Eco 5316 Time Series Econometrics

Lecture 6 Forecasting

Forecasting

three main components needed to produce a forecast

- information set $\mathcal{I}_t = \{y_0, y_1, \ldots, y_t\}$ at forecast origin t
- forecast horizon h
- loss function $L(y_{t+h} - f_{t,h})$ or $L(e_{t,h})$

where $f_{t,h}$ is the h-step ahead forecast at forecast origin t given information set $\mathcal{I}_t$ and $e_{t,h} = y_{t+h} - f_{t,h}$ is the forecast error

optimal forecast: forecaster wants to construct a forecast f^*_{t+h} that minimizes the expected loss

$$E\left[L(y_{t+h} - f_{t,h})|\mathcal{I}_t\right] = \int L(y_{t+h} - f_{t,h}) f(y_{t+h}|\mathcal{I}_t) dy_{t+h}$$

thus

$$f^*_{t,h} = \arg\min_{f_{t,h}} E\left[L(y_{t+h} - f_{t,h})|\mathcal{I}_t\right]$$

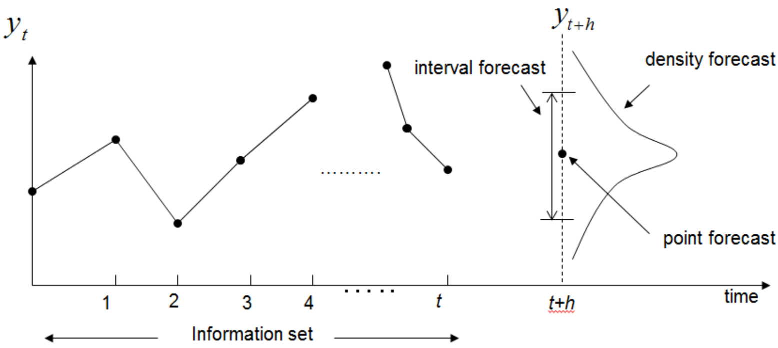
Point, Interval and Density Forecasts

first, we need conditional distribution and moments for y_{t+h} given information set $\mathcal{I}_t$

- conditional probability density function $f(y_{t+h}|\mathcal{I}_t)$
- conditional mean $\mu_{t+h|t} = E_t(y_{t+h}|\mathcal{I}_t)$
- conditional variance $\sigma^2_{t+h|t} = var_t(y_{t+h}|\mathcal{I}_t)$

these will be used to build the point, interval and density forecasts

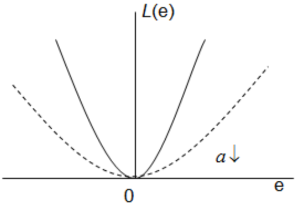
Point, Interval and Density Forecasts



Symmetric Loss Function

Quadratic loss function

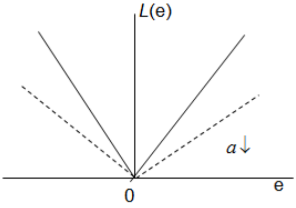
$$L(e) = ae^2, \quad a > 0$$



$$L(e) = L(-e)$$

Absolute value loss function

$$L(e) = a|e|, \quad a > 0$$



$$L(e) = L(-e)$$

Point, Interval and Density Forecasts

suppose that conditional density $f(y_{t+h}|\mathcal{I}_t)$ is $N(\mu_{t+h|t}, \sigma^2_{t+h|t})$ then density forecast is $N(\mu_{t+h|t}, \sigma^2_{t+h|t})$ and

1. if loss function is quadratic $L(e_{t,h}) = ae^2_{t,h}$

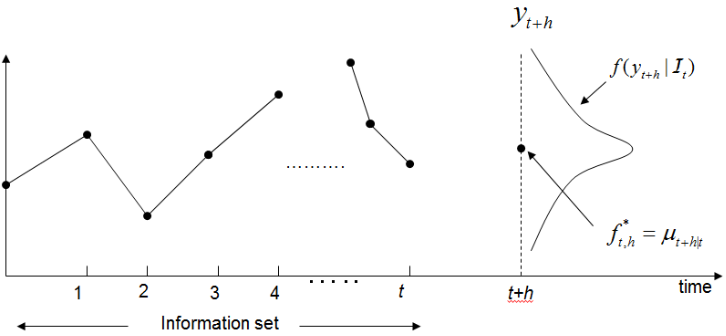
- optimal point forecast is $f^*_{t+h} = \mu_{t+h|t}$
- 95% interval forecast is $\mu_{t+h|t} \pm 1.96\sigma_{t+h|t}$

2. if loss function is absolute value $L(e_{t,h}) = a|e_{t,h}|$

- optimal point forecast is the conditional median $f^*_{t+h} = median(y_{t+h}|\mathcal{I}_t)$

note: if $f(y_{t+h}|\mathcal{I}_t)$ is symmetric then mean and median coincide

Quadratic Loss Function



Example: AR(1) model

suppose that y_t follows an AR(1) model $y_t = \phi_0 + \phi_1 y_{t-1} + \varepsilon_t$ with $\varepsilon_t \sim N(0, \sigma_\varepsilon^2)$ and that $L(e_{t,h}) = ae_{t,h}^2$ then:

- for conditional mean we have

$$\mu_{t+1|t} = E_t(y_{t+1}|\mathcal{I}_t) = \phi_0 + \phi_1 y_t$$

- for conditional variance

$$\sigma_{t+1|t}^2 = var_t(y_{t+1}|\mathcal{I}_t) = var(\varepsilon_{t+1}) = \sigma_\varepsilon^2$$

- thus the 1 step ahead point forecast of y_{t+1} is

$$f_{t,1} = \mu_{t+1|t} = \phi_0 + \phi_1 y_t$$

- the conditional density forecast for y_{t+1} is $N(\phi_0 + \phi_1 y_t, \sigma_\varepsilon^2)$
- the 95% interval forecast is $\mu_{t+1|t} \pm 1.96\sigma_{t+1|t}$ that is $\phi_0 + \phi_1 y_t \pm 1.96\sigma_\varepsilon$

Example: AR(1) model

for forecast step $h \in \{1,2,3,\ldots\}$

- for conditional mean we have

$$\mu_{t+1|t} = E_t(y_{t+1}|\mathcal{I}_t) = \phi_0 + \phi_1 y_t$$
$$\mu_{t+2|t} = E_t(y_{t+2}|\mathcal{I}_t) = \phi_0 + \phi_1 E_t(y_{t+1}|\mathcal{I}_t) = (1+\phi_1)\phi_0 + \phi_1^2 y_t$$
$$\mu_{t+3|t} = E_t(y_{t+3}|\mathcal{I}_t) = \phi_0 + \phi_1 E_t(y_{t+2}|\mathcal{I}_t) = (1+\phi_1+\phi_1^2)\phi_0 + \phi_1^3 y_t$$
$$\vdots$$

  and so $\mu_{t+h|t} \to \frac{\phi_0}{1-\phi_1}$ as $h \to \infty$

- for conditional variance

$$\sigma^2_{t+1|t} = var_t(y_{t+1}|\mathcal{I}_t) = var(\varepsilon_{t+1}) = \sigma^2_\varepsilon$$
$$\sigma^2_{t+2|t} = var_t(y_{h+2}|\mathcal{I}_t) = var(\phi_1 y_{t+1} + \varepsilon_{t+2}|\mathcal{I}_t) = (1+\phi_1^2)\sigma^2_\varepsilon$$
$$\sigma^2_{t+3|t} = var_t(y_{h+3}|\mathcal{I}_t) = var(\phi_1 y_{h+2} + \varepsilon_{t+3}|\mathcal{I}_t) = (1+\phi_1^2+\phi_1^4)\sigma^2_\varepsilon$$
$$\vdots$$

  and so $\sigma^2_{t+h|t} \to \frac{\sigma^2_\varepsilon}{1-\phi_1^2}$ as $h \to \infty$

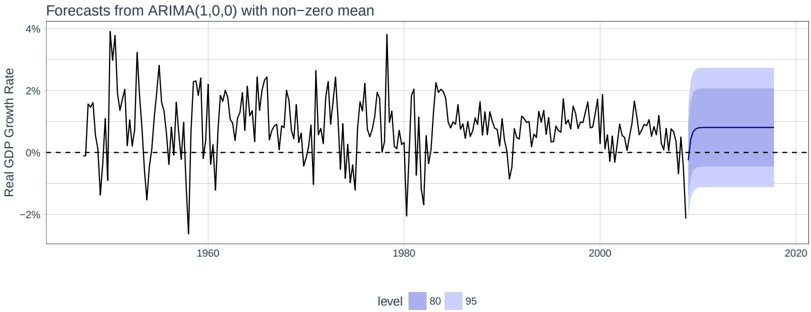
conditional mean thus converges to the unconditional mean, conditional variance converges to the unconditional variance

Example: AR(1) model

```r
library(tidyquant)
library(timetk)

# obtain data on real GDP, construct its log change
data.tbl <-
    tq_get("GDPC1",
           get = "economic.data",
           from = "1947-01-01",
           to = "2017-12-31") %>%
    rename(y = price) %>%
    mutate(dly = log(y) - lag(log(y)))

# split sample - estimation subsample dates
fstQ <- 1947.00  # 1947Q1
lstQ <- 2008.75  # 2008Q4

# convert data into ts, which is the format that Acf, auto.arima and forecast expect
data.ts <- data.tbl %>%
    tk_ts(select = dly, start = fstQ, frequency = 4)

# split sample - estimation and prediction subsamples
data.ts.1 <- data.tbl %>%
    tk_ts(select = dly, start = fstQ, end = lstQ, frequency = 4)
data.ts.2 <- data.ts %>%
    window(start = lstQ + 0.25)
```

Example: AR(1) model

```
# create 1,2,..,h step ahead forecasts, with 2008Q4 as forecast origin
library(forecast)
m1 <- Arima(data.ts.1, order=c(1,0,0))
m1.f.1.to.hmax <- forecast(m1, length(data.ts.2))

# plot the forecast
library(ggplot2)
theme_set(theme_tq())
autoplot(m1.f.1.to.hmax) +
    geom_hline(yintercept = 0, linetype="dashed") +
    scale_y_continuous(labels = scales::percent) +
    labs(x = "", y = "Real GDP Growth Rate")
```



Example: MA(2) model

suppose that y_t follows an MA(2) model $y_t = \phi_0 + \varepsilon_t + \theta_1 \varepsilon_{t-1} + \theta_2 \varepsilon_{t-2}$ with $\varepsilon_t \sim N(0, \sigma_\varepsilon^2)$ and that $L(e_{t,h}) = a e_{t,h}^2$ then:

- for conditional mean we have

$$\begin{aligned}
\mu_{t+1|t} &= E_t(y_{t+1}|\mathcal{I}_t) = \phi_0 + \theta_1 \varepsilon_t + \theta_2 \varepsilon_{t-1} \\
\mu_{t+2|t} &= E_t(y_{t+2}|\mathcal{I}_t) = \phi_0 + \theta_2 \varepsilon_t \\
\mu_{t+3|t} &= E_t(y_{t+3}|\mathcal{I}_t) = \phi_0
\end{aligned}$$

- for conditional variance

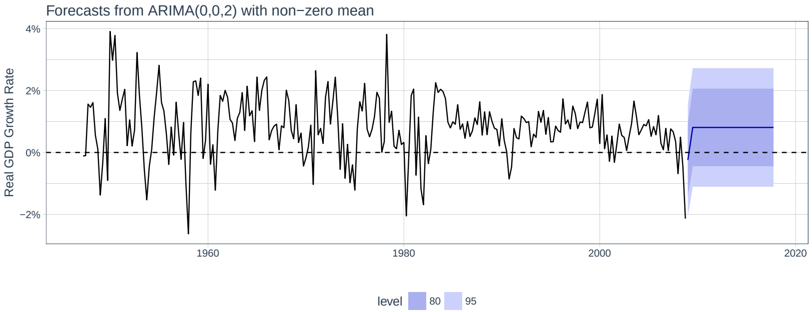
$$\begin{aligned}
\sigma_{t+1|t}^2 &= var_t(y_{t+1}|\mathcal{I}_t) = var(\varepsilon_{t+1}) = \sigma_\varepsilon^2 \\
\sigma_{t+2|t}^2 &= var_t(y_{h+2}|\mathcal{I}_t) = var(\varepsilon_{t+2} + \theta_1 \varepsilon_{t+1}) = (1 + \theta_1^2)\sigma_\varepsilon^2 \\
\sigma_{t+3|t}^2 &= var_t(y_{h+3}|\mathcal{I}_t) = var(\varepsilon_{t+3} + \theta_1 \varepsilon_{t+2} + \theta_2 \varepsilon_{t+1}) = (1 + \theta_1^2 + \theta_2^2)\sigma_\varepsilon^2
\end{aligned}$$

- the 1, 2, and 3 step ahead point forecasts are thus

$$\begin{aligned}
f_{t,1} &= \mu_{t+1|t} = \phi_0 + \theta_1 \varepsilon_t + \theta_2 \varepsilon_{t-1} \\
f_{t,2} &= \mu_{t+2|t} = \phi_0 + \theta_2 \varepsilon_t \\
f_{t,3} &= \mu_{t+3|t} = \phi_0
\end{aligned}$$

Example: MA(2) model

```r
m2 <- Arima(data.ts.1, order=c(0,0,2))
m2.f.1.to.hmax <- forecast(m2, length(data.ts.2))

autoplot(m2.f.1.to.hmax) +
    geom_hline(yintercept = 0, linetype="dashed") +
    scale_y_continuous(labels = scales::percent) +
    labs(x = "", y = "Real GDP Growth Rate")
```



Forecasting using ARMA(p, q) models

ARMA(p, q) models are mostly suitable for forecasts with a small step h, forecasts of distant future are not particularly accurate

forecast based on an AR(p) model:

- conditional mean converges to unconditional mean gradually
- conditional variance converges to unconditional variance gradually

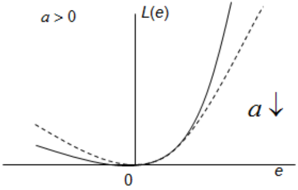
forecast based on an MA(q) model:

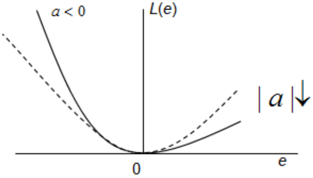
- once $h > q$ the conditional mean jumps straight to unconditional mean
- once $h > q$ the conditional variance jumps straight to unconditional variance

Asymmetric Loss Function

Linex function

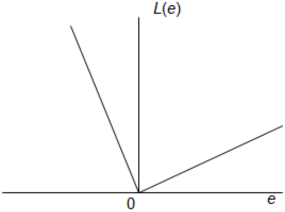
$$L(e) = \exp(ae) - ae - 1, \quad a \neq 0$$





Lin-lin function

$$L(e) = \begin{cases} a|e| & e > 0 \\ b|e| & e \leq 0 \end{cases}$$



$$b > a \rightarrow L(-e) > L(e)$$

Point, Interval and Density Forecasts

suppose that conditional density $f(y_{t+h}|\mathcal{I}_t)$ is $N(\mu_{t+h|t}, \sigma^2_{t+h|t})$ so that density forecast is $N(\mu_{t+h|t}, \sigma^2_{t+h|t})$ and

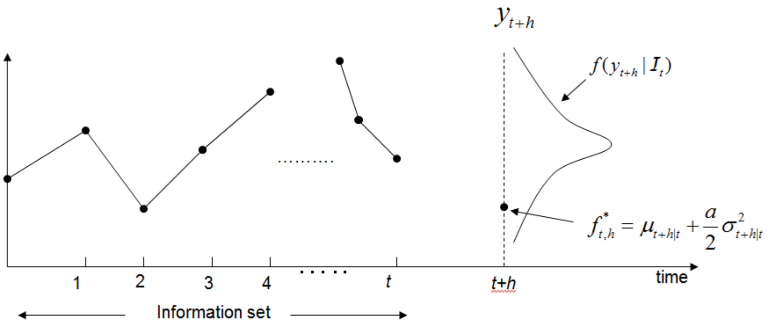
1. if loss function is linex $L(e_{t,h}) = exp(ae_{t,h}) - ae_{t,h} - 1$
   - optimal point forecast is $f^*_{t+h} = \mu_{t+h|t} + \frac{a}{2}\sigma^2_{t+h|t}$
2. if loss function is linlin

$$L(e_{t,h}) = \begin{cases} a|e_{t,h}| & \text{if } e_{t,h} < 0 \\ (1-a)|e_{t,h}| & \text{if } e_{t,h} \geq 0 \end{cases}$$

   - optimal point forecast is conditional quintile $f^*_{t+h} = q_a(y_{t+h}|\mathcal{I}_t)$

thus **for asymmetric loss function optimal forecast is actually biased** - on average forecast error is either positive or negative

Linex Loss Function



Example: AR(1) model

suppose that y_t follows an AR(1) model $y_t = \phi_0 + \phi_1 y_{t-1} + \varepsilon_t$ with $\varepsilon_t \sim N(0, \sigma_\varepsilon^2)$ and that $L(e_{t,h}) = exp(ae_{t,h}) - ae_{t,h} - 1$ then:

- for conditional mean we have

$$\mu_{t+1|t} = E_t(y_{t+1}|\mathcal{I}_t) = \phi_0 + \phi_1 y_t$$

- for conditional variance

$$\sigma^2_{t+1|t} = var_t(y_{t+1}|\mathcal{I}_t) = var(\varepsilon_{t+1}) = \sigma^2_\varepsilon$$

- thus the 1 step ahead point forecast of y_{t+1} is

$$f_{t,1} = \mu_{t+1|t} + \frac{a}{2}\sigma^2_{t+1|t} = \phi_0 + \phi_1 y_t + \frac{a}{2}\sigma^2_\varepsilon$$

- the conditional density forecast for y_{t+1} is $N(\phi_0 + \phi_1 y_t, \sigma^2_\varepsilon)$

Evaluating Accuracy of Forecasts

general idea:

- split sample into two parts:
  estimation sample $y_1, \ldots, y_t$
  prediction sample $y_{t+1}, \ldots, y_T$
- estimate the model using the first subsample
- evaluate **in-sample accuracy** - compare fitted values $\hat{y}_1, \ldots, \hat{y}_t$ with actual values $y_1, \ldots, y_t$
- use the second subsample to construct set of h step ahead forecasts $f_{t,h}, f_{t+1,h}, \ldots, f_{T-h,h}$
- evaluate **out-of-sample accuracy** - compare forecasts $f_{t,h}, f_{t+1,h}, \ldots, f_{T-h,h}$ with actual values $y_{t+h}, y_{t+1+h}, \ldots, y_T$

In-Sample Evaluation of Accuracy

given the fitted values $\hat{y}_j$ from the model, and in sample residuals $e_j = y_j - \hat{y}_j$

Mean Error - measure of the average bias

$$ME = \frac{1}{t}\sum_{j=0}^{t} e_j$$

Mean Squared Error - sample average loss for quadratic loss function

$$MSE = \frac{1}{t}\sum_{j=0}^{t} e_j^2$$

Mean Absolute Error - sample average loss for absolute value loss function

$$MAE = \frac{1}{t}\sum_{j=0}^{t} |e_j|$$

Mean Absolute Percentage Error

$$MAPE = \frac{1}{t}\sum_{j=0}^{t} \left|\frac{e_j}{y_j}\right|$$

Mean Absolute Scaled Error - compares in sample MAE of the model forecast with in sample MAE for one-step naive forecast method $\hat{y}_{j+1} = y_j$

$$MASE = \frac{\frac{1}{t}\sum_{j=0}^{t} |e_j|}{\frac{1}{t-1}\sum_{j=1}^{t-1} |y_{j+1} - y_j|}$$

In-Sample Evaluation of Accuracy

```r
m1 <- Arima(data.ts.1, order = c(1,0,0))
accuracy(m1)
```

```
##                        ME        RMSE         MAE      MPE     MAPE      MASE        ACF1
## Training set 1.429256e-05 0.009159989 0.006768226 -44.3292 166.0401 0.6358981 -0.03028121
```

Out-of-Sample Evaluation of Accuracy

given out of sample forecast errors $e_{t,h}, e_{t+1,h}, \ldots, e_{T-h,h}$

Mean Error

$$ME = \frac{1}{T-t-h+1} \sum_{j=0}^{T-h-t} e_{t+j,h}$$

Mean Squared Error

$$MSE = \frac{1}{T-t-h+1} \sum_{j=0}^{T-h-t} e_{t+j,h}^2$$

Mean Absolute Error

$$MAE = \frac{1}{T-t-h+1} \sum_{j=0}^{T-h-t} |e_{t+j,h}|$$

Mean Absolute Percentage Error

$$MAPE = \frac{1}{T-t-h+1} \sum_{j=0}^{T-h-t} \left| \frac{e_{t+j,h}}{y_{t+j+h}} \right|$$

Mean Absolute Scaled Error

$$MASE = \frac{\frac{1}{T-l-h+1} \sum_{j=0}^{T-h-t} |e_{t+j,h}|}{\frac{1}{t-h} \sum_{j=1}^{t-h} |y_{j+h} - y_j|}$$

Out-of-Sample Evaluation of Accuracy - Forecasting Schemes

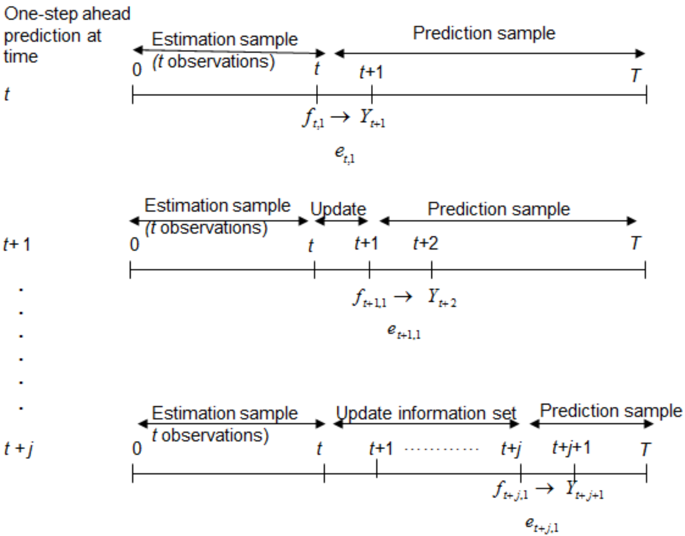
out of sample forecasts and forecast errors used to calculate ME, MSE, MAE, MPE, MAPE, ... can be constructed using one of the three schemes:

- fixed scheme
- recursive scheme
- rolling scheme

Forecasting Schemes

Fixed scheme example for one step ahead forecast:

model is estimated only once, each one step ahead forecast is constructed using same parameters



Out-of-Sample Evaluation of Accuracy - Fixed Scheme

```r
# estimate AR(1) model 1947Q2 to 2008Q4
m1 <- Arima(y = data.ts.1, order = c(1,0,0))
# create 1-step ahead forecasts - forecast origin is moving from 2008Q4 to 2017Q3
# but always use same estimated model m1 so this is a fixed forecasting scheme
m1.f.1 <- Arima(y = data.ts, model = m1)
# evaluate accuracy of 1-step ahead forecast throughout the whole sample 1947Q2 to 2016Q4
accuracy(fitted(m1.f.1), data.ts)
```

```
##                     ME        RMSE         MAE       MPE     MAPE        ACF1 Theil's U
## Test set -0.0002269938 0.008719256 0.006378762 -51.41388 166.8595 -0.02181353 0.9346668
```

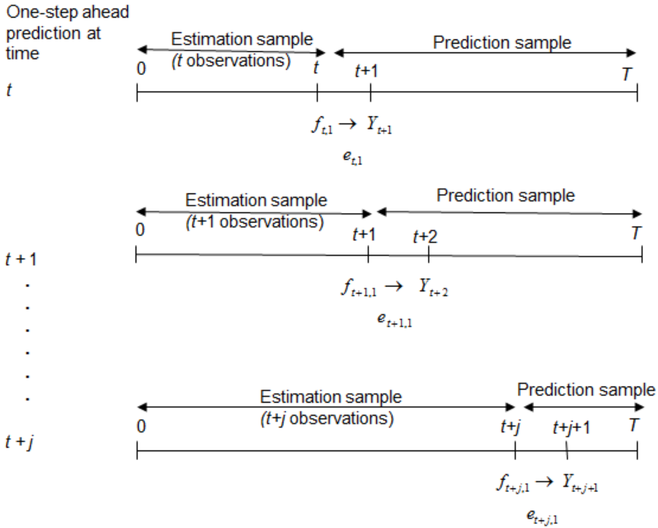
```r
# evaluate accuracy of out-of-sample 1-step ahead forecasts
accuracy(fitted(m1.f.1), data.ts.2)
```

```
##                    ME        RMSE         MAE       MPE     MAPE       ACF1 Theil's U
## Test set -0.001882486 0.004686156 0.003706608 -100.0227 172.4811 -0.3309034 0.3467054
```

Forecasting Schemes

Recursive scheme example for one step ahead forecast:

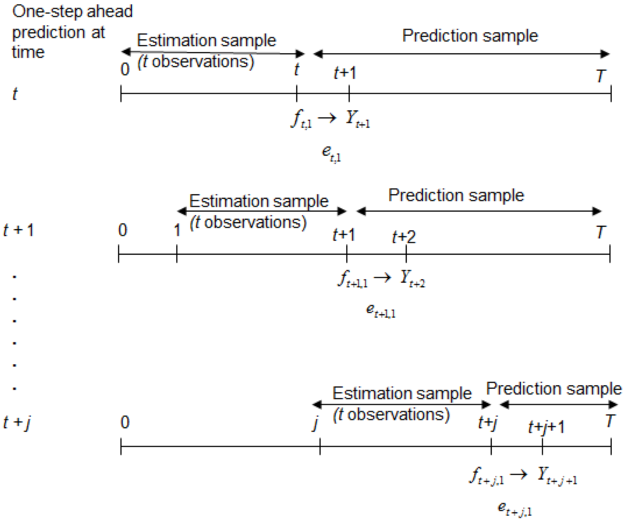
estimation sample keeps expanding and model is re-estimated again when each new observation is added to the estimation sample



Forecasting Schemes

Rolling scheme example for one step ahead forecast:

estimation sample always contains the same number of observation and model is re-estimated again within each rolling sample

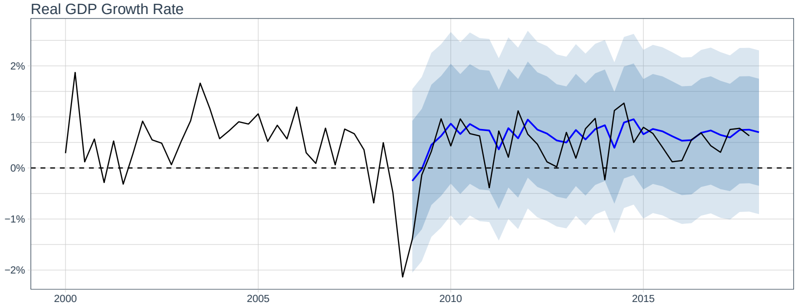


Out-of-Sample Evaluation of Accuracy - Rolling scheme

```r
library(tibbletime)
library(sweep)

# window size
window.length <- length(data.ts.1)
# create rolling ARMA function with rollify from tibbletime package
roll_Arima <- rollify(~Arima(.x, order=c(1,0,0)), window = window.length, unlist = FALSE)

# estimate rolling ARMA model, create 1 period ahead rolling forecasts
h <- 1
results <-
    data.tbl %>%
    as_tbl_time(index = date) %>%
    mutate(arma.model = roll_Arima(dly)) %>%
    filter(!is.na(arma.model)) %>%
    mutate(arma.f = map(arma.model, (. %>% forecast(h) %>% sw_sweep())))

# extract the 1 period ahead rolling forecasts
m1.f.1.rol <-
    results %>%
    select(date, arma.f) %>%
    unnest(arma.f) %>%
    filter(key == "forecast") %>%
    mutate(date = date %m+% months(3))
```

Out-of-Sample Evaluation of Accuracy - Rolling Scheme

```
# plot the 1 period ahead rolling forecasts
m1.f.1.rol %>%
    ggplot(aes(x = date, y = value)) +
        geom_ribbon(aes(ymin = lo.95, ymax = hi.95), color = NA, fill = "steelblue", alpha = 0.2) +
        geom_ribbon(aes(ymin = lo.80, ymax = hi.80), color = NA, fill = "steelblue", alpha = 0.3) +
        geom_line(size = 0.7, col = "blue") +
        geom_line(data = (data.tbl %>% filter(year(date) > 1999)), aes(x = date, y = dly)) +
        geom_hline(yintercept = 0, linetype="dashed") +
        scale_y_continuous(labels = scales::percent) +
        scale_color_manual(values = c("black","darkblue")) +
        labs(x = "", y = "", title = "Real GDP Growth Rate") +
        theme(legend.position = "none")
```



Forecasting Schemes - Comparison

advantages and disadvantages of the three schemes:

fixed scheme

- fast and convenient because - there is one and only one estimation
- does not allow for parameter updating, so again problem with structural breaks and model's stability

recursive scheme

- incorporates as much information as possible in the estimation of the model
- advantageous if the model is stable over time
- if the data have structural breaks, model's stability is compromised and so is the forecast

rolling scheme

- avoids the potential problem with the model's stability
- more robust against structural breaks in the data
- does not make use of all the data

Comparison

```r
# multistep forecast
accuracy(m1.f.1.to.hmax$mean, data.ts.2)
```

```
##                    ME        RMSE         MAE       MPE     MAPE        ACF1 Theil's U
## Test set -0.002916416 0.004895071 0.003811905 -144.8972 222.8371 -0.03828393 0.3542732
```

```r
# 1 step ahead fixed scheme forecast
accuracy(fitted(m1.f.1), data.ts.2)
```

```
##                    ME        RMSE         MAE       MPE     MAPE       ACF1 Theil's U
## Test set -0.001882486 0.004686156 0.003706608 -100.0227 172.4811 -0.3309034 0.3467054
```

```r
# 1 step ahead rolling scheme forecast
accuracy(m1.f.1.rol$value, data.ts.2)
```

```
##                    ME       RMSE        MAE       MPE     MAPE      ACF1 Theil's U
## Test set -0.001667329 0.00463774 0.00366208 -94.00967 166.0779 -0.335279 0.3705797
```