VIRTUAL *K*-FOLD CROSS VALIDATION:
A COMPUTATION-AWARE METHOD FOR ACCURACY ASSESSMENT

Cesare Alippi, Manuel Roveri
DEI, Politecnico di Milano, Piazza L. Da Vinci 32, 20133 Milano, Italy
Alippi, Roveri@elet.polimi.it

LOO and k -fold cross validation are widely used and effective model accuracy evaluation methods but suffer from the high computational load associated with the required training of multiple models. Such an issue has been solved for LOO in [1], where the concept of Virtual LOO has been suggested. There, accuracy is estimated without requiring re-train of different models (parameters and performance are estimated by relying on leverages).

Based on those outcomes, we provide a not trivial extension of the virtual LOO approach to generate a virtual estimate of the k -fold cross validation method for model assessment. Again, the estimation process is virtual in the sense that no training phases are required to evaluate the k -fold cross validation performance of the model.

Removal of a sample: Virtual LOO

The operational framework is the traditional one of regression analysis with N $\{(\mathbf{x}_1,y_1),\ldots,(\mathbf{x}_N,y_N)\}=Z_N$ *i.i.d.* samples $(\mathbf{x},y)$, $\mathbf{x}\in R^d$, $y\in R$ used to estimate the unknown $f(\mathbf{x})$ function with the parameterized family of models $\{g(\mathbf{x},\boldsymbol{\theta}),\ \boldsymbol{\theta}\in R^p\}$. We assume the traditional additive signal plus noise framework $y=f(\mathbf{x})+\eta$, η being a zero-mean random variable accounting for the noise. The performance figure of merit is the squared error (SE) but other figures of merit can be easily considered $L=\mathbf{r}^T\mathbf{r}$, T denotes the matrix transposition operator and $\mathbf{r}$ is the $N-$dimensional column vector of residuals $r_i=y_i-g(\mathbf{x_i},\boldsymbol{\theta})$, $i=1,\ldots,N$.

Consider a linear model $g(\mathbf{x},\boldsymbol{\theta})=\mathbf{x^T}\boldsymbol{\theta}$ and its least square solution $\hat{\boldsymbol{\theta}}_{\mathbf{o}}=\left(\mathbf{X^T X}\right)^{\mathbf{-1}}\mathbf{X^T y}$ where $\mathbf{y}$ is the $N\times 1$ vector of the y s and $\mathbf{X}$ the $N\times d$ matrix containing the observed $\mathbf{x}$ vectors. Under the assumption that inputs are independent, $\mathbf{X^T X}$ is a positive definite matrix. Removal of a generic i -th sample of the training set provides the new data set $Z_N^{(-i)}=Z_N\setminus\{(\mathbf{x_i},y_i)\}$ ($\setminus$ is the set subtraction operator) from which we derive a new parameter vector (or model)

$$\hat{\boldsymbol{\theta}}_{\mathbf{o}}^{(\mathbf{-i})}=\hat{\boldsymbol{\theta}}_{\mathbf{o}}-r_i\left(\mathbf{X^T X}-\mathbf{x_i}\mathbf{x_i^T}\right)^{-1}\mathbf{x_i} \tag{1}$$

with an associated ($N\times 1$) residual vector $\mathbf{r^{(-i)}}=\mathbf{y}-\mathbf{X}\hat{\boldsymbol{\theta}}_{\mathbf{o}}^{(\mathbf{-i})}=\mathbf{r}-\mathbf{X}\left(\hat{\boldsymbol{\theta}}_{\mathbf{o}}^{(\mathbf{-i})}-\hat{\boldsymbol{\theta}}_{\mathbf{o}}\right)$. The effect of the i -th training sample on the sole residual is [1],

$$r_i^{(-i)}=\frac{r_i}{1-h_{ii}} \tag{2}$$

where $h_{ii}=\mathbf{x_i^T}\left(\mathbf{X^T X}\right)^{\mathbf{-1}}\mathbf{x_i}$ is the diagonal element of the orthogonal projection matrix $\mathbf{H}=\mathbf{X}\left(\mathbf{X^T X}\right)^{\mathbf{-1}}\mathbf{X^T}$. Said that, the virtual LOO estimate of LOO can be defined as

$$e_{V-LOO}=\frac{1}{N}\sum_{i=1}^{N}\left(r_i^{(-i)}\right)^2=\frac{1}{N}\sum_{i=1}^{N}\left(\frac{r_i}{1-h_{ii}}\right)^2 \tag{3}$$

In the case of a non linear model we linearize it by considering a second-order Taylor expansion in the neighborhood of $\hat{\boldsymbol{\theta}}_{\mathbf{o}}$ and evaluate it on $\hat{\boldsymbol{\theta}}_{\mathbf{o}}^{(\mathbf{-i})}$: $\mathbf{g}\left(\mathbf{x},\hat{\boldsymbol{\theta}}^{(\mathbf{-i})}\right)\cong\mathbf{g}\left(x,\hat{\boldsymbol{\theta}}\right)+\mathbf{J}\left(\hat{\boldsymbol{\theta}}^{(\mathbf{-i})}-\hat{\boldsymbol{\theta}}\right)$. The relationship holds under the assumption that withdrawal of a sample from the training set has a small effect on $\hat{\boldsymbol{\theta}}_{\mathbf{o}}$; $\mathbf{J}$ is the Jacobian matrix of $g(\mathbf{x},\boldsymbol{\theta})$. Under the previous assumption [1][3], $\mathbf{H}$ to be considered is $\mathbf{H}=\mathbf{J}\left(\mathbf{J^T J}\right)^{\mathbf{-1}}\mathbf{J^T}$.

Multiple removal of samples: Virtual k-Fold Cross Validation

Whereas the virtual LOO requires removal of a sample, the virtual k-fold cross validation needs to *simultaneously* remove $l = \lceil N/k \rceil, l \geq 1$ samples and estimate the model performance. Unfortunately, we cannot simply iteratively apply removal of a sample *l* times and then use results (1) to (3) since removal of a sample has an impact not only on its residual but also on other residuals. As such, we need at first to derive the recurrent formula evaluating the impact of removal of the *i*-th sample on the residual of a generic *j*-th sample and then iterate such procedure up to *l*. Starting from (1) and considering the Sherman-Morrison matrix inversion lemma we derive that,

$$r_j^{(-i)} = r_j + r_i \frac{h_{ij}}{1-h_{ii}} \quad , \quad \text{with } j = 1,..,N \tag{4}$$

and h_{ij} is the *i,j*-th element of $\mathbf{H}$. Of course, when *j*=*i* (4) reduces to the (2). From the Sherman-Morrison Lemma, the effect of removal of the i-th sample on the projection matrix $\mathbf{H}$ is

$$\mathbf{H^{(-i)}} = \mathbf{\widetilde{H}^{(-i)}} + \frac{1}{1-h_{ii}} \mathbf{v_i v_i^T} \tag{5}$$

where $\mathbf{\widetilde{H}^{(-i)}} = \mathbf{X^{(-i)}}\left(\mathbf{X^T X}\right)^{\mathbf{-1}} \mathbf{X^{(-i)T}}, \mathbf{X^{(-i)}} = \left[\mathbf{x_1} \ \ \mathbf{..} \ \ \mathbf{x_{i-1}} \ \ \mathbf{0} \ \ \mathbf{x_{i+1}} \ \ \mathbf{...} \ \ \mathbf{x_N}\right]^T$ and $\mathbf{v_i} = \left[h_{1i} \ \ \ldots \ \ h_{i-1,i} \ \ 0 \ \ h_{i+1,i} \ \ \cdots \ \ h_{N,i}\right]^T$. Having evaluated the full effect of the removal of a sample on all residuals we extend the procedure to evaluate the effects on $\mathbf{H}$ induced by the removal of multiple samples. As such, having extracted $\mathbf{dim(t)}$ samples ($\mathbf{t}$ is the vector containing the indexes of the previously removed samples), we recursively remove the additional *i*-th sample and obtain

$$\mathbf{H}^{(-\mathbf{t}-i)} = \mathbf{\widetilde{H}}^{(-\mathbf{t}-i)} + \frac{1}{1-h_{ii}^{(-\mathbf{t})}} \mathbf{v_i^{(-t)}}\left(\mathbf{v_i^{(-t)}}\right)^T . \tag{6}$$

At the beginning, when $\mathbf{t}$ is void, (6) reduces to (5). Similarly, the procedure is applied to cover residuals and parameters updates and produces the recurrent expressions

$$r_j^{(-\mathbf{t}-i)} = r_j^{(-\mathbf{t})} + r_i^{(-\mathbf{t})} \frac{h_{ij}^{(-\mathbf{t})}}{1-h_{ii}^{(-\mathbf{t})}} \tag{7}$$

$$\hat{\boldsymbol{\theta}}_{\mathbf{0}}^{(-\mathbf{t}-i)} = \hat{\boldsymbol{\theta}}_{\mathbf{0}}^{(-\mathbf{t})} - \left(\left(\mathbf{X}^{(-\mathbf{t})}\right)^{\mathbf{T}} \mathbf{X}^{(-\mathbf{t})}\right)^{-1} \left(\mathbf{x_i}\right)^T \frac{r_i^{(-\mathbf{t})}}{1-h_{ii}^{(-\mathbf{t})}} \tag{8}$$

Obviously, at the beginning, $\mathbf{t}$ is void and Eq . (7) and (8) reduce to (4) and (1), respectively.
In the virtual k-fold cross validation case, Z_N is initially randomly split into k disjoint subsets of size N/k. In this framework define $\mathbf{d_j}$, $j = 1,..,k$ to be the vector containing the indexes of training samples that belong to the j-th subset. For each subset $\mathbf{d_j}$, we compute $\hat{\boldsymbol{\theta}}_{\mathbf{0}}^{(-\mathbf{d_j})}$ with Eq. (8) and the virtual k-fold cross validation estimate is

$$e_{Vk-CV} = \frac{1}{k} \sum_{j=1}^{k} \left(\mathbf{y_{d_j}} - \mathbf{g}\left(\mathbf{X_{d_j}}, \hat{\boldsymbol{\theta}}_{\mathbf{0}}^{(-\mathbf{d_j})}\right)\right)^2$$

where $\left\{\mathbf{X_{D_j}}, \mathbf{y_{D_j}}\right\}$ are the N/k training input-output pairs of the elements belonging to $\mathbf{d_j}$.
The extension to non linear models follows the approach already introduced in the previous section; the $\mathbf{H}$ to be considered in equations (4-8) is $\mathbf{H} = \mathbf{J}\left(\mathbf{J^T J}\right)^{\mathbf{-1}} \mathbf{J^T}$.

Results, together with mathematical details, will be presented at the workshop.

References:

[1] G. Monari and G. Dreyfus. “Withdrawing an example from the training set: an analytic estimation of its effect on a nonlinear parameterised model”. *Neurocomputing*, 35: 195-201, 2000.

[2] David C. Hoaglin, Roy E. Welsch, "The Hat Matrix in Regression and ANOVA" The American Statistician, Vol. 32, No. 1 (Feb., 1978), pp. 17-22.

[3] Roy T. St. Laurent, R. Dennis Cook, "Leverage and Superleverage in Nonlinear Regression", Journal of the American Statistical Association, Vol. 87, No. 420 (Dec., 1992), pp. 985-990