# h2o: : CHEAT SHEET

## Dataset Operations

### DATA IMPORT / EXPORT

**h2o.downloadCSV:** Download a H2O dataset to a CSV file on local disk.

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

**h2o.importFile:** Import a file from the local path and parse it.

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

**h2o.uploadFile:** Upload a file from the local drive and parse it.

### NATIVE R TO H2O COERCION

**as.h2o:** Convert an 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:** Create an H2O data frame, with optional randomization.

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

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

### 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.

## General Operations

### SUBSCRIPTING

Subscripting example to pull pieces from data object.

```
x[j]  ## column J
x[i, j]
x[[i]]
x$name
```

Selection

```
x[i]  <- value
x[i, j, ...] <- value
x[[i]] <- value
x$i <- value
```

Value Assignation

### SUBSETTING

**head, tail:** Return the First or Last Part of an Object

### CONCATENATION

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

**h2o.cbind:** Take a sequence of H2O datasets and combine them by column.

### DATA ATTRIBUTES

**colnames:** Return column names for a parsed H2O data object.

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

**names:** Get the name of an object.

**names<-:** Set the name of an object.

**dim:** Retrieve the dimension of an object.

**length:** Get the length of vectors (including lists) and factors.

**nrow:** Return a count of the number of rows in an H2OParsedData object.

**ncol:** Return a count of the number of columns in an H2OParsedData object.

**h2o.anyFactor:** Check if an H2O parsed data object has any categorical data columns.

**is.factor:** Check if a given column contains categorical data.

### DATA TYPE COERCION

**as.factor:** Convert a column from numeric to factor.

**as.Date:** Converts a column from factor to date.

## Methods from Group Generics: Math

### 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).

**sqrt:** Computes the principal square root of x, $\sqrt{x}$.

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

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

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

**log:** Compute logarithms (by default, natural logarithms).

**exp:** Compute the exponential function.

### MATH (GENERIC)

**cummax:** Display a vector of the cumulative maxima of the elements of the argument.

**cummin:** Display a vector of the cumulative minima of the elements of the argument.

**cumprod:** Display a vector of the cumulative products of the elements of the argument.

**cumsum:** Display a vector of the cumulative sums of the elements of the argument.

**log10:** Compute common (i.e., base 10) logarithms.

**log2:** Compute binary (i.e., base 2) logarithms.

**log1p:** Compute log(1+x) accurately also for |x|<< 1.

### MATH (GENERIC)

**acos:** Compute the trigonometric arc-cosine.

**acosh:** Compute the hyperbolic arc-cosine.

**asin:** Compute the trigonometric arc-sine.

**asinh:** Compute the hyperbolic arc-sine.

**atan:** Compute the trigonometric arc-tangent.

**atanh:** Compute the hyperbolic arc-tangent.

**expm1:** Compute exp(x) - 1 accurately also for |x|<< 1.

**cos:** Compute the trigonometric cosine.

**cosh:** Compute the hyperbolic cosine.

**cospi:** Compute the trigonometric two-argument arc-cosine.

**sin:** Compute the trigonometric sine.

**sinh:** Compute the hyperbolic sine.

**sinpi:** Compute the trigonometric two-argument arc-sine.

**tan:** Compute the trigonometric tangent.

**tanh:** Compute the hyperbolic tangent.

**tanpi:** Compute the trigonometric two-argument arc-tangent.

**gamma:** Display the gamma function γx

**lgamma:** Display the natural logarithm of the absolute value of the gamma function.

**digamma:** Display the first derivative of the logarithm of the gamma function.

**trigamma:** Display the second derivative of the logarithm of the gamma function.

### MATH2 (H2O)

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

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

# h2o: : CHEAT SHEET

## Methods from Group Generics: Summary

**SUMMARY (H2O)**

**max:** Display the maximum of all the input arguments.

**min:** Display the minimum of all the input arguments.

**range:** Display a vector containing the minimum and maximum of all the given arguments.

**sum:** Calculate the sum of all the values present in its arguments.

**SUMMARY (GENERIC)**

**prod:** Display the product of all values present in its arguments.

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

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

## Other Aggregations

**NON-GROUP GENERIC SUMMARIES**

**mean:** Generic function for the (trimmed) arithmetic mean.

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

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

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

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

**ROW / COLUMN AGGREGATION**

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

**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 Munging

**GENERAL COLUMN MANIPULATION**

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

**ELEMENT INDEX SELECTION**

**h2o.which:** Display the row numbers for which the condition is true.

**CONDITIONAL ELEMENT VALUE SELECTION**

**h2o.ifelse:** Apply conditional statements to numeric vectors in H2O parsed data objects.

**NUMERIC COLUMN MANIPULATIONS**

**h2o.cut:** Convert H2O Numeric Data to Factor.

**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.

**FACTOR LEVEL MANIPULATIONS**

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

**DATE MANIPULATIONS**

**h2o.month:** Convert the entries of a H2OParsedData object from milliseconds to months (on a 0 to 11 scale).

**h2o.year:** Convert the entries of a H2OParsedData object from milliseconds to years, indexed starting from 1900.

**MATRIX OPERATIONS**

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

**t:** Given a matrix or data.frame x, t returns the transpose of x.

## 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**

**ho2.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.

## 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 H2O objects from the server where the instance of H2O is running, but does not remove it from the R environment.

**H2O OBJECT SERIALIZATION**

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

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

**H2O CLUSTER CONNECTION**

**h2o.init ( nthreads = -1 ):** 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 LOAD BALANCING**

**h2o.rebalance:** Rebalance (repartition) an existing H2O dataset into given number of chunks (per Vec), for load-balancing across multiple threads or nodes.

**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 the local 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.