`latex-math` Macros

compiled: 2024-10-21

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
`\setzo`	$\{0, 1\}$	set 0, 1
`\setmp`	$\{-1, +1\}$	set -1, 1
`\unitint`	$[0, 1]$	unit interval
`\xt`	$\tilde{x}$	x tilde
`\argmax`	$arg\,max$	argmax
`\argmin`	$arg\,min$	argmin
`\argminlim`	$\operatorname{arg\,min}$	argmax with limits
`\argmaxlim`	$\operatorname{arg\,max}$	argmin with limits
`\sign`	sign	sign, signum
`\I`	$\mathbb{I}$	I, indicator
`\order`	$\mathcal{O}$	O, order
`\bigO`	$\mathcal{O}$	Big-O Landau
`\littleo`	o	Little-o Landau
`\pd`	$\frac{\partial \#1}{\partial \#2}$	partial derivative
`\floorlr`	$\lfloor \#1 \rfloor$	floor
`\ceillr`	$\lceil \#1 \rceil$	ceiling
`\indep`	$\perp\!\!\!\perp$	independence symbol
`\sumin`	$\sum\limits_{i=1}^{n}$	summation from i=1 to n
`\sumim`	$\sum\limits_{i=1}^{m}$	summation from i=1 to m
`\sumjn`	$\sum\limits_{j=1}^{n}$	summation from j=1 to p
`\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
`\summM`	$\sum\limits_{m=1}^{M}$	summation from m=1 to M

`\meanin`	$\frac{1}{n}\sum_{i=1}^{n}$	mean from i=1 to n
`\meanim`	$\frac{1}{m}\sum_{i=1}^{m}$	mean from i=1 to n
`\meankg`	$\frac{1}{g}\sum_{k=1}^{g}$	mean from k=1 to g
`\meanmM`	$\frac{1}{M}\sum_{m=1}^{M}$	mean from m=1 to M
`\prodin`	$\prod_{i=1}^{n}$	product from i=1 to n
`\prodkg`	$\prod_{k=1}^{g}$	product from k=1 to g
`\prodjp`	$\prod_{j=1}^{p}$	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`	$\langle \#1, \#2 \rangle$	<.,.>, scalarproduct
`\mat`	$\begin{pmatrix}\#1\end{pmatrix}$	short pmatrix command
`\Amat`	$\mathbf{A}$	matrix A
`\Deltab`	$\boldsymbol{\Delta}$	error term for vectors
`\P`	$\mathbb{P}$	P, probability
`\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`	$\overset{\#1}{\sim}$	... is distributed as ...

Back to contents

basic-ml

Macro	Notation	Comment
`\Xspace`	$\mathcal{X}$	X, input space
`\Yspace`	$\mathcal{Y}$	Y, output space
`\Zspace`	$\mathcal{Z}$	Z, space of sampled datapoints
`\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
`\xv`	$\mathbf{x}$	vector x (bold)
`\xtil`	$\tilde{\mathbf{x}}$	vector x-tilde (bold)
`\yv`	$\mathbf{y}$	vector y (bold)
`\xy`	$(\mathbf{x}, y)$	observation (x, y)
`\xvec`	$(x_1, \ldots, x_p)^\top$	(x1, ..., xp)
`\Xmat`	$\mathbf{X}$	Design matrix
`\allDatasets`	$\mathbb{D}$	The set of all datasets
`\allDatasetsn`	$\mathbb{D}_n$	The set of all datasets of size n
`\D`	$\mathcal{D}$	D, data
`\Dn`	$\mathcal{D}_n$	D_n, data of size n
`\Dtrain`	$\mathcal{D}_{\text{train}}$	D_train, training set
`\Dtest`	$\mathcal{D}_{\text{test}}$	D_test, test set
`\xyi`	$\left(\mathbf{x}^{(\#1)}, y^{(\#1)}\right)$	(x^i, y^i), i-th observation
`\Dset`	$\left(\left(\mathbf{x}^{(1)}, y^{(1)}\right), \ldots, \left(\mathbf{x}^{(n)}, y^{(n)}\right)\right)$	{(x1,y1)), ..., (xn,yn)}, data
`\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
`\xdat`	$\left\{\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(n)}\right\}$	{x1, ..., xn}, input data
`\ydat`	$\left\{\mathbf{y}^{(1)}, \ldots, \mathbf{y}^{(n)}\right\}$	{y1, ..., yn}, input data
`\yvec`	$\left(y^{(1)}, \ldots, y^{(n)}\right)^\top$	(y1, ..., yn), vector of outcomes
`\greekxi`	$\mathbf{x}^{(i)}$	Greek letter xi
`\xi`	$\mathbf{x}^{(\#1)}$	x^i, i-th observed value of x
`\yi`	$y^{(\#1)}$	y^i, i-th observed value of y
`\xivec`	$\left(x_1^{(i)}, \ldots, x_p^{(i)}\right)^\top$	(x1^i, ..., xp^i), i-th observation vector
`\xj`	$\mathbf{x}_j$	x_j, j-th feature
`\xjvec`	$\left(x_j^{(1)}, \ldots, x_j^{(n)}\right)^\top$	(x^1_j, ..., x^n_j), j-th feature vector
`\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`	$\left(\bigcup_{n \in \mathbb{N}} (\mathcal{X} \times \mathcal{Y})^n\right) \times \boldsymbol{\Lambda}$	Set of all datasets times the hyperparameter space
`\preimageInducerShort`	$\mathbb{D} \times \boldsymbol{\Lambda}$	Set of all datasets times the hyperparameter space
`\ind`	$\mathcal{I}$	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
`\fdomains`	$f: \mathcal{X} \rightarrow \mathbb{R}^g$	f with domain and co-domain
`\Hspace`	$\mathcal{H}$	hypothesis space where f is from
`\fbayes`	f^*	Bayes-optimal model
`\fxbayes`	$f^*(\mathbf{x})$	Bayes-optimal model
`\fkx`	$f_{\#1}(\mathbf{x})$	f_j(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
`\fhDnlam`	$\hat{f}_{\mathcal{D}_n, \boldsymbol{\lambda}}$	model learned on Dn with hp lambda
`\fhDlam`	$\hat{f}_{\mathcal{D}, \boldsymbol{\lambda}}$	model learned on D with hp lambda
`\fhDnlams`	$\hat{f}_{\mathcal{D}_n, \boldsymbol{\lambda}^*}$	model learned on Dn with optimal hp lambda
`\fhDlams`	$\hat{f}_{\mathcal{D}, \boldsymbol{\lambda}^*}$	model learned on D with optimal hp lambda
`\hx`	$h(\mathbf{x})$	h(x), discrete prediction function
`\hh`	$\hat{h}$	h hat
`\hxh`	$\hat{h}(\mathbf{x})$	hhat(x)
`\hxt`	$h(\mathbf{x} \| \boldsymbol{\theta})$	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)
`\hbayes`	h^*	Bayes-optimal classification model
`\hxbayes`	$h^*(\mathbf{x})$	Bayes-optimal classification model
`\yh`	$\hat{y}$	yhat for prediction of target
`\yih`	$\hat{y}^{(i)}$	yhat^(i) for prediction of ith targiet
`\resi`	$y^{(i)} - \hat{y}^{(i)}$	
`\thetah`	$\hat{\theta}$	theta hat
`\thetav`	$\boldsymbol{\theta}$	theta vector
`\thetavh`	$\hat{\boldsymbol{\theta}}$	theta vector hat
`\thetat`	$\boldsymbol{\theta}^{[\#1]}$	theta^[t] in optimization
`\thetatn`	$\boldsymbol{\theta}^{[\#1+1]}$	theta^[t+1] in optimization
`\thetahDnlam`	$\hat{\boldsymbol{\theta}}_{\mathcal{D}_n, \boldsymbol{\lambda}}$	theta learned on Dn with hp lambda
`\thetahDlam`	$\hat{\boldsymbol{\theta}}_{\mathcal{D}, \boldsymbol{\lambda}}$	theta learned on D with hp lambda
`\mint`	$\min_{\boldsymbol{\theta} \in \Theta}$	min problem theta
`\argmint`	$\arg\min_{\boldsymbol{\theta} \in \Theta}$	argmin theta
`\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`	$\pi\left(\mathbf{x}^{(\#1)} \mid \boldsymbol{\theta}\right)$	pi(x^i|theta), pdf of x given theta

`\pixii`	$\pi\left(\mathbf{x}^{(\#1)}\right)$	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`	$p(\mathbf{x} \vert y = \#1)$	p(x | y = k)
`\lpdfxyk`	$\log p(\mathbf{x} \vert y = \#1)$	log p(x | y = k)
`\pdfxiyk`	$p\left(\mathbf{x}^{(i)} \vert y = \#1\right)$	p(x^i | y = k)
`\pik`	$\pi_{\#1}$	pi_k, prior
`\lpik`	$\log \pi_{\#1}$	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
`\postk`	$\mathbb{P}(y = \#1 \mid \mathbf{x})$	P(y = k | y), post. prob for y=k
`\pidomains`	$\pi : \mathcal{X} \to [0, 1]$	pi with domain and co-domain
`\pibayes`	π^*	Bayes-optimal classification model
`\pixbayes`	$\pi^*(\mathbf{x})$	Bayes-optimal classification model
`\pix`	$\pi(\mathbf{x})$	pi(x), P(y = 1 | x)
`\piv`	$\boldsymbol{\pi}$	pi, bold, as vector
`\pikx`	$\pi_{\#1}(\mathbf{x})$	pi_k(x), P(y = k | x)
`\pikxt`	$\pi_{\#1}(\mathbf{x} \mid \boldsymbol{\theta})$	pi_k(x | theta), P(y = k | x, theta)
`\pixh`	$\hat\pi(\mathbf{x})$	pi(x) hat, P(y = 1 | x) hat
`\pikxh`	$\hat\pi_{\#1}(\mathbf{x})$	pi_k(x) hat, P(y = k | x) hat
`\pixih`	$\hat\pi(\mathbf{x}^{(i)})$	pi(x^(i)) with hat
`\pikxih`	$\hat\pi_{\#1}(\mathbf{x}^{(i)})$	pi_k(x^(i)) with hat
`\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)
`\bayesrulek`	$\frac{\mathbb{P}(\mathbf{x} \vert y=\#1)\mathbb{P}(y=\#1)}{\mathbb{P}(\mathbf{x})}$	Bayes rule
`\muk`	$\boldsymbol{\mu}_{\#1}$	mean vector of class-k Gaussian (discr analysis)
`\eps`	ϵ	residual, stochastic
`\epsv`	$\boldsymbol{\epsilon}$	residual, stochastic, as vector
`\epsi`	$\epsilon^{(i)}$	epsilon^i, residual, stochastic
`\epsh`	$\hat\epsilon$	residual, estimated
`\epsvh`	$\hat{\boldsymbol{\epsilon}}$	residual, estimated, vector
`\yf`	$y f(\mathbf{x})$	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`	$L(y, f)$	L(y, f), loss function
`\Lypi`	$L(y, \pi)$	L(y, pi), loss function
`\Lxy`	$L(y, f(\mathbf{x}))$	L(y, f(x)), loss function
`\Lxyi`	$L\left(y^{(i)}, f\left(\mathbf{x}^{(i)}\right)\right)$	loss of observation
`\Lxyt`	$L(y, f(\mathbf{x} \mid \boldsymbol{\theta}))$	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
`\Lpiv`	$L\left(y, \boldsymbol{\pi}\right)$	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)
`\lone`	$\lvert y - f(\mathbf{x})\rvert$	L1 loss
`\ltwo`	$\left(y - f(\mathbf{x})\right)^2$	L2 loss
`\lbernoullimp`	$\ln(1 + \exp(-y \cdot f(\mathbf{x})))$	Bernoulli loss for -1, +1 encoding
`\lbernoullizo`	$-y \cdot f(\mathbf{x}) + \log(1 + \exp(f(\mathbf{x})))$	Bernoulli loss for 0, 1 encoding
`\lcrossent`	$-y \log\left(\pi(\mathbf{x})\right) - (1-y)\log\left(1 - \pi(\mathbf{x})\right)$	cross-entropy loss
`\lbrier`	$\left(\pi(\mathbf{x}) - y\right)^2$	Brier score
`\risk`	$\mathcal{R}$	R, risk
`\riskbayes`	$\mathcal{R}^*$	
`\riskf`	$\mathcal{R}(f)$	R(f), risk
`\riskdef`	$\mathbb{E}_{y\mid\mathbf{x}}\left(L\left(y, f(\mathbf{x})\right)\right)$	risk def (expected loss)
`\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
`\LLtx`	$\mathcal{L}(\boldsymbol{\theta}\vert\mathbf{x})$	L(theta|x), likelihood
`\logl`	ℓ	l, log-likelihood
`\loglt`	$\ell(\boldsymbol{\theta})$	l(theta), log-likelihood
`\logltx`	$\ell(\boldsymbol{\theta}\vert\mathbf{x})$	l(theta|x), log-likelihood
`\errtrain`	$\text{err}_{\text{train}}$	training error
`\errtest`	err_{test}	test error
`\errexp`	$\overline{\text{err}_{\text{test}}}$	avg training error
`\thx`	$\boldsymbol{\theta}^\top \mathbf{x}$	linear model
`\olsest`	$(\mathbf{X}^\top \mathbf{X})^{-1}\mathbf{X}^\top \mathbf{y}$	OLS estimator in LM

Back to contents

ml-ensembles

Macro	Notation	Comment
`\bl`	$b^{[\#1]}$	baselearner, default m
`\blh`	$\hat{b}^{[\#1]}$	estimated base learner, default m
`\blx`	$b^{[\#1]}(\mathbf{x})$	baselearner, default m
`\blf`	$f^{[\#1]}$	baselearner: scores, default m
`\blfh`	$\hat{f}^{[\#1]}$	estimated baselearner: scores, default m
`\blfhx`	$\hat{f}^{[\#1]}(\mathbf{x})$	estimated baselearner: scores of x, default m
`\bll`	$h^{[\#1]}$	baselearner: hard labels, default m
`\bllh`	$\hat{h}^{[\#1]}$	estimated baselearner: hard labels, default m
`\bllhx`	$\hat{h}^{[\#1]}(\mathbf{x})$	estimated baselearner: hard labels of x, default m
`\blp`	$\pi^{[\#1]}$	baselearner: probabilities, default m
`\blph`	$\hat{\pi}^{[\#1]}$	estimated baselearner: probabilities, default m
`\blphxk`	$\hat{\pi}_k^{[\#1]}(\mathbf{x})$	estimated baselearner: probabilities of x for class k, default m
`\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
`\betam`	$\beta^{[\#1]}$	weight of basemodel m
`\betamh`	$\hat{\beta}^{[\#1]}$	weight of basemodel m with hat
`\betaM`	$\beta^{[M]}$	last baselearner
`\ib`	IB	In-Bag (IB)
`\ibm`	$\mathrm{IB}^{[m]}$	In-Bag (IB) for m-th bootstrap
`\oob`	OOB	Out-of-Bag (OOB)
`\oobm`	$\mathrm{OOB}^{[m]}$	Out-of-Bag (OOB) for m-th bootstrap
`\fm`	$f^{[\#1]}$	prediction in iteration m
`\fmh`	$\hat{f}^{[\#1]}$	prediction in iteration m
`\fmd`	$f^{[\#1-1]}$	prediction m-1
`\fmdh`	$\hat{f}^{[\#1-1]}$	prediction m-1
`\errm`	$\mathrm{err}^{[\#1]}$	weighted in-sample misclassification rate
`\wm`	$w^{[\#1]}$	weight vector of basemodel m
`\wmi`	$w^{[\#1](i)}$	weight of obs i of basemodel m
`\thetam`	$\boldsymbol{\theta}^{[\#1]}$	parameters of basemodel m
`\thetamh`	$\hat{\boldsymbol{\theta}}^{[\#1]}$	parameters of basemodel m with hat
`\blxt`	$b(\mathbf{x}, \boldsymbol{\theta}^{[\#1]})$	baselearner, default m
`\ens`	$\sum_{m=1}^M \beta^{[m]} b(\mathbf{x}, \boldsymbol{\theta}^{[m]})$	ensemble
`\rmm`	$\tilde{r}^{[\#1]}$	pseudo residuals
`\rmi`	$\tilde{r}^{[\#1](i)}$	pseudo residuals
`\Rtm`	$R_t^{[\#1]}$	terminal-region
`\Tm`	$T^{[\#1]}$	terminal-region
`\ctm`	$c_t^{[\#1]}$	mean, terminal-regions
`\ctmh`	$\hat{c}_t^{[\#1]}$	mean, terminal-regions with hat
`\ctmt`	$\tilde{c}_t^{[\#1]}$	mean, terminal-regions

`\Lp`	L'	
`\Ldp`	L''	
`\Lpleft`	L'_{left}	
`\ts`	$\boldsymbol{\theta}^\star$	theta*
`\bljt`	$b^{[j]}(\mathbf{x}, \boldsymbol{\theta})$	BL j with theta
`\bljts`	$b^{[j]}(\mathbf{x}, \boldsymbol{\theta}^\star)$	BL j with theta*

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`	$n_{\text{test},\#1}$	size of the i-th test set
`\ntraini`	$n_{\text{train},\#1}$	size of the i-th train set
`\Jtrain`	J_{train}	index vector train data
`\Jtest`	J_{test}	index vector test data
`\Jtraini`	$J_{\text{train},\#1}$	index vector i-th train dataset
`\Jtesti`	$J_{\text{test},\#1}$	index vector i-th test dataset
`\Dtraini`	$\mathcal{D}_{\text{train},\#1}$	D_train,i, i-th training set
`\Dtesti`	$\mathcal{D}_{\text{test},\#1}$	D_test,i, i-th test set
`\JSpace`	$\{1, \ldots, n\}^{\#1}$	space of train indices of size n_train
`\JtrainSpace`	$\{1, \ldots, n\}^{n_{\text{train}}}$	space of train indices of size n_train
`\JtestSpace`	$\{1, \ldots, n\}^{n_{\text{test}}}$	space of train indices of size n_test
`\yJ`	$\mathbf{y}_{\#1}$	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)
`\Itrainlam`	$\mathcal{I}(\mathcal{D}_{\text{train}}, \boldsymbol{\lambda})$	
`\GE`	GE	GE
`\GEh`	$\widehat{\text{GE}}$	GE-hat
`\GEfull`	$\text{GE}(\mathcal{I}, \boldsymbol{\lambda}, \#1, \rho)$	GE full
`\GEhholdout`	$\widehat{\text{GE}}_{J_{\text{train}}, J_{\text{test}}}(\mathcal{I}, \boldsymbol{\lambda}, \lvert J_{\text{train}} \rvert, \rho)$	GE hat holdout
`\GEhholdouti`	$\widehat{\text{GE}}_{J_{\text{train},\#1}, J_{\text{test},\#1}}(\mathcal{I}, \boldsymbol{\lambda}, \lvert J_{\text{train},\#1} \rvert, \rho)$	GE hat holdout i-th set
`\GEhlam`	$\widehat{\text{GE}}(\boldsymbol{\lambda})$	GE-hat(lam)
`\GEhlamsubIJrho`	$\widehat{\text{GE}}_{\mathcal{I}, \mathcal{J}, \rho}(\boldsymbol{\lambda})$	GE-hat_I,J,rho(lam)
`\GEhresa`	$\widehat{\text{GE}}(\mathcal{I}, \mathcal{J}, \rho, \boldsymbol{\lambda})$	GE-hat_I,J,rho(lam)
`\GErhoDef`	$\lim_{n_{\text{test}} \to \infty} \mathbb{E}_{\mathcal{D}_{\text{train}}, \mathcal{D}_{\text{test}} \sim \mathbb{P}_{xy}} \left[\rho\left(\mathbf{y}_{J_{\text{test}}}, \boldsymbol{F}_{J_{\text{test}}, \mathcal{I}(\mathcal{D}_{\text{train}}, \boldsymbol{\lambda})}\right)\right]$	GE formal def
`\agr`	agr	aggregate function
`\GEf`	$\text{GE}\left(\hat{f}\right)$	GE of a fitted model
`\GEfh`	$\widehat{\text{GE}}\left(\hat{f}\right)$	GEh of a fitted model
`\GEfL`	$\text{GE}\left(\hat{f}, L\right)$	GE of a fitted model wrt loss L
`\Lyfhx`	$L\left(y, \hat{f}(\mathbf{x})\right)$	pointwise loss of fitted model
`\GEnf`	$GE_n\left(\hat{f}_{\#1}\right)$	GE of a fitted model
`\GEind`	$GE_n\left(\mathcal{I}_{L,O}\right)$	GE of inducer
`\GED`	$\text{GE}_{\mathcal{D}}$	GE indexed with data
`\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
`\F`	$\boldsymbol{F}$	matrix of prediction scores

`\Fi`	$\boldsymbol{F}^{(\#1)}$	i-th row vector of the predscore mat
`\FJ`	$\boldsymbol{F}_{\#1}$	predscore mat idxvec J
`\FJf`	$\boldsymbol{F}_{J,f}$	predscore mat idxvec J and model f
`\FJtestfh`	$\boldsymbol{F}_{J_{\mathrm{test}},\hat{f}}$	predscore mat idxvec Jtest and model f hat
`\FJtestftrain`	$\boldsymbol{F}_{J_{\mathrm{test}},\mathcal{I}(\mathcal{D}_{\mathrm{train}},\boldsymbol{\lambda})}$	predscore mat idxvec Jtest and model f
`\FJtestftraini`	$\boldsymbol{F}_{J_{\mathrm{test},\#1},\mathcal{I}(\mathcal{D}_{\mathrm{train},\#1},\boldsymbol{\lambda})}$	predscore mat i-th idxvec Jtest and model f
`\FJfDef`	$\left(f(\mathbf{x}^{(J^{(1)})}),\ldots,f(\mathbf{x}^{(J^{(m)})})\right)$	def of predscore mat idxvec 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 HP 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
`\fvec`	$\left[f\left(\mathbf{x}^{(1)}\right), \ldots, f\left(\mathbf{x}^{(n)}\right)\right]$	function vector
`\fv`	$\mathbf{f}$	function vector
`\kv`	$\mathbf{k}$	cov matrix partition
`\kxxp`	$k\left(\mathbf{x}, \mathbf{x}'\right)$	cov of x, x'
`\kxij`	$k\left(\mathbf{x}^{(i)}, \mathbf{x}^{(j)}\right)$	cov of x_i, x_j
`\mv`	$\mathbf{m}$	GP mean vector
`\Kmat`	$\mathbf{K}$	GP cov matrix
`\gaussmk`	$\mathcal{N}(\mathbf{m}, \mathbf{K})$	Gaussian w/ mean vec, cov mat
`\gp`	$\mathcal{GP}\left(m(\mathbf{x}), k\left(\mathbf{x}, \mathbf{x}'\right)\right)$	Gaussian Process Definition
`\ls`	ℓ	length-scale
`\sqexpkernel`	$\exp\left(-\frac{\lVert\mathbf{x}-\mathbf{x}'\rVert^2}{2\ell^2}\right)$	squared exponential kernel
`\fstarvec`	$\left[f\left(\mathbf{x}_*^{(1)}\right), \ldots, f\left(\mathbf{x}_*^{(m)}\right)\right]$	pred function vector
`\kstar`	$\mathbf{k}_*$	cov of new obs with x
`\kstarstar`	$\mathbf{k}_{**}$	cov of new obs
`\Kstar`	$\mathbf{K}_*$	cov mat of new obs with x
`\Kstarstar`	$\mathbf{K}_{**}$	cov mat of new obs
`\preddistsingle`	$f_* \mid \mathbf{x}_*, \mathbf{X}, \mathbf{f}$	predictive distribution for single pred
`\preddistdefsingle`	$\mathcal{N}(\mathbf{k}_*^\top\mathbf{K}^{-1}\mathbf{f}, \mathbf{k}_{**} - \mathbf{k}_*^\top\mathbf{K}^{-1}\mathbf{k}_*)$	Gaussian distribution for single pred
`\preddist`	$f_* \mid \mathbf{X}_*, \mathbf{X}, \mathbf{f}$	predictive distribution
`\preddistdef`	$\mathcal{N}(\mathbf{K}_*^\top\mathbf{K}^{-1}\mathbf{f}, \mathbf{K}_{**} - \mathbf{K}_*^\top\mathbf{K}^{-1}\mathbf{K}_*)$	Gaussian predictive distribution

Back to contents

ml-hpo

Macro	Notation	Comment
`\Ilam`	$\mathcal{I}_{\boldsymbol{\lambda}}$	inducer with HP
`\LamS`	$\tilde{\boldsymbol{\Lambda}}$	search space
`\lami`	$\boldsymbol{\lambda}^{(\#1)}$	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
`\lamp`	$\boldsymbol{\lambda}^+$	proposed lambda
`\clamp`	$c(\boldsymbol{\lambda}^+)$	c of proposed lambda
`\archive`	$\mathcal{A}$	archive
`\archivet`	$\mathcal{A}^{[\#1]}$	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, perf measure, 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`	$\left(\lambda^{[1]}, \ldots, \lambda^{[m_{\text{init}}]}\right)$	vector of different inputs
`\minit`	m_{init}	size of the initial design
`\lambu`	λ_{budget}	single lambda_budget component HP
`\lamfid`	λ_{fid}	single lambda fidelity
`\lamfidl`	$\lambda_{\text{fid}}^{\text{low}}$	single lambda fidelity lower
`\lamfidu`	$\lambda_{\text{fid}}^{\text{upp}}$	single lambda fidelity upper
`\etahb`	η_{HB}	HB multiplier eta

Back to contents

ml-infotheory

Macro	Notation	Comment
`\entx`	$-\sum_{x\in\mathcal{X}} p(x)\cdot\log p(x)$	entropy of x
`\dentx`	$-\int_{\mathcal{X}} f(x)\cdot\log f(x)dx$	diff entropy of x
`\jentxy`	$-\sum_{x\in\mathcal{X}} p(x,y)\cdot\log p(x,y)$	joint entropy of x, y
`\jdentxy`	$-\int_{\mathcal{X},\mathcal{Y}} f(x,y)\cdot\log f(x,y)dxdy$	joint diff entropy of x, y
`\centyx`	$-\sum_{x\in\mathcal{X}} p(x)\sum_{y\in\mathcal{Y}} p(y\|x)\cdot\log p(y\|x)$	cond entropy y|x
`\cdentyx`	$-\int_{\mathcal{X},\mathcal{Y}} f(x,y)\cdot\log f(y\|x)dxdy$	cond diff entropy y|x
`\xentpq`	$-\sum_{x\in\mathcal{X}} p(x)\cdot\log q(x)$	cross-entropy of p, q
`\kldpq`	$D_{KL}(p\\|q)$	KLD between p and q
`\kldpqt`	$D_{KL}(p\\|q_{\boldsymbol{\theta}})$	KLD divergence between p and parameterized q
`\explogpq`	$\mathbb{E}_p\left[\log\frac{p(X)}{q(X)}\right]$	expected LLR of p, q (def KLD)
`\sumlogpq`	$\sum_{x\in\mathcal{X}} p(x)\cdot\log\frac{p(x)}{q(x)}$	expected LLR of p, q (def KLD)

Back to contents

ml-interpretable

Macro	Notation	Comment
`\pert`	$\tilde{\#1}^{\#2\|\#3}$	command to express that for #1 the subset #2 was perturbed given subset #3
`\fj`	f_j	marginal function f_j, depending on feature j
`\fnj`	f_{-j}	marginal function f_{-j}, depending on all features but 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
`\fhnj`	$\hat{f}_{-j}$	marginal function fh_{-j}, depending on all features but 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
`\Xnj`	$\mathbf{X}_{-j}$	Design matrix subset -j = {1, .., j-1, j+1, ..., p}
`\fhice`	$\hat{f}_{\#1,ICE}$	ICE function
`\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
`\Gower`	d_G	Gower distance

Back to contents

ml-mbo

Macro	Notation	Comment
`\xvsi`	$\mathbf{x}^{[\#1]}$	x at iteration i
`\ysi`	$y^{[\#1]}$	y at iteration i
`\Dt`	$\mathcal{D}^{[\#1]}$	archive at iteration t
`\Dts`	$\mathcal{D}^{[t]} = \{(\mathbf{x}^{[i]}, y^{[i]})\}_{i=1,\dots,t}$	archive at iteration t fully
`\fh`	$\hat{s}$	surrogate mean
`\sh`	$\hat{s}$	surrogate se
`\fmin`	$f_{\min}$	current best

Back to contents

ml-multitarget

Macro	Notation	Comment
`\Tspace`	$\mathcal{T}$	
`\tv`	$\mathbf{t}$	
`\tim`	$\mathbf{t}_m^{(i)}$	
`\yim`	$y_m^{(i)}$	

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`	$R_{reg}(\theta\|X, y)$	regularized objective function
`\Ounreg`	$R_{emp}(\theta\|X, y)$	unconstrained objective function
`\Pen`	$\Omega(\theta)$	penalty
`\Oregweight`	$R_{reg}(w\|X, y)$	regularized objective function with weight
`\Oweight`	$R_{emp}(w\|X, y)$	unconstrained objective function with weight
`\Oweighti`	$R_{emp}(w_i\|X, y)$	unconstrained objective function with weight w_i
`\Oweightopt`	$J(w^*\|X, y)$	unconstrained objective function withoptimal weight
`\Oopt`	$\hat{J}(\theta\|X, y)$	optimal objective function
`\Odropout`	$J(\theta, \mu\|X, y)$	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-online

Macro	Notation	Comment
`\Aspace`	$\mathcal{A}$	
`\norm`	$\|\|\#1\|\|_2$	
`\llin`	$L^{\mathtt{lin}}$	
`\lzeroone`	L^{0-1}	
`\lhinge`	$L^{\mathtt{hinge}}$	
`\lexphinge`	$\widetilde{L^{\mathtt{hinge}}}$	
`\lconv`	$L^{\mathtt{conv}}$	
`\FTL`	`FTL`	
`\FTRL`	`FTRL`	
`\OGD`	`OGD`	
`\EWA`	`EWA`	
`\REWA`	`REWA`	
`\EXPthree`	`EXP3`	
`\EXPthreep`	`EXP3P`	
`\reg`	ψ	
`\Algo`	`Algo`	

Back to contents

ml-regu

Macro	Notation	Comment
`\thetas`	$\boldsymbol{\theta}^*$	theta star
`\thetaridge`	$\boldsymbol{\theta}_{\mathrm{ridge}}$	theta (RIDGE)
`\thetalasso`	$\boldsymbol{\theta}_{\mathrm{LASSO}}$	theta (LASSO)
`\thetaols`	$\boldsymbol{\theta}_{\mathrm{OLS}}$	theta (RIDGE)

Back to contents

ml-survival

Macro	Notation	Comment
`\Ti`	$T^{(\#1)}$	??
`\Ci`	$C^{(\#1)}$	??
`\oi`	$o^{(\#1)}$	??
`\ti`	$t^{(\#1)}$	??
`\deltai`	$\delta^{(\#1)}$	
`\Lxdi`	$L\left(\boldsymbol{\delta}, f(\mathbf{x})\right)$	

Back to contents

ml-svm

Macro	Notation	Comment
`\sv`	SV	supportvectors
`\sl`	ζ	slack variable
`\slvec`	$\left(\zeta^{(1)}, \zeta^{(n)}\right)$	slack variable vector
`\sli`	$\zeta^{(\#1)}$	i-th slack variable
`\scptxi`	$\left\langle\boldsymbol{\theta}, \mathbf{x}^{(i)}\right\rangle$	scalar prodct of theta and xi
`\svmhplane`	$y^{(i)}\left(\left\langle\boldsymbol{\theta}, \mathbf{x}^{(i)}\right\rangle + \theta_0\right)$	SVM hyperplane (normalized)
`\alphah`	$\hat{\alpha}$	alpha-hat (basis fun coefficients)
`\alphav`	$\boldsymbol{\alpha}$	vector alpha (bold) (basis fun coefficients)
`\alphavh`	$\hat{\boldsymbol{\alpha}}$	vector alpha-hat (basis fun coefficients)
`\dualobj`	$\sum\limits_{i=1}^{n} \alpha_i - \frac{1}{2} \sum\limits_{i=1}^{n} \sum\limits_{j=1}^{n} \alpha_i \alpha_j y^{(i)} y^{(j)} \left\langle\mathbf{x}^{(i)}, \mathbf{x}^{(j)}\right\rangle$	min objective in lin svm dual
`\HS`	Φ	H, hilbertspace
`\phix`	$\phi(\mathbf{x})$	feature map x
`\phixt`	$\phi(\tilde{\mathbf{x}})$	feature map x tilde
`\kxxt`	$k(\mathbf{x}, \tilde{\mathbf{x}})$	kernel fun x, x tilde
`\scptxifm`	$\left\langle\boldsymbol{\theta}, \phi(\mathbf{x}^{(i)})\right\rangle$	scalar prodct of theta and xi

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`	$\pi_{\#1}^{(\mathcal{N})}$	class probability node N
`\pikNh`	$\hat{\pi}_{\#1}^{(\mathcal{N})}$	estimated class probability node N
`\pikNlh`	$\hat{\pi}_{\#1}^{(\mathcal{N}_1)}$	estimated class probability left node
`\pikNrh`	$\hat{\pi}_{\#1}^{(\mathcal{N}_2)}$	estimated class probability right node

Back to contents