Algorithm	Time complexity	Description
Hash table search	1	
Array	N	N: Number of elements
Binary tree	Log N	N: Number of elements
KD tree	N	N: Number of elements
Recursion with one element reduction	N^2	N: Number of elements
Recursion with halving reduction	LogN	N: Number of elements
Recursion with halving reduction	N	N: Number of elements
Naïve Bayes	N.D.C	N: Number of data points D: Number of features (dimension) C: Number of classes
Nearest neighbors search	M.logk.N.LogN	M: number of features k: number of neighbors N: number of observations
Matrix multiplication (m, n) x (n, d)	m.n.d	m, n, d matrices resp. dimension
Matrix multiplication (n, n)	n^3	n matrix dimension
Matrix multiplication (n, n) Strassen	$n^{2.8}$	n matrix dimension
Partial eigenvalues extraction (n, n)	$e.N^2$	e: number of eigenvalues N: number of observations
Complete eigenvalues extraction (n, n)	N^3	N: number of observations
Minimum spanning tree Prim linear queue	V^2	V: number vertices
Minimum spanning tree Prim binary heap	(E + V).LogV	E: number of edges V: number vertices
Minimum spanning tree Prim Fibonacci heap	V.LogV	E: number of edges V: number vertices
Shortest paths Graph Dijkstra linear sorting	V^2	V: number of vertices
Shortest paths Graph Dijkstra binary heap	(E + V).logV	V: number of vertices
Shortest paths Graph Dijkstra Fibonacci heap	V.log	E: number of edges V: number of vertices
Shortest paths kNN Graph - Dijkstra	$(k + logN).N^2$	k: number of neighbors N: number of observations
Shortest paths kNN Graph - Floyd-Warshall	N^3	N: number of observations
Fast Fourier transform	N.LogN	N: Number of observations
Batched gradient descent	N.P.I	N: Number of observations P: number of parameters I: number of iterations

Stochastic gradient descent	N.P.I	N: number of observations P: Number of variables I: number of epochs
Newton with Hessian computation	$N3.I$	N: number of observations I: number iterations
Conjugate gradient	N.m.sqrt(k)	N: number of observations m: number of non-zero k condition of the matrix
L-BFGS	N.M.I	N: number of observations M: estimated number of grads I: number of iterations
K-means (*)	C.N.M.I	C: Number of clusters M: Dimension N: number observations I: number of iterations
Decision tree	$M.D^2$	M: Number of data points D: Number of Attributes
Lasso regularization - LARS(*)	N.M.min(N,M)	M: Dimension N: number observations
Hidden Markov Model Forward-backward pass	$N^2.M$	N: number of states M: number of observations
Multilayer Perceptron (*)	n.M.P.N.e	n: input variables M: number hidden neurons P: number output values N: number of observations e: Number of epochs
Support vector machine (*) Newton	N^3	N: number of observations
Support vector machine (*) Cholesky	N^2	N: number of observations
Support vector machine (*) - SMO	N^2	N: number of observations
Principal Components Analysis (*)	$M^2N + N^3$	N: Number of observations M: number of features
Expectation-Maximization (*)	M^2N	N: Number of observations M: number of variables
Laplacian Eigenmaps	$M.logk.N.logN + m.N.k^2 + d.N^2$	N: Number of observations M: number of variables
Genetic algorithms	P.logP.I.C	C: number of genes P: population size I: Number of iterations
Feed forward neural network	N.(I + H.M).M	N: Number of observations I: Number of input unit H: Number of hidden layers M: Average number of units per hidden layer

(*): Model training