Stalzer and Mentzel *SpringerPlus (2016)5:1266*
DOI 10.1186/s40064-016-2888-8

SpringerPlus

REVIEW Open Access

A preliminary review of influential works in data-driven discovery

Mark Stalzer* and Chris Mentzel

*Correspondence:
stalzer@caltech.edu
Science Program, Gordon and Betty Moore Foundation, Palo Alto, CA 94304, USA

Abstract

The Gordon and Betty Moore Foundation ran an Investigator Competition as part of its Data-Driven Discovery Initiative in 2014. We received about 1100 applications and each applicant had the opportunity to list up to five *influential works* in the general field of "Big Data" for scientific discovery. We collected nearly 5000 references and 53 works were cited at least six times. This paper contains our preliminary findings.

Background

The long-term goal of the Gordon and Betty Moore Foundation's Data-Driven Discovery Initiative (DDD) is to foster and advance the people and practices of data-intensive science to take advantage of the increasing volume, velocity, and variety of scientific data to make new discoveries. Data-intensive science is inherently multidisciplinary, combining natural sciences with methods from statistics and computer science.

In January 2014 the DDD launched an Investigator Competition (IC) to identify some of the leading innovators in data-driven discovery. These scientists are striking out in new directions and are willing to take risks with the potential of huge payoffs in some aspect of data-intensive science. As part of the competition we collected several thousand references, which we call *influential works*, to the literature, software, and data sets that the applicants listed as one of the top five most important works in data-intensive science or data science.

This paper is a preliminary review of what we found. The next section presents the methodology and some statistics from the references. "Clusters of influential works" section contains several natural clusters of the works, some are obvious like genomics and machine learning. Others like the impact of Google's work, and questions about the scientific method are perhaps of more general interest. This paper ends with some limitations and next steps.

Influential works at a top level

In the competition pre-application stage we asked for up to five *influential works* in data-driven discovery. Specifically, as stated in the competition FAQ:

> *The (up to) five Influential Works on the pre-application web form are for you to reference work that you think has helped define the field of data science. This may or may not be your own work. Taken collectively, across all the DDD IC pre-applications, these works will give the foundation a snapshot of data intensive science.*

A total of 1095 applications were received in late February 2014, containing 4790 references.

The raw data is not available for public release since it was collected with the Foundation's promise of anonymity to get a better sampling. Specifically, from the competition FAQ:

> *Members of the DDD staff intend to write a review paper that summarizes these findings, and information will only be used in an aggregate form.*

Presented in this paper is an aggregate form, via an automated sorting process that is described in the "Appendix", for works cited at least six times. There are 53 of these works; and the ones cited at least ten times are in Table 1. This automatic approach works very well for papers and books, which have a well established citation form, but not so well for resources and tools and this will be discussed further in the limitations part of the concluding remarks.

A plot of the reference index for all works versus the reference count fitted to a power law is shown in Fig. 1. The correlation of about 0.99 is very good agreement. The h-index

Table 1 Works that were cited at least ten times, with count, year, and citation

Count	Year	Citation
63	2008	MapReduce (Dean and Ghemawat 2008)
51	2009	*Fourth paradigm* (Hey et al. 2009)
43	2009	*Elements of statistical learning* (Hastie et al. 2009)
30	2001	Initial sequencing of the human genome (Lander et al. 2001)
24	1948	A mathematical theory of communication (Shannon 2001)
23	2000	Sloan Digital Sky Survey (York et al. 2000)
20	1990	BLAST (Altschul et al. 1990)
19	1996	Lasso (Tibshirani 1996)
19	2003	Latent Dirichlet allocation (Blei et al. 2003)
17	1977	EM algorith (Dempster et al. 1977)
17	1995	Support vector networks (Cortes and Vapnik 1995)
15	2001	Random forests (Breiman 2001)
14	2006	*Pattern recognition* (Bishop et al. 2006)
14	1998	Anatomy of web search engine (Brin and Page 1998)
13	2007	*Numerical recipes* (Press 2007)
11	1979	Bootstrap methods (Efron 1979)
11	1953	Equation of state calculations (Metropolis et al. 1953)
11	1977	Exploratory data analysis (Tukey 1977)
11	1988	*Probabilistic reasoning* (Pearl 1988)
10	1999	PageRank (Page et al. 1999)
10	2013	*Bayesian data analysis* (Gelman et al. 2013)
10	2009	Unreasonable effectiveness of data (Halevy et al. 2009)

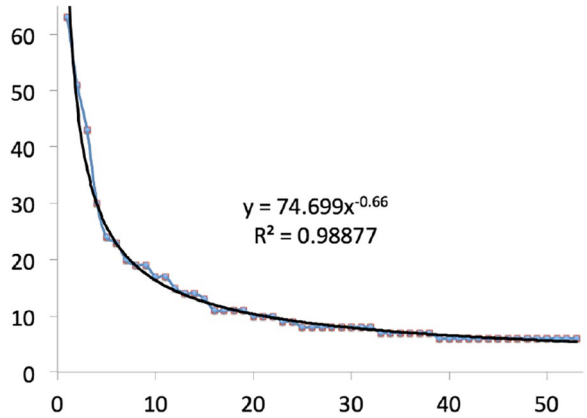

Fig. 1 Fit of the influential works to a power law (x is index, y is count). The correlation coefficient is $R^2 = 0.989$

of the works is 14; this is the subset of the works cited as least as often as their rank by the number of times cited (Hirsch 2005)⋆.[1]

The data set is 1.7 MB and is difficult to examine directly, but the sorting process was manually validated on some references that have rare words. For example, MapReduce (Dean and Ghemawat 2008) is reported here with 63 citations and a hand count shows 64, Latent Dirichlet allocation (Blei et al. 2003) is a perfect 19 for 19, and The Fourth Paradigm (Hey et al. 2009) is 51 for 58 and this mostly was due to sloppy citations. The counts reported here can be considered good lower bounds on the real counts.

Clusters of influential works

The works were manually organized into clusters by natural science domain, methodologies, tools, and the scientific method as shown in Table 2. Each cluster has some key topics as described below and all influential works are cited with varying levels of description.

Domain sciences

Astronomy

The Sloan Digital Sky Survey (SDSS) (York et al. 2000) is a widely cited resource (www.sdss.org).[2] The current release is SDSS-III DR12 that has observations through 14 July 2014 and contains 469,053,874 unique, primary, sources from several datasets. Generally, online astronomical datasets are being federated via interoperability standards created by organizations such as the International Virtual Observatory Alliance (www.ivoa.net). The result is a virtual telescope, and astronomers have been pioneers in making observations openly available and accessible.

New instruments are also showing that data-driven discovery is not just about the volume of data, but also the "velocity". One of the major challenges with the Large Synoptic

[1] References that provide background information, and not in the 53 influential works found as part of the competition, are denoted by a ⋆.

[2] SDSS was cited as both a resource and an associated technical summary paper. The intent was clear so we grouped all the citations together.

Table 2 A clustering of the 53 influential works with associated sections

Count	Cluster	Key topics
7	"Domain sciences"	Astronomy Genomics
29	"Methodologies"	Theory Statistical methods Machine learning
9	"General tools and applications"	Google General tools
8	"Centrality of the scientific method"	
53	ALL	

Survey Telescope (LSST), which should start doing science runs about 2020, is that the number of alerts to interesting objects may overwhelm the available follow up resources. Good object classification ("Machine learning" section) and prioritization will be crucial to the science output.

Genomics

It is very clear that genomics and the Human Genome Project (HGP) have been the main driver of data driven discovery in the life sciences. The two primary works are the "Initial sequencing and analysis of the Human genome" (Lander et al. 2001) and the related paper by Venter et al. (2001). These papers report the sequencing of the approximately 3 billion nucleotides that make up the human genome. The project was considered essentially complete in April 2003 and according to the NIH's HGP factsheet, it has enabled the discovery of over 1800 disease related genes and many other applications. An example is the Thousand Genomes Project (2012) which, as of 2012, had completed a variety of sequences from 1092 individuals from 14 populations. This allows comparative analysis of the sequences, which is at the core of bioinformatics-based discovery.

Consider two sequences a, b composed from the alphabet $\{A, C, G, T\}$—DNA nucleotides. We want to find the optimal alignments, essentially a string matching problem, of a, b. In general, however, the alignments are not perfect string matches due to missing data and other factors. Instead, a distance metric is defined and the alignments are optimized with respect to that metric. For example, under a certain metric two good alignments of `GACTAC` are `-ACG-C` and `-AC-GC`. This can be done optimally using dynamic programming in time $O(|a|\ |b|)$. However, if a must be aligned with many b taken from a database search, the computational expense is prohibitive. A key bioinformatics tool is the "Basic Local Alignment Search Tool" (Altschul et al. 1990) (BLAST). BLAST uses heuristics to reduce the time complexity and make large-scale searches practical.

There are many other applications besides human health. For example, population groupings can be inferred using Bayesian clustering methods from *multiloci* genotype information (Pritchard et al. 2000). This is an early form of Latent Dirichlet allocation (LDA) which is described more fully in the next section. It can be thought of as running LDA on genetic data, rather than on text: it clusters *individuals* into *population* rather

than documents into topics.[3] Another emerging example is the use of bioinformatics methods in ecology (Jones et al. 2006). A major challenge here is the heterogeneous natures of the data, from individuals to the biosphere, and their interactions.

The Protein Data Bank (Berman et al. 2000) was established at Brookhaven National Lab in 1971 as an archive for structural genomics data: essentially the shapes of biologically active molecules. These shapes and other information is determined experimentally by X-ray diffraction, NMR, and sometimes theoretical modeling. These experiments require special facilities and can be costly, so there was clearly a motivation in the community to build an archive to minimize duplication of effort. In 2000 there were 10,714 structures and this has grown to 106,710 by early 2015. The data bank supports sophisticated query mechanisms to assist researchers in finding structures with certain properties, such as atomic locations.

It is interesting that the two most referenced natural science domains are astronomy and genomics, and they can differ in length scales of phenomena by up to 33 orders of magnitude. The fact that humanity can probe over such a large range, and even further with high-energy physics experiments, is simply amazing.

Methodologies

Foundational theory

Reverend Bayes' essay on the Doctrine of Chances in 1763 (Bayes 1763) is the earliest commonly cited paper and it is truly foundational for data science (a popular modern text is Gelman et al. 2013). The work introduces "Bayes Law" which gives the likelihood of a condition A being present given that condition B is present, denoted as the conditional probably $P(A \mid B)$, as

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)} \tag{1}$$

where $P(A)$ and $P(B)$ are the so called prior probabilities, or the frequencies of occurrence of the conditions. Please note that the wording is careful to not confuse coincidence with causality: the "law" is just a statement of an existing closed population. This equation is optimal under a crucial assumption and this can be seen since it is the unique generalization (up to an integration constant) of *modus ponens* for probabilistic inference (Jaynes 2003). The crucial assumption is that the priors are known very well. There are extensions to Eq. 1 known as maximal-entropy methods that are based on ideas from statistical mechanics, i.e. how much information can be contained in all the possible ensembles of states in a closed system; again Jaynes is a good reference (Jaynes 2003).[4]

Shannon's seminal work on how much information can be transmitted over a communications channel is also based on entropic ideas (Shannon 2001). Recently, Donoho (2006) wrote on "Compressed sensing", with an application to image analysis, but the development is a more general result in information theory. Let x be an unknown vector of size $|m|$ and that we plan to make n measurements of x in a variety of ways. It is

[3] The method can be used back in time since DNA can be preserved; population studies have been done on Darwin's finches from the Galápagos in 1835 using specimens from British museums (Petren et al. 2010)*.

[4] It should be noted this may be a data anomaly as one of the authors cited this work on his homepage. He also cited *Sports Illustrated* which may explain random references to sports statistics.

shown that only $n = O(m^{1/4} \log^{5/2}(m))$ measurements are needed for a bounded error. This is a very interesting result because it shows that with clever measurement, we do not need to collect nearly as much data *if* there is an underlying sparse representation of what is being measured (another way to look at this is that there can be a lot of redundancy in representations). As will be seen in "Machine learning" section, some forms of compression can be automatically learned.

The Metropolis Algorithm (Metropolis et al. 1953) gives a way for sampling large spaces for computing high-dimensional integrals with a bounded convergence rate. *Probabilistic reasoning in intelligent systems: Networks of plausible inference* by Pearl (1988) also covers Bayesian inference, Bayesian and Markov networks, and more advanced topics of interest to the artificial intelligence community. We suspect that the use of automated reasoning techniques will grow in data science, although there are issues of scalability. Pearl has also written extensively on coincidence and causality.

Classical statistical methods

Any section on classical statistical methods must begin with linear models of data, such as fitting a line to a set of points using an ordinary least squares (OLS) estimate. The lasso (Tibshirani 1996), for "least absolute shrinkage and selection operator," can improve on the prediction accuracy of OLS *and* also helps with interpretation since it identifies key coefficients in the estimate.

Consider a sample of size N, we can certainly compute basic statistics such as the average. With *bootstrap* methods (Efron 1979), the sample is re-sampled multiple times with replacement to generate better statistics, and this is useful with complicated distributions. Extensions to the original 1979 approach use Bayesian methods (Rubin 1981)*. This is further developed in "A decision-theoretic generalization of on-line learning and an application to boosting" (Freund and Schapire 1995). It is an example of combining multiple *strategies*, even if they are individually weak, to build robust models. The authors use many example, including betting on horses.

Incomplete data is a very common problem and it can be formalized as follows. Let $\mathbf{x} \in \mathcal{X}$ be the *complete data* and $\mathbf{y} \in \mathcal{Y}$ be the (possibly incomplete) *observed data*, and assume there is a mapping such that $\mathcal{X}(\mathbf{y})$ gives all possible $\mathbf{x}$ for an observation $\mathbf{y}$. Given a set of parameters Φ, the family of complete sampling distributions $f(\mathbf{x}|\Phi)$ is related to the incomplete family $g(\mathbf{y}|\Phi)$ by

$$g(\mathbf{y}|\Phi) = \int_{\mathcal{X}(\mathbf{y})} f(\mathbf{x}|\Phi) \tag{2}$$

Dempster, Laird, and Rubin present a method for computing maximum likelihood estimates from incomplete data called the *EM* Algorithms (for Expectation-Maximization) (Dempster et al. 1977); it does this by adjusting the parameters to maximize g given the observations. The paper has many examples including missing value situations, truncated data, etc. It was read before the Royal Statistical Society and there is extensive commentary in its published form. One comment in particular, by R. J. A. Little, is a fine summary: "Other advantages of the EM approach are (a) because it is stupid, it is safe, (b) it is easy to program, and often allows simple adaptation of complete data methods, and (c) it provides fitted values for missing data." An application of the EM algorithm

and Bayesian statistics is "Latent Dirichlet allocation" (Blei et al. 2003) that build a multi-level model for "collections of discrete data such as text corpora."[5]

Consider a set of features used to classify objects. A "random forest" (Breiman 2001) is a collection of decision trees where each tree uses some subset of the features to do a classification; the trees then vote to determine the final class. It is shown that forests are not subject to overtraining, which can be a problem with machine learning methods (see next section).

The Elements of Statistical Learning (Hastie et al. 2009, Chapters 3, 10, 8, 15) by Hastie, Tibshirani, and Friedman covers lasso, bootstrap methods, the EM algorithm, and random forests. It is also has chapters on machine learning which is covered in the next section; it is a popular text. An earlier text (Breiman et al. 1984), also covers regression and tree methods.

When there are multiple hypotheses a standard approach is to control the familywise error rate (FWER)—closely related to Type I errors. This is a common problem in determining the efficacy of medical procedures. Benjamini and Hochberg suggest (Benjamini and Hochberg 1995), instead, to control the number of falsely rejected hypotheses—the false discovery rate (FDR). FDR can be more powerful when some (null) hypotheses are non-true.

Isomap (Tenenbaum et al. 2000) is an algorithm for reducing the dimensionality of input spaces, e.g. face recognition. It is broadly applicable whenever non-linear geometry complicates the use of techniques such as Principal Component Analysis (PCA). Another paper on non-linear reduction (Roweis and Saul 2000) presents a local, piecewise, linear method for modeling non-linear data. An interesting example is that using PCA on a logarithmic spiral, to first order, just yields a linear fit; yet the curve can be parameterized by its length and maintain its structure.

Machine learning

Methods for machine learning are crucial for data-driven discovery and are used for both classification and regression analysis. There are several standards texts (Mitchell 1997; Duda et al. 1999; Bishop et al. 2006; Murphy 2012). Here we will focus on two common methods and some recent advances.

Consider the classification problem $f : X \mapsto \{-1, 1\}$ where X is an observation space and f decides if a member of X belongs to one of two categories. For example, in an astronomical image, find all of the quasars with a redshift greater than some value. Machine learning methods take a set of example observations from X and use some *generalization* process to build an f.

One of the most rigorously-founded ways is to form a "Support Vector Machine" (Vapnik 1998; Cortes and Vapnik 1995) (SVM). The construction of an SVM attempts to build a hyperplane that divides the examples into the -1 or $+1$ spaces. In general, the examples are not completely separable and so a kernel $K(x, x')$ is used to project an element x of X into a higher dimensional space where the separation is more complete. It is useful to look at this in more detail, since it clearly shows data, mathematical formulations, and clever algorithms coming together to form an f.

A common kernel is $K(x, x') = e^{-(x-x')^2/2\sigma^2}$ where σ is determined from the data. The selection of a kernel generally requires some insight, particularly when the data is

[5] It would be interesting to apply LDA to the 53 influential works.

heterogeneous. Consider l training examples $(x_1, y_1) \ldots (x_l, y_l)$ where the $y_i \in \{-1, +1\}$. To construct an SVM, solve the following optimization problem for α:

$$\sum_{i=1}^{l} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{l} y_i y_j \alpha_i \alpha_j K(x_i, x_j) \tag{3}$$

subject to $\sum_{i=1}^{l} y_i \alpha_i = 0$ and all $\alpha_i \geq 0$. This can be done via quadratic programming which is generally $\mathcal{NP}$-hard, but due to some constraints in the formulation the optimization can be done quickly using Sequential Minimal Optimization (SMO) (Platt 1999)*. The decision function is then $f(x) = \text{sgn}(b + \sum_{i=1}^{l} y_i \alpha_i K(x, x_i))$ where b is the scalar category separator and can be computed directly given the α_i.

Recently SVMs, called SVM+, have been extended to work with an auxiliary "privileged information" set $X^\star$ that is *available only during classifier construction* (Vapnik and Vashist 2009)*. An example is to use a protein structure prediction code, during training, to help train a classifier. An SVM+ classifier typically performs better than regular SVM. Constructing an SVM+ can also be done fairly quickly using SMO (Pechyony and Vapnik 2012)*. There is an interesting analogy to Shannon's work that is based on the information available in a closed system. With SVM+, the classifier gets trained with access to another system, Vapnik calls it a teacher ($X + X^\star$), and then works independently (X) in operation.

Another common classification method are artificial Neural Networks (NN), and the basic ideas go back to 1943 (McCulloch and Pitts 1943)*. Here the input vector is fed into sigmoid nodes that make a choice in some shade of gray $[-1, 1]$ and the outputs move onto the next network layers. A purely feed-forward network, where there are no backward arcs, can be trained efficiently using back-propagation (Rumelhart et al. 2002) where classification errors are used to adjust the network weights backwards layer by layer.

In large NNs, such as those used in image processing, there can be a failure to generalize due to over fitting of the very large number of weights. One approach is to use a middle "coding" layer that is relatively small that forces the network to learn the key generalizations (Hinton and Salakhutdinov 2006). Recently, the so-called "dropout" algorithm has been developed that trains only subsets of the network on each example and this helps generalization too (Srivastava et al. 2014)*.

Closely related to NNs are logistic belief networks, where the nodes switch from 0 to 1 as a function of the probability of the weighted inputs. Hinton et al. (2006) present a particular form of a multilayer belief network where the initial layers are feed-forward and the final two layers are interconnected in such a way to form an associative memory. An efficient training algorithm is developed that trains the individual layers using a greedy algorithm, and then refines the weights for the whole network. For a standard handwriting recognition benchmark (the MNIST database of handwritten digits) the error rate was 1.25 % which was better than that obtained by other standard machine learning techniques (SVM was second best at 1.4 %). However, if you train a standard NN using slight perturbations of the training data, i.e. moving pixels around a bit, error rates as low as 0.4 % have been reported as of 2006. Table 1 of the reference shows some nice comparative data on methods and error rates.

Krizhevsky, Sutskever, and Hinton present their results from the ImageNet LSVRC-2010 and LSVRC-2012 contests (Krizhevsky et al. 2012).[6] The goal was to classify images

[6] The existence of standard data sets and contests has been very important in the development of machine learning algorithms.

into categories, and the training data set has roughly 1000 images in each of 1000 categories for a total of about a million images. The authors trained a convolutional neural network having 60 million parameters using several optimizations to make the problem tractable (the input layers of CNNs are not fully connected, they "focus" on overlapping zones of the visual field much like biological systems). The resulting network, for LSVRC-2012, had an error rate of 15.3 % compared to the second-best entry's rate of 26.2 %.

There is substantial anecdotal evidence that NNs and SVMs are the most powerful classifiers if trained properly, and that is why their use is so widespread. Hastie et al. (2009, Chapters 12 and 11) contains chapters on SVMs and NNs. The classic text of Duda et al. on pattern classification (Duda et al. 1999) also covers NNs, genetic algorithms, and many other machine learning algorithms.

Finally, hidden Markov models (Rabiner 1989) are transition networks where each transition is labeled with a probability of happening. They are common in natural language processing, but can also be applied to problems such as representing various biological (e.g. regulatory) networks.

General tools and applications

The section describes some general tools and applications that appeared in the works due to their wide applicability. It opens with Google, which was somewhat surprising to the authors, but the company clearly has an impact on the thinking of data scientists. The section closes with several general tools, such as R and IPython.

Google

PageRank (Page et al. 1999) is an algorithm for ranking pages in web searches and was the first used by Google. It is an important example of applied computer science, where two good intuitions are combined in a mathematically rigorous way to produce an algorithm of high utility. The first intuition is that the importance of a page is proportional to the number of pages that link to it. Ultimately, the sum of the importance for all pages is one. The second, and more mathematically interesting is that there is a damping factor which is denoted d. The idea is that a person will only wander so far (click) from a search result before getting bored and moving on to something else. In practice, $d \approx 0.85$ (Brin and Page 1998), and this $0 < d < 1$ helps to give rapid convergence.

Consider N web pages where the PageRank of page i is denoted r_i and define $R^T = \{r_1, r_2, \ldots, r_N\}$. Further define the matrix $M_{ij} = \delta_{ij}/L_j$, where L_j is the number of outbound links from page j and $\delta_{ij} = 1$ if pages i, j are linked, otherwise it is zero. With the identity matrix I, R is given in the steady state by

$$R = (I - dM)^{-1} \frac{1-d}{N} I \tag{4}$$

In practice, the solution is computed iteratively and converges quickly.

Conceptually, MapReduce (Dean and Ghemawat 2008) transforms an input set X of key:value pairs with keys in K_1 to an output set Y of pairs with keys in K_2 using a three stage Map–Shuffle–Reduce process. The Map step applies a function to every element of X producing an intermediate list X containing new pairs with keys in K_2. This X is then Shuffled

to group the values corresponding to a given key in X together so that they can then be Reduced using another function into the output Y. In the canonical example of counting the number of times a distinct word appears in a set of files, the elements of K_1 are filenames and K_2 contains words, the associated values are file contents and word counts.

If general, if X and the post-shuffled X are distributed across many nodes, the map and reduce stages can be done in parallel on local data. Production implementations have many optimizations to deal with issues like load balancing, data positioning and replication, minimizing communications, and fault tolerance. PageRank can be formulated in a way that yields an efficient MapReduce implementation. In the context of data-intensive discovery, it is very common to combine MapReduce with machine learning and classification ("Machine learning" section) to parallelize the processes.

The fact that a commercial enterprise is making such an impact on science is wonderful! However, we must add a note of caution: "Big Data" is not just the massive application of machine learning methods with large, blunderbuss, clusters; it is more subtle and widespread ("Centrality of the scientific method" section). Hadoop, an open implementation of MapReduce, was also cited by some. It should be noted that Google has largely moved onto systems such as BigTable (Chang et al. 2008)* and Cloud Dataflow for storing and processing data (https://cloud.google.com/dataflow/).

General tools

A strong cluster of references emerged around tools, programming languages and methods for understanding data. These works represent a cross section of non-domain specific methods that researchers from a variety of disciplines are utilizing to process data to information to understanding.

Numerical Recipes (Press 2007) is the most widely used reference for numerical algorithms and it covers a broad range of topics from linear algebra to optimization. There have been multiple editions since 1986, and the most recent edition (2007) has been expanded to cover topics such as classification and inference. The series web site, www.nr.com, considers itself one of the oldest pages on the Web, and provides paid access to all algorithms in various programming languages.

The R language (R Development Core Team 2008) is one of the leading statistical programming languages, and was referenced a significant number of times in the dataset. R was created as a free and open source implementation of the S statistical programming language with influences from Scheme. R focuses on ease of use, tight integration with publication quality graphics and charts, data processing, and modular extensions to go beyond the core functionality. It has its own mathematical formula expression language, like LaTeX, and provides users convenient tools converting formulas into executable code.

The IPython Notebook project (Perez and Granger 2007) (now Jupyter at www.jupyter.org) is noteworthy as one of a few open source software toolkits for both programming and data analysis that is not a database, algorithm or programming language. Jupyter is an "architecture for interactive computing and computational narratives in any programming language." It provides both a programming and documentation environment which ultimately allows for sharing of so-called narratives in an executable notebook, all available via the web. It is language agnostic; processing R, Python, Julia and provides basic workflow/reproducibility and collaboration capabilities. It is being used in a wide variety of scientific applications.

The Visual Display of Quantitative Information (Tufte 2001) by Tufte is a seminal work on data visualization, with a focus that uses very powerful human perceptive systems that are not likely to be automated soon. The famous chart, of course, is Napoleon's foolish march and then retreat from Russia. The authors feel that, perhaps, all talks should be speeches and perhaps simply summarized in a few charts. Tukey in a 1977 work (Tukey 1977) also emphasizes the use of graphs and tables to explore data.

Finally, Codd (1970) introduced relational databases in a brilliant Tour de Force of computer science, coupling theory with practice. No longer were databases to be ad-hoc, they have a theory that could be used to make them better. This is at the core of all relational database systems, and Codd won the A.M. Turing award in 1981 for his work.

An observation is the power of open source software. R, IPython, and Apache Hadoop which contains an implementation of MapReduce, are all available under various open source licenses. This allows the free use, inspection, and extension of the codes and greatly lowers barriers to entry, particularly for academic research purposes.

Centrality of the scientific method

One of the most cited influential works was *The Fourth Paradigm* (Hey et al. 2009), a collection of papers on data intensive scientific discovery produced by Microsoft in honor of Jim Gray, one of the first modern data scientists. The collection has had a catalytic effect based on the number of references, from researchers in a wide variety of fields. Another influential work is on the unreasonable effectiveness of data (Halevy et al. 2009), which is a nice play on the unreasonable effectiveness of mathematics. We must distinguish between *tools,* or instruments, and the scientific method. In the Fourth the argument is made that science has progressed from the 1. empirical stage (observation-only), to the 2. theory stage, and on to 3. simulation based science, and finally 4. big data science. It was at stage 2 that the scientific method became fully formed, and Newton deserves a lot of credit although Maxwell showed the raw power of theory to explain phenomena beyond human senses. The tools that Newton used were the calculus, which he had to invent, inclined planes, and dropping fruit. Now we use computers in stages 3 (theory) and 4 (observation). The scientific method stays the same, technology just allows better tools which begets deeper science and then new technology and tools.

There have also been claims that "Big Data" will eliminate science, we just need to use powerful methods to classify the data and from that we will know everything. The trouble is confusing classification, like botany, with science: predictive theories with bounded errors. Let us consider training a classifier to near Bayesian optimal. It could be a NN or a SVM, but the advantage with an SVM is that we can extract out the key support vectors, the prime x_j, and examine them. *Does this tell us anything?* The trouble is that if the experiment is changed, the support vectors will likely change too so where is the insight? Another take on this is by Breiman, in "Statistical modeling: The two cultures" (Breiman et al. 2001), where he contrasts what is called classical statistical methods in this paper with algorithmic models. The comments associated with the paper are enlightening.

As a concrete example, it may be within current computing and algorithmic technology to infer the Maxwell Equations directly from data given knowledge of vector calculus. This would be a formidable achievement. Indeed, the kinematic laws of the double pendulum problem can be inferred using symbolic regression from observations

(Schmidt and Lipson 2009). Latent in the Equations, however, is special relativity but it requires a mental shift to tease this out: specifically, Einstein's axiom that the speed of light is constant in all inertial reference frames. Making this brilliant leap seems hard to do by computing at this time. Perhaps we need a new Turing test, one not susceptible to linguistic parlor tricks: *given just the data and some fundamental theorems from analysis, discover special relativity and general relativity.*

A recent paper (2013) by Dhar, "Data science and prediction" (Dhar 2013), defines *data science* as

> *...the study of generalizable extraction of knowledge from data.*
> *A common epistemic requirement in assessing whether new knowledge is actionable for decision making is its predictive power, not just its ability to explain the past.*

This view is entirely consistent with the scientific method, however it does not mean that the *way* scientists do science is fixed. Indeed, in the delightful book *Reinventing discovery* Nielsen (2012) argues that network effects in scientific communications and access to data will dramatically accelerate scientific discovery. This prediction is almost certainly true.

Finally, there were a few general references such as (Han and Kamber 2011) and the National Academy of Sciences report on the *Frontiers of Massive Data Analysis* (National Research Council 2013).

Conclusions

Limitations It must be noted that the competition was for efforts in the natural sciences and methodologies, and therefor references important to social sciences are underrepresented in this sample. Indeed, the social sciences are potentially one of the most impactful areas for big data and we encourage funders in these fields to run an investigator competition in this broad area.

As mentioned in the introduction, we asked applicants to tag works as papers, books, or resources. The matching algorithm works very well for papers and books, but not so well for data resources and software tools. The fundamental problem is that there is no commonly accepted way of citing resources unless there is an associated paper (e.g. IPython) or the authors are very specific about how to cite the tool (e.g. R). We are sure that if we went through the nearly 5000 citations by hand, we would find more resources but we decided to stay with our deterministic, repeatable, methodology. Efforts to attach Digital Object Identifiers (DOIs) to resources are underway; however we believe one reason that articles and books are easier to reference is that they also have a standard, *human understandable*, way to identify themselves and not just some cryptic number.

Next steps The concepts behind "Big Data" are not new, and go back to at least 1609 with Kepler's *Astronomia Nova* (Kepler 1609)*. The great, early, data scientist reduced Tycho Brahe's voluminous observational data into just three laws, the most famous probably being that bodies move in ellipses about a mass center.[7]

[7] It may also be interesting to note that the publication of *Nova* was delayed by about 4 years, from 1605 to 1609, due to an intellectual property argument surrounding Mr. Brahe's data.

Our longer term plan is to perform further study of the influential works. The BibTeX file has been released as supplemental information, and the authors hope that a primary value of this work is in education.

Authors' contributions

CM and MS ran the Foundation's Investigator Competition and collected the data. MS reduced the data using the method described in the "Appendix". MS and CM organized the paper; with MS taking the lead writing role and CM providing parts and editing. Both authors read and approved the final manuscript.

Acknowledgements

The authors thank the 1095 applicants to the DDD Investigator Competition, the advisory panel, commentators on the arXiv paper, and the members of the Moore Foundation Science Program and its (former) Chief Program Officer, Vicki Chandler. We would also like to thank Joshua Greenberg of the Alfred P. Sloan Foundation for useful discussions and initial motivation for this paper. M. Stalzer thanks the Aspen Center for Physics and the NSF Grant #1066293 for hospitality during the editing of this paper

Competing interests

The authors declare that they have no competing interests.

Appendix

A total of 1095 applications were received in late February 2014, containing 4790 references. The author, title, etc. of each reference was broken into a bag of words and these bags were assigned to *buckets* based on reference similarity using weighted word frequency by a sorting process. Specifically, the weight of a word i that occurred N_i times is

$$\ln N_w/N_i \tag{5}$$

where N_w is the total number of unique words. In other words, words of lesser frequency carry somewhat more weight, leading to higher matching value. An obvious example is "paradigm". Words must be of length four or greater and appear twice or more; this eliminates stop words, e.g. "and, the", in English and words with no matching value, although it does throw out a bit of information.

References were sorted into the buckets based on the bucket's *signature.* A signature keeps the top eight words in a bucket by Eq. 5, although when buckets are merged in the sorting process (see below) all words in both buckets are used to recompute the new merged signature so that signatures are refined over time.

The sorting algorithm is straightforward. Begin by assigning each reference to its own bucket and compute its signature. Take the first bucket and find a bucket whose signature matches to within a threshold; if there is a match, merge the two buckets. Repeat with the second bucket, and so on. The threshold is manually adjusted to produce strong groupings, with few extraneous references in each bucket. If the threshold is too high, nothing groups, and if it is too low, everything groups into one bucket. Some manual edits were done to clean up the buckets. Papers, books, and resources were treated separately (this is done by giving the type tags high weights).

Four or five words of each signature were then submitted to Google Scholar to get BibTeX entries. Google Scholar almost always listed the right work first, although the quality of the BibTeX entries is highly variable and often needed to be fixed.

A note on the references

Each reference contains a note on the number of times it was cited ("n 63"), and the number of applicants that self-identified in a field, such as computer science, that cited the reference ("CS 41"). Table 3 is the key to fields.

Table 3 Key to reference tags and fields

Tag	Field
ACM	Applied and computational mathematics
AG	Agriculture
APHYS	Applied physics
ASPC	Aerospace
ASTRO	Astronomy and astrophysics
ASTROB	Astrobiology
ATMOS	Atmospheric science
BCS	Brain and cognitive science
BIO	Biology
BIOE	Bioengineering
BIOI	Bioinformatics
CBIO	Computational biology
CE	Computer engineering
CHEM	Chemistry
CHEME	Chemical engineering
CIVE	Civil engineering
CLI	Climate science
CS	Computer science
CSS	Computational social science
CSYS	Complex systems
DM	Data mining
EBIO	Evolutionary biology
ECO	Ecology
EE	Electrical engineering
ENGR	Engineering (general)
EPS	Earth and planetary science
ESE	Environmental science and engineering
EST	Energy science and technology
GENE	Genetics
GENOM	Genomics
GEOP	Geophysics
MATH	Mathematics
MATS	Materials science
MBIO	Biochemistry and molecular biophysics
ME	Mechanical engineering and solid mechanics
MED	Medicine
MMO	Marine microbiology and oceanography
NEURO	Neuroscience
OPSR	Operations research
PHYS	Physics
REMS	Remote sensing
SBIO	Systems biology
SML	Statistics and machine learning

Received: 28 January 2016 Accepted: 20 July 2016
Published online: 05 August 2016

References

Altschul SF, Gish W, Miller W, Myers EW, Lipman DJ (1990) Basic local alignment search tool. J Mol Biol 215(3):403–410 (n 20 BIOI 15 SML 11 GENE 10 CS 9 BIO 9 MBIO 6 MMO 3 ESE 2 CHEME 2 ACM 2 SBIO 1 OPSR 1 EBIO 1 CHEM 1 BCS 1)

Bayes M (1763) A letter from the late Reverend Mr. Thomas Bayes, F. R. S. to John Canton, M. A. and F. R. S. Philos Trans (1683–1775) 53(1763):269–271 (n 8 SML 5 BIOI 4 BIO 4 CS 3 GENE 2 ACM 2 MBIO 1 MATS 1 GEOP 1 ESE 1 BCS 1 ASTRO 1 APHYS 1)

Benjamini Y, Hochberg Y (1995) Controlling the false discovery rate: a practical and powerful approach to multiple testing. J R Stat Soc Ser B (Methodol) 57:289–300 (n 7 SML 7 BIOI 6 GENE 3 BIO 3 CS 2 ACM 2 MBIO 1 MATH 1 ESE 1 EE 1)

Berman HM, Westbrook J, Feng Z, Gilliland G, Bhat T, Weissig H, Shindyalov IN, Bourne PE (2000) The protein data bank. Nucleic Acids Res 28(1):235–242 (n 6 MBIO 4 CS 4 BIOI 4 ACM 3 SML 2 GENE 1 CHEM 1 BIO 1 BCS 1)

Bishop CM et al (2006) Pattern recognition and machine learning. Springer, New York (n 14 CS 12 SML 11 BIOI 5 ACM 4 BCS 3 MATH 2 EE 2 SBIO 1 MBIO 1 MATS 1 GEOP 1 EST 1 BIO 1)

Blei DM, Ng AY, Jordan MI (2003) Latent Dirichlet allocation. J Mach Learn Res 3:993–1022 (n 19 SML 13 CS 13 BIOI 4 GENE 3 EE 3 BCS 2 ASTRO 2 ACM 2 MBIO 1 MATH 1 ESE 1 ENGR 1 CSS 1 CE 1 BIOE 1)

Breiman L, Friedman J, Stone CJ, Olshen RA (1984) Classification and regression trees. CRC Press, Boca Raton (n 7 SML 5 BIOI 5 CS 3 PHYS 2 BIO 2 BCS 2 MMO 1 MBIO 1 MATH 1 ESE 1 EE 1 ECO 1 CHEM 1 ACM 1)

Breiman L (2001) Random forests. Mach Learn 45(1):5–32 (n 15 SML 10 CS 6 ESE 4 BIOI 4 ACM 3 MATH 2 BIO 2 ASTRO 2 PHYS 1 OPSR 1 GEOP 1 EE 1 DM 1 CLI 1 BCS 1 APHYS 1)

Breiman L et al (2001) Statistical modeling: the two cultures (with comments and a rejoinder by the author). Stat Sci 16(3):199–231 (n 6 SML 3 ESE 2 CS 2 BIOI 2 ENGR 1 EE 1 BIO 1 ACM 1)

Brin S, Page L (1998) The anatomy of a large-scale hypertextual web search engine. Comput Netw ISDN Syst 30(1):107–117 (n 14 CS 9 SML 8 BIOI 6 MATH 5 BIO 4 ACM 4 GENE 3 BIOE 2 BCS 2 PHYS 1 OPSR 1 MED 1 ME 1 MBIO 1 ENGR 1 CIVE 1 CHEM 1 ASTRO 1 ASPC 1 APHYS 1)

Chang F, Dean J, Ghemawat S, Hsieh WC, Wallach DA, Burrows M, Chandra T, Fikes A, Gruber RE (2008) BigTable: a distributed storage system for structured data. ACM Trans Comput Syst 26(2):4

Codd EF (1970) A relational model of data for large shared data banks. Commun ACM 13(6):377–387 (n 8 CS 5 SML 4 BIOI 3 BIO 3 PHYS 2 MATH 2 GENE 2 ASTRO 2 MBIO 1 ESE 1 BCS 1 APHYS 1 ACM 1)

Cortes C, Vapnik V (1995) Support-vector networks. Mach Learn 20(3):273–297 (n 17 CS 11 SML 9 BIOI 7 ENGR 3 EE 3 BIOE 3 BCS 3 ASTRO 3 ACM 3 BIO 2 PHYS 1 ME 1 MBIO 1 GENE 1 EST 1 ESE 1 CLI 1 CHEM 1 CE 1 APHYS 1)

Dean J, Ghemawat S (2008) MapReduce: simplified data processing on large clusters. Commun ACM 51(1):107–113 (n 63 CS 41 SML 32 BIOI 21 ACM 12 PHYS 8 ESE 7 CE 7 BIO 7 ASTRO 7 MBIO 5 MATH 5 GEOP 5 GENE 5 EE 5 BIOE 5 APHYS 5 MMO 4 ENGR 4 CIVE 3 BCS 3 ME 2 EST 2 CHEME 2 MED 1 MATS 1 CSS 1 ASPC 1)

Dempster AP, Laird NM, Rubin DB (1977) Maximum likelihood from incomplete data via the EM algorithm. J R Stat Soc Ser B (Methodol) 1–38 (n 17 SML 13 BIOI 8 CS 7 MATH 3 GENE 3 ACM 3 PHYS 2 ENGR 2 EE 2 MBIO 1 CHEME 1 CHEM 1 BIOE 1 BIO 1 BCS 1)

Dhar V (2013) Data science and prediction. Commun ACM 56(12):64–73 (n 6 SML 4 CS 3 BIOI 2 ME 1 MATS 1 GENE 1 ESE 1 CE 1 BIO 1 ASTRO 1 ACM 1)

Donoho DL (2006) Compressed sensing. IEEE Trans Inf Theory 52(4):1289–1306 (n 7 SML 5 BIOI 3 MATH 2 EE 2 ME 1 MBIO 1 GEOP 1 ENGR 1 CS 1 BIO 1 BCS 1 ACM 1)

Duda RO, Hart PE, Stork DG (1999) Pattern classification. Wiley, London (n 8 CS 8 SML 6 BIOI 3 MATH 2 ESE 2 BCS 2 SBIO 1 MBIO 1 GEOP 1 GENE 1 BIOE 1 BIO 1 ACM 1)

Efron B (1979) Bootstrap methods: another look at the jackknife. Ann Stat 7(1):1–26 (n 11 SML 10 BIOI 7 GENE 3 MBIO 2 CS 2 ACM 2 NEURO 1 MATH 1 ENGR 1 EE 1 DM 1 BIOE 1 BIO 1 ASTRO 1)

Freund Y, Schapire RE (1995) A decision-theoretic generalization of on-line learning and an application to boosting. In: Vitányi P (ed) Computational Learning Theory. Lecture Notes in Computer Science, vol 904. Springer, Berlin pp 23–37 (n 6 SML 5 CS 3 MATH 2 BIOI 2 GENE 1 CLI 1 CE 1 BIO 1 ACM 1)

Gelman A, Carlin JB, Stern HS, Dunson DB, Vehtari A, Rubin DB (2013) Bayesian data analysis. CRC Press, Boca Raton (n 10 SML 6 ASTRO 4 ESE 2 MMO 1 MBIO 1 GENE 1 CS 1 CLI 1 BIOI 1 BIO 1 ACM 1)

Halevy A, Norvig P, Pereira F (2009) The unreasonable effectiveness of data. Intell Syst 24(2):8–12 (n 10 CS 7 SML 5 PHYS 2 ESE 2 BIOI 2 BCS 2 ASTRO 2 ACM 2 EE 1 CE 1)

Han J, Kamber M (2011) Data mining: concepts and techniques, 3rd edn. Morgan Kaufmann, Burlington (n 6 SML 4 CS 3 ESE 2 PHYS 1 OPSR 1 MMO 1 MATS 1 MATH 1 GEOP 1 BIOI 1 BIO 1)

Hastie T, Tibshirani R, Friedman J (2009) The elements of statistical learning. Springer, Berlin (n 43 SML 33 BIOI 19 CS 15 ACM 10 BIO 8 MBIO 5 GENE 5 MATH 4 EE 3 ESE 2 ASTRO 2 PHYS 1 MATS 1 ENGR 1 CHEM 1 BCS 1)

Hey AJ, Tansley S, Tolle KM et al (2009) The fourth paradigm: data-intensive scientific discovery. Microsoft Research, Redmond (n 51 CS 27 SML 16 BIOI 16 BIO 14 ESE 13 ASTRO 8 CE 7 PHYS 6 MATS 5 MMO 4 MBIO 3 MATH 3 GENE 3 ENGR 3 EE 3 CHEM 3 ECO 2 CIVE 2 ACM 2 SBIO 1 REMS 1 ME 1 GEOP 1 EST 1 CSS 1 CHEME 1 BCS 1 ASTROB 1 ASPC 1 APHYS 1)

Hinton G, Osindero S, Teh Y-W (2006) A fast learning algorithm for deep belief nets. Neural Comput 18(7):1527–1554 (n 6 SML 4 CS 3 BIOI 3 BCS 3 APHYS 2 PHYS 1 NEURO 1 MMO 1 MED 1 MBIO 1 MATH 1 EE 1 CHEM 1 BIO 1 ACM 1)

Hinton GE, Salakhutdinov RR (2006) Reducing the dimensionality of data with neural networks. Science 313(5786):504–507 (n 7 SML 4 BCS 4 EE 3 CS 3 BIOI 3 ACM 3 BIO 2 PHYS 1 MATH 1 CHEM 1 CE 1 BIOE 1 ASTRO 1 ASPC 1)

Hirsch JE (2005) An index to quantify an individual's scientific research output. Proc Natl Acad Sci USA 102(46):16569–16572

Jaynes ET (2003) Probability theory: the logic of science. Cambridge University Press, Cambridge (n 8 SML 4 CS 4 ACM 4 GENE 3 BIOI 3 BIO 3 PHYS 2 MBIO 2 MATH 2 EE 1 CHEM 1 BIOE 1 BCS 1 ASTRO 1)
Jones MB, Schildhauer MP, Reichman O, Bowers S (2006) The new bioinformatics: integrating ecological data from the gene to the biosphere. Annu Rev Ecol Evol Syst 37:519–544 (n 8 BIO 5 ESE 4 SML 2 BIOI 2 ME 1 MATS 1 CS 1 CE 1 ACM 1)
Kepler J (1609) Astronomia nova E-lib.ch 10.3931/e-rara-558
Krizhevsky A, Sutskever I, Hinton GE (2012) ImageNet classification with deep convolutional neural networks. In: Pereira F, Burges CJC, Bottou L, Weinberger KQ (eds) Advances in neural information processing systems. Curran Associates, Inc., pp 1097–1105 (n 6 SML 3 BCS 3 CS 2 ENGR 1 EE 1 CE 1)
Lander ES, Linton LM, Birren B, Nusbaum C, Zody MC, Baldwin J, Devon K, Dewar K, Doyle M, FitzHugh W et al (2001) Initial sequencing and analysis of the human genome. Nature 409(6822):860–921 (n 30 BIOI 18 BIO 12 SML 11 CS 10 GENE 8 MATH 4 PHYS 3 MBIO 3 MATS 3 ESE 3 ACM 3 ME 2 GEOP 2 EE 2 SBIO 1 MMO 1 ENGR 1 CIVE 1 CHEM 1 CE 1 BIOE 1 ASTRO 1 ASPC 1 APHYS 1)
McCulloch WS, Pitts W (1943) A logical calculus of the ideas immanent in nervous activity. Bull Math Biophys 5(4):115–133
Metropolis N, Rosenbluth AW, Rosenbluth MN, Teller AH, Teller E (1953) Equation of state calculations by fast computing machines. J Chem Phys 21(6):1087–1092 (n 11 SML 5 ACM 5 CS 4 BIOI 3 MBIO 2 EE 2 BIOE 2 PHYS 1 MATS 1 MATH 1 ENGR 1 CHEM 1 CBIO 1 BIO 1 ASTRO 1 APHYS 1)
Mitchell TM (1997) Machine learning, vol 45. McGraw Hill, New York (n 6 CS 5 SML 3 MMO 1 MBIO 1 EE 1 BIOI 1 BIO 1 BCS 1)
Murphy KP (2012) Machine learning: a probabilistic perspective. MIT Press, Cambridge n 6 SML 6 CS 3 ESE 2 NEURO 1 BIO 1 BCS 1 ACM 1
National Research Council (2013) Frontiers in massive data analysis. The National Academies Press, Keck (n 9 SML 6 BIOI 4 ESE 2 EE 2 CS 2 PHYS 1 MATH 1 GENE 1 ENGR 1 ASTRO 1)
Nielsen M (2012) Reinventing discovery: the new era of networked science. Princeton University Press, Princeton (n 8 CS 4 ESE 3 MMO 2 BIOI 2 BIO 2 SML 1 REMS 1 PHYS 1 EST 1 ENGR 1 EE 1 CE 1 BCS 1 ASPC 1)
Page L, Brin S, Motwani R, Winograd T (1999) The PageRank citation ranking: bringing order to the web. Stanford InfoLab, Technical report, no. 10 (CS 8 BIOI 4 SML 2 BIOE 2 BCS 2 OPSR 1 MBIO 1 MATH 1 GEOP 1 EST 1 ENGR 1 EE 1 CSS 1 BIO 1 ACM 1)
Pearl J (1988) Probabilistic reasoning in intelligent systems: Networks of plausible inference. Morgan Kaufmann, Burlington (n 11 SML 8 CS 7 BIOI 5 BIO 3 MMO 1 MBIO 1 MATH 1 GENE 1 ESE 1 CBIO 1 BIOE 1 BCS 1)
Pechyony D, Vapnik V (2012) Fast optimization algorithms for solving SVM+. CRC Press, Boca Raton
Perez F, Granger BE (2007) IPython: a system for interactive scientific computing. Comput Sci Eng 9(3):21–29 (n 6 BIO 4 CS 3 BIOI 3 SML 2 GENE 2 ESE 2 PHYS 1 MMO 1 MATH 1 BIOE 1 BCS 1 ASTRO 1)
Petren K, Grant PR, Grant BR, Clack AA, Lescano NV (2010) Multilocus genotypes from Charles Darwin's finches: biodiversity lost since the voyage of the Beagle. Philos Trans R Soc B Biol Sci 365(1543):1009–1018
Platt J (1999) Fast training of support vector machines using sequential minimal optimization. MIT Press, Cambridge
Press WH (2007) Numerical recipes: the art of scientific computing. Cambridge University Press, Cambridge (n 13 ASTRO 9 CS 4 PHYS 3 ACM 3 SML 2 EE 2 MBIO 1 MATH 1 GENE 1 BIOI 1 BIOE 1 BIO 1 BCS 1 APHYS 1)
Pritchard JK, Stephens M, Donnelly P (2000) Inference of population structure using multilocus genotype data. Genetics 155(2):945–959 (n 6 GENE 6 BIOI 5 BIO 4 SML 3 ESE 1)
R Development Core Team (2008) R: a language and environment for statistical computing. R Foundation for Statistical Computing (n 6 SML 5 BIOI 4 BIO 3 GENE 2 CS 2 ACM 2 OPSR 1 MBIO 1) [Online]. http://www.R-project.org
Rabiner L (1989) A tutorial on hidden Markov models and selected applications in speech recognition. Proc IEEE 77(2):257–286 (n 7 SML 5 CS 4 ENGR 2 BIOI 2 ACM 2 PHYS 1 MMO 1 MATH 1 ESE 1 EE 1 BCS 1)
Roweis ST, Saul LK (2000) Nonlinear dimensionality reduction by locally linear embedding. Science 290(5500):2323–2326 (n 8 SML 7 CS 4 MATH 3 BIOI 3 ACM 3 BCS 2 REMS 1 PHYS 1 MMO 1 ESE 1 EE 1 CLI 1 CHEME 1 CE 1 CBIO 1 BIO 1 APHYS 1)
Rubin DB (1981) The Bayesian bootstrap. Ann Stat 9(1):130–134
Rumelhart DE, Hinton GE, Williams RJ (2002) Learning representations by back-propagating errors. In: Polk TA, Seifert CM (eds) Cognitive modeling. MIT Press, Cambridge, pp 213–220 (n 6 SML 5 CS 5 BIOI 3 GENE 2 EE 2 BIOE 2 ACM 2 MBIO 1 GEOP 1 ESE 1 ENGR 1 BIO 1 BCS 1 APHYS 1)
Schmidt M, Lipson H (2009) Distilling free-form natural laws from experimental data. Science 324(5923):81–85 (n 6 PHYS 3 SML 2 ENGR 2 ASTRO 2 APHYS 2 MMO 1 MATS 1 ESE 1 EE 1 CS 1 BIOI 1)
Shannon CE (2001) A mathematical theory of communication. Reprinted in ACM SIGMOBILE Mob Comput Commun Rev 5(1):3–55 (n 24 CS 15 SML 12 BIO 11 BIOI 10 ACM 8 MBIO 6 EE 6 PHYS 4 MATH 3 GENE 3 BIOE 3 MMO 2 ESE 2 ENGR 2 CHEME 2 CHEM 2 CE 2 ASTRO 2 APHYS 2 OPSR 1 GEOP 1 EST 1 DM 1 BCS 1 ASPC 1)
Srivastava N, Hinton G, Krizhevsky A, Sutskever I, Salakhutdinov R (2014) Dropout: a simple way to prevent neural networks from overfitting. J Mach Learn Res 15:1929–1958
Tenenbaum JB, De Silva V, Langford JC (2000) A global geometric framework for nonlinear dimensionality reduction. Science 290(5500):2319–2323 (n 6 ACM 5 SML 4 BIOI 3 PHYS 2 MATH 2 CS 2 REMS 1 MMO 1 GENE 1 ESE 1 ENGR 1 CLI 1 CHEME 1 BIO 1 BCS 1 ASTRO 1 APHYS 1)
The 1000 Genomes Project Consortium (2012) An integrated map of genetic variation from 1,092 human genomes. Nature 491(7422):56–65 (n 7 BIO 6 BIOI 5 GENE 4 SML 2 PHYS 1 MBIO 1 CS 1 CBIO 1 ACM 1)
Tibshirani R (1996) Regression shrinkage and selection via the lasso. J R Stat Soc Ser B (Methodol) 267–288 (n 19 SML 17 BIOI 6 CS 5 ACM 5 MATH 4 BCS 3 BIO 2 MBIO 1 ENGR 1 EE 1)
Tufte ER (2001) The visual display of quantitative information, 2nd edn. Graphics Press, Cheshire (n 9 CS 5 SML 4 ASTRO 4 PHYS 2 GENE 2 ESE 2 CE 2 BIOI 2 BIO 2 ACM 2 MMO 1 MBIO 1 BCS 1 APHYS 1)
Tukey JW (1977) Exploratory data analysis. Pearson, London (n 11 SML 7 CS 4 BIOI 4 ACM 4 GENE 3 BIO 3 ESE 2 PHYS 1 MMO 1 MBIO 1 CE 1 BCS 1)
Vapnik V (1998) Statistical learning theory, vol 2. Wiley, New York

Vapnik V, Vashist A (2009) A new learning paradigm: Learning using privileged information. Neural Netw 22(5):544–557
Venter JC, Adams MD, Myers EW, Li PW, Mural RJ, Sutton GG, Smith HO, Yandell M, Evans CA, Holt RA et al (2001) The sequence of the human genome. Science 291(5507):1304–1351 (n 8 BIOI 7 CS 5 SML 3 BIO 3 ACM 2 PHYS 1 MBIO 1 GENOM 1 GENE 1 CHEM 1 BCS 1)
York DG, Adelman J, Anderson JE Jr, Anderson SF, Annis J, Bahcall NA, Bakken J, Barkhouser R, Bastian S, Berman E et al (2000) The Sloan Digital Sky Survey: technical summary. Astron J 120(3):1579 (n 23 ASTRO 20 PHYS 7 SML 6 CS 6 EE 2 BIO 1 ACM 1)