facts:

- (1) engineer's acceptance of the concept of *probable information in an uncertain world* [2];
- (2) the power and applicability of *statistical learning theory* (see a few notes below); and
- (3) the success of *deep learning* [3,4] (see end of this chapter).

ML is grounded in Statistical Learning Theory (SLT) which provides a large framework for studying fundamental questions of learning and inference, extracting knowledge, making predictions and decisions and constructing formal models from data. Ultimately, SLT contributes to help to design better learning algorithms [5–7].

Uncertainty and Probabilistic Reasoning. The basis for the great success of ML was set more than 250 years ago by THOMAS BAYES (1701–1761), whose work on decision making under uncertainty was communicated after his dead by RICHARD PRICE (1723–1791) [8]. However, it was actually PIERRE SIMON DE LAPLACE (1749–1827) some 20 years later [9], who generalized these ideas and made the field of probabilistic reasoning accessible, usable and useful for computational approaches today. A further success factor was the predictive power of Gaussian Processes, which have been successfully used for dealing with stochastic processes in time [10]. A *Gaussian process (GP)* can be seen as a generalization of the normal probability distribution, which is named after CARL FRIEDRICH GAUSS (1777–1855), and which can be used as a prior probability distribution over functions [11]. This idea is surprisingly useful now for us dealing with high-dimensional data, because Bayesian inference can be easily applied, consequently it unites a consistent view with computability. Moreover, it is fascinating that the probabilistic reasoning approach fits well to explanations of human learning and problem [12–14]. Furthermore, much practical value provides the use of *probabilistic programming*. This programming concept is different from traditional programming, in a way that parts of the program are not fixed in advance; instead they take on values generated at runtime by random sampling procedures. A good example for this approach is the combination of probabilistic programming and Particle Markov Chain Monte Carlo (PMCMC), which allows automatic Bayesian inference on probabilistic models including stochastic recursion [15]; for an implementation in Python see [16]. The two additional powerful constructs to functional or imperative programming concepts include [17]:

- (1) the ability to draw values at random from probability distributions, and
- (2) the ability to condition values of variables in a program via observations.

Many real-world problems of our daily life can be harnessed by probabilistic programs due to the applicability of probabilistic inference, i.e., computing an explicit representation of the probability distribution implicitly specified by a probabilistic program. Depending on the application, the desired output from the inference may vary, e.g., if we want to estimate the expected value of a function f with respect to the distribution, or the mode of the distribution, or a set of samples drawn from this distribution [1].

Artificial Generation of Knowledge from Experience. ML as a field of computer science started seven decades ago with ideas on developing algorithms that can automatically learn from data to gain knowledge from experience and to gradually improve their learning behaviour. The original definition was *"the artificial generation of knowledge from experience"*, and first studies have been performed with games [18]. While statistics aimed to provide a human the tools to analyze data manually, the aim of ML was from the beginning to replace the human, and similarly as we humans do, to learn automatically from data to make predictions and decisions. Consequently, ML was always a field of overlapping

interest between cognitive science and computer science [19]. The field progressed enormously in the last two decades with application successes in various fields, ranging from Astronomy to Zoology, mostly due to the availability of what is called "Big Data", collected by satellites, telescopes, high throughput machines, sensor networks, smart phones, etc. [20]. The best practice examples today include autonomous vehicles, recommender systems, or natural language understanding [21]. Finally, the convincing successes of deep belief network approaches [4,22] made the field very prominent (see below).

Meanwhile industry from Amazon to Zalando is investing a lot into research as they envision enormous business potential in the near future which also stimulates fruitful cooperation between academia and industry, and even small companies have identified the value of ML for solving a large variety of business relevant problems [23]. Health informatics is among the greatest application challenges, which is not surprising, because medicine is a good example for a domain full of uncertainty, where we are constantly confronted with probabilistic, unknown, incomplete, heterogenous, noisy, dirty, erroneous, inaccurate, and missing data sets in arbitrarily high dimensional spaces, which poses grand challenges to ML [24,25].

Inverse Probability Allows to Infer Unknowns and to Make Predictions. ML builds mainly on three pillars of mathematics: linear algebra, optimization and probability theory, although many other mathematical areas are involved, see e.g., [26]. Probability theory [27] provides the mathematical language for representing of and dealing with uncertainty, similarly as calculus is the language for representing of and dealing with rates of change (refer to ZHOUBIN GHAHRAMANI (2013) [28]. The typical data organization is in form of n-dimensional arrays, where the rows represent the samples (data items) and the columns represent the attributes (features), which can be seen as a n-dimensional vector of attributes and the array as a matrix. We can learn from data—even from high-dimensional data in $\mathbb{R}^n$—by transformation of the prior probability distributions into posterior probability distributions. To illustrate this learning process let us show a simple example here in $\mathbb{R}^2$.

Note: events are labeled with capital letters A; A random variable is also denoted by capital X and may take values in small letters x; the probability of an event is capital $P(A)$. A connection between values and events is in the case "$X = x$", i.e., the event X takes on the value x; A discrete random variable has a probability mass function small $p(x)$, and the connection between P and p is that $P(X = x) = p(x)$. Note also that a continuous random variable has a probability density function $f(x)$, and the connection between P and f is that $P(a \leqq X \leqq b) = \int_a^b f(x)dx$; in the following we use a small h_n to indicate a hypothesis n, and small θ to indicate the hypothesized value of a model parameter; we use capital letters $\mathcal{D}$ when talking about data as events, and small x when talking about data as values. The expression $p(x)$ with $0 \leqq p(x) \leqq 1$ denotes the probability that x is true. Following BAYES we can now instead of x, y denote d for data and h for the hypothesis, and with capital $\mathcal{H} = \{h_1, h_2, ..., h_n\}$ define the hypotheses space; then $\forall(h, d)$

$$P(h|d) = \frac{P(d|h)P(h)}{\sum_{h' \in \mathcal{H}} P(d|h')P(h')} \tag{1}$$

We can now use the ML notation by replacing the symbols: we replace d by $\mathcal{D}$ to denote our observed data set, and we replace h with $p(\theta)$ to denote the (yet) unknown parameters of our model. $\vec{\theta}$ is called the parameter vector (set of parameters that generated (x, y)), and the goal is to estimate θ from given x and y. Let us consider n data contained in a set $\mathcal{D} = x_{1:n} = \{x_1, x_2, ..., x_n\}$, and let be the likelihood $p(\mathcal{D}|\theta)$ and specify a prior $p(\theta)$, consequently we can compute the posterior:

$$p(\theta|\mathcal{D}) = \frac{p(\mathcal{D}|\theta)p(\theta)}{p(\mathcal{D})} \tag{2}$$

Figure 1 illustrates this learning process: We receive the posterior probability function (green) by multiplying the prior probability (red) times the likelihood (blue), divided by the evidence (normalization—in high-dimensional spaces this is a challenge to computation). In short: the posterior

is the likelihood times the prior through the evidence and the *inverse probability* allows us to learn from data, to infer unknowns and to make predictions [29].

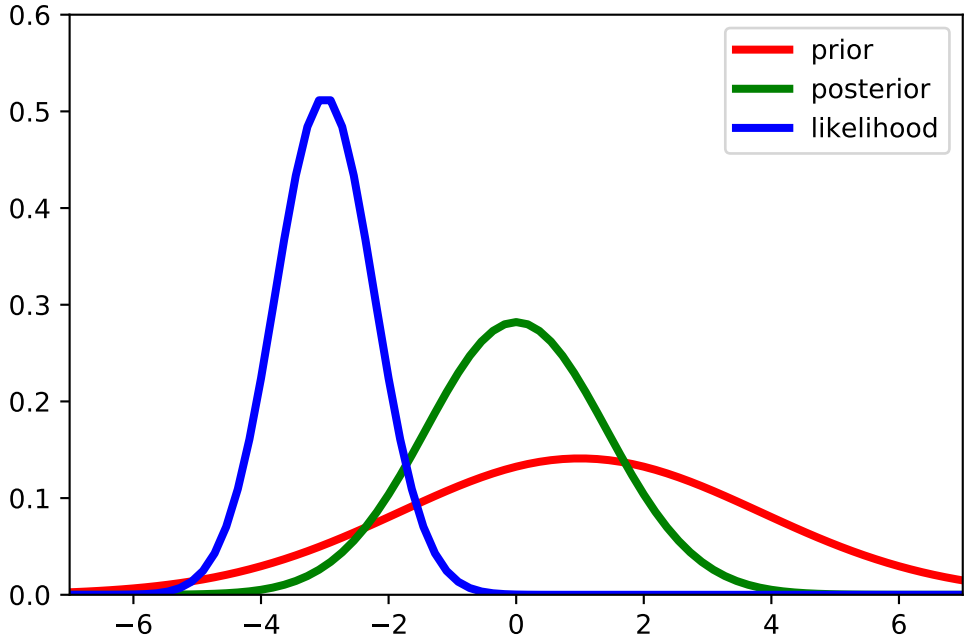

Figure 1. The posterior probability function (green) is received by multiplying the prior probability (red) times the likelihood (blue), divided by the evidence.

Representation Learning and Context. The performance of any ML algorithm is dependent on the choice of the *data representations*. Consequently, these data representations aka features are key for learning and understanding (see also Section 3), hence much effort in ML goes into the design of preprocessing pipelines and in data transformations and data mappings that result in a respective representation which supports effective ML. Current learning algorithms have still an enormous weakness: they are unable to *extract the discriminative knowledge* from the data. Consequently, it is of utmost importance to expand the universal applicability of learning algorithms, hence, to make them less dependent on (hand crafted) feature engineering. BENGIO, COURVILLE and VINCENT (2013) [30] argue that this can only be achieved if the algorithms can learn to identify and to *disentangle the underlying exploratory factors* already existent among the low-level data. That entails that a truly intelligent algorithm is required to understand the *context*, and to be able to discriminate between relevant and irrelevant features—similarly as we humans can do. "What is interesting?" and "What is relevant?" are hard questions, and as long as we cannot achieve this grand goal with automatic approaches, we have to develop algorithms which can be applied by a domain expert. Such an expert is likely to be aware of what is interesting and relevant in his/her domain, thereby can design features more appropriately than a machine learning engineer, who is mostly no domain expert. This calls for a new kind of *algorithm usability* [31]. Switching back to our probabilistic perspective, this would mean that learning features from data can be seen as recovering a parsimonious set of latent random variables (i.e., according to OCCAMS'S razor, see [32] for a critical discussion), representing a distribution over the observed data to express a probabilistic model $p(x, h)$ over the joint space of the latent variables, h, and the observed data x. Also this approach fits well into the perspective of cognitive science [33].

Automatic ML vs. Interactive ML. The ultimate goal of the worldwide ML community is to develop algorithms/systems which can *automatically* learn from data *without any human-in-the-loop* [34]. This *automatic machine learning (aML)* works well when having large amounts of training data [35], consequently "Big Data" is beneficial for automatic approaches. However, sometimes we do not have

large amounts of data, and/or we are confronted with rare events and/or hard problems. The health domain is a representative example for a domain with many such complex data problems [24,36]. In such domains the application of fully automatic black-box approaches ("press the button and wait for the results") seems elusive in the near future. Again, a good example are Gaussian processes, where aML approaches (e.g., kernel machines [37]) struggle on function extrapolation problems, which are astonishingly trivial for human learners [38]. Consequently, interactive Machine Learning (iML) approaches, by integrating a human-in-the-loop (e.g., a human kernel [33]), or the involvement of a human directly into the machine-learning algorithm [39], thereby making use of human cognitive abilities, is a promising approach. iML-approaches can be of particular interest to solve problems, where we are lacking big data sets, deal with complex data and/or rare events, where traditional learning algorithms suffer of insufficient training samples. In the medical domain a "doctor-in-the-loop" can help with his/her expertise in solving problems which otherwise would remain NP-hard. A recent experimental work [40] demonstrates the usefulness on the Traveling Salesman Problem (TSP), which appears in a number of practical problems, e.g., the native folded three-dimensional conformation of a protein in its lowest free energy state; or both 2D and 3D folding processes as a free energy minimization problem belong to a large set of computational problems, assumed to be conditionally intractable [41]. As the TSP is about finding the shortest path through a set of points, it is an intransigent mathematical problem, where many heuristics have been developed in the past to find approximate solutions [42]. There is evidence that the inclusion of a human can be useful in numerous other problems in different application domains, see e.g., [43,44]. However, for clarification, iML means the integration of a human into the *algorithmic* loop, i.e., to open the black box approach to a glass box. Other definitions speak also of a human-in-the-loop, but it is what we would call classic supervised approaches [45], or in a total different meaning to put the human into physical feedback loops [46].

Deep Learning. Last but not least deep learning (DL) approaches should be briefly mentioned here, because they are currently heavily contributing to the popularity of ML in the broader community generally, and to the success of industrial applications specifically. A few sentences above we have discussed the importance of learning representations. Deep learning approaches can be seen as *representation learning* methods with *multiple levels of representations* consisting of a number of simple non-linear single levels, where each level transforms the respective level into a representation of a higher—more abstract—level. Important here is to emphasize that the features are *not* hand-crafted, instead fully automatically learned from the data, layer by layer, using a general-purpose learning procedure [4]. The practical value has been proven in different applications, e.g., in computer vision [47], natural language understanding [48], connectomics (study of brain circuits) [49], bioinformatics [50], health informatics [51–53], or in physics [54], to point only to a few examples. DL also contributes to advances in implementing human-level intelligence [55,56], hence contributes to cognitive science. For an excellent overview and a good explanation of the history of deep learning refer to SCHMIDHUBER (2015) [3]. Finally, it should be mentioned that deep learning as it achieves so fantastic performance on particular tasks, it has also serious limitations: they are black-box approaches, where it is currently difficult to explain *how and why* a result was achieved, consequently lacking transparency and trust, are prone to catastrophic forgetting, are demanding huge computational resources, and need enormous amounts of training data (often millions of training samples), most of all they are poor at representing uncertainties.

Bayesian Deep Learning. Neural network approaches have achieved surprising success in certain application areas (e.g., machine vision, machine reading, machine hearing to mention three), however, simply being able to see, read, and hear is far from being truly intelligent, being able to understand the context. A good example is medical decision making: the medical professional looks at visible symptoms (e.g., on medical images), reads the corresponding report in the patient record, and hears the ailments of the patient. Now the medical doctor has to look for relations among different information, infer the etiology and to make predictions and finally decisions. A human can deal with uncertainties due to his/her previous knowledge and experience within a short time [57,58]. One of the pioneers

in combining BAYESIAN networks with probabilistic approaches to mathematically model *causality* was JUDEA PEARL [2,59]. This insights call for merging probabilistic graphical models with deep learning approaches (see the survey by WANG and YEUNG (2016) [60]). Neural network approaches (applied e.g., for regression and classification) do not well represent uncertainty, but BAYESIAN models offer a mathematically grounded framework to reason about model uncertainty. Recently, YARIN and GHAHRAMANI (2016) [61], developed a new theoretical framework casting dropout training in deep neural networks as approximate BAYESIAN inference in deep Gaussian processes, which provides new tools to model uncertainty with dropout neural networks, consequently inspires future work. However, a remaining big problem of deep learning is catastrophic forgetting [62,63].

Deep Transfer Learning. A very recent work by LEE, KIM, LEE and YOON (2017) [64] advances on deep learning for graph-structured data by incorporating another key concept: transfer learning (more details see in Section 4): Convolutional Neuronal Networks (CNN) and Recurrent Neural Networks (RNN) extract data-driven features from input data (e.g., image, video, and audio data) structured in typically low-dimensional regular grids. Grid structures are often assumed to have statistical characteristics (e.g., stationarity, locality, etc.) to facilitate the modeling process. Learning algorithms can take advantage of this assumption and boost performance by simply reducing the complexity of the parameters [65]. By overcoming the common assumption that training and test data should always be drawn from the same feature space and distribution, the transfer learning between different task domains can alleviate the burden of collecting new data and new training models for a new task. Given the importance of structural characteristics in graph analysis, it is necessary to transfer the data-driven structural features learned by deep networks from a source domain to a target domain.

3. Knowledge Extraction (KE)

Stochastic Ontologies. The combination of ontologies with ML approaches is a hot topic and not yet extensively investigated, having great future potential, particularly in complex domains such as the health domain. This is due to the fact, that both ontologies and ML constitute two indispensable technologies for domain specific knowledge extraction, actively used in knowledge-based systems. Little is yet known about how the two can be successfully integrated. The reason is that the two technologies are mainly used separately, without direct connection.

TSYMBAL ET AL. (2007), [66], emphasized that the knowledge extracted by the two techniques is complementary, consequently significant benefits can be obtained with an integration of both. A solution to this problem is of highest interest for health informatics, where relevant data sets are complex and of high dimensionality with heterogeneous features [67], but where at the same time sophisticated bodies of knowledge are available for a long time, for example in the form of well-established classification systems including the unified medical language system (UMLS), the international classification of diseases (ICD), or the standard nomenclature of medical terms (SNOMED), as well as ontologies from the *omics data world including OMIM, GO, or FMA, just to mention a few.

Ontology learning is the trend towards the automatic ML-based creation of ontologies, because hand-crafting ontologies is extremely labor intensive and time consuming. One example has been presented by BALCAN ET AL. (2013) [68], where they present and analyze a theoretical model to understand and explain the effectiveness of ontologies for learning multiple related tasks from primarily unlabeled data. In this model they show that an ontology, which specifies the relationships between multiple outputs, in some cases is sufficient to completely learn a classification using a large unlabeled data source. Interestingly, the motivator for this work was the famous Never Ending Language Learning (NELL) project by the group of TOM MITCHELL (2010) [69].

Features are key to learning and understanding. ANDREW Y. NG emphasizes in his courses that practical machine learning is feature engineering. Feature extraction and selection have become the focus of heavy research in areas for which data sets with hundreds of thousands of variables are

available, e.g., in natural language processing, gene expression arrays, or combinatorial chemistry [70]. In the following sections an incomplete, personally biased, but consistent overview about interesting topics relating to KE in natural language processing (NLP) and natural language understanding (NLU) is presented with a focus on and how to put it into a (personal) *context*.

Data as Knowledge Triggers. In his Stanford NLP lecture series, CHRISTOPHER D. MANNING (see also: [71]) pointed out that human language in general is a symbolic/categorical signaling system; most information it conveys is *not contained in the words or sentences themselves.* Rather, it triggers within the brain of the recipient a whole slew of associations relating to that person's specific experiences as well as something we might call *world knowledge*. Moreover, there is empirical evidence that, in some cases, a representation of the speakers' intentions is helpful [72], and there is agreement that *understanding* language (not mere language processing) is more than the use of fixed conventions and/or decoding combinatorial structures and that probabilistic modeling may be helpful here [73].

Consequently, language interpretation depends on uncertain real world knowledge, common sense, *and* contextual knowledge, which explains the dominance of feature engineering tasks in the field of NLP and reduces the actual machine learning part to mere numerical optimization. Generally, the success of machine learning algorithms depend on feature learning, aka representation learning, because different representations can entangle the explanatory factors of variation behind the data [30].

This contextual knowledge is even significant for the meaning of individual words, as e.g., the word *king* triggers different associations depending on its usage within particular domains (history, chess, pop culture, pirate, etc.). Methods to automatically encode these conceptual peculiarities emerged only recently [74,75] and open up a multitude of new business application scenarios, especially pertaining to the analysis of small snippets of text which contain insufficient information for purely statistical analysis (bag-of-words methods, see e.g., [76] and compare with the feature hashing trick [77]—analogous to the kernel trick [37,78]).

However, aside from incorporating world knowledge into concept encodings the main problem is in lacking sufficient personal context to extract not only knowledge but also meaning from texts and to provide individual recommendations. Such information can be easily found in social graphs, embedding individual data within neighborhoods, whose structure encodes context. The problem with this approach, however, is the *incomplete* knowledge about the graph structure, either because the data is unavailable or has been anonymized for security reasons (e.g., due to the production of open data sets). This leads us to the question of minimal viable data sets and possible methods for reconstruction.

Partial Context and Model (re-)construction. In their work on Kronecker graphs (this is a generative model for networks) LESKOVEC ET AL. (2010) [79] asked themselves the interesting question *"How can we generate synthetic, but realistic looking, time-evolving graphs?"*. Although graph generators had been around for a while at that time, they were hitherto mostly unable to produce graphs displaying real-world properties, such as heavy tails for degree distributions or densification and shrinking diameters over time. Viable social network generators could also help with supplementing partially known graphs and therefore enabling ML approaches on much smaller or fragmented knowledge bases. In addition to generating realistic network structures, the task of re-populating anonymized feature vectors based on their structural embedding could prove crucial for practical ML: For instance, in SNAP (Stanford Network Analysis Platform [80]) anonymized FB (Facebook) ego graphs, features such as *university attended* are represented only as *anonymized feature xyz*. Whereas this obviously tells us that all people who attended *anonymized feature 223* attended the same university, we can only guess as to which school is represented by *anonymized feature 224*, which would result in an independent draw from whatever distribution we assume. Incorporating the social embedding of nodes into our model would make that draw depend on the values of connected nodes in the graph, allowing us to apply efficient sampling methods such as MCMC (Markov Chain Monte Carlo) [81] to the problem. As a result, ML performance on anonymized graphs could be boosted

without any personal re-identification attempts; a crucial advantage as more and more countries adopt stringent data privacy and security laws [82].

A slightly different problem in model construction is that of finding suitable formal constraints from unstructured information formats. As an example we can take the construction of Business process models from event-based data such as automatic log files, or Github commit messages, which usually only provide positive examples of event paths, but omit negative information including state transitions that were prevented from taking place. The authors of [83] developed an algorithm incorporating artificially generated negative events to act as additional constraints on the model, resulting in higher specificity—not allowing unintended, random behavior. Coupling this approach with semantic embeddings described above could result in automatic sequence model extraction from unstructured and un-processed data with tremendous potential in automatic exploration and sense-making of hitherto unspecified processes, e.g., disease stage development in the health sector or even research in underlying biological processes.

Federated Learning and Client-side Learning. As noted in [84] data are often not available in bulk but arrive sequentially over time, so it is necessary to update an already learned model in real-time (also called *sequential learning* or *online learning*). This furthermore holds the advantage of computational simplicity by not having to store the entire data structure for model updates, especially when those adaptations can be performed in a de-centralized manner. Taking the idea of knowledge extraction from partially known models to the extreme, one could propose learning schemes in which global models result partly or solely from a large number of clients possessing only fragmented views on raw data. In a world permeated by smart devices with tremendous computing power and ubiquitous network access, such an approach could soon be poised to combine the above ideas into a powerful global knowledge extraction "organism", which is the underlying idea of Google's new *federated learning* approach [85]. In a recent work they trained a deep neural network (for an overview of deep learning in neural networks refer to: [3]) in a federated learning model by application of distributed gradient descent across user-held training data on mobile devices [86], which is a current hot topic [87].

Taking a step back from those futuristic perspectives, LESKOVEC ET AL. (2006) [88] have conducted experiments on recommendation cascades, which are sequences of accepted and forwarded recommendations. Building on their insight that a vast majority of relevant recommendations within a social network originate from nodes within a radius of 1.2 and taking modern publish/subscribe architectures into account, we can arrive at the idea of a *local sphere* of data permanently residing (and kept up-to-date) on clients such as smart phones or even Web browsers. Thus scalable recommender systems could be implemented with only a fraction of the cost and algorithmic complexity required today, paving the way for even greater "democratization" of Machine Learning related markets in the future.

4. Selected three Future Research Challenges

Multi-Task Learning (MTL) aims to improve the prediction performance by learning a problem together with multiple, different but related other problems through shared parameters or a shared representation. The underlying principle is *bias learning* based on probable approximately correct learning (PAC learning) [89]. To find such a bias is still the hardest problem in any ML task and essential for the initial choice of an appropriate hypothesis space, which must be large enough to contain a solution, and small enough to ensure a good generalization from a small number of data sets. Existing methods of bias generally require the input of a human-expert-in-the-loop in the form of heuristics and domain knowledge to ensure the selection of an appropriate set of features, as such features are key to learning and understanding. However, such methods are limited by the accuracy and reliability of the expert' s knowledge (robustness of the human) and also by the extent to which that knowledge can be transferred to new tasks (see next subsection). BAXTER (2000) [90] introduced a model of bias learning which builds on the PAC learning model which concludes that learning

multiple related tasks reduces the sampling burden required for good generalization and bias that is learnt on sufficiently many training tasks is likely to be good for learning novel tasks drawn from the same environment (the problem of transfer learning to new environments is discussed in the next subsection). A practical example is *regularized MTL* [91], which is based on the minimization of regularization functionals similar to Support Vector Machines (SVMs, a good introduction can be found in [92]), that have been successfully used in the past for single-task learning. The regularized MTL approach allows to model the relation between tasks in terms of a novel kernel function that uses a task-coupling parameter and largely outperforms single-task learning using SVMs. However, multi-task SVMs are inherently restricted by the fact that SVMs require each class to be addressed explicitly with its own weight vector. In a multi-task setting this requires the different learning tasks to share the *same set of classes.* An alternative formulation for MTL is an extension of the large margin nearest neighbor algorithm (LMNN) [93]. Instead of relying on separating hyper-planes, its decision function is based on the nearest neighbor rule which inherently extends to many classes and becomes a natural fit for MTL. This approach outperforms state-of-the-art MTL classifiers, and here many research challenges remain open [94].

Transfer Learning is the ability to learn tasks permanently and this is crucial to the development of any artificial intelligence. Humans can do that very good—even very little children. A good counterexample are neural networks (deep learning) which in general are not capable of it and are considerably hampered by *catastrophic forgetting*.

The synaptic consolidation in human brains enables continual learning by reducing the plasticity of synapses that are vital to previously learned tasks. KIRKPATRICK ET AL. (2017) [95], implemented an algorithm that performs a similar operation in artificial neural networks by constraining important parameters to stay close to their old values. As known a deep neural network consists of multiple layers of linear projections followed by element-wise non-linearities. Learning a task consists basically of adjusting the set of weights and biases θ of the linear projections, consequently, many configurations of θ will result in the same performance which is relevant for the so-called elastic weight consolidation (EWC): over-parametrization makes it likely that there is a solution for task B, θ^*_B, that is close to the previously found solution for task A, θ^*_A. While learning task B, EWC therefore protects the performance in task A by constraining the parameters to stay in a region of low error for task A centered around θ^*_A. This constraint has been implemented as a quadratic penalty, and can therefore be imagined as a mechanical spring anchoring the parameters to the previous solution, hence the name elastic.

In order to justify this choice of constraint and to define which weights are most important for a task, it is useful to consider neural network training from a probabilistic perspective. From this point of view, optimizing the parameters is tantamount to finding their most probable values given some data $\mathcal{D}$. Interestingly, this can be computed as conditional probability $p(\theta|\mathcal{D})$ from the prior probability of the parameters $p(\theta)$ and the probability of the data $p(\mathcal{D}|\theta)$ by:

$$\log p(\theta|\mathcal{D}) = \log p(\mathcal{D}|\theta) + \log p(\theta) - \log p(\mathcal{D})$$

Here, the international research community is challenged to contribute on avoiding the problem of catastrophic forgetting, which is a hot topic with many open research avenues [63].

According to PAN and YANG (2010) [96] a major assumption in many ML algorithms is, that both the training data and future (unknown) data must be in the same feature space and required to have the same distribution. In many real-world applications, particularly in the health domain, this is not the case: Sometimes we have a classification task in one domain of interest, but we only have sufficient training data in another domain of interest, where the latter data may be in a completely different feature space or follows a different data distribution. In such cases transfer learning would greatly improve the performance of learning by avoiding much expensive data-labeling efforts, however, many open questions remain for future research [97].

Multi-Agent-Systems (MAS) are collections of many agents interacting with each other. They can either share a common goal (for example an ant colony, bird flock, or fish swarm etc.), or they can pursue their own interests (for example as in an open-market economy). MAS can be traditionally characterized by the facts that (a) each agent has incomplete information and/or capabilities for solving a problem, (b) agents are autonomous, so there is no global system control; (c) data is decentralized; and (d) computation is asynchronous [98]. For the health domain of particular interest is the *consensus problem*, which formed the foundation for distributed computing [99]. The roots are in the study of (human) experts in group consensus problems: Consider a group of humans who must act together as a team and each individual has a subjective probability distribution for the unknown value of some parameter; a model which describes how the group reaches agreement by pooling their individual opinions was described by DeGroot (1974) [100] and was used decades later for the aggregation of information with uncertainty obtained from multiple sensors [101] and medical experts [102]. On this basis Olfati-Saber et al. (2007) [103] presented a theoretical framework for analysis of consensus algorithms for networked multi-agent systems with fixed or dynamic topology and directed information flow. In complex real-world problems, e.g., for the epidemiological and ecological analysis of infectious diseases, standard models based on differential equations very rapidly become unmanageable due to too many parameters, and here MAS can also be very helpful [104]. Moreover, collaborative multi-agent reinforcement learning has a lot of research potential for machine learning [105], which is very suitable for collaborative interactive machine learning [106].

5. Benefits of the New Journal MAKE

There are excellent and well established top journals in the field, for example: Machine Learning (MACH), the Journal of Machine Learning Research (JMLR), or the Knowledge and Information Systems (KAIS) journal—just to mention three.

Springer **Machine Learning** (MACH) is in operation since 1986 and is an established international forum for research on computational approaches to learning. The journal publishes articles reporting substantive results on a wide range of learning methods applied to a variety of learning problems. In 2001, forty editors and members of the editorial board of Machine Learning resigned in order to support the **Journal of Machine Learning Research** (JMLR), which was at that time the pioneering journal in machine learning: online available, open access and the copyright remaining with the authors. The JMLR is now the top-end journal and *the* benchmark of the field.

Springer **Knowledge and Information Systems** (KAIS) is in operation since 1999 and provides an international professional forum for advances on all topics related to knowledge systems and information systems. The journal focuses on systems, including their theoretical foundations, infrastructure and enabling technologies.

The journal for **MAchine Learning & Knowledge Extraction** (MAKE) is a peer-reviewed open access journal and the copyright remains with the authors. The publisher is the Multidisciplinary Digital Publishing Institute (MDPI), headquartered in Basel (Switzerland) with offices in Europe and China.

Unique features include:

- *Promotion* of a cross-disciplinary *integrated machine learning* approach addressing seven sections to concert international efforts without boundaries, supporting collaborative, trans-disciplinary, and cross-domain collaboration between experts from these seven disciplines (see next section for details);
- *Appraisal* of these different fields shall foster diverse perspectives and opinions, hence offering a platform for the exchange of novel ideas and a fresh look on methodologies to put crazy ideas into business for the benefit of the human; additionally to foster education (see details below);
- *Stimulation* of replications and further research by inclusion of data and/or software regarding the full details of experimental work as supplementary material, if unable to be published in

a standard way, or by providing links to repositories (e.g., Github) shall provide a benefit for the international research community (see issues of availability, usability and acceptance, below).

Machine Learning Education. The advances of machine learning research and the practical success in many different domains call worldwide for a new kind of research-oriented graduates. To keep students up-to-date with most recent material in such an innovative field is not an easy task. In a recent talk NANDO DE FREITAS pointed out that alone deep learning research is like playing with a huge amounts of Lego blocks. Finding and putting together the right blocks is difficult. An integrative machine learning approach calls also for an *integrated teaching approach* and needs a concerted effort of the various disciplines.

In innovative and rapidly changing areas the application of Research-Based Teaching (RBT) approaches can be of great help [107], where e.g., the curriculum is designed around current research topics, always grounded in relevant and necessary fundamentals. A sample curriculum for a course of "machine learning in health informatics" is described in [108].

Consequently, the journal supports educational efforts, particularly in the form of valuable, concise, strictly peer-reviewed tutorial papers, similarly to the IEEE Signal Processing Magazine, which is doing an excellent job for the benefit of their community, see three examples [109–111].

Responsibility, Ethical/Social Issues, Law, Technology Assessment. Both scientists and engineers are responsible for their developments. This is particularly true for the field of machine learning and its implications on our society. The enormous future potential of machine learning specifically, and artificial intelligence generally, requires to take not only over social responsibility, but even maximising the social benefit of these technologies [112]. Here it is important not only to take care of ethics in the sense of how humans use computational approaches, instead to deal with machine learning ethics, which is concerning the ethical dimension to ensuring that the behavior of machines toward human users is ethically acceptable [113], which is of increasing importance in learning machines, autonomous systems and decision making [114–116].

Critical discussions of social implications are therefore of utmost importance, in combination with issues of regional, national, transnational and international laws, directives and regulations with a strong focus on privacy, data protection, safety and security (which is a own section of the integrated approach, see next chapter).

Availability, Usability, Acceptance. The value of machine learning algorithms for the progress of the international research community is to a large part dependent on three important issues:

- (1) availability of open source code associated with research papers [117];
- (2) reproducibility of available methods and tools which is a cornerstone in fundamental science;
- (3) usability and usefulness of that code for solving real-world problems.

The problem is still that much potential of sophisticated methods and tools can not be used due to lack of availability, interoperability, and reproducibility. Another huge obstacle is the lack of usability of available machine learning methods and tools, which often makes it hard for a domain expert to apply them. This calls for adequate machine learning usability [118].

It is well understandable that all these topics mentioned within the previous pages cannot be tackled within one single discipline; instead it needs an combined effort of various sections, brought together in a concerted integrative approach. This leads us to the last open question: What is this *"integrative machine learning"* approach?

6. Integrative Machine Learning

The meaning of the words integrative or integrated stems from Latin *integratus*, which means "make whole", i.e., *"to put together parts or elements and combine them into a harmonious, interrelated whole, so that constituent units function in a cooperatively manner"*.

Although machine learning has a lot of awesome theoretical aspects and is deeply grounded in the field of artificial intelligence (AI) [29,119], it should always be emphasized that machine learning is

a very practical field with many diverse application areas. Looking into the past, the field was just three decades ago a small niche with a few applications. Meanwhile, it evolved to a dominant field, constantly growing, with a lot of facets of enormous both width and depth.

Such a field needs an integrative approach.

Integrative/Integrated Machine Learning is based on the idea of combining the best of the two worlds dealing with understanding intelligence, which is manifested in the HCI–KDD approach: [120–122]: Human–Computer Interaction (HCI), rooted in cognitive science, particularly dealing with *human intelligence*, and Knowledge Discovery/Data Mining (KDD), rooted in computer science particularly dealing with *computational intelligence* [67]. This approach fosters a complete machine learning and knowledge extraction (MAKE) pipeline, ranging from the very physical issues of data pre-processing, mapping and fusion of arbitrarily high-dimensional data sets (see right side in Figure 2) up to the visualization of the results in a dimension accessible to a human end-user and making data interactively accessible and manipulable (left side in Figure 2).

Cognitive Science studies the principles of human learning from data to understand intelligence. The Motto of DEMIS HASSABIS from Google Deepmind is *"Solve intelligence—then solve everything else"* (see also: [55]). Our natural surrounding is in $\mathbb{R}^3$ and humans are excellent in perceiving patterns out of data sets with dimensions of ≤ 3. In fact, it is amazing how humans extract so much knowledge from so little data [19], which is a perfect motivator for the concept of interactive Machine Learning (iML), i.e., using the experience and knowledge of humans to help to solve problems which would otherwise remain computationally intractable. However, in most application domains, e.g., in the health informatics domain, we are challenged with data of arbitrarily high dimensions [25]. Within such data, relevant *structural* patterns and/or *temporal* patterns ("knowledge") are hidden, knowledge is difficult to extract, hence not accessible to a human. There is need to bring the results from high dimensions into the lower dimension, where humans are working on 2D surfaces on different devices (from tablet computers to large wall-displays), and hence the representation is limited to $\mathbb{R}^2$.

Computer Science studies the principles of computational learning from data to understand intelligence [21]. Computational learning has been of general interest for a very long time, but we are far away from solving intelligence: facts are not knowledge and descriptions are not insight. A good example is the famous book by Nobel prize winner ERIC KANDEL *"Principles of Neural Science"* [123] which doubled in volume every decade—effectively, the goal should be to make this book shorter.

At high-level, cognitive science and machine learning had little overlap in the past. Most computer engineers had their interest in their machines and were not interested in any human factors. At the same time cognitive scientists showed rarely interest in computational approaches. Actually, it was the great practical success of machine learning in the last two decades, which brought them both together. Many successful people of the community nowadays have a background in both cognitive science and computer science and are fostering a close collaboration of both fields.

Even at low-level, HCI and KDD did not harmonize in the past. HCI had its focus on specific experimental paradigms, embedded rather in psychological issues, aiming to be cognitively plausible and resulting in nagging at design issues. KDD had its focus on computational learning problems, embedded in engineering, thereby focusing on algorithm optimization at small scale, and rather ignoring any design issues concerning a possible end user.

Consequently, a concerted effort of both worlds along with a multi-disciplinary skill-set encompassing various specializations can be highly beneficial for tackling the challenges of the future to help to understand intelligence and to develop software which learns from experience – similarly as we humans do.

The MAKE-topics may be illustrated (see Figure 2) by seven sections with the aim to fertilize cross-disciplinary thinking. It is well known that scientific progress often emerges at the overlapping areas of seemingly distinct sections. In the following only a non-detailed high-level description is given (a description of challenges of each section is beyond the scope of this inaugural paper, and could be on the agenda for future work).

The MAKE-Topics may be illustrated by 7 sections (see Figure 2):

Section 1: Data: Data preprocessing, integration, mapping, fusion. This starts with understanding the physical aspects of raw data and fostering a deep understanding of the data ecosystem, particularly within an application domain.

Section 2: Learning: Algorithms. The core section deals with all aspects of learning algorithms, in the design, development, experimentation and evaluation of algorithms generally and in the application to application domains specifically.

Section 3: Visualization: Data visualization, visual analysis. At the end of the pipeline there is a human, who is limited to perceive information in dimensions $\leqq 3$. It is a hard task to map the results, gained in arbitrarily high dimensional spaces, down to the lower dimensions, ultimately to $\mathbb{R}^2$.

Section 4: Privacy: Data Protection, Safety & Security. Worldwide increasing demands on data protection laws and regulations (e.g., the new European Union data protection directions), privacy aware machine learning becomes a necessity not an add-on. New approaches, e.g., federated learning, glass-box approaches, will be important in the future. However, all these topics needs a strong focus on usability, acceptance and social issues.

Section 5: Network Science: Graph-Based Data Mining. Graph theory provides powerful tools to map data structures and to find novel connections between data objects and the inferred graphs can be further analyzed by using graph-theoretical, statistical and ML techniques.

Section 6: Topology: Topology-Based Data Mining. The most popular techniques of computational topology include *homology* and *persistence* and the combination with ML approaches would have enormous potential for solving many practical problems.

Section 7: Entropy: Entropy-Based Data Mining. Entropy can be used as a measure of *uncertainty in data*, thus provides a bridge to theoretical and practical aspects of information science (e.g., Kullback–Leibler Divergence for distance measure of probability distributions).

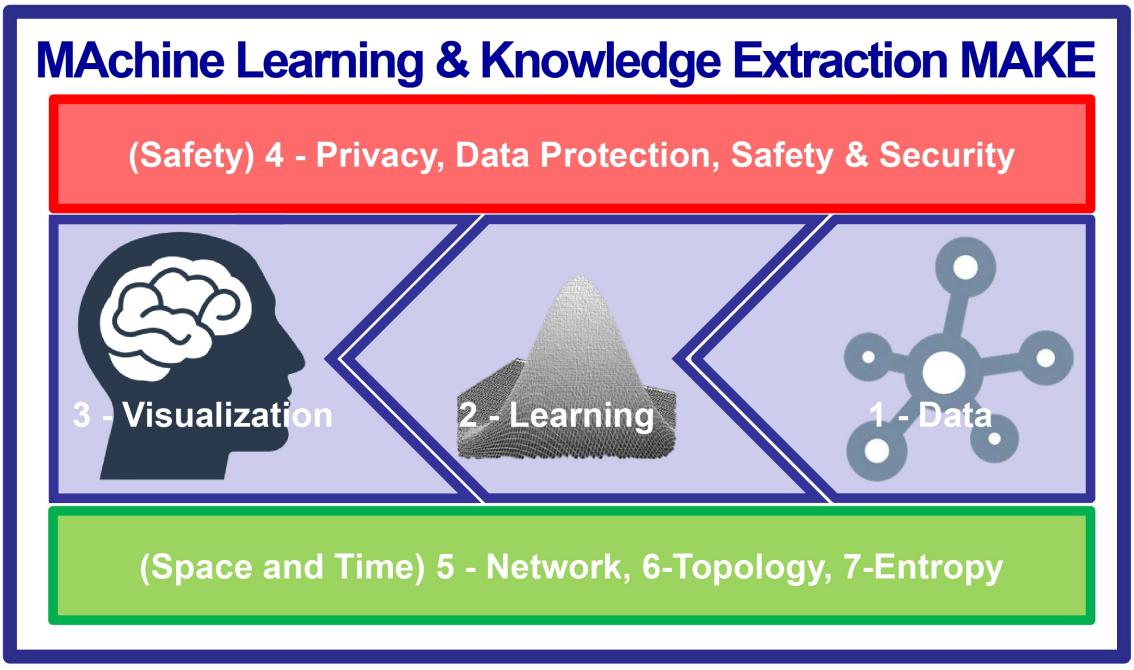

Figure 2. The big picture of the MAKE-Pipeline: The horizontal process chain (blue box) encompasses the whole machine learning pipeline from physical aspects of raw data, to human aspects of data visualization; while the vertical topics (green box) include important aspects of space/structure (graphs/networks/computational topology) and time (entropy); privacy, data protection, safety and security are mandatory topics for many application domains (e.g., health).

7. Conclusions

Machine learning deals with *understanding intelligence* for building algorithms that can learn from data, to gain knowledge from experience and improve their learning behaviour over time. The challenge is in knowledge extraction to discover *relevant* structural and/or temporal patterns ("knowledge") in data, which is often hidden in arbitrarily high dimensional spaces—not accessible to a human. Consequently, a combined view to MAchine Learning & Knowledge Extraction (MAKE) has enormous future potential for both academia and industry, e.g., in bridging probabilistic approaches with classic ontological approaches. Grand challenges are in sensemaking, in understanding the context, and in support of making decisions under uncertainty, which is needed for solving problems in various application domains from Astronomy to Zoology.

The ultimate goal is to design and develop human-level intelligent algorithms which can *automatically* learn from data, and improve with experience over time *without any human-in-the-loop*. However, the application of such fully automatic approaches in complex domains (e.g., health) seems elusive in the near future. A convincing example are Gaussian processes, where automatic approaches (e.g., kernel machines) struggle on function extrapolation problems, which are quite trivial for human learners. Consequently, interactive approaches, by integrating a human-into-the-loop (e.g., a human kernel), thereby making use of human cognitive abilities, seems to be a promising approach for the near future.

This is particularly useful to solve problems where we do not have "Big Data", instead are lacking large amounts of training data, deal with complex data and/or rare events, where traditional learning algorithms (e.g., deep learning) suffer due to insufficient training samples. Such an "expert-in-the-loop" can help to solve solve problems which otherwise would remain NP-hard.

For all these reasons there is much room for a new peer-reviewed open journal, complementary to the already existing established journals. The goal of the new journal "MAchine Learning & Knowledge extraction" (MAKE) is to provide an open platform to bring together researchers from diverse sections in a cross-disciplinary manner, without any boundaries, to stimulate fresh ideas and to encourage multi-disciplinary and cross-domain problem solving for the benefit of the human. Let's MAKE it!

Acknowledgments: The author cordially thanks the anonymous reviewers of this editorial paper for valuable feedback and comments. The author thanks the Holzinger group and all members of the organically grown international HCI-KDD network, fostering the international Cross-Domain Conference for Machine Learning & Knowledge Extraction (CD-MAKE), https://cd-make.net, supported by the International Federation of Information Processing (IFIP). Thanks to all international colleagues who support the idea of *"Science is to test crazy ideas—Engineering is to put these ideas into Business"*. Machine Learning & Knowledge Extraction is such an enormous broad and rapidly growing field that there is much room for a new journal, which is in no way in conflict of interest to any existing journal, instead it shall be complementary to them—for the benefit of our international research community—ultimately for the benefit of the human.

Conflicts of Interest: The author declares that there are no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript:

aML	Automatic Machine Learning
ANN	Artificial Neural Network
CNN	Convolutional Neural Network
DL	Deep Learning
EWC	Elastic Weight Consolitation
FMA	Foundational Model of Anatomy
FB	Facebook
GO	Gene Ontology
GP	Gaussian Processes
HCI–KDD	Human–Computer Interaction & Knowledge Discovery from Data
iML	Interactive Machine Learning

ICD	International Classification of Diseases
ML	Machine Learning
MAKE	MAchine Learning & Knowledge Discovery
MCMC	Markov Chain Monte Carlo
MDPI	Multidisciplinary Digital Publishing Institute
MTL	Multi Task Learning
NP	Nondeterministic Polynomial time
NELL	Never Ending Language Learning
NLU	Natural Language Understanding
OMIM	Online Mendelian Inheritance in Man
PAC	Probable Approximate Correct
PDF	Probability Density Function
RNN	Recurrent Neural Network; alternatively: Recursive Neural Network
SNAP	Stanford Network Analysis Platform
SVM	Support Vector Machine
SLT	Statistical Learning Theory
SNOMED	Systematized Nomenclature of Medicine
TSP	Traveling Salesman Problem
UMLS	Unified Medical Language System

1. Meijer, E. Making money using math. *Commun. ACM* **2017**, *60*, 36–42, doi:10.1145/3052935.
2. Pearl, J. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*; Morgan Kaufmann: San Francisco, CA, USA, 1988.
3. Schmidhuber, J. Deep learning in neural networks: An overview. *Neural Netw.* **2015**, *61*, 85–117, doi:10.1016/j.neunet.2014.09.003.
4. LeCun, Y.; Bengio, Y.; Hinton, G. Deep learning. *Nature* **2015**, *521*, 436–444, doi:10.1038/nature14539.
5. Vapnik, V.N.; Chervonenkis, A.Y. On the Uniform Convergence of Relative Frequencies of Events to Their Probabilities. *Theory Probab. Appl.* **1971**, *16*, 264–280, doi:10.1137/1116025.
6. Bousquet, O.; Boucheron, S.; Lugosi, G. Introduction to Statistical Learning Theory. In *Advanced Lectures on Machine Learning*; Bousquet, O., von Luxburg, U., Raetsch, G., Eds.; Springer: Berlin/Heidelberg, Germany, 2004; pp. 169–207, doi:10.1007/978-3-540-28650-9_8.
7. Hastie, T.; Tibshirani, R.; Friedman, J. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*, 2nd ed.; Springer: New York, NY, USA, 2009, doi:10.1007/978-0-387-84858-7.
8. Bayes, T. An Essay towards solving a Problem in the Doctrine of Chances (communicated by Richard Price). *Philos. Trans.* **1763**, *53*, 370–418, doi:10.1098/rstl.1763.0053.
9. Laplace, P.S. Mémoire sur les probabilités. *Mémoires de l'Académie Royale des sciences de Paris* **1781**, *1778*, 227–332. (In French)
10. Kolmogorov, A. Interpolation und extrapolation von stationaeren zufaelligen Folgen. *Izv. Akad. Nauk SSSR Ser. Mat.* **1941**, *5*, 3–14. (In German)
11. Rasmussen, C.E.; Williams, C.K. *Gaussian Processes for Machine Learning*; MIT Press: Cambridge, MA, USA, 2006.
12. Knill, D.C.; Pouget, A. The Bayesian brain: The role of uncertainty in neural coding and computation. *Trends Neurosci.* **2004**, *27*, 712–719, doi:10.1016/j.tins.2004.10.007.
13. Chater, N.; Tenenbaum, J.B.; Yuille, A. Probabilistic models of cognition: Conceptual foundations. *Trends Cogn. Sci.* **2006**, *10*, 287–291, doi:10.1016/j.tics.2006.05.007.
14. Doya, K.; Ishii, S.; Pouget, A.; Rao, R. *Bayesian Brain: Probabilistic Approaches to Neural Coding*; MIT Press: Boston, MA, USA, 2007; doi:10.7551/mitpress/9780262042383.001.0001.
15. Wood, F.; van de Meent, J.-W.; Mansinghka, V. A new approach to probabilistic programming inference. In Proceedings of the 17th International Conference on Artificial Intelligence and Statistics (AISTATS) 2014, Reykjavik, Iceland, 22–25 April 2014; pp. 1024–1032.
16. Salvatier, J.; Wiecki, T.V.; Fonnesbeck, C. Probabilistic programming in Python using PyMC3. *PeerJ Comput. Sci.* **2016**, *2*, e55, doi:10.7717/peerj-cs.55.

17. Gordon, A.D.; Henzinger, T.A.; Nori, A.V.; Rajamani, S.K. Probabilistic programming. In Proceedings of the on Future of Software Engineering, Hyderabad, India, 31 May–7 June 2014; pp. 167–181, doi:10.1145/2593882.2593900.
18. Samuel, A.L. Some studies in machine learning using the game of checkers. *IBM J. Res. Dev.* **1959**, *3*, 210–229, doi:10.1147/rd.33.0210.
19. Tenenbaum, J.B.; Kemp, C.; Griffiths, T.L.; Goodman, N.D. How to grow a mind: Statistics, structure, and abstraction. *Science* **2011**, *331*, 1279–1285, doi:10.1126/science.1192788.
20. Bell, G.; Hey, T.; Szalay, A. Beyond the Data Deluge. *Science* **2009**, *323*, 1297–1298, doi:10.1126/science.1170411.
21. Jordan, M.I.; Mitchell, T.M. Machine learning: Trends, perspectives, and prospects. *Science* **2015**, *349*, 255–260, doi:10.1126/science.aaa8415.
22. Hinton, G.E.; Osindero, S.; Teh, Y.W. A fast learning algorithm for deep belief nets. *Neural Comput.* **2006**, *18*, 1527–1554, doi:10.1162/neco.2006.18.7.1527.
23. Henke, N.; Bughin, J.; Chui, M.; Manyika, J.; Saleh, T.; Wiseman, B.; Sethupathy, G. *The Age of Analytics: Competing in a Data-Driven World*; Mckinsey Company: New York, NY, USA, 2016.
24. Holzinger, A.; Dehmer, M.; Jurisica, I. Knowledge Discovery and interactive Data Mining in Bioinformatics—State-of-the-Art, future challenges and research directions. *BMC Bioinform.* **2014**, *15* (Suppl. 6), I1, doi:10.1186/1471-2105-15-S6-I1.
25. Lee, S.; Holzinger, A. Knowledge Discovery from Complex High Dimensional Data. In *Solving Large Scale Learning Tasks. Challenges and Algorithms, Lecture Notes in Artificial Intelligence, LNAI 9580*; Michaelis, S., Piatkowski, N., Stolpe, M., Eds.; Springer: Cham, Switzerland, 2016; pp. 148–167, doi:10.1007/978-3-319-41706-6_7.
26. Simovici, D.A.; Djeraba, C. *Mathematical Tools for Data Mining*; Springer: London, UK, 2014.
27. Jaynes, E.T. *Probability Theory: The Logic of Science*; Cambridge University Press: Cambridge, UK, 2003.
28. Ghahramani, Z. Bayesian non-parametrics and the probabilistic approach to modelling. *Philos. Trans. R. Soc. A Math. Phys. Eng. Sci.* **2013**, *371*, 1–20, doi:10.1098/rsta.2011.0553.
29. Ghahramani, Z. Probabilistic machine learning and artificial intelligence. *Nature* **2015**, *521*, 452–459, doi:10.1038/nature14541.
30. Bengio, Y.; Courville, A.; Vincent, P. Representation learning: A review and new perspectives. *IEEE Trans. Pattern Anal. Mach. Intell.* **2013**, *35*, 1798–1828, doi:10.1109/TPAMI.2013.50.
31. Neumann, M.; Huang, S.; Marthaler, D.E.; Kersting, K. pyGPs: A Python library for Gaussian process regression and classification. *J. Mach. Learn. Res.* **2015**, *16*, 2611–2616.
32. Domingos, P. The Role of Occam's Razor in Knowledge Discovery. *Data Min. Knowl. Discov.* **1999**, *3*, 409–425, doi:10.1023/a:1009868929893.
33. Wilson, A.G.; Dann, C.; Lucas, C.G.; Xing, E.P. The Human Kernel. *arXiv* **2015**, arXiv:1510.07389.
34. Shahriari, B.; Swersky, K.; Wang, Z.; Adams, R.P.; de Freitas, N. Taking the human out of the loop: A review of Bayesian optimization. *Proc. IEEE* **2016**, *104*, 148–175, doi:10.1109/JPROC.2015.2494218.
35. Sonnenburg, S.; Rätsch, G.; Schaefer, C.; Schoelkopf, B. Large scale multiple kernel learning. *J. Mach. Learn. Res.* **2006**, *7*, 1531–1565.
36. Holzinger, A. *Biomedical Informatics: Computational Sciences Meets Life Sciences*; BoD: Norderstedt, Germany, 2012; p. 368.
37. Hofmann, T.; Schoelkopf, B.; Smola, A.J. Kernel methods in machine learning. *Ann. Stat.* **2008**, *36*, 1171–1220, doi:10.1214/009053607000000677.
38. Griffiths, T.L.; Lucas, C.; Williams, J.; Kalish, M.L. Modeling human function learning with Gaussian processes. In *Advances in Neural Information Processing Systems (NIPS 2008)*; Koller, D., Schuurmans, D., Bengio, Y., Bottou, L., Eds.; NIPS: San Diego, CA, USA, 2009; Volume 21, pp. 553–560.
39. Holzinger, A. Interactive Machine Learning for Health Informatics: When do we need the human-in-the-loop? *Brain Inform.* **2016**, *3*, 119–131, doi:10.1007/s40708-016-0042-6.
40. Holzinger, A.; Plass, M.; Holzinger, K.; Crisan, G.C.; Pintea, C.M.; Palade, V. Towards interactive Machine Learning (iML): Applying Ant Colony Algorithms to solve the Traveling Salesman Problem with the Human-in-the-Loop approach. In *Springer Lecture Notes in Computer Science LNCS 9817*; Springer: Berlin/Heidelberg, Germany, 2016; pp. 81–95, doi:10.1007/978-3-319-45507-5_6.

41. Crescenzi, P.; Goldman, D.; Papadimitriou, C.; Piccolboni, A.; Yannakakis, M. On the complexity of protein folding. *J. Comput. Biol.* **1998**, *5*, 423–465, doi:10.1016/S0092-8240(05)80170-3.
42. Macgregor, J.N.; Ormerod, T. Human performance on the traveling salesman problem. *Percept. Psychophys.* **1996**, *58*, 527–539, doi:10.3758/bf03213088.
43. Napolitano, F.; Raiconi, G.; Tagliaferri, R.; Ciaramella, A.; Staiano, A.; Miele, G. Clustering and visualization approaches for human cell cycle gene expression data analysis. *Int. J. Approx. Reason.* **2008**, *47*, 70–84, doi:10.1016/j.ijar.2007.03.013.
44. Amato, R.; Ciaramella, A.; Deniskina, N.; Del Mondo, C.; di Bernardo, D.; Donalek, C.; Longo, G.; Mangano, G.; Miele, G.; Raiconi, G.; et al. A multi-step approach to time series analysis and gene expression clustering. *Bioinformatics* **2006**, *22*, 589–596, doi:10.1093/bioinformatics/btk026.
45. Shyu, C.R.; Brodley, C.E.; Kak, A.C.; Kosaka, A.; Aisen, A.M.; Broderick, L.S. ASSERT: A Physician-in-the-Loop Content-Based Retrieval System for HRCT Image Databases. *Comput. Vis. Image Underst.* **1999**, *75*, 111–132, doi:10.1006/cviu.1999.0768.
46. Schirner, G.; Erdogmus, D.; Chowdhury, K.; Padir, T. The future of human-in-the-loop cyber-physical systems. *Computer* **2013**, *46*, 36–45, doi:10.1109/MC.2013.31.
47. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. Imagenet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems (NIPS 2012)*; Pereira, F., Burges, C.J., Bottou, L., Weinberger, K.Q., Eds.; NIPS: San Diego, CA, USA, 2012; pp. 1097–1105.
48. Mikolov, T.; Deoras, A.; Povey, D.; Burget, L.; Cernocky, J. Strategies for training large scale neural network language models. In Proceedings of the IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU 2011), Waikoloa, HI, USA, 11–15 December 2011; pp. 196–201, doi:10.1109/ASRU.2011.6163930.
49. Helmstaedter, M.; Briggman, K.L.; Turaga, S.C.; Jain, V.; Seung, H.S.; Denk, W. Connectomic reconstruction of the inner plexiform layer in the mouse retina. *Nature* **2013**, *500*, 168–174, doi:10.1038/nature12346.
50. Leung, M.K.; Xiong, H.Y.; Lee, L.J.; Frey, B.J. Deep learning of the tissue-regulated splicing code. *Bioinformatics* **2014**, *30*, i121–i129, doi:10.1093/bioinformatics/btu277.
51. Bar, Y.; Diamant, I.; Wolf, L.; Greenspan, H. Deep learning with non-medical training used for chest pathology identification. In Proceedings of the Medical Imaging 2015: Computer-Aided Diagnosis, Orlando, FL, USA, 21–26 February 2015; doi:10.1117/12.2083124.
52. Tajbakhsh, N.; Shin, J.Y.; Gurudu, S.R.; Hurst, R.T.; Kendall, C.B.; Gotway, M.B.; Liang, J. Convolutional Neural Networks for Medical Image Analysis: Full Training or Fine Tuning? *IEEE Trans. Med. Imaging* **2016**, *35*, 1299–1312, doi:10.1109/TMI.2016.2535302.
53. Havaei, M.; Guizard, N.; Larochelle, H.; Jodoin, P.M. Deep learning trends for focal brain pathology segmentation in MRI. In *Machine Learning for Health Informatics*; Holzinger, A., Ed.; Springer: Cham, Switzerland, 2016; pp. 125–148, doi:10.1007/978-3-319-50478-0_6.
54. Carrasquilla, J.; Melko, R.G. Machine learning phases of matter. *Nat. Phys.* **2017**, *13*, 431–434, doi:10.1038/nphys4035.
55. Mnih, V.; Kavukcuoglu, K.; Silver, D.; Rusu, A.A.; Veness, J.; Bellemare, M.G.; Graves, A.; Riedmiller, M.; Fidjeland, A.K.; Ostrovski, G.; et al. Human-level control through deep reinforcement learning. *Nature* **2015**, *518*, 529–533, doi:10.1038/nature14236.
56. Silver, D.; Huang, A.; Maddison, C.J.; Guez, A.; Sifre, L.; van den Driessche, G.; Schrittwieser, J.; Antonoglou, I.; Panneershelvam, V.; Lanctot, M.; Dieleman, S.; Grewe, D.; Nham, J.; Kalchbrenner, N.; Sutskever, I.; Lillicrap, T.; Leach, M.; Kavukcuoglu, K.; Graepel, T.; Hassabis, D. Mastering the game of Go with deep neural networks and tree search. *Nature* **2016**, *529*, 484–489, doi:10.1038/nature16961.
57. Gigerenzer, G.; Gaissmaier, W. Heuristic Decision Making. *Annu. Rev. Psychol.* **2011**, *62*, 451–482, doi:10.1146/annurev-psych-120709-145346.
58. Marewski, J.N.; Gigerenzer, G. Heuristic decision making in medicine. *Dialogues Clin. Neurosci.* **2012**, *14*, 77–89.
59. Pearl, J. *Causality: Models, Reasoning, and Inference*, 2nd ed.; Cambridge University Press: Cambridge, UK, **2009**, doi:10.1017/CBO9780511803161.
60. Wang, H.; Yeung, D.-Y. Bayesian deep learning: A framework and some existing methods. *IEEE Trans. Knowl. Data Eng.* **2016**, *28*, 3395–3408, doi:10.1109/TKDE.2016.2606428.

61. Gal, Y.; Ghahramani, Z. Dropout as a Bayesian Approximation: Representing Model Uncertainty in Deep Learning. In Proceedings of the 33rd International Conference on Machine Learning (ICML), New York, NY, USA, 19–24 June 2016, pp. 1050–1059.
62. McCloskey, M.; Cohen, N.J. Catastrophic interference in connectionist networks: The sequential learning problem. In *The Psychology of Learning and Motivation*; Bower G.H., Ed.; Academic Press: San Diego, CA, USA, 1989; Volume 24, pp. 109–165, doi:10.1016/S0079-7421(08)60536-8.
63. Goodfellow, I.J.; Mirza, M.; Xiao, D.; Courville, A.; Bengio, Y. An empirical investigation of catastrophic forgeting in gradient-based neural networks. *arXiv* **2015**, arXiv:1312.6211.
64. Lee, J.; Kim, H.; Lee, J.; Yoon, S. Intrinsic Geometric Information Transfer Learning on Multiple Graph-Structured Datasets. *arXiv* **2016**, arXiv:1611.04687.
65. Henaff, M.; Bruna, J.; LeCun, Y. Deep convolutional networks on graph-structured data. *arXiv* **2015**, arXiv:1506.05163.
66. Tsymbal, A.; Zillner, S.; Huber, M. Ontology—Supported Machine Learning and Decision Support in Biomedicine. In *Data Integration in the Life Sciences*; Cohen-Boulakia, S., Tannen, V., Eds.; Springer: Berlin/Heidelberg, Germany, 2007; Volume 4544, pp. 156–171, doi:10.1007/978-3-540-73255-6_14.
67. Holzinger, A.; Jurisica, I. Knowledge Discovery and Data Mining in Biomedical Informatics: The future is in Integrative, Interactive Machine Learning Solutions. In *Lecture Notes in Computer Science LNCS 8401*; Holzinger, A., Jurisica, I., Eds.; Springer: Burlin/Heidelberg, Germany, 2014; pp. 1–18, doi:10.1007/978-3-662-43968-5_1.
68. Balcan, N.; Blum, A.; Mansour, Y. Exploiting Ontology Structures and Unlabeled Data for Learning. In Proceedings of the 30th International Conference on Machine Learning, Atlanta, GA, USA, 16–21 June 2013; pp. 1112–1120.
69. Carlson, A.; Betteridge, J.; Kisiel, B.; Settles, B.; Hruschka, E.R., Jr.; Mitchell, T.M. Toward an Architecture for Never-Ending Language Learning (NELL). In Proceedings of the Twenty-Fourth AAAI Conference on Artificial Intelligence (AAAI-10), Atlanta, GA, USA, 11–15 July 2010.
70. Guyon, I.; Elisseeff, A. An introduction to variable and feature selection. *J. Mach. Learn. Res.* **2003**, *3*, 1157–1182, doi:10.1162/153244303322753616.
71. Manning, C.D.; Schuetze, H. *Foundations of Statistical Natural Language Processing*; MIT Press: Cambridge, MA, USA, 1999.
72. Frank, M.C.; Goodman, N.D.; Tenenbaum, J.B. Using speakers' referential intentions to model early cross-situational word learning. *Psychol. Sci.* **2009**, *20*, 578–585, doi:10.1111/j.1467-9280.2009.02335.x.
73. Goodman, N.D.; Frank, M.C. Pragmatic language interpretation as probabilistic inference. *Trends Cogn. Sci.* **2016**, *20*, 818–829, doi:10.1016/j.tics.2016.08.005.
74. Rong, X. Word2vec parameter learning explained. *arXiv* **2014**, arXiv:1411.2738.
75. Goldberg, Y.; Levy, O. word2vec Explained: deriving Mikolov et al.'s negative-sampling word-embedding method. *arXiv* **2014**, arXiv:1402.3722.
76. Wallach, H.M. Topic modeling: Beyond bag-of-words. In Proceedings of the 23rd International Conference on Machine Learning (ICML), Pittsburgh, PA, USA, 25–29 June 2006; pp. 977–984.
77. Weinberger, K.; Dasgupta, A.; Langford, J.; Smola, A.; Attenberg, J. Feature hashing for large scale multitask learning. In Proceedings of the 26th Annual International Conference on Machine Learning (ICML 2009), Montreal, QC, Canada, 14–18 June 2009; Bottou, L., Littman, M., Eds.; ACM: New York, NY, USA, 2009; pp. 1113–1120.
78. Guyon, I.; Boser, B.; Vapnik, V. Automatic capacity tuning of very large VC-dimension classifiers. In Proceedings of the 7th Advances in Neural Information Processing Systems Conference (NIPS 1993), Denver, CO, USA; Cowan, J., Tesauro, G., Alspector, J., Eds.; NIPS: San Diego, CA, USA, 1993; Volume 7, pp. 147–155.
79. Leskovec, J.; Chakrabarti, D.; Kleinberg, J.; Faloutsos, C.; Ghahramani, Z. Kronecker graphs: An approach to modeling networks. *J. Mach. Learn. Res.* **2010**, *11*, 985–1042, doi:10.1145/1756006.1756039.
80. Leskovec, J.; Sosic, R. SNAP: A general-purpose network analysis and graph-mining library. *ACM Trans. Intell. Syst. Technol.* **2016**, *8*, 1–20, doi:10.1145/2898361.
81. Wood, F.; Meent, J.W.; Mansinghka, V. A new approach to probabilistic programming inference. In Proceedings of the 17th International Conference on Artificial Intelligence and Statistics (AISTATS) 2014, Reykjavik, Iceland, 22–25 April 2014; pp. 1024–1032.

82. Malle, B.; Kieseberg, P.; Weippl, E.; Holzinger, A. The right to be forgotten: Towards Machine Learning on perturbed knowledge bases. In *Springer Lecture Notes in Computer Science LNCS 9817*; Springer: Heidelberg/Berlin, Germany; New York, NY, USA, 2016; pp. 251–266, doi:10.1007/978-3-319-45507-5_17.
83. Goedertier, S.; Martens, D.; Vanthienen, J.; Baesens, B. Robust process discovery with artificial negative events. *J. Mach. Learn. Res.* **2009**, *10*, 1305–1340, doi:10.1145/1577069.1577113.
84. Doucet, A.; De Freitas, N.; Gordon, N. An introduction to sequential Monte Carlo methods. In *Sequential Monte Carlo Methods in Practice*; Springer: Heidelberg/Berlin, Germany, 2001; pp. 3–14, doi:10.1007/978-1-4757-3437-9_1.
85. Konecný, J.; McMahan, H.B.; Ramage, D.; Richtárik, P. Federated Optimization: Distributed Machine Learning for On-Device Intelligence. *arXiv* **2016**, arXiv:1610.02527v1.
86. Bonawitz, K.; Ivanov, V.; Kreuter, B.; Marcedone, A.; McMahan, H.B.; Patel, S.; Ramage, D.; Segal, A.; Seth, K. Practical Secure Aggregation for Federated Learning on User-Held Data. *arXiv* **2016**, arXiv:1611.04482.
87. McMahan, B.; Moore, E.; Ramage, D.; Hampson, S.; y Arcas, B.A. Communication-efficient learning of deep networks from decentralized data. In Proceedings of the 20th International Conference on Artificial Intelligence and Statistics, Fort Lauderdale, FL, USA, 20–22 April 2017; Volume 54, pp. 1273–1282.
88. Leskovec, J.; Singh, A.; Kleinberg, J. Patterns of influence in a recommendation network. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*; Springer: Heidelberg/Berlin, Germany, 2006; pp. 380–389, doi:10.1007/11731139_44.
89. Valiant, L.G. A theory of the learnable. *Commun. ACM* **1984**, *27*, 1134–1142, doi:10.1145/1968.1972.
90. Baxter, J. A model of inductive bias learning. *J. Artif. Intell. Res.* **2000**, *12*, 149–198, doi:10.1613/jair.731.
91. Evgeniou, T.; Pontil, M. Regularized multi-task learning. In Proceedings of the Tenth ACM SIGKDD International Conference On Knowledge Discovery and Data Mining, Seattle, WA, USA, 22–25 August 2004; pp. 109–117, doi:10.1145/1014052.1014067.
92. Hearst, M.A.; Dumais, S.T.; Osuna, E.; Platt, J.; Schoelkopf, B. Support vector machines. *IEEE Intell. Syst. Appl.* **1998**, *13*, 18–28.
93. Weinberger, K.Q.; Saul, L.K. Distance metric learning for large margin nearest neighbor classification. *J. Mach. Learn. Res.* **2009**, *10*, 207–244.
94. Parameswaran, S.; Weinberger, K.Q. Large margin multi-task metric learning. In *Advances in Neural Information Processing Systems 23 (NIPS 2010)*; Lafferty, J., Williams, C., Shawe-Taylor, J., Zemel, R., Culotta, A., Eds.; NIPS: San Diego, CA, USA, 2010; pp. 1867–1875.
95. Kirkpatrick, J.; Pascanu, R.; Rabinowitz, N.; Veness, J.; Desjardins, G.; Rusu, A.A.; Milan, K.; Quan, J.; Ramalho, T.; Grabska-Barwinska, A.; Hassabis, D.; Clopath, C.; Kumaran, D.; Hadsell, R. Overcoming catastrophic forgetting in neural networks. *arXiv* **2016**, arXiv:1612.00796.
96. Pan, S.J.; Yang, Q.A. A Survey on Transfer Learning. *IEEE Trans. Knowl. Data Eng.* **2010**, *22*, 1345–1359, doi:10.1109/tkde.2009.191.
97. Taylor, M.E.; Stone, P. Transfer learning for reinforcement learning domains: A survey. *J. Mach. Learn. Res.* **2009**, *10*, 1633–1685.
98. Sycara, K.P. Multiagent systems. *AI Mag.* **1998**, *19*, 79.
99. Lynch, N.A. *Distributed Algorithms*; Morgan Kaufmann: San Francisco, CA, USA, 1996.
100. DeGroot, M.H. Reaching a consensus. *J. Am. Stat. Assoc.* **1974**, *69*, 118–121.
101. Benediktsson, J.A.; Swain, P.H. Consensus theoretic classification methods. *IEEE Trans. Syst. Man Cybern.* **1992**, *22*, 688–704, doi:10.1109/21.156582.
102. Weller, S.C.; Mann, N.C. Assessing rater performance without a gold standard using consensus theory. *Med. Decis. Mak.* **1997**, *17*, 71–79, doi:10.1177/0272989X9701700108.
103. Olfati-Saber, R.; Fax, J.A.; Murray, R.M. Consensus and cooperation in networked multi-agent systems. *Proc. IEEE* **2007**, *95*, 215–233, doi:10.1109/jproc.2006.887293.
104. Roche, B.; Guegan, J.F.; Bousquet, F. Multi-agent systems in epidemiology: A first step for computational biology in the study of vector-borne disease transmission. *BMC Bioinform.* **2008**, *9*, 435, doi:10.1186/1471-2105-9-435.
105. Kok, J.R.; Vlassis, N. Collaborative multiagent reinforcement learning by payoff propagation. *J. Mach. Learn. Res.* **2006**, *7*, 1789–1828.

106. Robert, S.; Büttner, S.; Röcker, C.; Holzinger, A. Reasoning Under Uncertainty: Towards Collaborative Interactive Machine Learning. In *Machine Learning for Health Informatics: State-of-the-Art and Future Challenges*; Holzinger, A., Ed.; Springer: Cham, Switzerland, 2016; pp. 357–376, doi:10.1007/978-3-319-50478-0_18.
107. Holzinger, A. *Successful Management of Research and Development*; BoD: Norderstedt, Germany, 2011; p. 112.
108. Holzinger, A. Machine Learning for Health Informatics. In *Machine Learning for Health Informatics: State-of-the-Art and Future Challenges, Lecture Notes in Artificial Intelligence LNAI 9605*; Holzinger, A., Ed.; Springer: Cham, Switzerland, 2016; pp. 1–24, doi:10.1007/978-3-319-50478-0_1.
109. Theodoridis, S.; Slavakis, K.; Yamada, I. Adaptive Learning in a World of Projections. *IEEE Signal Process. Mag.* **2011**, *28*, 97–123, doi:10.1109/msp.2010.938752.
110. Hinton, G.; Deng, L.; Yu, D.; Dahl, G.E.; Mohamed, A.R.; Jaitly, N.; Senior, A.; Vanhoucke, V.; Nguyen, P.; Sainath, T.N.; et al. Deep Neural Networks for Acoustic Modeling in Speech Recognition. *IEEE Signal Process. Mag.* **2012**, *29*, 82–97, doi:10.1109/msp.2012.2205597.
111. Wu, W.; Nagarajan, S.; Chen, Z. Bayesian Machine Learning. *IEEE Signal Process. Mag.* **2016**, *33*, 14–36, doi:10.1109/msp.2015.2481559.
112. Russell, S.; Dietterich, T.; Horvitz, E.; Selman, B.; Rossi, F.; Hassabis, D.; Legg, S.; Suleyman, M.; George, D.; Phoenix, S. Letter to the Editor: Research Priorities for Robust and Beneficial Artificial Intelligence: An Open Letter. Avaliable online: https://www.bibsonomy.org/bibtex/2185f9c84cb3aef91e7bb82eeb4728ce0/dblp (accessed on 25 June 2017).
113. Anderson, M.; Anderson, S.L. Machine ethics: Creating an ethical intelligent agent. *AI Mag.* **2007**, *28*, 15–25.
114. Boella, G.; van der Torre, L.; Verhagen, H. Introduction to the special issue on normative multiagent systems. *Auton. Agents Multi-Agent Syst.* **2008**, *17*, 1–10, doi:10.1007/s10458-008-9047-8.
115. Cervantes, J.A.; Rodriguez, L.F.; Lopez, S.; Ramos, F.; Robles, F. Autonomous Agents and Ethical Decision-Making. *Cogn. Comput.* **2016**, *8*, 278–296, doi:10.1007/s12559-015-9362-8.
116. Deng, B. The Robot's dilemma. *Nature* **2015**, *523*, 24–26, doi:10.1038/523024a.
117. Thimbleby, H. Explaining code for publication. *Softw. Pract. Exp.* **2003**, *33*, 975–1001, doi:10.1002/spe.537.
118. Sonnenburg, S.; Braun, M.L.; Ong, C.S.; Bengio, S.; Bottou, L.; Holmes, G.; LeCun, Y.; Muller, K.R.; Pereira, F.; Rasmussen, C.E.; et al. The need for open source software in machine learning. *J. Mach. Learn. Res.* **2007**, *8*, 2443–2466.
119. Michalski, R.S.; Carbonell, J.G.; Mitchell, T.M. *Machine Learning: An Artificial Intelligence Approach*; Springer: Berlin/Heidelberg, Germany, 1983.
120. Holzinger, A. On Knowledge Discovery and Interactive Intelligent Visualization of Biomedical Data—Challenges in Human–Computer Interaction & Biomedical Informatics. In Proceedings of the DATA 2012, International Conference on Data Technologies and Applications, Rome, Italy, 25–27 July 2012; pp. 5–16.
121. Holzinger, A. Human—Computer Interaction and Knowledge Discovery (HCI-KDD): What is the benefit of bringing those two fields to work together? In *Multidisciplinary Research and Practice for Information Systems, Springer Lecture Notes in Computer Science LNCS 8127*; Cuzzocrea, A., Kittl, C., Simos, D.E., Weippl, E., Xu, L., Eds.; Springer: Heidelberg/Berlin, Germany; New York, NY, USA, 2013; pp. 319–328, doi:10.1007/978-3-642-40511-2_22.
122. Holzinger, A. Trends in Interactive Knowledge Discovery for Personalized Medicine: Cognitive Science meets Machine Learning. *IEEE Intell. Inform. Bull.* **2014**, *15*, 6–14.
123. Kandel, E.R.; Schwartz, J.H.; Jessell, T.M.; Siegelbaum, S.A.; Hudspeth, A. *Principles of Neural Science*, 5th ed.; McGraw-Hill: New York, NY, USA, 2012; p. 1760.