Mathematical Notations for Course LV 185.A83 Machine Learning for Health Informatics

Andreas Holzinger
Vienna University of Technology, Austria
andreas.holzinger@univie.ac.at

First version April, 12, 2016; Current as of July, 6, 2016

Abstract

Machine learning (ML) is a very practical field. Consequently, the theoretical-mathematical content of the course 185.A83 "Machine Learning for Health Informatics" is kept to a minimum. It is hard to keep a consistent notation throughout the class to cover the extremely wide variety of data, models and algorithms discussed in this course. Definitions, conventions and usage of one and the same expression may be very different in mathematics and in computer science. This short document outlines some of the used mathematical notations. Note that one and the same symbol may have different meaning in different contexts.

1 Introduction and Motivation

Machine learning (ML) is the fastest growing technical field, at the intersection of informatics and statistics, tightly connected with data science and knowledge discovery [1], and health informatics (HI) is amongst the greatest challenges [2].

The goal of ML is to develop algorithms which can learn from data and improve over time and can be used for predictions. In automatic Machine learning (aML), great advances have been made, e.g., in speech recognition, recommender systems, or autonomous vehicles. Automatic approaches, e.g. deep learning, greatly benefit from big data with many training sets. In the health domain, sometimes we are confronted with a small number of data sets or rare events, where aML-approaches suffer of insufficient training samples. Here interactive Machine Learning (iML) may be of help, having its roots in Reinforcement Learning (RL), Preference Learning (PL) and Active Learning (AL). The term iML [3], [4] can be defined as algorithms that can interact with agents and can optimize their learning behaviour through these interactions, where the agents can also be human. This human-in-the-loop can be beneficial in solving computationally hard problems, e.g., subspace clustering [5], protein folding - which routed in the traveling salesman problem [6], or k-anonymization [7], where human expertise can help to reduce an exponential search space through heuristic selection of samples. Therefore, what would otherwise be an NP-hard problem reduces greatly in complexity through the input and the assistance of a human agent involved in the learning phase. However, although humans are excellent at pattern recognition in dimensions of 3; most biomedical data sets are in dimensions much higher than 3, making manual data analysis very hard. Successful application of machine learning in health informatics requires to consider the whole pipeline from data preprocessing to data visualization. Consequently, this course fosters the HCI-KDD approach, which encompasses a synergistic combination of methods from two areas to unravel such challenges: Human-Computer Interaction (HCI) and Knowledge Discovery/Data Mining (KDD), with the goal of supporting human intelligence with machine learning.

It should be mentioned that a central topic in HI is decision making, and decision making can be seen as a search in an high-dimensional problem space. However, biomedical data sets are full of uncertainty, incompleteness etc. [8], they can contain missing data, noisy data, dirty data, unwanted data, and most of all, some problems in the medical domain are hard, which makes the application of fully automated approaches difficult or even impossible, or at least the quality of results from automatic approaches might be questionable. Moreover, the complexity of sophisticated machine learning algorithms has detained non-experts from the application of such solutions. Consequently, the integration of the knowledge of a domain expert can sometimes be indispensable and the interaction of a domain expert with the data would greatly enhance the knowledge discovery process pipeline. Hence, *interactive* machine learning (iML) puts the "human-in-the-loop" to enable what neither a human nor a computer could do on their own. This idea is supported by a synergistic combination of methodologies of two areas that offer ideal conditions towards unraveling such problems: Human-Computer Interaction (HCI) and Knowledge Discovery/Data Mining (KDD), with the goal of supporting human intelligence with machine intelligence to discover novel, previously unknown insights into data (HCI-KDD approach [9]). 1) ML is extremely broad and deals with many problems. A central problem is the extraction of features from data to solve predictive tasks, including decision support, forecasting, ranking, classifying (e.g., in cancer diagnosis), detecting anomalies (e.g., virus mutations) or for sentiment analysis [10]. The grand challenge is to discover relevant *structural* patterns and/or *temporal* patterns ("knowledge") in such data, which are often hidden and not accessible to the human expert. Most of health data sets are weakly-structured and non-standardized,, and most data is in dimensions much higher than 3, and despite human experts are excellent in pattern recognition for dimensions ≤ 3, such data make manual analysis often impossible [11].

In this short document the student will find some notations of mathematical foundations, linear algebra, probability, specific ML notations, and graphical model notations, finally some recommendations to mathematical literature.

2 Mathematical Foundations

Symbol	**Meaning**
$\lfloor x \rfloor$	Floor of x, i.e., round down to nearest integer
$\lceil x \rceil$	Ceiling of x, i.e., round up to nearest integer
$\vec{x} \otimes \vec{y}$	Convolution of $\vec{x}$ and $\vec{y}$
$\vec{x} \odot \vec{y}$	Hadamard (elementwise) product of $\vec{x}$ and $\vec{y}$
$a \wedge b$	logical AND
$a \vee b$	logical OR
$\neg a$	logical NOT
$\mathbb{I}(x)$	Indicator function, $\mathbb{I}(x) = 1$ if x is true, else $\mathbb{I}(x) = 0$
∞	Infinity
$\rightarrow$	Tends towards, e.g., $n \rightarrow \infty$
$\leftarrow$	in an algorithm: assign to variable t the new value $t+1$, e.g., $t \leftarrow t+1$
$\propto$	Proportional to, so $y = ax$ can be written as $y \propto x$
$\lvert x \rvert$	Absolute value
$\lvert \mathcal{S} \rvert$	Size (cardinality) of a set
$n!$	Factorial function
∇	Vector of first derivatives
∇^2	Hessian matrix of second derivatives
$\triangleq$	Defined as
$O(\cdot)$	Big-O: roughly means order of magnitude
$\mathbb{R}$	The real numbers
$1:n$	Range (Matlab convention): $1:n = 1, 2, ..., n$
$\approx$	Approximately equal to
$\arg\max_x f(x)$	Argmax: the value x that maximizes f
$\arg\min_x f(x)$	Argmin: the value x that minimizes f
$B(a, b)$	Beta function, $B(a, b) = \frac{\Gamma(a)\Gamma(b)}{\Gamma(a+b)}$

$B(\vec{\alpha})$	Multivariate beta function, $\frac{\prod\limits_k \Gamma(\alpha_k)}{\Gamma(\sum\limits_k \alpha_k)}$
$n!$	n factorial $= n * (n-1) * (n-2) * ... * 1$
$\binom{n}{k} = \frac{n!}{k!(n-k)!}$	n choose k , equal to $n!/(k!(nk)!)$
$\delta(x)$	Dirac delta function,$\delta(x) = \infty$ if $x = 0$, else $\delta(x) = 0$
$\Gamma(x)$	Gamma function, $\Gamma(x) = \int_0^\infty u^{x-1}e^{-u}\mathrm{d}u$
$\Psi(x)$	Digamma function,$Psi(x) = \frac{d}{dx}\log\Gamma(x)$
$\mathcal{X}$	A set from which values are drawn (e.g.,$\mathcal{X} = \mathbb{R}^D$)
$\equiv$	equivalent to (or defined to be)
$\lim\limits_{a\to\infty} f(x)$	the value of $f(x)$ in the limit as x approaches a
$m \mod n$	m modulo n, the remainder when m is divided by n (e.g. $7mod5 = 2$)
ln	logarithm base e, or natural logarithm of x
log	logarithm base 10 of x
$\log_2$	logarithm base 2 of x
$\exp(x)$ or e^x	exponential of x, i.e., e raised the power of x
$\partial f(x)/\partial x$	partial derivative of f with respect to x
$\int_a^b f(x)dx$	the integral of $f(x)$ between a and b. If no limits are written, the full space is assumed.
$F(X;\theta)$	function of x, with implied dependence upon θ
$< x >$	expected value of random variable x
$\bar{x}$	mean or averrage value of x
$\mathcal{E}[f(x)]$	the expected value of function $f(x)$ where x is a random variable
$\mathcal{E}_y[f(x,y)]$	the expected value of function over several variables, $f(x)$, taken over a subset y of them
$\sum_{i=1}^n a_i$	the sum from $i = 1$ to $n : a_1 + a_2 + ... + a_n$
$\prod_{i=1}^n a_i$	the product from $i = 1$ to $n : a_1 * a_2 * ... * a_n$
$f(x) * g(x)$	convolution of $f(x)$ with $g(x)$
$\mathcal{A}, \mathcal{B}, \mathcal{C}, \mathcal{D}, ...$	"Calligraphic" font generally denotes sets or lists, e.g., data set $\mathcal{D} = x_1, ..., x_n$
$x \in \mathcal{D}$	x is an element of set $\mathcal{D}$
$x \notin \mathcal{D}$	x is not an element of set $\mathcal{D}$
$\mathcal{D} \cup \mathcal{D}$	x union of two sets, i.e., the set containing all elements of $\mathcal{D}$ and $\mathcal{D}$
$\|\mathcal{D}\|$	cardinality of set $\mathcal{D}$, i.e., the number of (possibly non-distinct) elements in it
$\max\limits_x[\mathcal{D}]$	the maximum x value in set $\mathcal{D}$
$dom(x)$	Domain of variable x
$x = x$	The variable x is in the state x
$dim(x)$	For a discrete variable x, this denotes the number of states x can take
$x_{a:b}$	$x_a, x_{a+1}, ..., x_b$
$\not<, \not>$	not less than; not greater than
$\neq$	not equal to
$\ll, \gg$	much less than; much greater than
d/dx	the derivative with respect to x
$\mathcal{M} \subset \mathcal{N}$	$\mathcal{M}$ is a subset of $\mathcal{N}$
$\mathcal{M} \supset \mathcal{N}$	$\mathcal{M}$ contains $\mathcal{N}$
$\mathcal{M} \bigcap \mathcal{N}$	intersection of $\mathcal{M}$ and $\mathcal{N}$
$\implies$	implies
$\longleftrightarrow$	equivalent to
$\exists$	there exists
$\forall$	for every

3 Linear algebra notations

We use boldface lower-case to denote vectors, such as $\vec{x}$, and boldface upper-case to denote matrices, such as $\vec{X}$. We denote entries in a matrix by non-bold upper case letters, such as X_{ij}.

Vectors are assumed to be column vectors, unless noted otherwise. We use $(x_1, \cdots, x_D)$ to denote a column vector created by stacking D scalars. If we write $\vec{X} = (\vec{x}_1, \cdots, \vec{x}_n)$, where the left hand side is a matrix, we mean to stack the $\vec{x}_i$ along the columns, creating a matrix.

Symbol	Meaning
$\vec{X} \succ 0$	$\vec{X}$ is a positive definite matrix
$tr(\vec{X})$	Trace of a matrix
$det(\vec{X})$	Determinant of matrix $\vec{X}$
$\lvert\vec{X}\rvert$	Determinant of matrix $\vec{X}$
$\vec{X}^{-1}$	Inverse of a matrix
$\vec{X}^{\dagger}$	Pseudo-inverse of a matrix
$\vec{X}^T$	Transpose of a matrix
$\vec{x}^T$	Transpose of a vector
$\mathrm{diag}(x)$	Diagonal matrix made from vector $\vec{x}$
$\mathrm{diag}(X)$	Diagonal vector extracted from matrix $\vec{X}$
$\vec{I}$ or $\vec{I}_d$	Identity matrix of size $d \times d$ (ones on diagonal, zeros of)
$\vec{1}$ or $\vec{1}_d$	Vector of ones (of length d)
$\vec{0}$ or $\vec{0}_d$	Vector of zeros (of length d)
$\lvert\lvert\vec{x}\rvert\rvert = \lvert\lvert\vec{x}\rvert\rvert_2$	Euclidean or ℓ_2 norm $\sqrt{\sum\limits_{j=1}^{d} x_j^2}$
$\lvert\lvert\vec{x}\rvert\rvert_1$	ℓ_1 norm $\sum\limits_{j=1}^{d} \lvert x_j\rvert$
$\vec{X}_{:,j}$	j'th column of matrix
$\vec{X}_{i,:}$	transpose of i'th row of matrix (a column vector)
$\vec{X}_{i,j}$	Element (i, j) of matrix $\vec{X}$
$\vec{x} \otimes \vec{y}$	Tensor product of $\vec{x}$ and $\vec{y}$
R^d	d-dimensional Euclidean space
x, A, ...	boldface is used for (column) vectors and matrices
$f(x)$	vector-valued function (note the boldface) of a scalars
$f(\chi)$	vector-valued function (note the boldface) of a vector
I	identity matrix, square matrix having 1s on the diagonal and 0 everywhere else
$\sum$	covariance matrix
λ	eigenvalue
$\mathbf{e}$	eigenvector
$\mathbf{u}_i$	unit vector in the ith direction in Euclidean space
$dim\ x$	The dimension of vector/matrix x

4 Probability notations

We denote random and fixed scalars by lower case, random and fixed vectors by bold lower case, and random and fixed matrices by bold upper case. Occasionally we use non-bold upper case to denote scalar random variables. Also, we use $p()$ for both discrete and continuous random variables

Symbol	Meaning
X, Y	Random variable
$P()$	Probability of a random event
$F()$	Cumulative distribution function(CDF), also called distribution function
$p(x)$	Probability mass function(PMF)
$f(x)$	probability density function(PDF)
$F(x, y)$	Joint CDF
$p(x, y)$	Joint PMF
$f(x, y)$	Joint PDF

$p(X\|Y)$	Conditional PMF, also called conditional probability
$f_{X\|Y}(x\|y)$	Conditional PDF
$X \perp Y$	X is independent of Y
$X \not\perp Y$	X is not independent of Y
$X \perp Y\|Z$	X is conditionally independent of Y given Z
$X \not\perp Y\|Z$	X is not conditionally independent of Y given Z
$X \sim p$	X is distributed according to distribution p
$\vec{\alpha}$	Parameters of a Beta or Dirichlet distribution
$\text{cov}[X]$	Covariance of X
$\mathbb{E}[X]$	Expected value of X
$\mathbb{E}_q[X]$	Expected value of X wrt distribution q
$\mathbb{H}(X)$ or $\mathbb{H}(p)$	Entropy of distribution $p(X)$
$\mathbb{I}(X;Y)$	Mutual information between X and Y
$\mathbb{KL}(p\|\|q)$	KL divergence from distribution p to q
$\ell(\vec{\theta})$	Log-likelihood function
$L(\theta, a)$	Loss function for taking action a when true state of nature is θ
λ	Precision (inverse variance) $\lambda = 1/\sigma^2$
Λ	Precision matrix $\Lambda = \Sigma^{-1}$
$\text{mode}[\vec{X}]$	Most probable value of $\vec{X}$
μ	Mean of a scalar distribution
$\vec{\mu}$	Mean of a multivariate distribution
Φ	cdf of standard normal
ϕ	pdf of standard normal
$\vec{\pi}$	multinomial parameter vector, Stationary distribution of Markov chain
ρ	Correlation coefficient
$\text{sigm}(x)$	Sigmoid (logistic) function, $\frac{1}{1+e^{-x}}$
σ^2	Variance
Σ	Covariance matrix
$\text{var}[x]$	Variance of x
ν	Degrees of freedom parameter
Z	Normalization constant of a probability distribution
˜	has the distribution, e.g., $p(x)$˜$N(\mu, \sigma^2)$
$N(\mu, \sigma^2)$	multidimensional normal or Gaussian distribution with mean μ and variance σ^2
$O(h(x))$	big oh order of $h(x)$
$\Theta(h(x))$	big theta order of $h(x)$
$\Omega(h(x))$	big omega order of $h(x)$
$\sup\limits_x f(x)$	the supremum value of $f(x)$-the global maximum of $f(x)$ over all values of x
$p(x = tr)$	Probability of variable x being in the state true
$p(x = fa)$	Probability of variable x being in the state false
$p(x \cap y)$	Probability of x and y
$p(x \cup y)$	Probability of x or y
$p(x\|\|y)$	Probability of x contioned on y
$\langle f(x) \rangle_{g(x)}$	The average of the function $f(x)$ with respect to the distribtioin $p(x)$
$\sigma(x)$	The logistic sigmoid $\frac{1}{(1+\exp(-x))}$
$erf(x)$	The (Gaussian) error function

5 Specific Machine learning notations

In general, we use upper case letters to denote constants, such as C, K, M, N, T, etc. We use lower case letters as dummy indexes of the appropriate range, such as $c = 1 : C$ to index classes, $i = 1 : M$ to index data cases, $j = 1 : N$ to index input features, $k = 1 : K$ to index states or clusters, $t = 1 : T$ to index time, etc.

We use x to represent an observed data vector. In a supervised problem, we use y or $\vec{y}$ to represent the desired output label. We use $\vec{z}$ to represent a hidden variable. Sometimes we also use q to represent a hidden discrete variable.

Symbol	Meaning
C	Number of classes
D	Dimensionality of data vector (number of features - of a feature vector gained)
N	Number of data cases
N_c	Number of examples of class c,$N_c = \sum_{i=1}^{N} \mathbb{I}(y_i = c)$
R	Number of outputs (response variables)
$\mathcal{D}$	Training data $\mathcal{D} = \{(\vec{x}_i, y_i) \| i = 1 : N\}$
$\mathcal{D}_{test}$	Test data
$\mathcal{X}$	Input space
$\mathcal{Y}$	Output space
K	Number of states or dimensions of a variable (often latent)
$k(x, y)$	Kernel function
$\vec{K}$	Kernel matrix
$\mathcal{H}$	Hypothesis space
L	Loss function
$J(\vec{\theta})$	Cost function
$f(\vec{x})$	Decision function
$P(y\|\vec{x})$	TODO
λ	Strength of ℓ_2 or $\ell_1 regularizer$
$\phi(x)$	Basis function expansion of feature vector $\vec{x}$
Φ	Basis function expansion of design matrix $\vec{X}$
$q()$	Approximate or proposal distribution
$Q(\vec{\theta}, \vec{\theta}_{old})$	Auxiliary function in EM
T	Length of a sequence
$T(\mathcal{D})$	Test statistic for data
$\vec{T}$	Transition matrix of Markov chain
$\vec{\theta}$	Parameter vector
$\vec{\theta}^{(s)}$	s'th sample of parameter vector
$\hat{\vec{\theta}}$	Estimate (usually MLE or MAP) of $\vec{\theta}$
$\hat{\vec{\theta}}_{MLE}$	Maximum likelihood estimate of $\vec{\theta}$
$\hat{\vec{\theta}}_{MAP}$	MAP estimate of $\vec{\theta}$
$\bar{\vec{\theta}}$	Estimate (usually posterior mean) of $\vec{\theta}$
$\vec{w}$	Vector of regression weights (called $\vec{\beta}$ in statistics)
b	intercept (called ε in statistics)
$\vec{W}$	Matrix of regression weights
x_{ij}	Component (i.e., feature) j of data case i ,for $i = 1 : N, j = 1 : D$
$\vec{x}_i$	Training case, $i = 1 : N$
$\vec{X}$	Design matrix of size $N \times D$
$\bar{\vec{x}}$	Empirical mean $\bar{\vec{x}} = \frac{1}{N} \sum_{i=1}^{N} \vec{x}_i$
$\tilde{\vec{x}}$	Future test case
$\vec{x}_*$	Feature test case
$\vec{y}$	Vector of all training labels $\vec{y} = (y_1, ..., y_N)$
z_{ij}	Latent component j for case i
S	Number of samples

6 Graphical model notations

In graphical models, we index nodes by $s, t, u \in V$, and states by $i, j, k \in \mathcal{X}$.

Symbol	Meaning
$\tilde{st}$	Node s is connected to node t
bel	Belief function
$\mathcal{C}$	Cliques of a graph
ch_j	Child of node j in a DAG (directed acyclic graph)
$desc_j$	Descendants of node j in a DAG
G	A graph
$\mathcal{E}$	Edges of a graph
mb_t	Markov blanket of node t
nbd_t	Neighborhood of node t
pa_t	Parents of node t in a DAG
$pred_t$	Predecessors of node t in a Direct Acyclic Graph (DAG) with respect to some ordering
$\psi_c(x_c)$	Potential function for clique c
$\mathcal{S}$	Separators of a graph
θ_{sjk}	prob. node s is in state k given its parents are in states j
$\mathcal{V}$	Nodes of a graph
$pa(x)$	The parents of x
$ch(x)$	The children of x
$ne(x)$	The neighbours of x
$\tilde{ij}$	The set of unique neighbouring edges on a graph

7 Recommended Literature (incomplete)

To strengthen the mathematical understanding the following textbooks can be recommended:

Dan SIMOVICI & Chabane DJERABA (2014) Mathematical Tools for Data Mining: Set Theory, Partial Orders, Combinatorics, Second Edition. London, Heidelberg, New York, Dordrecht: Springer [12]. This is a must-have book on every desk, a comprehensive compendium of the maths we need in our daily work, includes topologies and measures in metric spaces.

Keneth H. ROSEN (2013) Discrete Mathematics and its Applications. New York: McGraw-Hill [13]. This discrete mathematics course book spans a thread through mathematical reasoning, combinatorial analysis, discrete structures, algorithmic thinking and applications as well as modeling very recommendable.

Richard O. DUDA, Peter E. HART & David G. STORK (2001) Pattern Classification. New York: John Wiley [14]. This is THE classic work from Bayesian Decision Theory, Nonparametric Techniques, Linear Discriminant Functions and Stochastic Methods with a useful and applicable mathematical foundation. A must-have for any data scientist.

About the Lecturer

Andreas Holzinger is lead of the Holzinger Group HCI-KDD at the Institute for Medical Informatics, Statistics and Documentation at the Medical University Graz, and Associate Professor of Applied Computer Science at the Institute of Information Systems and Computer Media at Graz University of Technology. Currently, Andreas is Visiting Professor for Machine Learning in Health Informatics at the Faculty of Informatics at Vienna University of Technology. His research interests are in supporting human intelligence with machine learning to help to solve problems in the biomedical domain. Andreas obtained a Ph.D. in Cognitive Science from Graz University in 1998 and his Habilitation (second Ph.D.) in Computer Science from Graz University of Technology in 2003. Andreas was Visiting Professor in Berlin, Innsbruck, London (2 times), and Aachen. Andreas founded the international Expert Network HCI-KDD to foster a synergistic combination of methodologies of two areas that offer ideal conditions towards unraveling problems in understanding complex data: HumanComputer Interaction (HCI) and Knowledge Discovery from Data (KDD), with the goal of supporting human intelligence with machine learning for knowledge discovery. Andreas is Associate Editor of Knowledge and Information Systems (KAIS), and member of IFIP WG 12.9 Computational Intelligence. More details via: www.aholzinger.at

References

[1] M. I. Jordan and T. M. Mitchell. Machine learning: Trends, perspectives, and prospects. *Science*, 349(6245):255–260, 2015. doi: 10.1126/science.aaa8415. URL `http://www.sciencemag.org/content/349/6245/255.abstract`.

[2] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *Nature*, 521(7553):436–444, 2015. doi: 10.1038/nature14539.

[3] Andreas Holzinger. Interactive machine learning for health informatics: When do we need the human-in-the-loop? *Springer Brain Informatics (BRIN)*, 3(2):119–131, 2016. doi: 10.1007/s40708-016-0042-6. URL `http://dx.doi.org/10.1007/s40708-016-0042-6`.

[4] Andreas Holzinger. Interactive machine learning (iml). *Informatik Spektrum*, 39(1):64–68, 2016. doi: 10.1007/s00287-015-0941-6. URL `https://pure.tugraz.at/portal/files/3107876/iML.pdf`.

[5] Michael Hund, Werner Sturm, Tobias Schreck, Torsten Ullrich, Daniel Keim, Ljiljana Majnaric, and Andreas Holzinger. Analysis of patient groups and immunization results based on subspace clustering. In Yike Guo, Karl Friston, Faisal Aldo, Sean Hill, and Hanchuan Peng, editors, *Brain Informatics and Health, Lecture Notes in Artificial Intelligence LNAI 9250*, volume 9250, pages 358–368. Springer International Publishing, Cham, 2015. doi: 10.1007/978-3-319-23344-4_35.

[6] A Holzinger, M Plass, K Holzinger, GC Crisan, CM Pintea, and V Palade. Towards interactive machine learning (iml): Applying ant colony algorithms to solve the traveling salesman problem with the human-in-the-loop approach. In *IFIP International Cross Domain Conference and Workshop (CD-ARES)*, page in print. Springer, Heidelberg, Berlin, New York, 2016. URL `http://hci-kdd.org/project/iml`.

[7] Peter Kieseberg, Bernd Malle, Peter Frhwirt, Edgar Weippl, and Andreas Holzinger. A tamper-proof audit and control system for the doctor in the loop. *Brain Informatics*, pages 1–11, 2016. doi: 10.1007/s40708-016-0046-2. URL `http://dx.doi.org/10.1007/s40708-016-0046-2`.

[8] Andreas Holzinger. *Biomedical Informatics: Discovering Knowledge in Big Data*. Springer, New York, 2014. doi: 10.1007/978-3-319-04528-3.

[9] Andreas Holzinger. Human-computer interaction and knowledge discovery (HCI-KDD): What is the benefit of bringing those two fields to work together? In Alfredo Cuzzocrea, Christian Kittl, Dimitris E. Simos, Edgar Weippl, and Lida Xu, editors, *Multidisciplinary Research and Practice for Information Systems, Springer Lecture Notes in Computer Science LNCS 8127*, pages 319–328. Springer, Heidelberg, Berlin, New York, 2013.

[10] Gerald Petz, Micha Karpowicz, Harald Frschu, Andreas Auinger, Vclav Sttesk, and Andreas Holzinger. Computational approaches for mining users opinions on the web 2.0. *Information Processing & Management*, 51(4):510–519, 2015. doi: http://dx.doi.org/10.1016/j.ipm.2014.07.011. URL `http://www.sciencedirect.com/science/article/pii/S0306457315000655`.

[11] Andreas Holzinger, Matthias Dehmer, and Igor Jurisica. Knowledge discovery and interactive data mining in bioinformatics - state-of-the-art, future challenges and research directions. *BMC Bioinformatics*, 15(S6):I1, 2014. doi: doi:10.1186/1471-2105-15-S6-I1. URL `http://www.biomedcentral.com/1471-2105/15/S6/I1`.

[12] Dan A Simovici and Chabane Djeraba. *Mathematical tools for data mining. Second Edition*. 2014. doi: 10.1007/978-1-4471-6407-4.

[13] Kenneth H Rosen and Kamala Krithivasan. *Discrete mathematics and its applications. Seventh Edition*. McGraw-Hill, New York, 2012.

[14] Richard O. Duda, Peter E. Hart, and David G. Stork. *Pattern Classification. Second Edition*. Wiley, New York et al., 2000.