`latex-math` Macros

compiled: 2021-10-17

Latex macros like `\frac{#1}{#2}` with arguments are displayed as $\frac{\#1}{\#2}$.

Note that macro declarations may only span a single line to be displayed correctly in the below tables.

Contents

basic-math

Macro	Notation	Comment
`\N`	$\mathbb{N}$	N, naturals
`\Z`	$\mathbb{Z}$	Z, integers
`\Q`	$\mathbb{Q}$	Q, rationals
`\R`	$\mathbb{R}$	R, reals
`\C`	$\mathbb{C}$	C, complex
`\continuous`	$\mathcal{C}$	C, space of continuous functions
`\M`	$\mathcal{M}$	machine numbers
`\epsm`	ϵ_m	maximum error
`\xt`	$\tilde{x}$	x tilde
`\argmax`	$\arg\max$	argmax
`\argmin`	$\arg\min$	argmin
`\argminlim`	$\arg\min$	argmax with limits
`\argmaxlim`	$\arg\max$	argmin with limits
`\sign`	sign	sign, signum
`\I`	$\mathbb{I}$	I, indicator
`\order`	$\mathcal{O}$	O, order
`\fp`	$\frac{\partial \cdot}{\partial \cdot}$	partial derivative
`\pd`	$\frac{\partial \cdot}{\partial \cdot}$	partial derivative
`\sumin`	$\sum\limits_{i=1}^{n}$	summation from i=1 to n
`\sumim`	$\sum\limits_{i=1}^{m}$	summation from i=1 to m
`\sumjp`	$\sum\limits_{j=1}^{p}$	summation from j=1 to p
`\sumik`	$\sum\limits_{i=1}^{k}$	summation from i=1 to k
`\sumkg`	$\sum\limits_{k=1}^{g}$	summation from k=1 to g
`\sumjg`	$\sum\limits_{j=1}^{g}$	summation from j=1 to g
`\meanin`	$\frac{1}{n}\sum\limits_{i=1}^{n}$	mean from i=1 to n
`\meankg`	$\frac{1}{g}\sum\limits_{k=1}^{g}$	mean from k=1 to g
`\prodin`	$\prod\limits_{i=1}^{n}$	product from i=1 to n
`\prodkg`	$\prod\limits_{k=1}^{g}$	product from k=1 to g
`\prodjp`	$\prod\limits_{j=1}^{p}$	product from j=1 to p
`\one`	$\mathbf{1}$	1, unitvector
`\zero`	$\mathbf{0}$	0-vector
`\id`	$\boldsymbol{I}$	I, identity
`\diag`	diag	diag, diagonal
`\trace`	tr	tr, trace
`\spn`	span	span
`\scp`	$\langle \cdot, \cdot \rangle$	<.,.>, scalarproduct
`\mat`	$\begin{pmatrix}\cdot\end{pmatrix}$	short pmatrix command
`\Amat`	$\mathbf{A}$	matrix A
`\xv`	$\mathbf{x}$	vector x (bold)
`\xtil`	$\tilde{\mathbf{x}}$	vector x-tilde (bold)

\yv	$\mathbf{y}$	vector y (bold)
\Deltab	$\boldsymbol{\Delta}$	error term for vectors
\E	$\mathbb{E}$	E, expectation
\var	Var	Var, variance
\cov	Cov	Cov, covariance
\corr	Corr	Corr, correlation
\normal	$\mathcal{N}$	N of the normal distribution
\iid	$\overset{i.i.d}{\sim}$	dist with i.i.d superscript
\distas	$\dot{\sim}$	... is distributed as ...

Back to contents

basic-ml

Macro	Notation	Comment
\Xspace	$\mathcal{X}$	X, input space
\Yspace	$\mathcal{Y}$	Y, output space
\nset	$\{1, \ldots, n\}$	set from 1 to n
\pset	$\{1, \ldots, p\}$	set from 1 to p
\gset	$\{1, \ldots, g\}$	set from 1 to g
\Pxy	$\mathbb{P}_{xy}$	P__xy
\Exy	$\mathbb{E}_{xy}$	E__xy: Expectation over random variables xy
\xy	$(\mathbf{x}, y)$	observation (x, y)
\xvec	$(x_1, \ldots, x_p)^T$	(x1, ..., xp)
\Xmat	$\mathbf{X}$	Design matrix
\allDatasets	$\mathbb{D}$	The set of all datasets
\D	$\mathcal{D}$	D, data
\obsi	$\left(\mathbf{x}^{(\cdot)}, y^{(\cdot)}\right)$	observation (x^(i), y^(i))
\Dn	$\mathcal{D}_n$	D__n, data of size n
\allDatasetsn	$\mathbb{D}_n$	The set of all datasets of size n
\defAllDatasetsn	$(\mathcal{X} \times \mathcal{Y})^n$	Def. of the set of all datasets of size n
\defAllDatasets	$\bigcup_{n \in \mathbb{N}} (\mathcal{X} \times \mathcal{Y})^n$	Def. of the set of all datasets
\ydat	$\mathbf{y}$	y (bold), vector of outcomes
\yvec	$\left(y^{(1)}, \ldots, y^{(n)}\right)^T$	(y1, ..., yn), vector of outcomes
\yi	$y^{(\cdot)}$	y^i, i-th observed value of y
\xyi	$\left(\mathbf{x}^{(\cdot)}, y^{(\cdot)}\right)$	(x^i, y^i), i-th observation
\xivec	$\left(x_1^{(i)}, \ldots, x_p^{(i)}\right)^T$	(x1^i, ..., xp^i), i-th observation vector
\xj	$\mathbf{x}_j$	x__j, j-th feature
\xjvec	$\left(x_j^{(1)}, \ldots, x_j^{(n)}\right)^T$	(x^1__j, ..., x^n__j), j-th feature vector
\Dtrain	$\mathcal{D}_{\text{train}}$	D__train, training set
\Dtest	$\mathcal{D}_{\text{test}}$	D__test, test set
\phiv	ϕ	Basis transformation function phi
\phixi	$\phi^{(i)}$	Basis transformation of xi: phi^i := phi(xi)
\lamv	$\boldsymbol{\lambda}$	lambda vector, hyperconfiguration vector
\Lam	$\boldsymbol{\Lambda}$	Lambda, space of all hpos

\preimageInducer	$\left(\bigcup_{n \in \mathbb{N}}(\mathcal{X} \times \mathcal{Y})^n\right) \times \mathbf{\Lambda}$	Set of all datasets times the hyperparameter space
\preimageInducerShort	$\mathbb{D} \times \mathbf{\Lambda}$	Set of all datasets times the hyperparameter space
\inducer	$\mathcal{I}$	Inducer, inducing algorithm, learning algorithm
\ftrue	f_{true}	True underlying function (if a statistical model is assumed)
\ftruex	$f_{\text{true}}(\mathbf{x})$	True underlying function (if a statistical model is assumed)
\fx	$f(\mathbf{x})$	f(x), continuous prediction function
\Hspace	$\mathcal{H}$	hypothesis space where f is from
\fix	$f_i(\mathbf{x})$	f_i(x), discriminant component function
\fjx	$f_j(\mathbf{x})$	f_j(x), discriminant component function
\fkx	$f_k(\mathbf{x})$	f_k(x), discriminant component function
\fgx	$f_g(\mathbf{x})$	f_g(x), discriminant component function
\fh	$\hat{f}$	f hat, estimated prediction function
\fxh	$\hat{f}(\mathbf{x})$	fhat(x)
\fxt	$f(\mathbf{x} \mid \boldsymbol{\theta})$	f(x | theta)
\fxi	$f\left(\mathbf{x}^{(i)}\right)$	f(x^(i))
\fxih	$\hat{f}\left(\mathbf{x}^{(i)}\right)$	f(x^(i))
\fxit	$f\left(\mathbf{x}^{(i)} \mid \boldsymbol{\theta}\right)$	f(x^(i) | theta)
\fhD	$\hat{f}_{\mathcal{D}}$	fhat_D, estimate of f based on D
\fhDtrain	$\hat{f}_{\mathcal{D}_{\text{train}}}$	fhat_Dtrain, estimate of f based on D
\fhDnlambda	$\hat{f}_{\mathcal{D}_n, \boldsymbol{\lambda}}$	model learned on Dn with hp lambda
\fhDlambda	$\hat{f}_{\mathcal{D}, \boldsymbol{\lambda}}$	model learned on D with hp lambda
\fhDnlambdastar	$\hat{f}_{\mathcal{D}_n, \boldsymbol{\lambda}^*}$	model learned on Dn with optimal hp lambda
\fhDlambdastar	$\hat{f}_{\mathcal{D}, \boldsymbol{\lambda}^*}$	model learned on D with optimal hp lambda
\hx	$h(\mathbf{x})$	h(x), discrete prediction function
\hxv	$h(\mathbf{x})$	h(x), discrete prediction function with x (vector) as input
\hh	$\hat{h}$	h hat
\hxh	$\hat{h}(\mathbf{x})$	hhat(x)
\hxt	$h(\mathbf{x} \vert \boldsymbol{\theta})$	h(x | theta)
\hxi	$h\left(\mathbf{x}^{(i)}\right)$	h(x^(i))
\hxit	$h\left(\mathbf{x}^{(i)} \mid \boldsymbol{\theta}\right)$	h(x^(i) | theta)
\yh	$\hat{y}$	yhat for prediction of target
\yih	$\hat{y}^{(i)}$	yhat^(i) for prediction of ith targiet
\thetah	$\hat{\theta}$	theta hat
\thetab	$\boldsymbol{\theta}$	theta vector
\thetabh	$\hat{\boldsymbol{\theta}}$	theta vector hat
\thetat	$\boldsymbol{\theta}^{[t]}$	theta^[t] in optimization
\thetatn	$\boldsymbol{\theta}^{[t+1]}$	theta^[t+1] in optimization
\thxh	$\boldsymbol{\theta}^T \mathbf{x}$	linear combination with theta
\thetahDnlambda	$\hat{\boldsymbol{\theta}}_{\mathcal{D}_n, \boldsymbol{\lambda}}$	theta learned on Dn with hp lambda
\thetahDlambda	$\hat{\boldsymbol{\theta}}_{\mathcal{D}, \boldsymbol{\lambda}}$	theta learned on D with hp lambda
\pdf	p	p
\pdfx	$p(\mathbf{x})$	p(x)
\pixt	$\pi(\mathbf{x} \mid \boldsymbol{\theta})$	pi(x|theta), pdf of x given theta
\pixit	$\pi\left(\mathbf{x}^{(i)} \mid \boldsymbol{\theta}\right)$	pi(x^i|theta), pdf of x given theta
\pixii	$\pi\left(\mathbf{x}^{(i)}\right)$	pi(x^i), pdf of i-th x
\pdfxy	$p(\mathbf{x}, y)$	p(x, y)
\pdfxyt	$p(\mathbf{x}, y \mid \boldsymbol{\theta})$	p(x, y | theta)
\pdfxyit	$p\left(\mathbf{x}^{(i)}, y^{(i)} \mid \boldsymbol{\theta}\right)$	p(x^(i), y^(i) | theta)
\pdfxyk	$p(\mathbf{x} \vert y = k)$	p(x | y = k)
\pdfxyj	$p(\mathbf{x} \vert y = j)$	p(x | y = j)
\lpdfxyk	$\log p(\mathbf{x} \vert y = k)$	log p(x | y = k)
\pdfxiyk	$p\left(\mathbf{x}^{(i)} \vert y = k\right)$	p(x^i | y = k)

`\pik`	π_k	pi_k, prior
`\lpik`	$\log \pi_k$	log pi_k, log of the prior
`\pit`	$\pi(\boldsymbol{\theta})$	Prior probability of parameter theta
`\post`	$\mathbb{P}(y = 1 \mid \mathbf{x})$	P(y = 1 | x), post. prob for y=1
`\pix`	$\pi(\mathbf{x})$	pi(x), P(y = 1 | x)
`\postk`	$\mathbb{P}(y = k \mid \mathbf{x})$	P(y = k | y), post. prob for y=k
`\pikx`	$\pi_k(\mathbf{x})$	pi_k(x), P(y = k | x)
`\pikxt`	$\pi_k(\mathbf{x} \mid \boldsymbol{\theta})$	pi_k(x | theta), P(y = k | x, theta)
`\pijx`	$\pi_j(\mathbf{x})$	pi_j(x), P(y = j | x)
`\pigx`	$\pi_g(\mathbf{x})$	pi_g(x), P(y = g | x)
`\pdfygxt`	$p(y \mid \mathbf{x}, \boldsymbol{\theta})$	p(y | x, theta)
`\pdfyigxit`	$p\left(y^{(i)} \mid \mathbf{x}^{(i)}, \boldsymbol{\theta}\right)$	p(y^i |x^i, theta)
`\lpdfygxt`	$\log p(y \mid \mathbf{x}, \boldsymbol{\theta})$	log p(y | x, theta)
`\lpdfyigxit`	$\log p\left(y^{(i)} \mid \mathbf{x}^{(i)}, \boldsymbol{\theta}\right)$	log p(y^i |x^i, theta)
`\pixh`	$\hat{\pi}(\mathbf{x})$	pi(x) hat, P(y = 1 | x) hat
`\pikxh`	$\hat{\pi}_k(\mathbf{x})$	pi_k(x) hat, P(y = k | x) hat
`\pixih`	$\hat{\pi}(\mathbf{x}^{(i)})$	pi(x^(i)) with hat
`\pikxih`	$\hat{\pi}_k(\mathbf{x}^{(i)})$	pi_k(x^(i)) with hat
`\eps`	ϵ	residual, stochastic
`\epsi`	$\epsilon^{(i)}$	epsilon^i, residual, stochastic
`\epsh`	$\hat{\epsilon}$	residual, estimated
`\yf`	$yf(\mathbf{x})$	y f(x), margin
`\yfi`	$y^{(i)}f\left(\mathbf{x}^{(i)}\right)$	y^i f(x^i), margin
`\Sigmah`	$\hat{\Sigma}$	estimated covariance matrix
`\Sigmahj`	$\hat{\Sigma}_j$	estimated covariance matrix for the j-th class
`\Lyf`	$L\left(y, f\right)$	L(y, f), loss function
`\Lxy`	$L\left(y, f(\mathbf{x})\right)$	L(y, f(x)), loss function
`\Lxyi`	$L\left(y^{(i)}, f\left(\mathbf{x}^{(i)}\right)\right)$	loss of observation
`\Lxyt`	$L\left(y, f(\mathbf{x} \mid \boldsymbol{\theta})\right)$	loss with f parameterized
`\Lxyit`	$L\left(y^{(i)}, f\left(\mathbf{x}^{(i)} \mid \boldsymbol{\theta}\right)\right)$	loss of observation with f parameterized
`\Lxym`	$L\left(y^{(i)}, f\left(\tilde{\boldsymbol{x}}^{(i)} \mid \boldsymbol{\theta}\right)\right)$	loss of observation with f parameterized
`\Lpixy`	$L\left(y, \pi(\mathbf{x})\right)$	loss in classification
`\Lpixyi`	$L\left(y^{(i)}, \pi\left(\mathbf{x}^{(i)}\right)\right)$	loss of observation in classification
`\Lpixyt`	$L\left(y, \pi(\mathbf{x} \mid \boldsymbol{\theta})\right)$	loss with pi parameterized
`\Lpixyit`	$L\left(y^{(i)}, \pi\left(\mathbf{x}^{(i)} \mid \boldsymbol{\theta}\right)\right)$	loss of observation with pi parameterized
`\Lhxy`	$L\left(y, h(\mathbf{x})\right)$	L(y, h(x)), loss function on discrete classes
`\Lr`	$L\left(r\right)$	L(r), loss defined on residual (reg) / margin (classif)
`\risk`	$\mathcal{R}$	R, risk
`\riskf`	$\mathcal{R}(f)$	R(f), risk
`\riskt`	$\mathcal{R}(\boldsymbol{\theta})$	R(theta), risk
`\riske`	$\mathcal{R}_{\text{emp}}$	R_emp, empirical risk w/o factor 1 / n
`\riskeb`	$\bar{\mathcal{R}}_{\text{emp}}$	R_emp, empirical risk w/ factor 1 / n
`\riskef`	$\mathcal{R}_{\text{emp}}(f)$	R_emp(f)
`\risket`	$\mathcal{R}_{\text{emp}}(\boldsymbol{\theta})$	R_emp(theta)
`\riskr`	$\mathcal{R}_{\text{reg}}$	R_reg, regularized risk
`\riskrt`	$\mathcal{R}_{\text{reg}}(\boldsymbol{\theta})$	R_reg(theta)
`\riskrf`	$\mathcal{R}_{\text{reg}}(f)$	R_reg(f)
`\riskrth`	$\hat{\mathcal{R}}_{\text{reg}}(\boldsymbol{\theta})$	hat R_reg(theta)
`\risketh`	$\hat{\mathcal{R}}_{\text{emp}}(\boldsymbol{\theta})$	hat R_emp(theta)
`\LL`	$\mathcal{L}$	L, likelihood
`\LLt`	$\mathcal{L}(\boldsymbol{\theta})$	L(theta), likelihood
`\logl`	ℓ	l, log-likelihood
`\loglt`	$\ell(\boldsymbol{\theta})$	l(theta), log-likelihood

\LS	$\mathfrak{L}$	????????????
\TS	$\mathfrak{T}$	??????????????
\errtrain	$\text{err}_{\text{train}}$	training error
\errtest	err_{test}	training error
\errexp	$\overline{\text{err}_{\text{test}}}$	training error

Back to contents

ml-automl

Macro	Notation	Comment
\lambdav	$\boldsymbol{\lambda}$	lambda vector

Back to contents

ml-bagging

Macro	Notation	Comment
\bl	$b^{[\cdot]}(\mathbf{x})$	baselearner with argument for m
\blm	$b^{[m]}(\mathbf{x})$	baselearner without argument for m
\blmh	$\hat{b}^{[m]}(\mathbf{x})$	estimated base learner
\fM	$f^{[M]}(\mathbf{x})$	ensembled predictor
\fMh	$\hat{f}^{[M]}(\mathbf{x})$	estimated ensembled predictor
\ambifM	$\Delta\left(f^{[M]}(\mathbf{x})\right)$	ambiguity/instability of ensemble

Back to contents

ml-boosting

Macro	Notation	Comment
\fm	$f^{[m]}$	prediction in iteration m
\fmh	$\hat{f}^{[m]}$	prediction in iteration m
\fmd	$f^{[m-1]}$	prediction m-1
\fmdh	$\hat{f}^{[m-1]}$	prediction m-1

`\bmm`	$b^{[m]}$	basemodel m
`\bmmh`	$\hat{b}^{[m]}$	basemodel m with hat
`\betam`	$\beta^{[m]}$	weight of basemodel m
`\betamh`	$\hat{\beta}^{[m]}$	weight of basemodel m with hat
`\betai`	$\beta^{[\cdot]}$	weight of basemodel with argument for m
`\errm`	$\text{err}^{[m]}$	weighted in-sample misclassification rate
`\wm`	$w^{[m]}$	weight vector of basemodel m
`\wmi`	$w^{[m](i)}$	weight of obs i of basemodel m
`\thetam`	$\theta^{[m]}$	parameters of basemodel m
`\thetamh`	$\hat{\theta}^{[m]}$	parameters of basemodel m with hat
`\rmm`	$\tilde{r}^{[m]}$	pseudo residuals
`\rmi`	$\tilde{r}^{[m](i)}$	pseudo residuals
`\Rtm`	$R_t^{[m]}$	terminal-region
`\Tm`	$T^{[m]}$	
`\ctm`	$c_t^{[m]}$	mean, terminal-regions
`\ctmh`	$\hat{c}_t^{[m]}$	mean, terminal-regions with hat
`\ctmt`	$\tilde{c}_t^{[m]}$	mean, terminal-regions
`\fxk`	$f_k(x)$	f_k(x)
`\Lp`	L'	
`\Ldp`	L''	
`\Lpleft`	L'_{left}	

Back to contents

ml-eval

Macro	Notation	Comment
`\ntest`	n_{test}	size of the test set
`\ntrain`	n_{train}	size of the train set
`\ntesti`	$n_{\text{test},\cdot}$	size of the i-th test set
`\ntraini`	$n_{\text{train},\cdot}$	size of the i-th train set
`\Jtrain`	J_{train}	index vector associated to the train data
`\Jtest`	J_{test}	index vector associated to the test data
`\Jtraini`	$J_{\text{train},\cdot}$	index vector associated to the i-th train dataset
`\Jtesti`	$J_{\text{test},\cdot}$	index vector associated to the i-th test dataset
`\Dtraini`	$\mathcal{D}_{\text{train},\cdot}$	D_train,i, i-th training set
`\Dtesti`	$\mathcal{D}_{\text{test},\cdot}$	D_test,i, i-th test set
`\JSpace`	$\{1,\ldots,n\}^{\cdot}$	space of train indices of size m_train
`\JtrainSpace`	$\{1,\ldots,n\}^{n_{\text{train}}}$	space of train indices of size m_train
`\JtestSpace`	$\{1,\ldots,n\}^{n_{\text{test}}}$	space of train indices of size m_test
`\yJ`	$\mathbf{y}_{\cdot}$	output vector associated to index J
`\yJDef`	$\left(y^{(J^{(1)})},\ldots,y^{(J^{(m)})}\right)$	def of the output vector associated to index J
`\JJ`	$\mathcal{J}$	cali-J, set of all splits
`\JJset`	$((J_{\text{train},1},J_{\text{test},1}),\ldots,(J_{\text{train},B},J_{\text{test},B}))$	(Jtrain_1,Jtest_1) ...(Jtrain_B,Jtest_B)
`\GE`	GE	GE
`\GEh`	$\widehat{\text{GE}}$	GE-hat

`\GEfull`	$\mathrm{GE}(\mathcal{I}, \boldsymbol{\lambda}, \cdot, \rho)$	GE(I, lam, ?, rho)
`\GEhlam`	$\widehat{\mathrm{GE}}(\boldsymbol{\lambda})$	GE-hat(lam)
`\GEhlamsubIJrho`	$\widehat{\mathrm{GE}}_{\mathcal{I},\mathcal{J},\rho}(\boldsymbol{\lambda})$	GE-hat_I,J,rho(lam)
`\GEhresa`	$\widehat{\mathrm{GE}}(\mathcal{I}, \mathcal{J}, \rho, \boldsymbol{\lambda})$	GE-hat_I,J,rho(lam)
`\GErhoDef`	$\lim_{n_{\text{test}} \to \infty} \mathbb{E}\left[\rho\left(\mathbf{y}_{J_{\text{test}}}, \boldsymbol{F}_{J_{\text{test}}, \mathcal{I}(\mathcal{D}_{\text{train}}, \boldsymbol{\lambda})}\right)\right]$	GE formal def
`\agr`	agr	aggregate function
`\GEf`	$\mathrm{GE}\left(\hat{f}\right)$	Generalization error of a fitted model
`\GEind`	$GE_n\left(\mathcal{I}_{L,O}\right)$	Generalization error of a fitted model
`\GEnf`	$GE_n\left(\hat{f}_{\cdot}\right)$	Generalization error GE
`\GEhat`	$\widehat{\mathrm{GE}}$	Estimated train error
`\GED`	$\mathrm{GE}_{\mathcal{D}}$	Generalization error GE
`\EGEn`	EGE_n	expected GE
`\EDn`	$\mathbb{E}_{\lvert D \rvert = n}$	expectation wrt data of size n
`\rhoL`	ρ_L	perf. measure derived from pointwise loss function L
`\F`	$\boldsymbol{F}$	matrix of prediction scores
`\Fi`	$\boldsymbol{F}^{(\cdot)}$	i'th row vector of the prediction scores matrix
`\FJ`	$\boldsymbol{F}_{\cdot}$	predscore mat index vector J
`\FJf`	$\boldsymbol{F}_{J,f}$	predscore mat index vector J and model f
`\FJtestfh`	$\boldsymbol{F}_{J_{\text{test}},\hat{f}}$	predscore mat index vector Jtest and model f hat
`\FJtestftrain`	$\boldsymbol{F}_{J_{\text{test}},\mathcal{I}(\mathcal{D}_{\text{train}},\boldsymbol{\lambda})}$	predscore mat index vector Jtest and model f
`\FJtestftraini`	$\boldsymbol{F}_{J_{\text{test},\cdot},\mathcal{I}(\mathcal{D}_{\text{train},\cdot},\boldsymbol{\lambda})}$	predscore mat i-th index vector Jtest and model f
`\FJfDef`	$\left(f(\mathbf{x}^{(J^{(1)})}), \ldots, f(\mathbf{x}^{(J^{(m)})})\right)$	def of predscore mat index vector J and model f
`\preimageRho`	$\bigcup_{m \in \mathbb{N}} \left(\mathcal{Y}^m \times \mathbb{R}^{m \times g}\right)$	Set of all datasets times the hyperparameter space
`\np`	n_+	no. of positive instances
`\nn`	n_-	no. of negative instances
`\rn`	π_-	proportion negative instances
`\rp`	π_+	proportion negative instances
`\tp`	$\#\mathrm{TP}$	true pos
`\fap`	$\#\mathrm{FP}$	false pos (fp taken for partial derivs)
`\tn`	$\#\mathrm{TN}$	true neg
`\fan`	$\#\mathrm{FN}$	false neg

Back to contents

ml-feature-sel

Macro	Notation	Comment
`\xjNull`	x_{j_0}	
`\xjEins`	x_{j_1}	
`\xl`	$\mathbf{x}_l$	
`\pushcode`		

Back to contents

ml-gp

Macro	Notation	Comment
\gp	$\mathcal{GP}\left(m(\mathbf{x}), k\left(\mathbf{x}, \mathbf{x}'\right)\right)$	Gaussian Process Definition
\mvec	$\mathbf{m}$	Gaussian process mean vector
\Kmat	$\mathbf{K}$	estimated base learner
\kstarx	$\mathbf{k}_*(x)$	cov of new obs with x
\ls	ℓ	length-scale

Back to contents

ml-hpo

Macro	Notation	Comment
\Ilam	$\mathcal{I}_{\boldsymbol{\lambda}}$	I_lambda
\lami	$\boldsymbol{\lambda}^{(\cdot)}$	lambda i
\clam	$c(\boldsymbol{\lambda})$	c(lambda)
\clamh	$c(\hat{\boldsymbol{\lambda}})$	c(lambda-hat)
\lams	$\boldsymbol{\lambda}^*$	Theoretical min of c
\lamh	$\hat{\boldsymbol{\lambda}}$	returned lambda of HPO
\LamS	$\tilde{\boldsymbol{\Lambda}}$	search space
\lamp	$\boldsymbol{\lambda}^+$	proposed lambda
\clamp	$c(\boldsymbol{\lambda}^+)$	c of proposed lambda
\archive	$\mathcal{A}$	archive at time step t
\archivet	$\mathcal{A}^{[\cdot]}$	archive at time step t
\tuner	$\mathcal{T}$	tuner
\tunerfull	$\mathcal{T}_{\mathcal{I},\tilde{\boldsymbol{\Lambda}},\rho,\mathcal{J}}$	tuner with inducer, search space, performance measure and resampling strategy
\chlam	$\hat{c}(\boldsymbol{\lambda})$	post mean of SM
\shlam	$\hat{\sigma}(\boldsymbol{\lambda})$	post sd of SM
\vhlam	$\hat{\sigma^2}(\boldsymbol{\lambda})$	post var of SM
\ulam	$u(\boldsymbol{\lambda})$	acquisition function
\lambdaopt	λ^*	Minimum of the black box function Psi
\metadata	$\left\{\left(\boldsymbol{\lambda}^{(i)}, \Psi^{[i]}\right)\right\}$	Metadata for the Gaussian process
\lamvec	$\left(\lambda^{[1]}, \dots, \lambda^{[m_{\text{init}}]}\right)$	Vector of different inputs
\minit	m_{init}	Size of the initial design
\lambu	λ_{budget}	single lambda_budget komponent HP
\lamfid	λ_{fid}	single lambda_budget komponent HP
\lamfidl	$\lambda_{\text{fid}}^{\text{low}}$	single lambda_budget komponent HP
\lamfidu	$\lambda_{\text{fid}}^{\text{upp}}$	single lambda_budget komponent HP
\etahb	η_{HB}	HB multiplier eta
\costs	$\mathcal{C}$	costs
\Celite	$\boldsymbol{\theta}^*$	elite configurations
\instances	$\mathcal{I}$	sequence of instances
\budget	$\mathcal{B}$	computational budget

Back to contents

ml-interpretable

Macro	Notation	Comment
`\fj`	f_j	marginal function f__j, depending on feature j
`\fS`	f_S	marginal function f__S depending on feature set S
`\fC`	f_C	marginal function f__C depending on feature set C
`\fhj`	$\hat{f}_j$	marginal function fh__j, depending on feature j
`\fhS`	$\hat{f}_S$	marginal function fh__S depending on feature set S
`\fhC`	$\hat{f}_C$	marginal function fh__C depending on feature set C
`\XSmat`	$\mathbf{X}_S$	Design matrix subset
`\XCmat`	$\mathbf{X}_C$	Design matrix subset
`\Scupj`	$S \cup \{j\}$	coalition S but without player j
`\Scupk`	$S \cup \{k\}$	coalition S but without player k
`\SsubP`	$S \subseteq P$	coalition S subset of P
`\SsubPnoj`	$S \subseteq P \setminus \{j\}$	coalition S subset of P without player j
`\SsubPnojk`	$S \subseteq P \setminus \{j,k\}$	coalition S subset of P without player k
`\phiij`	$\hat{\phi}_j^{(i)}$	Shapley value for feature j and observation i
`\Gspace`	$\mathcal{G}$	Hypothesis space for surrogate model
`\neigh`	$\phi_{\mathbf{x}}$	Proximity measure
`\zv`	$\mathbf{z}$	Sampled datapoints for surrogate
`\Zspace`	$\mathcal{Z}$	Space of sampled datapoints
`\Gower`	d_G	Gower distance

Back to contents

ml-nn

Macro	Notation	Comment
`\neurons`	$z_1, \dots, z_M$	vector of neurons
`\hidz`	$\mathbf{z}$	vector of hidden activations
`\biasb`	$\mathbf{b}$	bias vector
`\biasc`	c	bias in output
`\wtw`	$\mathbf{w}$	weight vector (general)
`\Wmat`	$\mathbf{W}$	weight vector (general)
`\wtu`	$\mathbf{u}$	weight vector of output neuron
`\Oreg`	$R_{reg}(\theta \vert X, y)$	regularized objective function
`\Ounreg`	$R_{emp}(\theta \vert X, y)$	unconstrained objective function
`\Pen`	$\Omega(\theta)$	penalty
`\Oregweight`	$R_{reg}(w \vert X, y)$	regularized objective function with weight
`\Oweight`	$R_{emp}(w \vert X, y)$	unconstrained objective function with weight
`\Oweighti`	$R_{emp}(w_i \vert X, y)$	unconstrained objective function with weight w__i
`\Oweightopt`	$J(w^* \vert X, y)$	unconstrained objective function withoptimal weight
`\Oopt`	$\hat{J}(\theta \vert X, y)$	optimal objective function
`\Odropout`	$J(\theta, \mu \vert X, y)$	dropout objective function
`\Loss`	$L(y, f(\mathbf{x}, \boldsymbol{\theta}))$	
`\Lmomentumnest`	$L(y^{(i)}, f(x^{(i)}, \boldsymbol{\theta} + \varphi \boldsymbol{\nu}))$	momentum risk
`\Lmomentumtilde`	$L(y^{(i)}, f(x^{(i)}, \tilde{\boldsymbol{\theta}}))$	Nesterov momentum risk

`\Lmomentum`	$L(y^{(i)}, f(x^{(i)}, \boldsymbol{\theta}))$	
`\Hess`	$\mathbf{H}$	
`\nub`	$\boldsymbol{\nu}$	
`\uauto`	$L(x, g(f(x)))$	undercomplete autoencoder objective function
`\dauto`	$L(x, g(f(\tilde{x})))$	denoising autoencoder objective function
`\deltab`	$\boldsymbol{\delta}$	
`\Lossdeltai`	$L(y^{(i)}, f(\mathbf{x}^{(i)} + \boldsymbol{\delta}\|\boldsymbol{\theta}))$	
`\Lossdelta`	$L(y, f(\mathbf{x} + \boldsymbol{\delta}\|\boldsymbol{\theta}))$	

Back to contents

ml-rf

Macro	Notation	Comment
`\betaM`	$\beta^{[M]}$	baselearner with argument for M
`\betai`	$\beta^{[1]}$	baselearner with argument for 1

Back to contents

ml-survival

Macro	Notation	Comment
`\Ti`	$T^{(\cdot)}$	??
`\Ci`	$C^{(\cdot)}$	??
`\oi`	$o^{(\cdot)}$	??
`\ti`	$t^{(\cdot)}$	??
`\deltai`	$\delta^{(\cdot)}$	
`\Lxdi`	$L\left(\boldsymbol{\delta}, f(\mathbf{x})\right)$	

Back to contents

ml-svm

Macro	Notation	Comment
\sv	SV	supportvectors
\HS	Φ	H, hilbertspace
\slvec	$\left(\zeta^{(1)}, \zeta^{(n)}\right)$	slack variables (SVM)
\sli	$\zeta^{(i)}$	slack variable (SVM)
\alphah	$\hat{\alpha}$	alpha-hat
\alphav	$\boldsymbol{\alpha}$	vector alpha (bold)
\alphavh	$\hat{\boldsymbol{\alpha}}$	vector alpha-hat
\phix	$\phi(\mathbf{x})$	\phi(x)
\phixt	$\phi(\tilde{\mathbf{x}})$	\phi(x-tilde)

Back to contents

ml-trees

Macro	Notation	Comment
\Np	$\mathcal{N}$	(Parent) node N
\Npk	$\mathcal{N}_k$	node N_k
\Nl	$\mathcal{N}_1$	Left node N_1
\Nr	$\mathcal{N}_2$	Right node N_2
\pikN	$\pi_k^{(\mathcal{N})}$	class probability node N
\pikNh	$\hat{\pi}.^{(\mathcal{N})}$	estimated class probability node N
\pikNlh	$\hat{\pi}.^{(\mathcal{N}_1)}$	
\pikNrh	$\hat{\pi}.^{(\mathcal{N}_2)}$	

Back to contents

probmodel

Macro	Notation	Comment
\muk	$\boldsymbol{\mu_k}$	

Back to contents