h2omlpostestframe — Specify frame for postestimation analysis

Description

`h2omlpostestframe` is a convenience command for setting an H2O frame to be used by `h2oml` postestimation commands to report results after `h2oml` *gbm* and `h2oml` *rf*. `h2omlpostestframe` does not physically change the current frame to the specified frame; see `_h2oframe change`.

`h2omlpostestframe` affects all but the following postestimation commands: `h2omlestat gridsummary`, `h2omlselect`, `h2omlexplore`, `h2omlestat cvsummary`, `h2omlgraph varimp`, `h2omlgraph scorehistory`, and `h2omltree`.

Quick start

Specify a generic frame named `mytest` to be used by postestimation commands, and label it as "Testing" in the output

```
h2omlpostestframe mytest
```

Specify a predefined validation frame to be used by postestimation commands

```
h2omlpostestframe _valid
```

Specify a frame named `auto` and label it

```
h2omlpostestframe auto, label(Auto dataset)
```

Switch back to the default frame specific to each postestimation command

```
h2omlpostestframe _default
```

Menu

Statistics > H2O machine learning

Syntax

Specify generic frame to be used by postestimation commands to report the results

`h2omlpostestframe` *framename* [, `notest label(`*string*`)`]

Specify prespecified frame to be used by postestimation commands to report the results

`h2omlpostestframe` *frametype* [, `label(`*string*`)`]

frametype	Description
`_default`	default frame; varies across commands
`_train`	training frame
`_valid`	validation frame
* `_cv`	cross-validation "frame"

* `_cv` does not correspond to an actual H2O frame; it is not applicable for some postestimation commands. See *Remarks and examples*.

`label()` is not allowed with `_default` or `_cv`.

Options

Options

`notest` specifies that the generic frame should not be considered a testing frame. By default, the specified frame is assumed to be a testing frame. This frame will be used whenever option `test` is specified with `h2oml` postestimation commands that support this option. However, if option `notest` is specified with `h2omlpostestframe`, then option `test` may not be used with the postestimation commands.

`label(`*string*`)` labels frame as *string* in the output.

Remarks and examples

The `h2omlpostestframe` command is designed to simplify machine learning postestimation analysis. If neither the `cv()` nor `validframe()` option is specified during estimation, the `h2oml` postestimation commands perform computations using the training frame. If the `validframe()` option is specified, they use the validation frame. And if the `cv()` option is specified, they use the cross-validation results for computation.

Sometimes, we may want to use a different frame for postestimation analysis such as a testing frame. The `h2oml` postestimation commands support options that allow you to specify a different frame. Alternatively, we can use the `h2omlpostestframe` command to specify the desired frame once for all postestimation analyses. By default, the specified frame is assumed to be a testing frame and thus will be labeled correspondingly in the output. You can use the `notest` option to suppress this and use the `label()` option to provide your own frame label.

Instead of a generic frame name, we can also specify `_train`, `_valid`, or `_cv` with the `h2omlpostestframe` command to use the respective training, validation, or cross-validation results for all postestimation analyses, provided the appropriate options were specified during estimation. The `_cv` specification does not correspond to an actual H2O frame and is not supported by `h2omlpredict`, `h2omlgraph pdp`, `h2omlgraph ice`, `h2omlgraph shapvalues`, and `h2omlgraph shapsummary` postestimation commands.

At any point during your postestimation analyses, you can specify _default to switch back to using the default frame, which is specific to each postestimation command.

Below, we demonstrate various uses of h2omlpostestframe on auto.dta.

▷ Example 1: Using h2omlpostestframe

Suppose we want to perform various postestimation analyses using the testing frame. 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. We use the _h2oframe split command to randomly split the auto frame into a training frame (80%) and a testing frame (20%), which we name train and test, respectively. We also change the current frame to train. 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/r19/auto
(1978 automobile data)
. h2o init
. _h2oframe put, into(auto)
. _h2oframe split auto, into(train test) split(0.8 0.2) rseed(19)
. _h2oframe change train
```

Next we perform random forest binary classification using cross-validation.

```
. h2oml rfbinclass foreign price mpg length, cv(3, modulo) h2orseed(19)
```
(*output omitted*)

We want to use the testing frame test for all postestimation analyses. We type

```
. h2omlpostestframe test
(testing frame test is now active for h2oml postestimation)
```

The command reported that test is assumed to be a testing frame.

Now we can use any of the postestimation commands that work with a testing frame, and the test frame will be used in computations automatically:

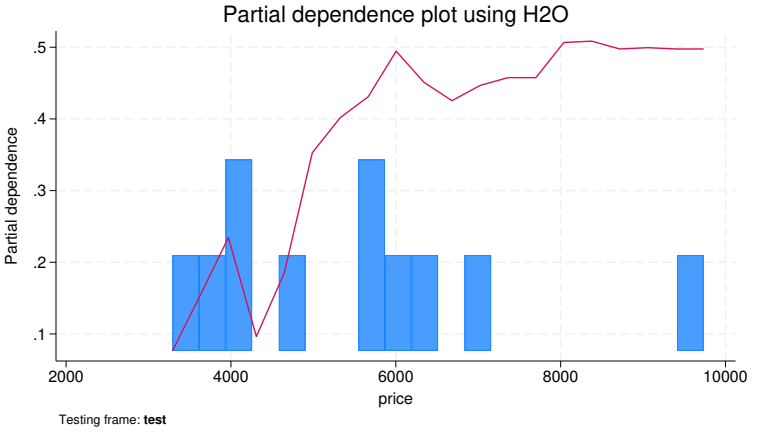
```
. h2omlestat confmatrix
Confusion matrix using H2O
Testing frame: test
```

	Predicted				
foreign	Domestic	Foreign	Total	Error	Rate
Domestic	6	1	7	1	.143
Foreign	0	4	4	0	0
Total	6	5	11	1	.091

```
Note: Probability threshold .52 that maximizes F1 metric
      used for classification.
```

or

```
. h2omlgraph pdp price
```



And to compute predictions for the testing frame `test`, we can simply type

```
. h2omlpredict foreignhat, class
```

Note that `h2omlpostestframe` does not physically change the current frame to `test`. To access the predicted classes, we will need to change the working frame to `test` with `_h2oframe change test`.

Instead of using `h2omlpostestframe`, we could have specified the `test(test)` options with each command above. For instance, we could have typed

```
. h2omlestat confmatrix, test(test)
Confusion matrix using H2O
Testing frame: test
```

foreign	Predicted Domestic	Foreign	Total	Error	Rate
Domestic	6	1	7	1	.143
Foreign	0	4	4	0	0
Total	6	5	11	1	.091

```
Note: Probability threshold .52 that maximizes F1 metric
      used for classification.
```

But this would require more typing.

If we need to switch back to postestimation commands using their default frames, we can specify _default instead of the frame name. For instance, because we specified the cv() option during estimation, by default, h2omlestat confmatrix would have reported the results based on cross-validation. We can still obtain these results by specifying the cv option with the command:

```
. h2omlestat confmatrix, cv
Cross-validation confusion matrix using H2O
```

	Predicted				
foreign	Domestic	Foreign	Total	Error	Rate
Domestic	34	11	45	11	.244
Foreign	2	16	18	2	.111
Total	36	27	63	13	.206

```
Note: Probability threshold .22 that maximizes F1 metric
      used for classification.
```

Or we can use h2omlpostestframe to restore the default frame for all postestimation commands by typing

```
. h2omlpostestframe _default
(cross-validation results are now active for h2oml postestimation)
```

We can also specify one of the predefined frames with h2omlpostestframe to be used for h2oml postestimation analysis: _train to use the training frame, _valid to use the validation frame when the validframe() option is specified during estimation, and _cv to use cross-validation results when the cv() option is specified during estimation. For instance, we can type

```
. h2omlpostestframe _train
(training frame train is now active for h2oml postestimation)
```

The above is also equivalent to specifying the train option with h2omlestat confmatrix:

```
. h2omlestat confmatrix, train
  (output omitted)
```

Also, because we previously used h2omlpostestframe to define a testing frame, we can use the test option with the postestimation commands that support this option to obtain results for the testing frame:

```
. h2omlestat confmatrix, test
Confusion matrix using H2O
Testing frame: test
```

	Predicted				
foreign	Domestic	Foreign	Total	Error	Rate
Domestic	6	1	7	1	.143
Foreign	0	4	4	0	0
Total	6	5	11	1	.091

```
Note: Probability threshold .52 that maximizes F1 metric
      used for classification.
```

◁

Stored results

`h2omlpostestframe` stores the following in `r()`:

Macros

`r(postest_frame)`	name of the frame
`r(postest_label)`	frame label

Also see

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

[H2OML] **h2oml postestimation** — Postestimation tools for h2oml gbm and h2oml rf