

## Dataset Operations

### DATA IMPORT / EXPORT

**h2o.uploadFile:** Upload a file into H2O from a client-side path, and parse it.

**h2o.downloadCSV:** Download a H2O dataset to a client-side CSV file.

**h2o.importFile:** Import a file into H2O from a server-side path, and parse it.

**h2o.exportFile:** Export H2O Data Frame to a server-side file.

**h2o.parseRaw:** Parse a raw data file.

### NATIVE R TO H2O COERCION

**as.h2o:** Convert R object to an H2O object.

### H2O TO NATIVE R COERCION

**as.data.frame:** Check if an object is a data frame, or coerce it if possible.

### DATA GENERATION

**h2o.createFrame:** Creates a data frame in H2O with real-valued, categorical, integer, and binary columns specified by the user, with optional randomization.

**h2o.runif:** Produce a vector of random uniform numbers.

**h2o.interaction:** Create interaction terms between categorical features of an H2O Frame.

**h2o.target\_encode\_apply:** Target encoding map to an H2O Data Frame, which can improve performance of supervised learning models for high cardinality categorical columns.

### DATA SAMPLING / SPLITTING

**h2o.splitFrame:** Split an existing H2O dataset according to user-specified ratios.

### MISSING DATA HANDLING

**h2o.impute:** Impute a column of data using the mean, median, or mode.

**h2o.insertMissingValues:** Replaces a user-specified fraction of entries in a H2O dataset with missing values.

**h2o.na\_omit:** Remove Rows With NAs.

## General Operations

### SUBSCRIPTING

Subscripting example to pull (/push) pieces from (/to) a H2O Parsed Data object.

<pre>x[j] ## column J x[i, j] x[[i]] x\$name</pre>	<pre>x[i] &lt;- value x[i, j, ...] &lt;- value x[[i]] &lt;- value x\$i &lt;- value</pre>
Selection	Value Assignment

### SUBSETTING

**h2o.head, h2o.tail:** Object's Start or End.

### DATA ATTRIBUTES

**h2o.names:** Return column names for a parsed H2O data obj. Also: **h2o.colnames**

**names<-:** Set the row or column names of a matrix-like object. Also: **colnames<-**

**h2o.dim:** Retrieve object dimensions.

**h2o.length:** Length of vector, list or factor.

**h2o.nrow:** Number of H2O Frame rows.

**h2o.ncol:** Number of H2O Frame columns.

**h2o.anyFactor:** Check if an H2OFrame object has any categorical data columns.

**is.factor, is.character, is.numeric:** Check Column's Data Type.

**DATA TYPE COERCION:** Convert to:

**h2o.asfactor, as.factor:** Factor.

**h2o.as\_date, as.Date:** Date.

**h2o.ascharacter, as.character:** Character.

**h2o.asnumeric, as.numeric:** Numeric.

### BASIC DATA MANIPULATION

**c:** Combine Values into a Vector or List.

**h2o.cbind; h2o.rbind:** Combine a sequence of H2O datasets by column (cbind) or rows (rbind).

**h2o.merge:** Merges 2 H2OFrames.

**h2o.arrange:** Sorts H2OFrame by columns.

### ELEMENT INDEX SELECTION

**h2o.which:** True Condition's Row Numbers

### CONDITIONAL VALUE SELECTION

**h2o.ifelse:** Apply conditional statements to numeric vectors in a H2OFrame.

## Math Operations

(math) vectorized function

### MATH

**h2o.abs:** Compute the absolute value of x.

**sign:** Return a vector with the signs of the corresponding elements of x (the sign of a real number is 1, 0, or -1 if the number is positive, zero, or negative, respectively).

**h2o.sqrt:** Principal Square Root of x,  $\sqrt{x}$ .

**h2o.ceiling:** Take a single numeric argument x and return a numeric vector containing the smallest integers not less than the corresponding elements of x.

**h2o.floor:** Take a single numeric argument x and return a numeric vector containing the largest integers not greater than the corresponding elements of x.

**h2o.trunc:** Take a single numeric argument x and return a numeric vector containing the integers formed by truncating the values in x toward 0.

**h2o.log:** Compute natural logarithms. See also: **h2o.log10, h2o.log2, h2o.log1p**

**h2o.exp:** Compute the exponential function

**h2o.cos, h2o.cosh, h2o.acos, h2o.sin, h2o.tan, h2o.tanh, Math:** ?groupGeneric

**&&** (Vectorized AND), **||** (Vectorized OR), **!x, %in%, Ops:** +, -, \*, /, ^, %%, %/%, ==, !=, <, <=, >=, >, &, |, !

### CUMULATIVE

**h2o.cummax:** Vector of the cumulative maxima of the elements of the argument.

**h2o.cummin:** Vector of the cumulative minima of the elements of the argument.

**h2o.cumprod:** Vector of the cumulative products of the elements of the argument.

**h2o.cumsum:** Vector of the cumulative sums of the elements of the argument.

### PRECISION

**h2o.round:** Round values to the specified number of decimal places. The default is 0.

**h2o.signif:** Round values to the specified number of significant digits.

## Group By Summaries

(group by) summary function

**nrow:** Count the number of rows.

**max:** All input argument's Maximum.

**min:** All input argument's Minimum.

**sum:** All argument values Sum.

**mean:** (Trimmed) arithmetic mean.

**sd:** Calculate the standard deviation of a column of continuous real valued data.

**var:** Compute the variance of x.

## Generic Summaries

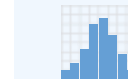
### NON-GROUP\_BY SUMMARIES

**h2o.median:** Calculate the median of x.

**h2o.range:** Input argument's Min/Max Vector

**h2o.cor:** Correlation Matrix of H2OFrames.

**h2o.quantile:** Obtain and display quantiles for H2O parsed data.



**h2o.hist:** Compute a histogram over a numeric column.

**h2o.prod:** Product of all arguments values.

**h2o.any:** Given a set of logical vectors, determine if at least one of the values is true.

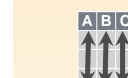
**h2o.all:** Given a set of logical vectors, determine if all of the values are true.

### NON-GROUP\_BY SUMMARIES: GENERIC

**h2o.summary:** Produce result summaries of the results of various model fitting functions.

## Aggregations

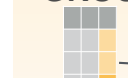
### ROW / COLUMN AGGREGATION



By Rows By Cols

**apply:** Apply a function over an H2O parsed data object (an array) margins.

### GROUP BY AGGREGATION



**h2o.group\_by:** Apply an aggregate function to each group of an H2O dataset.

### TABULATION



**h2o.table:** Use the cross-classifying factors to build a table of counts at each combination of factor levels.

## Data Modeling

### MODEL TRAINING: SUPERVISED LEARNING

**h2o.deeplearning:** Deep Learning Neural Networks.

**h2o.gbm:** Gradient Boosted Classification Trees and Gradient Boosted Regression Trees.

**h2o.glm:** Generalized Linear Model, fit by specifying a response variable, a set of predictors, and a description of the error distribution.

**h2o.naiveBayes:** Naive Bayes Classifier.

**h2o.randomForest:** Random Forest Classification.

**h2o.xgboost:** Extreme Gradient Boosted Model.

### MODEL TRAINING: UNSUPERVISED LEARNING

**h2o.prcomp:** Principal Components Analysis.

**h2o.kmeans:** k-means Clustering.

**h2o.anomaly:** Detect anomalies using a H2O deep learning model with auto-encoding.

**h2o.deepfeatures:** Extract the non-linear features using a H2O deep learning model.

### GRID SEARCH

**h2o.grid:** Efficient method to build multiple models with different hyperparameters.

### MODEL SCORING

**h2o.predict:** Obtain predictions from various fitted H2O model objects.

### MODEL METRICS

**h2o.model metrics:** Given predicted values (target for regression, class-1 probabilities, or binomial or per-class probabilities for multinomial), compute a model metrics object.

### REGRESSION MODEL HELPER

**h2o.mse:** Display the mean squared error calculated from a column of predicted responses and a column of actual (reference) responses.

### CLASSIFICATION MODEL HELPERS

**h2o.accuracy:** Between cluster sum of squares.

**h2o.auc:** AUC (area under ROC curve).

**h2o.confusionMatrix:** Display prediction errors for classification data (predicted vs reference).

**h2o.hit\_ratio\_table:** Retrieve the Hit Ratios.

**h2o.performance:** Evaluate the predictive performance of a model via various measures.

### CLUSTERING MODEL HELPER

**h2o.betweenss:** Between Cluster Sum of Squares.

**h2o.centers:** Retrieve the Model Centers.

## Data Munging

### GENERAL COLUMN MANIPULATION

**is.na:** Display missing elements.

### FACTOR LEVEL MANIPULATIONS

**h2o.levels:** Display a list of the unique values found in a categorical data column.

**h2o.relevel:** Reorders levels of an H2O factor, similarly to standard R's relevel.

**h2o.setLevels:** Set Levels of H2O Factor.

### NUMERIC COLUMN MANIPULATIONS

**h2o.cut:** Convert H2O Numeric Data to Factor by breaking it into Intervals.

### CHARACTER COLUMN MANIPULATIONS

**h2o.strsplit:** "String Split": Splits the given factor column on the input split.

**h2o.tolower:** Convert the characters of a character vector to lower case.

**h2o.toupper:** Convert the characters of a character vector to lower case.

**h2o.trim:** "Trim spaces": Remove leading and trailing white space.

**h2o.gsub:** Match a pattern & replace *all* instances (occurrences) of the matched pattern with the replacement string globally.

**h2o.sub:** Match a pattern & replace the *first* instance (occurrence) of the matched pattern with the replacement string.

### DATE MANIPULATIONS

**h2o.month:** Convert Milliseconds to Months in H2O Datasets (Scale: 0 to 11).

**h2o.year:** Convert Milliseconds to Years in H2O Datasets, indexed starting from 1900.

**h2o.day:** Convert Milliseconds to Day of Month in H2O Datasets (Scale: 1 to 31).

**h2o.hour:** Convert Milliseconds to Hour of Day in H2O Datasets (Scale: 0 to 23).

**h2o.dayOfWeek:** Convert Milliseconds to Day of Week in a H2OFrame (Scale: 0 to 6)

### MATRIX OPERATIONS

**%\*%:** Multiply two conformable matrices.

**t:** Returns the transpose of an H2O Frame.

## Cluster Operations

### H2O KEY VALUE STORE ACCESS

**h2o.assign:** Assign H2O hex.keys to R objects.

**h2o.getFrame:** Get H2O dataset Reference.

**h2o.getModel:** Get H2O model reference.

**h2o.ls:** Display a list of object keys in the running instance of H2O.

**h2o.rm:** Remove specified H2O Objects from the H2O server, but not from the R environment.

**h2o.removeAll:** Remove All H2O Objects from the H2O server, but not from the R environment.

### H2O MODEL IMPORT / EXPORT

**h2o.loadModel:** Load H2OModel from disk.

**h2o.saveModel:** Save H2OModel object to disk.

**h2o.download\_pojo:** Download the Scoring POJO (Plain Old Java Object) of an H2O Model.

**h2o.download\_mojo:** Download the model in MOJO format.

### H2O CLUSTER CONNECTION

**h2o.init:** Connect to a running H2O instance using all CPUs on the host.

**h2o.shutdown:** Shut down the specified H2O instance. All data on the server will be lost!

### H2O CLUSTER INFORMATION

**h2o.clusterInfo:** Display the name, version, uptime, total nodes, total memory, total cores and health of a cluster running H2O.

**h2o.clusterStatus:** Retrieve information on the status of the cluster running H2O.

### H2O LOGGING

**h2o.clearLog:** Clear all H2O R command and error response logs from the local disk.

**h2o.downloadAllLogs:** Download all H2O log files to the local disk.

**h2o.logAndEcho:** Write a message to the H2O Java log file and echo it back.

**h2o.openLog:** Open existing logs of H2O R POST commands and error responses on disk.

**h2o.getLogPath:** Get the file path for the H2O R command and error response logs.

**h2o.startLogging:** Begin logging H2O R POST commands and error responses.

**h2o.stopLogging:** Stop logging H2O R POST commands and error responses.