FORECASTING FOR BUSINESS & ECONOMICS EBC2089

Prof. dr. Alain Hecq
a.hecq@maastrichtuniversity.nl

February 2023

Prediction

- The estimated regression equation by OLS

$$\hat{y} = X\hat{\beta} = \hat{\beta}_0 + \hat{\beta}_1 x_1 ... + \hat{\beta}_k x_k$$

  is used for predicting the value of y given values of X, say X^0.
- Then we predict (point prediction) the corresponding value y^0 of y by (OLS predictions are unbiased)

$$\hat{y}^0 = X^{0'}\hat{\beta}$$

- the true value y^0 is given by

$$y^0 = X^{0'}\beta + \varepsilon^0$$

  where ε^0 is the error term

Prediction

- Forecast errors

$$
\begin{aligned}
e^0 &= y^0 - \hat{y}^0 = (\beta - \hat{\beta})' X^0 + \varepsilon^0 \\
E(e^0) &= 0 \\
\\
V(e^0) &= V((\beta - \hat{\beta})' X^0) + \sigma^2 \\
&= X^{0'} (\sigma^2 (X'X)^{-1}) X^0 + \sigma^2 \\
&= \sigma^2 (1 + X^{0'} (X'X)^{-1} X^0) \\
\\
&= s^2 (1 + X^{0'} (X'X)^{-1} X^0)
\end{aligned}
$$

- Confidence interval

$$\hat{y}^0 \pm z_{\alpha/2}\sqrt{s^2(1 + X^{0\prime}(X'X)^{-1}X^0)}$$

- $z_{\alpha/2} \approx 2$ for a 95% confidence interval.
- Accuracy of forecasts

$$\begin{aligned} RMS(F)E &: \sqrt{\frac{1}{n^0}\sum(y_i - \hat{y}_i)^2} \\ MAE &: \frac{1}{n^0}\sum|y_i - \hat{y}_i| \end{aligned}$$

Forecasting concepts: in sample vs. out of sample

- In sample forecasts: estimate 1980q1 to 2019q3, evaluate forecasts on e.g. 2011q1 to 2019q3

$$t \;=\; \overbrace{1, 2, ..T_0, T_{0+1}, ...T}^{\text{estimation period}},$$
$$\overbrace{T_{0+1}, ..., T}^{\text{forecast eval.}}$$

This is very close to a R^2.

- Out of sample forecast (better): estimate 1980q1 to 2010q4, fix parameters to their estimated values, evaluate forecasts on 2011q1 to 2019q3

$$t = \overbrace{1, 2, ...T_0}^{\text{estimation period}}, \overbrace{T_{0+1}, ..., T}^{\text{forecast eval.}}$$

Forecasting concepts: static vs. dynamic forecasts

- Static vs. dynamic forecast in case of lagged dependent variable
- Let us consider a population equation

$$y_t = \alpha + \beta x_t + \rho y_{t-1} + \varepsilon_t$$

- That I estimate on $t = 1..T$ using e.g. OLS

$$\hat{y}_t = \hat{\alpha} + \hat{\beta} x_t + \hat{\rho} y_{t-1}$$

Forecasting concepts: static vs. dynamic

- Dynamic forecasts: Multi step ahead forecasts

$$\hat{y}_{T+H} = \hat{\alpha} + \hat{\beta} x_{T+H} + \hat{\rho} \hat{y}_{T+H-1}$$

  where x_{T+H} is given (in the data base) but $\hat{y}_{T+H-1}$ is forecasted
- Static forecasts: a sequence of one step ahead

$$\hat{y}_{T+H} = \hat{\alpha} + \hat{\beta} x_{T+H} + \hat{\rho} y_{T+H-1}$$

  where x_{T+H} AND $\hat{y}_{T+H-1}$ are given (in the data base).
- In both cases the estimation of the parameters is not updated.
- The first forecasted value is the same.

Comparing forecast accuracy

- Let's assume we aim at forecasting $\{y_t\}$ using two models, say A and B.
- They produce two forecasts, $\hat{y}^A_{t+h}$ and $\hat{y}^B_{t+h}$ given information up to moment t
- Maybe A is a set of regressor from economic theory, whereas B is using variables from your intuition.
- Computing the forecast errors amounts to

$$\hat{\epsilon}^A_{t+h} = y_{t+h} - \hat{y}^A_{t+h}$$

$$\hat{\epsilon}^B_{t+h} = y_{t+h} - \hat{y}^B_{t+h}$$

- Repeat it in order to have a sequence of n forecasts (static or dynamic), i.e. a vector of forecasts.

Comparing forecast accuracy

- Problem: How to assess which model is better?
- Even if all forecast errors of model A are smaller than the ones from B, is that difference large "enough" to conclude superior forecasting performance of model A?
- In other words: is RMSE (RMSFE) enough for comparing models?
- We need a formal test to check that
- Another problem: Since the h-step forecasts use overlapping data, the forecast error vectors will be serially correlated
- The test should be able to incorporate that

Loss functions

- We measure forecast accuracy by a loss function, i.e.,

$$L(y_{t+h}, \hat{y}^i_{t+h}) = L(\hat{\epsilon}^i_{t+h}), i = A, B$$

- Examples include

$$L(\hat{\epsilon}^i_{t+h}) = (\hat{\epsilon}^i_{t+h})^2 \text{ Squared error loss}$$

$$L(\hat{\epsilon}^i_{t+h}) = |\hat{\epsilon}^i_{t+h}| \text{ Absolute error loss}$$

- Obviously, the null hypothesis we are interested in is:

$$H_0 : \mathbb{E}[L(\hat{\epsilon}^A_{t+h})] = \mathbb{E}[L(\hat{\epsilon}^B_{t+h})] \text{ against}$$

$$H_A : \mathbb{E}[L(\hat{\epsilon}^A_{t+h})] \neq \mathbb{E}[L(\hat{\epsilon}^B_{t+h})]$$

The Diebold-Mariano Test

- Consider the loss differential

$$d_t = L(\hat{\epsilon}^A_{t+h}) - L(\hat{\epsilon}^B_{t+h})$$

- Then the null of interest is $H_0 : \mathbb{E}[d_t] = 0$ and the Diebold-Mariano test statistic is noting but a regression of d_t on an intercept

$$d_t = c + \varepsilon_t$$

and

$$H_0 : c = 0$$

with maybe a HAC s.e.

- Note: The models should be non-nested (otherwise test degenerate in large sample);
- Otherwise, we need encompassing tests (Giacomini and White (2005))
- Or "do it" and say that we have a small sample anyway...

The Diebold-Mariano test in practice using EViews DM

EViews examples

- Step 0: Estimate 2 models excluding the forecasting period,
- Step 1: Obtain forecasts from those 2 different models say for the last 12 observations (or more in practice)
- Step 2: open the actual (true) series (show)
- Step 3: go to **View/Forecast evaluation**
- Step 4: type the 2 forecasts series from the two models and the span of the forecasting sample.
- You will get DM test.

The Diebold-Mariano test in practice using EViews by hand

- Step 0: Estimate 2 models excluding the forecasting period,
- Step 1: Obtain forecasts from those 2 different models say for the last 12 observations (or more in practice)
- Step 2: compute the forecast errors e_t of the two models on the 12 observations (true minus forecasts)
- Step 3: square the forecast errors
- Step 4: build $d_t = e^2_{At} - e^2_{Bt}$
- Step 5: carry out on the 12 observation an OLS regression from d_t on an intercept and test the null that it is zero.
- You will get DM test.

Forecast combinations

- The forecast literature is filled with "horse races", i.e., studies in which among several competing models, one is chosen to be superior in terms of forecasting performance
  - Sometimes even simply $\text{RMSFE}_A < R\text{MSFE}_B$
  - Tests is better (e.g. DM test)
- In finance or insurances: Maybe a portfolio of assets (pool of insurances) leads to diversification gains and thereby minimized overall risk (risk sharing)
- Application here in forecasting: Maybe we can combine forecasts in such a way as to improve the overall forecasts and limit the individual models' poor performances for some forecasts by offsetting them with other well-performing models

Forecast combinations

- Two groups of forecast combination methods: Variance and Regression (where the former is a special case of the latter actually)
- As before, let's assume we have forecasts $\hat{y}^A_{t+h}$ and $\hat{y}^B_{t+h}$ at our disposal
- A simple way to combine them is:

$$\hat{y}^{Comb}_{t+h} = w_a \hat{y}^A_{t+h} + w_b \hat{y}^B_{t+h}$$

- Note that $w_a + w_b$ should equal one to yield unbiasedness $\Rightarrow w_b = 1 - w_a$
- A simple approach would be to set $w_a = 1/2$
- Combined forecast error: $\hat{\epsilon}^{Comb}_{t+h} = w_a \hat{\epsilon}^A_{t+h} + (1 - w_a)\hat{\epsilon}^B_{t+h}$ with variance

$$\sigma^2_{Comb} = (w_a)^2 \sigma^2_{AA} + (1 - w_a)^2 \sigma^2_{BB} + 2 w_a (1 - w_a) \sigma^2_{AB}$$

Forecast combinations

- Plugging in $w_a = 0.5$ yields $\sigma^2_{Comb} = 0.25\sigma^2_{AA} + 0.25\sigma^2_{BB} + 0.5\sigma^2_{AB}$
- Let us assume $\sigma^2_{AA} = 1.5, \sigma^2_{BB} = 0.5$ and $\sigma^2_{AB} = 1$
- Then, $w_a = 0.5$ leads to $\sigma^2_{Comb} = 1$, whereas $w_a = 0.25$ gives $\sigma^2_{Comb} = 0.75$
- What is the optimal w_a? Let's minimize σ^2_{Comb} w.r.t. w_a:

$$\Rightarrow w_a^* = \frac{\sigma^2_{BB} - \sigma^2_{AB}}{\sigma^2_{AA} + \sigma^2_{BB} - 2\sigma^2_{AB}}$$

Forecast combinations

- Nothing to loose by combining: $\sigma^2_{Comb} \leq \min\{\sigma^2_{AA}, \sigma^2_{BB}\}$
- Replace unknown (co)variances by consistent estimates: $\hat{\sigma}^2_{ij} = \sum_t \hat{\epsilon^i_{t+h}} \hat{\epsilon^j_{t+h}}$
- Naturally (often used in practice) , for uncorrelated forecast errors $\sigma^2_{AB} = 0$ and the formula simplifies

$$\Rightarrow w_a^* = \frac{\sigma^2_{BB}}{\sigma^2_{AA} + \sigma^2_{BB}}$$

- We can straightforwardly extend the analysis to combinations of more than two models

Forecast combinations

- The regression method for forecast combinations relies on regressing the actual observation on the forecasts:

$$y_{t+h} = \beta_1 \hat{y}^A_{t+h} + (1 - \beta_1)\hat{y}^B_{t+h} + u_{t+h}$$

- In practice it may be beneficial to include an intercept and relax the sum-to-unity restriction $\Rightarrow$ bias correction
- Possible to allow for time-varying weights and non-linear combining regressions
- Allow for MA($h-1$) or general ARMA(p,q) structure to account for serial correlation when $h > 1$ and to capture dynamics in y
- Equality constraints may introduce bias, but eliminate sampling variation in the combining weights

Model selection with EViews (Varsel)

- **Schizophrenia**: we want to have a model that make sense (intuitively, economically) but also to get the best forecast whatever the model.
- We use tools that may be used to automatically determine the variables used as regressors in a least squares regression.
- The explosion in available data over recent decades, coupled with increases in computing power, has led to growing popularity of methods that allow the data themselves to suggest the most appropriate combination of regressors to use in estimation.
- These techniques allow the researcher to provide a set of candidate variables for the model, rather than specifying a specific model.

Model selection with EViews (Varsel)

- We will ask for the help of selection models
  - Uni-Directional forward/backward
  - Stepwise forward/Stepwise backward
  - Swapwise R2 increments
  - Gets
  - Lasso (machine learning)

Uni-Directional-Forwards

- The Uni-directional-Forwards method uses either a lowest *p-value* or largest *t-statistic* criterion for adding variables.
- The method begins with no added regressors.
- If using the *p-value* criterion, we select the variable that would have the lowest *p-value* were it added to the regression.
- If the *p-value* is lower than the specified stopping criteria, the variable is added.

Uni-Directional-Forwards

- The selection continues by selecting the variable with the next lowest *p-value*, given the inclusion of the first variable.
- The procedure stops when the lowest *p-value* of the variables not yet included is greater than the specified forwards stopping criterion, or the number of forward steps or number of added regressors reach the optional user specified limits.
- If using the largest *t-statistic* criterion, the same variables are selected, but the stopping criterion is specified in terms of the statistic value instead of the *p-value*.

Uni-Directional-Backwards

- The Uni-directional-Backwards method is analogous to the Uni-directional-Forwards method,
- but begins with all possible added variables included, and then removes the variable with the highest *p-value*.
- The procedure continues by removing the variable with the next highest *p-value*, given that the first variable has already been removed.
- This process continues until the highest *p-value* is less than the specified backwards stopping criteria, or the number of backward steps or number of added regressors reach the optional user specified limits.
- The largest *t-statistic* may be used in place of the lowest *p-value* as a selection criterion

Stepwise-Forwards

- The Stepwise-Forwards method is a combination of the Uni-directional-Forwards and Backwards methods.
- Stepwise-Forwards begins with no additional regressors in the regression, then adds the variable with the lowest p-value. The variable with the next lowest p-value given that the first variable has already been chosen, is then added.
- Next both of the added variables are checked against the backwards *p-value* criterion. Any variable whose p-value is higher than the criterion is removed.

Stepwise-Forwards

- Once the removal step has been performed, the next variable is added. At this, and each successive addition to the model, all the previously added variables are checked against the backwards criterion and possibly removed.
- The Stepwise-Forwards routine ends when the lowest p-value of the variables not yet included is greater than the specified forwards stopping criteria (or the number of forwards and backwards steps or the number of added regressors has reached the corresponding optional user specified limit).

Stepwise-backwards

- The Stepwise-Backwards procedure reverses the Stepwise-Forwards method.
- All possible added variables are first included in the model. The variable with the highest *p-value* is first removed. The variable with the next highest p-value, given the removal of the first variable, is also removed.
- Next both of the removed variables are checked against the forwards *p-value* criterion. Any variable whose *p-value* is lower than the criterion is added back in to the model.

Stepwise-backwards

- Once the addition step has been performed, the next variable is removed. This process continues where at each successive removal from the model, all the previously removed variables are checked against the forwards criterion and potentially re-added.
- The Stepwise-Backwards routine ends when the largest p-value of the variables inside the model is less than the specified backwards stopping criterion, or the number of forwards and backwards steps or number of regressors reaches the corresponding optional user specified limit.
- The largest t-statistic may be used in place of the lowest p-value as a selection criterion.

Swapwise-Max R-Squared Increment

- The Max R-squared Increment method starts by adding the variable which maximizes the resulting regression R-squared.
- The variable that leads to the largest increase in R-squared is then added.
- Next each of the two variables that have been added as regressors are compared individually with all variables not included in the model, calculating whether the R-squared could be improved by swapping the "inside" with an "outside" variable. If such an improvement exists then the "inside" variable is replaced by the "outside" variable.

Swapwise-Max R-Squared Increment

- If there exists more than one swap that would improve the R-squared, the swap that yields the largest increase is made.
- Once a swap has been made the comparison process starts again. Once all comparisons and possible swaps are made, a third variable is added, with the variable chosen to produce the largest increase in R-squared. The three variables inside the model are then compared with all the variables outside the model and any R-squared increasing swaps are made. This process continues until the number of variables added to the model reaches the user-specified limit

Auto-Search / GETS

- The auto-search/GETS algorithm is similar to the uni-directional-backwards method.
- The model with all search variables (termed the general unrestricted model, GUM) is estimated, and checked with a set of diagnostic tests.
- A number of search paths are defined, one for each insignificant search variable in the GUM.
- For each path, the insignificant variable is removed and then a series of further variable removal steps is taken, each time removing the most insignificant variable, and each time checking whether the current model passes the set of diagnostic tests.

Auto-Search / GETS

- If the diagnostic tests fail after the removal of a variable, that variable is placed back into the model and prevented from being removed again along this path.
- Variable removal finishes once there are no more insignificant variables, or it is impossible to removal a variable without failing the diagnostic tests.
- Once all paths have been calculated the final models produced by the paths are compared using an information criteria selection. The best model is then selected.

Lasso

- The Lasso (Least Absolute Shrinkage and Selection Operator) estimator is the OLS estimator with an L1 penalty (i.e. absolute values) term that is designed to guard against overfitting:

$$\hat{\beta}_{lasso} = \underbrace{\frac{1}{n}\sum_{j=1}^{n}(y_t - \beta_0 - \sum_{i=1}^{k}\beta_i x_{it})^2}_{OLS} + \lambda \sum_{i=1}^{k}|\beta_i|$$

  with the parameter λ determining the extent of the penalty and the resulting shrinkage toward zero of the parameters.
- We see here that the standard least squares objective function is augmented with a function that penalizes for non-zero coefficients, pushing coefficients toward zero. This is also called shrinkage.