Mathematical Notations for Learning from Data Course

Islem Rekik*

Istanbul Technical University

Abstract. **Inspiring quotes**. *"You'll need to set small, specific goals to master a skill, but first you'll want to be sure of the basics."* Source: *Learn better* by Ulrich Boser.
"Learning is an iterative process that requires that you revisit what you have learnt." Source: *Make it stick* by Henry L. Roediger III and Mark A. McDaniel.
"The good news is that we now know of simple and practical strategies that anybody can use; at any point in life, to learn better and remember longer: various forms of retrieval practice, such as low-stakes quizzing and self-testing, spacing out practice, interleaving the practice of different but related topics or skills, trying to solve a problem before being taught the solution, distilling the underlying principles or rules that differentiate types of problems, and so on." Source: *Make it stick* by Henry L. Roediger III and Mark A. McDaniel.
Pre-requisites: Linear algebra. To refresh your memory, you can check 3blue1brown YouTube playlist[a].
Machine Learning Blinks. Check the YouTube link below to watch the lectures[b].

* Corresponding author: irekik@itu.edu.tr; http://basira-lab.com

[a] https://www.youtube.com/watch?v=fNk_zzaMoSs&list=PLZHQObOWTQDPD3MizzM2xVFitgF8hE_ab

[b] https://www.youtube.com/watch?v=HyWmnlahXAA&list=PLug43ldmRSo1LDlvQOPzgoJ6wKnfmzimQ

Table 1: *Major mathematical notations used in lecture 1.*

Mathematical notation	Definition
$\mathcal{D}$	dataset
n	number of samples in a dataset $\mathcal{D}$
d	number of features
$\mathbf{x} \in \mathbb{R}^{d\times 1}$	feature vector or data point (sample)
$\mathbf{x}_{feature}^{sample}$	–
$\mathbf{x}^i \in \mathbb{R}^{d\times 1}$	i^{th} sample in the population
$\mathbf{x}_j^i \in \mathbb{R}$	j^{th} feature of i^{th} sample in the population
$\mathcal{D} = \{\mathbf{x}^i, y^i\}_{i=1}^n$	training dataset where $\mathbf{x}^i \in \mathbb{R}^d$ denotes the feature vector for the i^{th} sample and $y^i \in \mathbb{R}$ denotes its score
$\mathbf{X} \in \mathbb{R}^{d\times n}$	data matrix stacking all samples vertically
f	mapping or transformation function to learn
$f : \mathbb{R} \mapsto \mathbb{R}$	one-to-one mapping
$f : \mathbb{R} \mapsto \mathbb{R}^m$	one-to-many mapping
$f : \mathbb{R}^p \mapsto \mathbb{R}$	many-to-one mapping
$f : \mathbb{R}^p \mapsto \mathbb{R}^m$	many-to-many mapping

Table 2: *Major mathematical notations used in lecture 3.*

Mathematical notation	Definition
$\mathcal{D}$	dataset
n	number of samples in a dataset $\mathcal{D}$
d	number of features
$\mathbf{x} \in \mathbb{R}^{d\times 1}$	feature vector or data point (sample)
$\mathbf{x}_{feature}^{sample}$	–
$\mathbf{x}^i \in \mathbb{R}^{d\times 1}$	i^{th} sample in the population
$\mathbf{x}_j^i \in \mathbb{R}$	j^{th} feature of i^{th} sample in the population
$\mathbf{\Sigma} \in \mathbb{R}^{d\times d}$	covariance matrix of data population $\{\mathbf{x}^i\}_{i=1}^n$
$\lvert\mathbf{A}\rvert \in \mathbb{R}$	determinant of matrix A
$p(x) = \frac{1}{\sqrt{2\pi\sigma}} exp(\frac{-1}{2}\frac{\lvert\lvert x-\mu\rvert\rvert_2^2}{\sigma^2})$	probability density function of a variable $x \in \mathbb{R}$
$p(\mathbf{x}) = \frac{1}{(2\pi)^{d/2}\lvert\mathbf{\Sigma}\rvert^{1/2}} exp[\frac{-1}{2}(\mathbf{x}-\mu)^T\mathbf{\Sigma}^{-1}(\mathbf{x}-\mu)]$	probability density function of a multidimensional variable $\mathbf{x} \in \mathbb{R}^{d\times 1}$
$\mu \in \mathbb{R}^{d\times 1}$	sample mean $\mu = \frac{1}{n}\sum_{i=1}^n \mathbf{x}^i$
$\lvert\lvert\mathbf{x}-\mu\rvert\rvert_{\mathbf{\Sigma}^{-1}} = (\mathbf{x}-\mu)^T\mathbf{\Sigma}^{-1}(\mathbf{x}-\mu) \in \mathbb{R}$	Mahalanobis distance between $\mathbf{x}$ and μ
$\mathbf{I}_{d\times d} \in \mathbb{R}^{d\times d}$	identify matrix of size d ×d
$\lvert\lvert\mathbf{x}-\mu\rvert\rvert_{\mathbf{I}_{d\times d}} = (\mathbf{x}-\mu)^T(\mathbf{x}-\mu) \in \mathbb{R}$	Euclidean distance between $\mathbf{x}$ and μ
	also noted as L_2 norm $\lvert\lvert\cdot\rvert\rvert_2$
$g_i(\mathbf{x}) = \mathbf{w}_i^T\mathbf{x} + \mathbf{w}_{i0}$	discriminant Bayes function for class i when $\mathbf{\Sigma}_i = \sigma_i^2\mathbf{I}$
	general case: when $\sigma_i^2 \neq \sigma_j^2$ for classes i and j (i.e., different means $\mu_i \neq \mu_j$
	but constant variance for all data features in each class)
	special case: when $\sigma_i^2 = \sigma_j^2$ for classes i and j (i.e., different means $\mu_i \neq \mu_j$
	but constant variances across all classes)
	(i.e., lines connecting means of different classes are perpendicular to decision boundaries)
	if $ln(p(c_i)) = ln(p(c_j))$, $g_i(\mathbf{x}) = -\lvert\lvert\mathbf{x}-\mu_i\rvert\rvert_2^2$
$g_i(\mathbf{x}) = \mathbf{w}_i^T\mathbf{x} + \mathbf{w}_{i0}$	discriminant Bayes function for class i when $\mathbf{\Sigma}_i = \mathbf{\Sigma}_j = \mathbf{\Sigma}$
	(i.e., constant data feature covariance $\mathbf{\Sigma}$ across classes)
	(i.e., lines connecting means of different classes are not perpendicular to decision boundaries)
	if $ln(p(c_i)) = ln(p(c_j))$, $g_i(\mathbf{x}) = -\frac{1}{2}\lvert\lvert\mathbf{x}-\mu_i\rvert\rvert_{\mathbf{\Sigma}^{-1}}^2$
$g_i(\mathbf{x}) = \mathbf{x}^T\mathbf{W}\mathbf{x} + \mathbf{w}_i^T\mathbf{x} + \mathbf{w}_{i0}$	quadratic discriminant function (decision boundaries are nonlinear)

Table 3: *Major mathematical notations used in lectures 4.*

Mathematical notation	Definition
$arg\ min_{\mathbf{w}} \mathcal{L}(\mathbf{w}) = \frac{1}{n} \sum_{(\mathbf{x}^i, y^i) \in \mathcal{D}} E(f_{\mathbf{w}}(\mathbf{x}^i), y^i) + R(\dots)$	supervised learning energy cost (loss function)
f	the mapping function to learn from $\mathbf{x}^i \rightarrow y^i$
E	the error function between the predicted target by f and the ground truth observation y^i
R	regularization term to avoid overfitting and control model complexity
$\mathbf{w}$	optimization parameters (weight vector)
	set of parameters that minimize the loss function $\mathcal{L}(\mathbf{w})$
$h(w) = l(v) + l'(v)(w - v)$	first-order Taylor approximation of the loss function l at point w (in 1-dimensional space)
$l'(w)$	first derivative of function l evaluated at point w
$h(w) = l(v) + l'(v)(w - v) + \frac{1}{2} l''(v)(w - v)^2$	second-order Taylor approximation of the loss function l at point w (in 1-dimensional space)
$l''(w)$	second derivative of function l evaluated at point w
$h(\mathbf{w}) = l(\mathbf{v}) + \nabla l(\mathbf{w})(\mathbf{w} - \mathbf{v})$	first-order Taylor approximation of high-dimensional loss function l at vector point $\mathbf{w} \in \mathbb{R}^d$
$\mathbf{w} = [w_1\ w_2 \dots w_N] \in \mathbb{R}^d$	weight vector to learn
$\nabla l(\mathbf{w}) \in \mathbb{R}^{d \times 1}$	gradient vector of the multivariate loss function l at location $\mathbf{w}$
	note that $\nabla l(\mathbf{w})^T$ is $\in \mathbb{R}^{1 \times d}$ (row vector)
	$\nabla l(\mathbf{v}) = [\frac{\partial}{\partial w_1} l(\mathbf{v})\ \frac{\partial}{\partial w_2} l(\mathbf{v}) \dots \frac{\partial}{\partial w_d} l(\mathbf{v})]^T$
$h(\mathbf{w}) = l(\mathbf{v}) + \nabla l(\mathbf{w})(\mathbf{w} - \mathbf{v}) + \frac{1}{2}(\mathbf{w} - \mathbf{v})^T \nabla^2 l(\mathbf{w})(\mathbf{w} - \mathbf{v})$	second-order Taylor approximation of high-dimensional loss function l at vector point $\mathbf{w} \in \mathbb{R}^d$
$\nabla^2 l(\mathbf{w}) \in \mathbb{R}^{d \times d}$	Hessian symmetric matrix of second derivatives of l along all its dimensions (variables)
	l is many times differentiable at the vector valued input $\mathbf{w}$
$l'(w) = 0$	stationary point (min, max or saddle) for a 1-dimensional function
$\nabla l(\mathbf{w}) = \mathbf{0}_{d \times 1}$	stationary point (min, max or saddle) for an N-dimensional function
	(all elements of the gradient are zero)
$\mathbf{Q} = \mathbf{Q}^T$	$\mathbf{Q}$ is symmetric
$\frac{\partial \mathbf{a}^T \mathbf{x}}{\partial \mathbf{x}} = \mathbf{a}$	matrix cookbook[a] (also check[b])
$\frac{\partial \mathbf{a}^T \mathbf{X} \mathbf{a}}{\partial \mathbf{a}} = 2\mathbf{X}\mathbf{a}$ (for $X = X^T$)	matrix cookbook
$l''(w) > 0$	convex function (facing upward)
$l''(w) < 0$	concave function (facing downward)
$\frac{\partial \mathbf{A}\mathbf{x}}{\partial \mathbf{x}} = \frac{1}{2}(\mathbf{A}^T + \mathbf{A})$	matrix cookbook
$\nabla^2 l(\mathbf{w}) = \frac{1}{2}(\mathbf{Q}^T + \mathbf{Q}) \in \mathbb{R}^{d \times d}$	is the Hessian matrix of l equal to $l(\mathbf{w}) = \frac{1}{2}\mathbf{w}^T \mathbf{Q} \mathbf{w} + \mathbf{r}^T \mathbf{w} + d$
	$d \in \mathbb{R}$ and $\mathbf{r} \in \mathbb{R}^{d \times 1}$
$\nabla l(\mathbf{w}) = \mathbf{Q}\mathbf{w} + \mathbf{r} \in \mathbb{R}^{d \times 1}$	is the gradient vector of l equal to $l(\mathbf{w}) = \frac{1}{2}\mathbf{w}^T \mathbf{Q} \mathbf{w} + \mathbf{r}^T \mathbf{w} + d$
$\mathbf{w}^k = \mathbf{w}^{k-1} - \alpha_k \nabla l(\mathbf{w}^{\mathbf{k-1}})$	gradient descent for finding the optimal $\mathbf{w}$

[a] https://www.math.uwaterloo.ca/~hwolkowi/matrixcookbook.pdf

[b] https://ninova.itu.edu.tr/en/courses/institute-of-science-and-technology/1580/blg-527e/ekkaynaklar?g179937