

# 1 Strategies of martix construction

Suppose a large set of time series  $\mathfrak{D} = \{\mathbf{s}\}$  is given. The “object-feature” matrix  $\mathbf{X}^*$  for the multiscale autoregressive problem statement is composed of row-vectors

$$\mathbf{s}'_i = [\mathbf{y}'_i, \mathbf{x}'_i] = [\underbrace{s(t_i), \dots, s(t_i - \Delta t_r)}_{\mathbf{y}'_i}, \underbrace{s(t_i - \Delta t_r - \Delta t_p), \dots, s(t_i - \Delta t_r - \Delta t_p - \Delta t_p)}_{\mathbf{x}'_i}],$$

where  $s(t)$  is an element of time series  $\mathbf{s}$ . Consider several strategies to decompose time series  $\mathbf{s}$  into segments  $\Delta t_i = (t_i, \dots, t_i - \Delta t_r - \Delta t_p)$  to construct matrix  $\mathbf{X}^*$ .

1. Row vectors  $\mathbf{s}_i$  cover time series without intersections. Let  $\{T_{\max}, \dots, 1\}$  be the set of indices of tim series  $\mathbf{s}$ , then thee strategy of selecting  $t_i$  holds the following:

$$\{T_{\max}, \dots, 1\} = \bigsqcup_{i=1}^M \Delta t_i. \quad (1)$$

It follows from (1) that  $|t_{i+1} - t_i| > \Delta t_r + \Delta t_p$  for any  $i = 1, \dots, M - 1$ .

2. Row vectors  $\mathbf{s}_i = [\mathbf{y}_i, \mathbf{x}_i]$  overlap, but target parts  $\mathbf{y}_i$  do not intersect:

$$\{T_{\max}, \dots, 1\} = \bigsqcup_{i=1}^M M - 1 \{t_i, \dots, t_i - \Delta t_r\} \Rightarrow |t_{i+1} - t_i| > \Delta t_r. \quad (2)$$

3. For each time stamp  $t_i$  of the least frequent regular sampling there is correspondent row vector  $\mathbf{s}_i$  in  $\mathbf{X}^*$ :

$$\{T_{\max}, \dots, 1\} = \bigcup_i t_i.$$

4. Time intervals  $\Delta t_i$  are selected randomly.

5. Potentially, other sensible strategies are possible.

Vector  $\boldsymbol{\varepsilon} \in \mathbb{R}^{\Delta t_r}$  of model residuals at time stamp  $t_i$  is given by

$$\boldsymbol{\varepsilon}_i = \mathbf{y}_i - \hat{\mathbf{y}}_i.$$

Dependent on the way the matrix  $\mathbf{X}^*$  is designed, there might be dependencies between components of subsequent vectors  $\mathbf{y}_i, \mathbf{y}_{i+1}$ . If there are such  $i, i' \in \mathcal{B}$  that  $|t_i - t_{i'}| < \Delta t_r$ , vectors  $\boldsymbol{\varepsilon}_i$  and  $\boldsymbol{\varepsilon}_{i'}$  overlap or contain residuals for the same time stamp. In this case define the test vector of residuals as

$$\boldsymbol{\varepsilon}(\mathcal{B}) = \left\{ \bar{\varepsilon}_t \left| t \in \bigcup_{i \in \mathcal{B}} \{i - \Delta t_r, \dots, i\} = \{t_{i_{\min}} - \Delta t_r, \dots, t_{i_{\max}}\} \right. \right\},$$

where  $\bar{\varepsilon}_t$  is the average residual for the time stamp  $t$ .

To avoid these issues, we fix the second strategy of the  $\mathbf{X}$  construction.

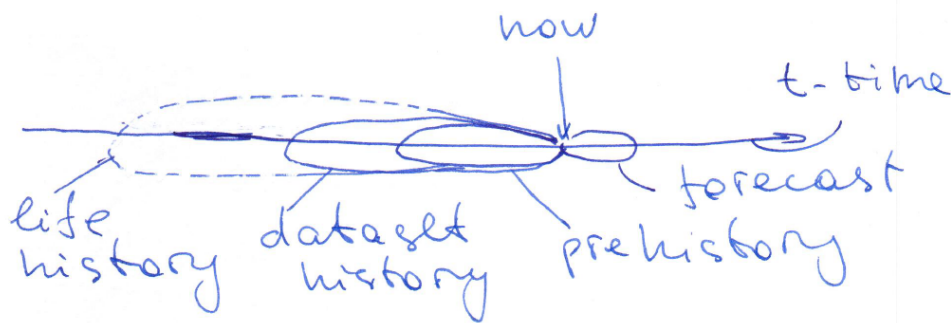


Figure 1: Online forecasting paradigm.

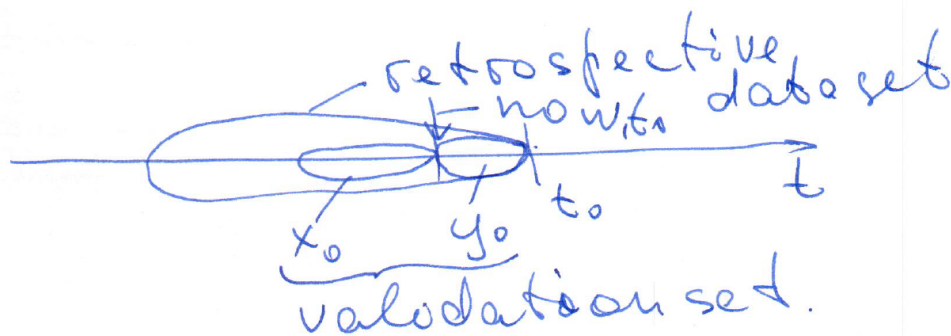


Figure 2: Retrospective forecast includes most recent samples in data set.

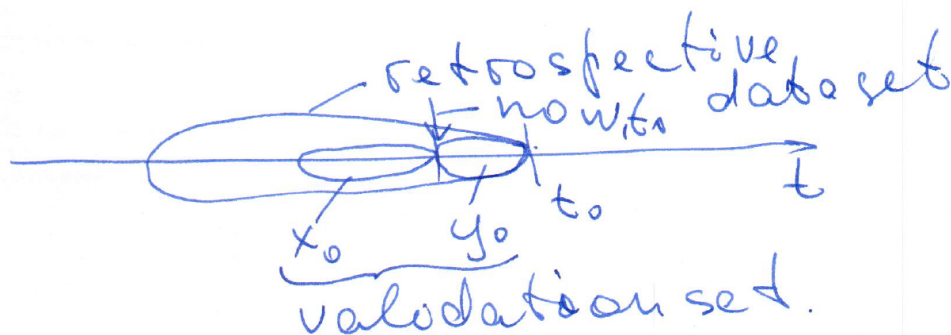


Figure 3: Retrospective forecast includes most recent samples in data set.

## 2 Forecast analysis

We consider the forecast testing procedure, given by the algorithm 1. Our ultimate goal is to construct a forecasting model be able to obtain forecasts at any given time  $t$  ???. To construct such model we have to imitate this setting, using the so-called retrospective forecast 1 (rolling forecasts, walking ahead predictions). Here we conceal most recent

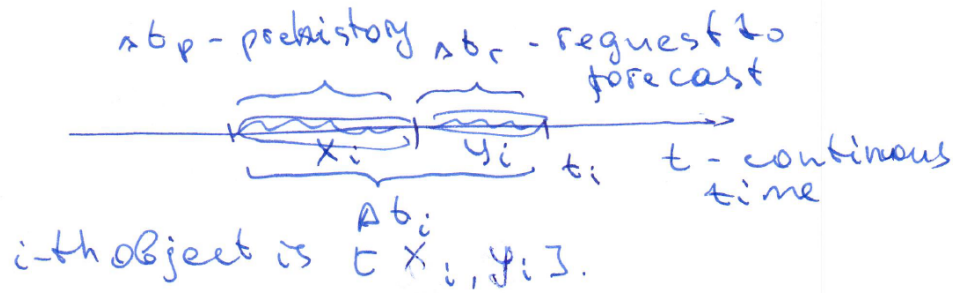


Figure 4: Draw an object from time series history.

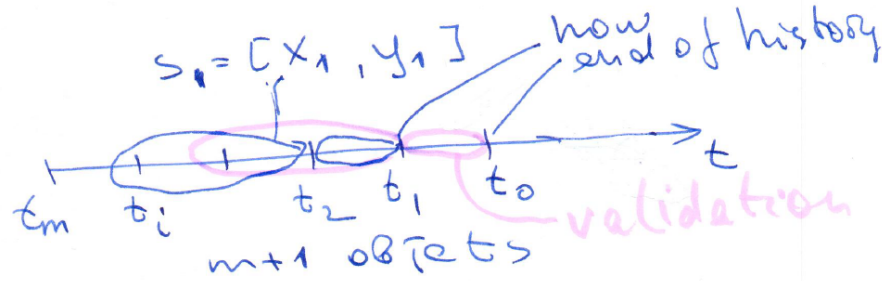


Figure 5: Design matrix generation.

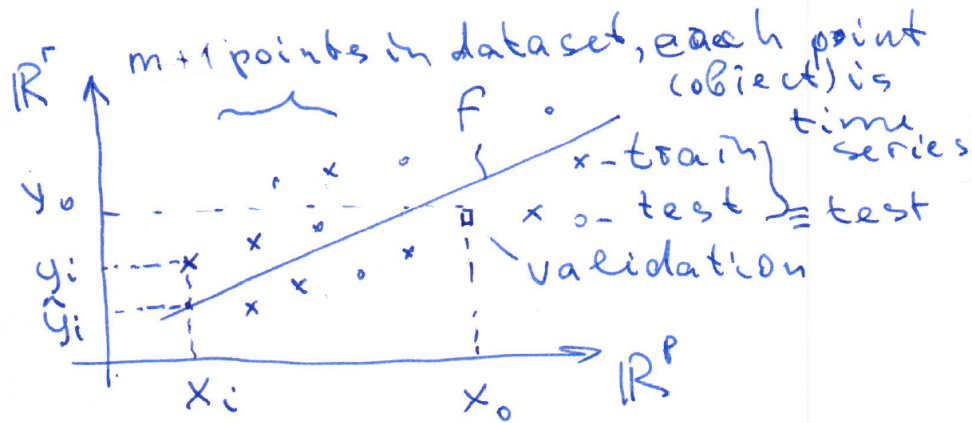


Figure 6: Forecasting as regression problem.

historical samples and make predictions as if they were unknown. Then the quality of the model is evaluated according to its performance on these concealed samples. In this paper we convert the forecasting problem to regression problem, considering the segments of time series as objects of the design matrix  $\mathbf{X}^*$  1. Sliding 1 the "current" time point  $t$  according to the chosen strategy (1–5), we obtain the design matrix  $\mathbf{X}$  and further we solve the regression problem instead of forecasting problem ??.

### ***ComputeForecastingErrors()***

**Data:**  $\mathbf{X}^* \in \mathbb{R}^{M \times (\Delta t_r + \Delta t_p)}$ . Parameters: sample size  $m$ , train to test ratio  $\alpha$ .

**Result:** Forecasting quality: root-mean-squared error.

**while**  $n \leq M - m$ : **do**

define,  $\mathbf{X}_n^* = [\mathbf{x}_n^*, \dots, \mathbf{x}_{m+n-1}^*]^\top$  ;

$\mathbf{X}_{\text{train}}, \mathbf{X}_{\text{test}}, \mathbf{X}_{\text{val}} = \text{TrainTestSplit}(\mathbf{X}_n^*, \alpha)$ ;

train forecasting model  $\mathbf{f}(\mathbf{x}, \hat{\mathbf{w}}_n)$ , using  $\mathbf{X}_{\text{train}}$  and  $\mathbf{X}_{\text{test}}$ ;

obtain vector of residuals  $\boldsymbol{\varepsilon} = [\varepsilon_T, \dots, \varepsilon_{T-\Delta t_r+1}]$  with respect to  $\mathbf{X}_{\text{val}}$  ;

compute forecasting quality:

$$\text{RMSE}(n) = \sqrt{\frac{1}{\Delta t_r} \sum_{t=0}^{\Delta t_r} \varepsilon_{T-t}^2};$$

$n = n + 1$  ;

**end**

Average RMSE by data splits.

### ***TrainTestSplit()***

**Data:** Object-feature matrix  $\mathbf{X}^* \in \mathbb{R}^{m \times (\Delta t_r + \Delta t_p)}$ . Train to test ratio  $\alpha$ .

**Result:** Train, test, validation matrices  $\mathbf{X}_{\text{train}}^*, \mathbf{X}_{\text{test}}^*, \mathbf{X}_{\text{val}}^*$ .

Set train set and test set sizes:

$$m_{\text{train}} = \lfloor \alpha(m-1) \rfloor ;$$

$$m_{\text{test}} = m - 1 - m_{\text{train}} ;$$

Decompose matrix  $\mathbf{X}^*$  into train, test, validation matrices  $\mathbf{X}_{\text{train}}^*, \mathbf{X}_{\text{test}}^*, \mathbf{X}_{\text{val}}^*$ :

$$\mathbf{X}_{\text{train}}^* = \left[ \begin{array}{c} \mathbf{x}_{\text{val}}^* \in \mathbb{R}^{1 \times (\Delta t_r + \Delta t_p)} \\ \mathbf{X}_{m_{\text{test}}}^* \in \mathbb{R}^{m_{\text{test}} \times (\Delta t_r + \Delta t_p)} \\ \mathbf{X}_{m_{\text{train}}}^* \in \mathbb{R}^{m_{\text{train}} \times (\Delta t_r + \Delta t_p)} \end{array} \right] = \left[ \begin{array}{c|c} \mathbf{y}_{\text{val}} & \mathbf{x}_{\text{val}} \\ \mathbf{Y}_{m_{\text{test}}} & \mathbf{X}_{m_{\text{test}}} \\ \mathbf{Y}_{m_{\text{train}}} & \mathbf{X}_{m_{\text{train}}} \end{array} \right]$$

**Algorithm 1:** Train-test split.

## **2.1 Ensuring forecast model validity**

A valid forecast model must meet the following conditions:

- Mean of residuals equals to zero.
- Residuals are stationary.
- Residuals are not autocorrelated.

If the forecast does not meet any of these conditions, then it can be further improved by simply adding a constant (minus residual mean) to the model, balancing variance or including more lags. Additionally, desirable properties are normality and homoscedasticity

of residuals. These properties are not necessary for an adequate model, but allow to obtain theoretical estimations of the confidence interval.

## **2.2 Forecasting quality**