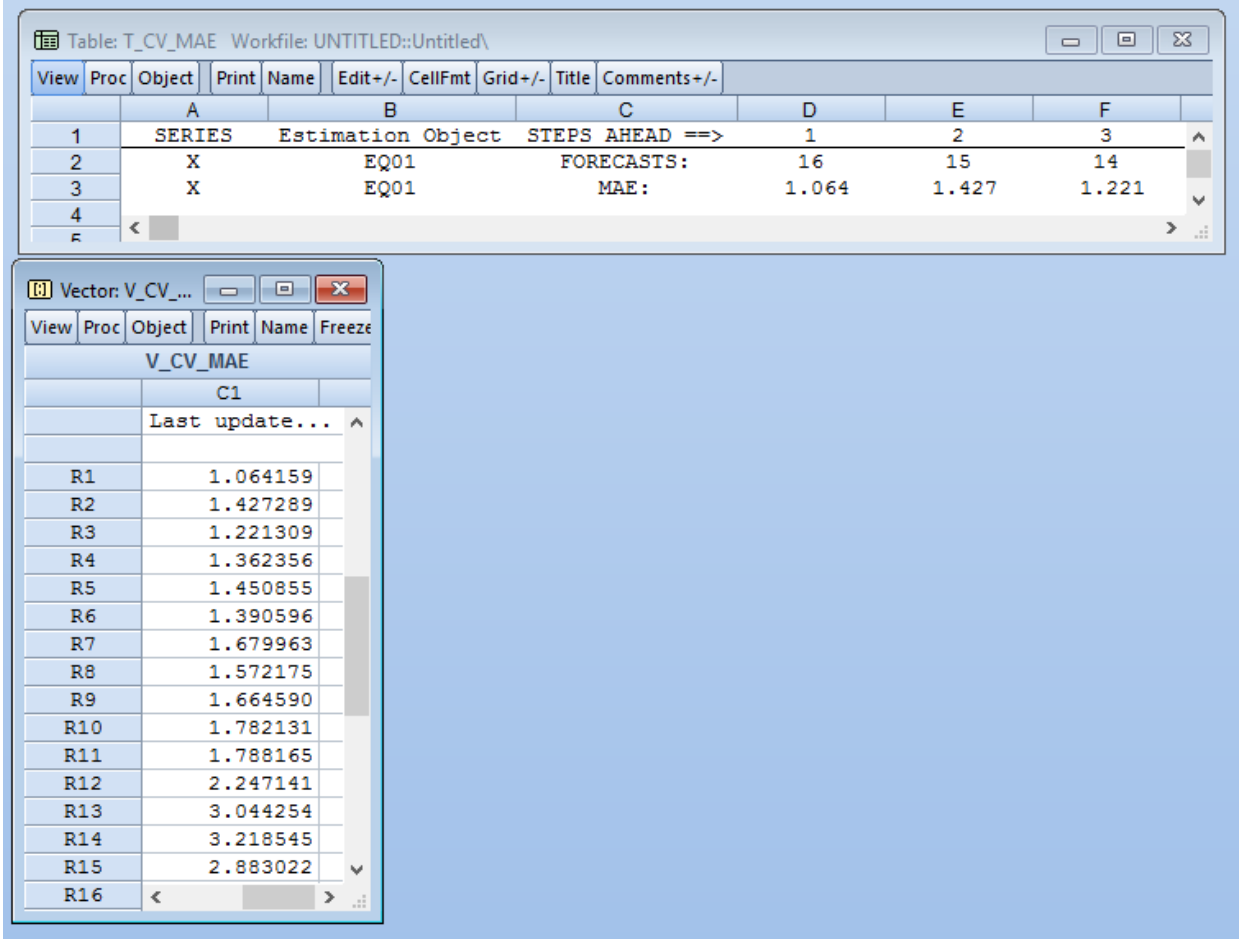
tscval	Equation Proc, VAR Proc

Perform time-series cross-validation.

`tscval` performs rolling estimation and out-of-sample forecast evaluation for equation and VAR objects. If called from an equation object, `tscval` will return cross-validation results for forecasts of the base (e.g. non-transformed) forms of the dependent variable. If called from a VAR object, `tscval` will return cross-validation results for forecasts of the base forms of all endogenous variables.

The add-in will return table objects and vector objects (one per variable and error measure) containing cross-validation results by horizon. For example:



The object t_cv_mae shows MAE results from the equation object “EQ_01”. The table shows that the average 1-period-ahead MAE was 1.064, and that this average was calculated over 16 forecasts. The vector object v_cv_mae mirrors the table data, and is provided for the convenience of users who might want to use cross-validation results for model weighting or selection.

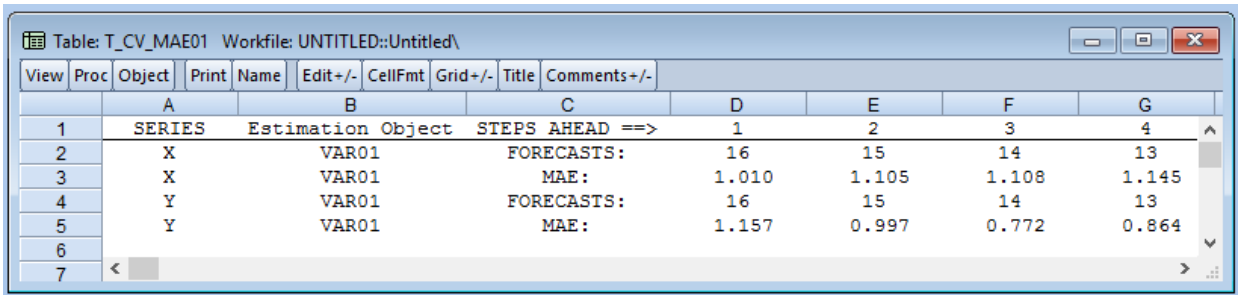


Table: T_CV_MAE01 Workfile: UNTITLED::Untitled\

View | Proc | Object | Print | Name | Edit+/- | CellFmt | Grid+/- | Title | Comments+/-

	A	B	C	D	E	F	G
1	SERIES	Estimation Object	STEPS AHEAD ==>	1	2	3	4
2	X	VAR01	FORECASTS:	16	15	14	13
3	X	VAR01	MAE:	1.010	1.105	1.108	1.145
4	Y	VAR01	FORECASTS:	16	15	14	13
5	Y	VAR01	MAE:	1.157	0.997	0.772	0.864
6							
7							

For VAR output, `tscval` will return one table per error type, with the endogenous variables stacked on top of each other. For example:

The object t_cv_mae01 shows MAE results from the VAR object “VAR01”. The table shows that the 1-period-ahead MAE for series X was 1.010 and the 1-period-ahead MAE for series Y was 1.157. For VAR objects, `tscval` will return one vector object per endogenous variable and error type.

For both equation and VAR cross-validation, the returned objects contain two useful metadata attributes to aid in manipulation/combination of the results. The table and vector objects all contain the attribute “Estimation_Object”, which gives the name of the workfile object (equation or VAR) used to produce forecasts. In addition, vector objects contain the attribute “Series”, which gives the name of the workfile series object to which the error estimates pertain.

Syntax

{%equation}.tscval(*options)*
{%var}.tscval(*options*)

Options

SAMPLE = *arg*	The range within which cross-validation is performed. Arguments should be strings in valid EViews sample form (e.g., “1999m01 2015m12”). If omitted, SAMPLE will default to the workfile sample
H = *arg*	Maximum proportion of the training range (specified in SAMPLE) to holdout. Arguments should be real numbers which satisfy 0 < H <= 1. If omitted, H will default to 0.1 (10% of the training range).
ERR = *arg(s)*	One or more error measures to return. *arg* takes values of “MSE” (mean squared error), “MAE” (mean absolute error), “RMSE” (root mean squared error), “MSFE” (mean squared forecast error), “medAE” (median absolute error), “MAPE” (mean absolute percent error), “SMAPE” (symmetric MAPE), “MPE” (mean

	percent error), “MSPE” (mean squared percent error), “RMSPE” (root mean squared percent error), “medPE” (median percent error), “medSPE” (median squared percent error), “SIGN” (count of times that the model forecast the correct direction of change over a given horizon), and “SIGNP” (SIGN, expressed as a percentage of forecasts produced). To specify multiple errors, pass in any combination of the arguments above as a space-delimited list (e.g. “MAE MAPE MSE”). Defaults to “MAE”.
KEEP_MATS = *arg*	Return raw matrices of all the errors and forecasts (“t”). If ignored or anything other than “t” is passed, raw matrices will not be returned in the workfile.

Examples

The commands

```
equation eq01.ls d(ip) c d(gdp)
eq01.tscval
```

estimate an equation object called `EQ01` and generate cross-validation results using the add-in defaults. The workfile will contain a table object and vector object containing out-of-sample forecast error results for ip (the difference operator will be unwound in forecasting).

You may wish to customize the results more. For example, the commands:

```
equation eq01.ls d(ip) c d(gdp)
eq01.tscval(SAMPLE=”2006m01 2015m12”, H=0.4,
     ERR = “MAE MSE MAPE”)
```

estimate an equation object called `EQ01` and perform time-series cross-validation. Given the settings from `SAMPLE` and `H`, the first equation in the cross-validation exercise will be trained on the sample “2006m01 2011m12” (60% of the argument passed to `SAMPLE`) and tested over the sample “2012m01 2015m12” (the holdout sample, the remaining 40% of the argument passed to `SAMPLE`). Three table objects and three vector objects will be created, one for each error measure.

The commands

```
var var01.LS 1 2 LOG(X) LOG(Y) @ C @LAG(LOG(Z),1) @TREND
var01.tscval
```

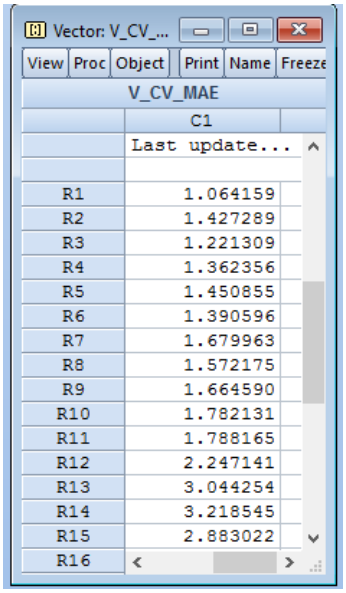
will produce a similar output with a table for each error type selected in the program command. Each table will contain error estimates for every endogenous variable in the VAR object.

tscv_score	Vector Proc

Score a vector of time-series cross-validation results

`tscv_score` computes a scalar score summarizing a vector of cross-validation error estimates. In the default case, this scalar will be simply an equally-weighted average of the errors across all horizons. The score is returned as a metadata attribute in the cross validation vector.

Recall that the vectors returned from running `tscval` are of the following form:



`tscv_score` will summarize this vector of results in a scalar, using either equal or user-provided weights.

Syntax

{%vector}.tscv_score(*options)*

Options

attr = *arg*	Name to use in assigning the score into the vector's metadata. If omitted, attr will default to "tscv_score".
npers = *arg*	Number of horizons to compute the score over, expressed as an integer. If omitted, npers will default to the length of the vector this proc is called from.
score_vec = *arg*	Vector object of weights. If provided, the proc will use these weights in the scoring procedure. If omitted, the default assumption is an equally-

	weighted average across horizons.

Examples

The commands

```
equation eq01.ls d(ip) c d(gdp)
eq01.tscval(ERR = "MAE")
```

estimate an equation object called `EQ01` and generate cross-validation results using the add-in defaults. The workfile will contain a vector object called "v_cv_mae" with out-of-sample forecast error results (MAEs) for ip (the difference operator will be unwound in forecasting). The vector v_cv_mae will be referenced in all following examples

The default case, an equally-weighted average across all horizons, can be computed with the command:

```
v_cv_mae.tscv_score
```

The command

```
v_cv_mae.tscv_score(npers = 5)
```

will compute a score by averaging across only the first 5 horizons from cross-validation.

The command

```
v_cv_mae.tscv_score(npers = 5, attr = thing)
```

will compute a score by averaging across only the first 5 horizons from cross-validation and store that score in the attribute "thing" in v_cv_mae's label.

The commands

```
vector(5) v_scr = 0.1
v_scr(1) = 0.6
v_cv_mae.tscv_score(npers = 5, score_vec = v_scr)
```

will compute a score across only the first 5 horizons from cross-validation, using the weights v_scr. In other words, the score returned will be 60% * [1-period MAE] + 10% * [2-5 month MAEs]. This option allows users to tailor scoring to their forecasting objectives (e.g. when testing a trading strategy, one might want to favor methods that perform well over longer horizons).

Contact Information

Please send any questions, comments, criticisms, or complaints to jaylamb20@gmail.com. If you'd like to contribute to the project, please feel free to send a pull request to https://github.com/jameslamb/ML4EVIEWS/tree/master/tscval.

References

[1] Hyndman, Rob J. “Why every statistician should know about cross-validation”. *Hyndsight blog*, October 2010. Retrieved from: http://robjhyndman.com/hyndsight/crossvalidation/

[2] Hyndman, Rob J. “Time series cross-validation: and R example”. *Hyndsight blog*, August 2011. Retrieved from: http://robjhyndman.com/hyndsight/tscvexample/

[3] Fomby, Tom. “Scoring Measures for Prediction Problems”. Retrieved from: http://faculty.smu.edu/tfomby/eco5385_eco6380/lecture/Scoring%20Measures%20for%20Prediction%20Problems.pdf

[4] Hyndman, Rob J. “Errors on percentage errors”. *Hyndsight blog*, April 2014. Retrieved from: http://robjhyndman.com/hyndsight/smape/

[5] Hyndman, Rob J. “Another Look at Forecast Accuracy Metrics for Intermittent Demand”. *Foresight: The International Journal of Applied Forecasting* 4.4 (2006): 43-46. Retrieved from: http://robjhyndman.com/papers/foresight.pdf

[6] Z. Chen and Y. Yang, 2004, Assessing forecast accuracy measures, Preprint Series, n. 2004-2010, p. 2004-10. Retrieved from: https://www.researchgate.net/profile/Zhuo_Chen5/publication/228774888_Assessing_forecast_accuracy_measures/links/02bfe50da5f6e9d263000000.pdf