h2omlgraph shapsummary — Produce SHAP beeswarm plot[+]

[+]This command includes features that are part of StataNow.

Description

`h2omlgraph shapsummary` produces the beeswarm plot of Shapley additive explanation (SHAP) values after regression or binary classification performed by `h2oml gbregress`, `h2oml rfregress`, `h2oml gbbinclass`, or `h2oml rfbinclass`. SHAP values indicate the contributions of predictors to the prediction for a given observation. The beeswarm plot allows visualization of SHAP values for many observations by placing them in a one-dimensional scatterplot for each predictor where the overlapping observations are separated (or jittered) so that each SHAP value is visible.

SHAP values are considered a unified measure for variable importance and machine learning model explanation. For an overview of SHAP values, see *Remarks and examples* in [H2OML] **h2omlgraph shapvalues**.

Quick start

Plot SHAP summary

```
h2omlgraph shapsummary
```

As above, but plot the summary for predictors `x1`, `x2`, and `x3`

```
h2omlgraph shapsummary x1-x3
```

Plot the summary for the top 5 highest SHAP-important predictors

```
h2omlgraph shapsummary, top(5)
```

Menu

Statistics > H2O machine learning

Syntax

```
h2omlgraph shapsummary [predictors] [ , options ]
```

options	Description
Main	
top(#)	display the top # highest SHAP-important predictors; default is top(20)
samples(#)	specify the number of observations to be randomly sampled to estimate the SHAP approximation; default is samples(1000)
rseed(#)	set random-number seed to #
savedata(*filename*[, replace])	save plot data to *filename*
Plot options	
norefline	suppress vertical reference line identifying the origin
rlopts(*line_options*)	affect rendition of reference line
startcolor(*colorstyle*)	determine starting color for the color legend
endcolor(*colorstyle*)	determine ending color for the color legend
jitter(#)	affect the magnitude of jitter of overlapped observations
twoway_options	any option other than by() documented in [G-3] ***twoway_options***
train	specify that the SHAP summary be reported using training results
valid	specify that the SHAP summary be reported using validation results
test	specify that the SHAP summary be computed using testing frame
test(*framename*)	specify that the SHAP summary be computed using data in testing frame *framename*
frame(*framename*)	specify that the SHAP summary be computed using data in H2O frame *framename*
framelabel(*string*)	label frame as *string* in the output

train, valid, test, test(), frame(), and framelabel() do not appear in the dialog box.

Options

Main

top(#) specifies the number of highest SHAP-important predictors to be included in the plot. Up to 20 top important predictors are included by default. top() is not allowed if *predictors* are specified.

samples(#) specifies the maximum number of observations to be randomly sampled with replacement to approximate the estimate of the contribution function. The default is samples(1000).

rseed(#) specifies the random-number seed for reproducibility.

savedata(*filename*[, replace]) saves the plot data to a Stata data file (.dta file). replace specifies that *filename* be overwritten if it exists.

Plot options

norefline suppresses the vertical reference line identifying the origin. The line is included by default.

rlopts(*line_options*) affects the rendition of the reference line. See [G-3] ***line_options***.

`startcolor(`*colorstyle*`)` determines the starting color of the color legend. The color legend shows whether the value of the given predictor for the observation is low (starting color) or high (ending color). See [G-4] ***colorstyle***.

`endcolor(`*colorstyle*`)` determines the ending color of the color legend. The color legend shows whether the value of the given predictor for the observation is low (starting color) or high (ending color). See [G-4] ***colorstyle***.

`jitter(`#`)` adds spherical random noise to the data before plotting. # represents the size of the noise as a percentage of the graphical area.

twoway_options are any of the options documented in [G-3] ***twoway_options***, excluding `by()`. These include options for titling the graph (see [G-3] ***title_options***) and options for saving the graph to disk (see [G-3] ***saving_option***).

The following options are available with `h2omlgraph shapsummary` but are not shown in the dialog box:

`train`, `valid`, `test`, `test()`, and `frame()` specify the H2O frame for which SHAP summary is reported. Only one of `train`, `valid`, `test`, `test()`, or `frame()` is allowed.

`train` specifies that SHAP summary be reported using training results. This is the default when validation is not performed during estimation and when a postestimation frame has not been set with `h2omlpostestframe`.

`valid` specifies that SHAP summary be reported using validation results. This is the default when validation is performed during estimation and when a postestimation frame has not been set with `h2omlpostestframe`. `valid` may be specified only when the `validframe()` option is specified with `h2oml` *gbm* or `h2oml` *rf*.

`test` specifies that SHAP summary be computed on the testing frame specified with `h2omlpostestframe`. This is the default when a testing frame is specified with `h2omlpostestframe`. `test` may be specified only after a testing frame is set by using `h2omlpostestframe`. `test` is necessary only when a subsequent `h2omlpostestframe` command is used to set a default postestimation frame other than the testing frame.

`test(`*framename*`)` specifies that SHAP summary be computed using data in testing frame *framename* and is rarely used. This option is most useful when running a single postestimation command on the named frame. If multiple postestimation commands are to be run on the same test frame, it is more computationally efficient and convenient to specify the testing frame by using `h2omlpostestframe` instead of specifying `test(`*framename*`)` with individual postestimation commands.

`frame(`*framename*`)` specifies that SHAP summary be computed using the data in H2O frame *framename*.

`framelabel(`*string*`)` specifies the label to be used for the frame in the output.
stata.com

Remarks and examples

We assume you have read the introduction to explainable machine learning in *Interpretation and explanation* in [H2OML] **Intro** and [H2OML] **h2omlgraph shapvalues**.

Additional examples can be found in example 6 of [H2OML] **h2oml** and example 2 of [H2OML] **h2omlgraph shapvalues**.

SHAP values explain the predictions of a model by measuring the contribution of each predictor to those predictions. For an overview of SHAP values and how they are computed, see *Remarks and examples* in [H2OML] **h2omlgraph shapvalues**. SHAP values can be computed for each observation in the dataset. The `h2omlgraph shapvalues` command allows you to plot SHAP values for one observation at a time. The `h2omlgraph shapsummary` command discussed here provides a summary beeswarm plot for evaluating the contribution of predictors across many observations.

▷ Example 1: Interpreting a SHAP summary plot

In this example, we interpret a SHAP summary plot after performing random forest regression.

We start by opening the 1978 automobile data (`auto.dta`) in Stata and then putting the data into an H2O frame. Recall that `h2o init` initiates an H2O cluster, `_h2oframe put` loads the current Stata dataset into an H2O frame, and `_h2oframe change` makes the specified frame the current H2O frame. For details, see *Prepare your data for H2O machine learning in Stata* in [H2OML] **h2oml** and [H2OML] **H2O setup**.

```
. use https://www.stata-press.com/data/r18/auto
(1978 automobile data)
. h2o init
  (output omitted)
. _h2oframe put, into(auto)
Progress (%): 0 100
. _h2oframe change auto
```

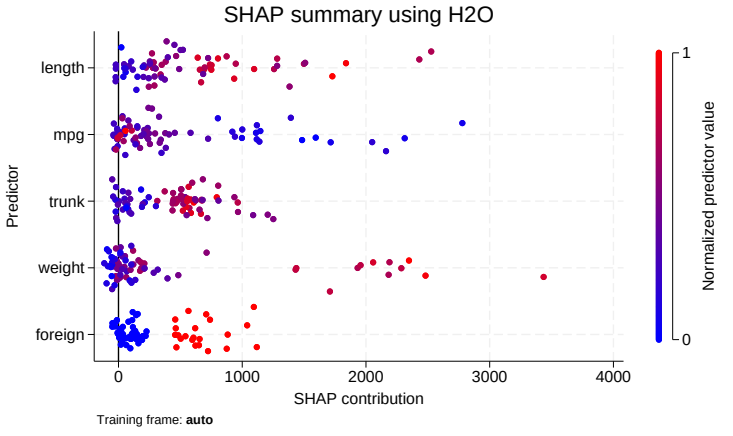
For simplicity, we save the predictor names in the global macro `predictors` in Stata. We then perform random forest regression with 100 trees and limit the maximum depth of the trees to 5.

```
. global predictors foreign mpg trunk weight length
. h2oml rfregress price $predictors, h2orseed(19) ntrees(100) maxdepth(5)
Progress (%): 0 100
Random forest regression using H2O
Response: price
Frame:                                  Number of observations:
  Training: auto                                  Training =     74
Model parameters
Number of trees      = 100
              actual = 100
Tree depth:                             Pred. sampling value =     -1
           Input max =   5              Sampling rate        =   .632
                 min =   2              No. of bins cat.     =  1,024
                 avg = 5.0              No. of bins root     =  1,024
                 max =   5              No. of bins cont.    =     20
Min. obs. leaf split =   1              Min. split thresh.   = .00001
Metric summary
```

Metric	Training
Deviance	3129378
MSE	3129378
RMSE	1769.005
RMSLE	.2315556
MAE	1229.955
R-squared	.6353542

Finally, we use the `h2omlgraph shapsummary` command to plot the SHAP summary. The `samples(300)` option specifies that 300 randomly sampled observations be used, and the `rseed(19)` option is for reproducibility.

```
. h2omlgraph shapsummary, samples(300) rseed(19)
```



The summary plot is a beeswarm plot that provides a summary of how the predictors in a dataset affect the model's predictions. In the graph, for each predictor, each observation is represented as a dot. The horizontal location shows the contributed SHAP value for a specific observation. Colors show whether the predictor has high (red) or low (blue) observed values. For example, smaller observed values of weight are mostly associated with smaller SHAP contributions and a smaller predicted price. On the other hand, smaller observed values of `mpg` mostly imply larger SHAP contributions and a larger predicted price.

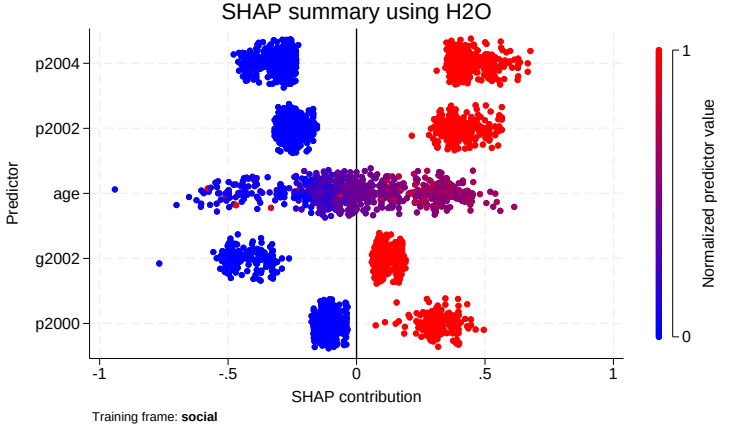
`h2omlgraph shapsummary` offers a number of options to control the look of this graph. The start color and end color for the normalized predictions can be changed by using the `scolor()` and `ecolor()` options. We can specify the `jitter()` option to control how much the observations overlap. We can also specify the `sample()` option to control the maximum number of observations to be sampled from the dataset.

▷ Example 2: Explaining voting behavior

In example 2 of [H2OML] **h2omlgraph shapvalues**, we used local SHAP explanation to study voting behavior for a specific observation. In this example, we use `h2omlgraph shapsummary` to explain voting behavior from a global perspective.

We assume that the `h2oml gbbinclass` command in example 2 of [H2OML] **h2omlgraph shapvalues** has been run to perform gradient boosting binary classification. Here we focus on the SHAP summary plot for the top 5 SHAP-important predictors.

```
. h2omlgraph shapsummary, top(5) rseed(19)
Progress (%): 0 100
```



For binary classification, the explanation is with respect to the positive class, which in our case is `vote = Yes`. We see that being young (represented by blue points for `age`) has a negative effect on the probability of voting because lower ages are mostly associated with negative SHAP contributions. The `p2000`, `p2002`, `p2004`, and `g2002` variables are indicators for voting in primary and general elections. We see that the previous voting behavior of the subjects has a substantial effect on future voting behavior.

Also see

[H2OML] **h2oml** — Introduction to commands for Stata integration with H2O machine learning+

[H2OML] **h2omlgraph shapvalues** — Produce SHAP values plot for individual observations+