Performance Evaluation

Pedro Larrañaga, Concha Bielza

Computational Intelligence Group
Artificial Intelligence Department
Universidad Politécnica de Madrid

Machine Learning

Master in Data Science + Master HMDA

Outline

1. The supervised classification learning problem

2. Performance measures

3. Performance estimation

Outline

1. The supervised classification learning problem
2. Performance measures
3. Performance estimation

The supervised classification learning problem

Three components

1. An instance space $\Omega_{\mathbf{X}}$
   - Random vectors $\mathbf{x} = (x_1, ..., x_n) \in \mathbb{R}^n$ are drawn independently according to some fixed but unknown probability distribution, $p(\mathbf{x})$
   - The *i*-th component of $\mathbf{x}$, x_i, has been drawn from the subspace Ω_{X_i} and contains the value of the *i*-th predictor variable, X_i, for one specific instance
   - $\Omega_{\mathbf{X}} = \Omega_{X_1} \times \cdots \times \Omega_{X_n}$
2. A label space, Ω_C, containing for each vector $\mathbf{x} = (x_1, ..., x_n)$ the value, c, of its label. The labels are obtained from a random variable, C
   - The conditional distribution of labels for a given vector of the instance space, $p(c|\mathbf{x})$, and
   - The joint distribution, $p(\mathbf{x}, c)$, of cases (instances + labels) are also unknown
3. A learning algorithm that implements a set of functions over the instance space, whose outputs are in the label space. The application of the learning algorithm to a data set of labelled instances, $\mathcal{D} = \{(\mathbf{x}^1, c^1), ..., (\mathbf{x}^N, c^N)\}$, will provide a supervised classification model

The supervised classification learning problem

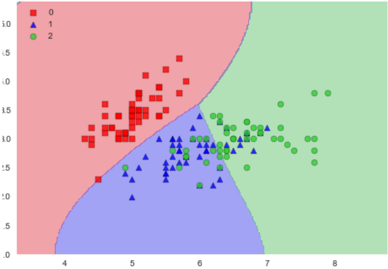
The data set $\mathcal{D}$ as a table

	X_1	$\dots$	X_n	C
$(\boldsymbol{x}^{(1)}, c^{(1)})$	$x_1^{(1)}$	$\dots$	$x_n^{(1)}$	$c^{(1)}$
$(\boldsymbol{x}^{(2)}, c^{(2)})$	$x_1^{(2)}$	$\dots$	$x_n^{(2)}$	$c^{(2)}$
$\dots$		$\dots$		$\dots$
$(\boldsymbol{x}^{(N)}, c^{(N)})$	$x_1^{(N)}$	$\dots$	$x_n^{(N)}$	$c^{(N)}$
$\boldsymbol{x}^{(N+1)}$	$x_1^{(N+1)}$	$\dots$	$x_n^{(N+1)}$	???

The supervised classification learning problem

Decision regions and decision boundaries

- The supervised classification model partitions the instance space into decision regions, one per class label. $\mathbf{x}$ is in the decision region associated with c if $\phi(\mathbf{x}) = c$.
- These regions are separated by decision boundaries, surfaces in the instance space corresponding to pairs of class labels reaching the same ϕ value
- The more flexible the decision boundaries, the better performance the classifier will have



Loss and risk functions

Loss and risk functions

- The loss function, $L(c, \phi(\mathbf{x}))$, is a quantitative measure of the loss when the label c of the vector $\mathbf{x}$ is different from the label assigned by the classifier, $\phi(\mathbf{x})$

$$\begin{array}{ccc} \Omega_C \times \Omega_C & \xrightarrow{L} & \mathbb{R}^+ \\ (c, \phi(\mathbf{x})) & \rightarrow & L(c, \phi(\mathbf{x})) \end{array}$$

  The zero-one loss function is $L(c, \phi(\mathbf{x})) = 1$ when $c \neq \phi(\mathbf{x})$ and 0 otherwise

- The expected risk of the classifier ϕ, $R(\phi) = \int L(c, \phi(\mathbf{x}))\mathrm{d}p(\mathbf{x}, c)$ computes the expectation of the loss (risk) function over the unknown distribution, $p(\mathbf{x}, \mathbf{c})$
  For the zero-one loss function, the expected risk associated with a classifier ϕ is calculated as $R_{0-1}(\phi) = p(C \neq \phi(\mathbf{X}))$ with cases drawn according to $p(\mathbf{x}, c)$
- The expected risk should be estimated using the information in $\mathcal{D} = \{(\mathbf{x}^1, c^1), ..., (\mathbf{x}^N, c^N)\}$, by the empirical risk function, $R_{\mathcal{D}}(\phi)$, according to

$$R_{\mathcal{D}}(\phi) = \frac{1}{N} \sum_{i=1}^{N} L(c^i, \phi(\mathbf{x}^i))$$

Loss and risk functions

	X_1	$\ldots$	X_n	C	$\phi(\mathbf{x})$
$(\mathbf{x}^1, c^1)$	7.2	$\ldots$	10.4	P	I
$(\mathbf{x}^2, c^2)$	7.1	$\ldots$	11.7	P	P
$(\mathbf{x}^3, c^3)$	6.4	$\ldots$	13.2	P	P
$(\mathbf{x}^4, c^4)$	6.7	$\ldots$	10.1	P	P
$(\mathbf{x}^5, c^5)$	8.9	$\ldots$	8.4	I	P
$(\mathbf{x}^6, c^6)$	9.2	$\ldots$	7.9	I	I
$(\mathbf{x}^7, c^7)$	10.7	$\ldots$	5.9	I	I
$(\mathbf{x}^8, c^8)$	8.1	$\ldots$	8.8	I	I
$(\mathbf{x}^9, c^9)$	9.9	$\ldots$	7.2	I	I
$(\mathbf{x}^{10}, c^{10})$	11.5	$\ldots$	6.9	I	I

- The output of the classifier is incorrect for Cases 1 and 5
- If each class is equally important, the loss associated with both types of mistakes is the same, and we have $L(c^i, \phi(\mathbf{x}^i)) = 1$ for $i \in \{1, 5\}$ and $L(c^j, \phi(\mathbf{x}^j)) = 0$ for $j \in \{2, 3, 4, 6, 7, 8, 9, 10\}$
- The empirical risk for this zero-one loss function would then be $R_{\mathcal{D}}(\phi) = 1/10 \times 2 = 0.20$. This empirical risk represents an estimation of the probability of the classifier being wrong

Outline

1. The supervised classification learning problem

2. Performance measures

3. Performance estimation

Binary classification

Two possible values for the class variable, C, represented, for example, as positive, $+$, and negative, $-$. $|\Omega_C| = 2$ = $|\Omega_{\phi(\mathbf{x})}| = 2$

Confusion matrix

$$\begin{array}{cc} & \phi(\mathbf{x}) \\ & \begin{array}{cc} + & - \end{array} \\ C \begin{array}{c} + \\ - \end{array} & \begin{pmatrix} \mathtt{TP} & \mathtt{FN} \\ \mathtt{FP} & \mathtt{TN} \end{pmatrix} \end{array}$$

- TP: true positives
- FP: false positives
- FN: false negatives
- TN: true negatives

Binary classification

Performance measures

Name	Formula
Accuracy	$\frac{\text{TP}+\text{TN}}{\text{TP}+\text{FN}+\text{FP}+\text{TN}}$
Sensitivity or Recall	$\frac{\text{TP}}{\text{TP}+\text{FN}}$
Specificity	$\frac{\text{TN}}{\text{FP}+\text{TN}}$
Positive predictive value or Precision	$\frac{\text{TP}}{\text{TP}+\text{FP}}$
Negative predictive value	$\frac{\text{TN}}{\text{TN}+\text{FN}}$
F_1-measure	$\frac{2\text{Prec}(\phi)\text{Rec}(\phi)}{\text{Prec}(\phi)+\text{Rec}(\phi)}$
Cohen's kappa statistic	$\frac{(\frac{\text{TP}}{N}+\frac{\text{TN}}{N})-[(\frac{\text{FN}+\text{TP}}{N})(\frac{\text{FP}+\text{TP}}{N})+(\frac{\text{FP}+\text{TN}}{N})(\frac{\text{FN}+\text{TN}}{N})]}{1-[(\frac{\text{FN}+\text{TP}}{N})(\frac{\text{FP}+\text{TP}}{N})+(\frac{\text{FP}+\text{TN}}{N})(\frac{\text{FN}+\text{TN}}{N})]}$

- The F_1 measure (van Rijsbergen, 1979) is the harmonic mean of the precision and recall measures
- Cohen's kappa statistic (Cohen, 1960) corrects the accuracy measure considering the result of a mere chance match between the classifier, $\phi(\mathbf{x})$, and the label generation process, C
- All measure values fall within the interval [0, 1], where values close to 1 are preferred

Binary classification

Performance measures. An example

$$\begin{array}{cc} & \phi(\mathbf{x}) \\ & \begin{array}{cc} + & \quad - \end{array} \\ C \begin{array}{c} + \\ - \end{array} & \begin{pmatrix} 120 & 8 \\ 60 & 139 \end{pmatrix} \end{array}$$

The values for the seven performance measures are:

1. $\mathrm{Acc}(\phi) = 0.79$
2. $\mathrm{Sensitivity}(\phi) = \mathrm{Rec}(\phi) = 0.94$
3. $\mathrm{Specificity}(\phi) = 0.74$
4. $\mathrm{PPV}(\phi) = \mathrm{Prec}(\phi) = 0.67$
5. $\mathrm{NPV}(\phi) = 0.95$
6. $F_1(\phi) = 0.78$
7. $\kappa(\phi) = 0.59$

Binary classification

Cost matrix

$$\begin{array}{ccc} & \multicolumn{2}{c}{\phi(\mathbf{x})} \\ & + & - \\ C \begin{array}{c} + \\ - \end{array} & \multicolumn{2}{c}{\begin{pmatrix} 0 & \mathrm{L}(+,-) \\ \mathrm{L}(-,+) & 0 \end{pmatrix}.} \end{array}$$

- Total cost error: $\mathrm{TCE}(\phi) = \mathtt{FN} \cdot L(+,-) + \mathtt{FP} \cdot L(-,+)$
- Total cost error in terms of the empirical risk as $\mathrm{TCE}(\phi) = N \cdot R_{\mathcal{D}}(\phi)$
- The total cost error verifies $0 \leq \mathrm{TCE}(\phi) \leq N \cdot \max\{L(+,-), L(-,+)\}$
- If the domain expert is not able to provide this information, costs are assumed to be symmetric: $L(+,-) = L(-,+)$

Binary classification

- The Brier score (Brier, 1950) measures the accuracy of probabilistic classifications over cases
- Measure of the calibration of a set of probabilistic predictions or as a quadratic cost function

Brier score

$$\mathrm{Brier}(\phi) = \frac{1}{N} \sum_{i=1}^{N} d^2 \left(p_\phi(\mathbf{c}|\mathbf{x}^i), \mathbf{c}^i \right)$$

- N denotes the number of cases in $\mathcal{D}$
- $p_\phi(\mathbf{c}|\mathbf{x}^i)$ is the vector $(p_\phi(+|\mathbf{x}^i), p_\phi(-|\mathbf{x}^i))$ containing the output of the probabilistic classifier
- $\mathbf{c}^i = (1, 0)$ or $\mathbf{c}^i = (0, 1)$ when the label of the i-th instance is $+$ or $-$, respectively
- The difference between the predicted probability assigned to the possible outcomes for each instance and its actual label is measured with the squared Euclidean distance, $d^2 \left(p_\phi(\mathbf{c}|\mathbf{x}^i), \mathbf{c}^i \right)$
- The Brier score for a binary classification problem verifies $0 \leq \mathrm{Brier}(\phi) \leq 2$

Binary classification

Brier score

	X_1	$\ldots$	X_n	C	$p_\phi(\mathbf{c}\vert\mathbf{x})$
$(\mathbf{x}^1, c^1)$	7.2	$\ldots$	10.4	P	(0.20, 0.80)
$(\mathbf{x}^2, c^2)$	7.1	$\ldots$	11.7	P	(0.65, 0.35)
$(\mathbf{x}^3, c^3)$	6.4	$\ldots$	13.2	P	(0.70, 0.30)
$(\mathbf{x}^4, c^4)$	6.7	$\ldots$	10.1	P	(0.87, 0.13)
$(\mathbf{x}^5, c^5)$	8.9	$\ldots$	8.4	I	(0.55, 0.45)
$(\mathbf{x}^6, c^6)$	9.2	$\ldots$	7.9	I	(0.25, 0.75)
$(\mathbf{x}^7, c^7)$	10.7	$\ldots$	5.9	I	(0.12, 0.88)
$(\mathbf{x}^8, c^8)$	8.1	$\ldots$	8.8	I	(0.07, 0.93)
$(\mathbf{x}^9, c^9)$	9.9	$\ldots$	7.2	I	(0.37, 0.63)
$(\mathbf{x}^{10}, c^{10})$	11.5	$\ldots$	6.9	I	(0.18, 0.82)

$$\text{Brier}(\phi) = \frac{1}{10}\left[(0.20-1)^2 + (0.80-0)^2 + \cdots + (0.18-0)^2 + (0.82-1)^2\right] = 0.2971$$

Multi-class classification

Confusion matrix

$$\begin{array}{rc}
 & \begin{array}{ccccc} & & \phi(\mathbf{x}) & & \\ c_1 & \cdots & c_j & \cdots & c_R \end{array} \\
\begin{array}{r} c_1 \\ \cdots \\ C\,c_j \\ \cdots \\ c_R \end{array} & \left(\begin{array}{ccccc} N_{11} & \cdots & N_{1j} & \cdots & N_{1R} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ N_{j1} & \cdots & N_{jj} & \cdots & N_{jR} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ N_{R1} & \cdots & N_{Rj} & \cdots & N_{RR} \end{array}\right)
\end{array}$$

Measures from the confusion matrix

Name	Notation	Formula
Accuracy	$\mathrm{Acc}(\phi)$	$\frac{\sum_{i=1}^{R} N_{ii}}{N}$
PPV or Prec for class c_j	$\mathrm{PPV}_j(\phi) = \mathrm{Prec}_j(\phi)$	$\frac{N_{jj}}{\sum_{i=1}^{R} N_{ij}}$
Total cost error	$\mathrm{TCE}(\phi)$	$\sum_{i=1}^{R} \sum_{j>i}^{R} N_{ij} \cdot L(c_i, c_j)$
Brier score	$\mathrm{Brier}(\phi)$	$\frac{1}{N} \sum_{i=1}^{N} d^2 \left(p_\phi(\mathbf{c}\|\mathbf{x}^i), \mathbf{c}^i\right)$

$$N = \sum_{i=1}^{R} \sum_{i=1}^{R} N_{ij}$$

Binary classification

- A receiver operating characteristic (ROC), or simply ROC curve (Lusted, 1960), is a graphical plot that illustrates the performance of a binary classifier system as its discrimination threshold is varied
- The discrimination threshold is a cutoff value for the posterior probability $p_\phi(c|\mathbf{x})$ for which the predicted label is $+$
- A given discrimination threshold returns a point of the plot
- The ROC curve is created by plotting (on the Y-axis) the true positive rate ($\texttt{TPR}= \frac{\texttt{TP}}{\texttt{TP}+\texttt{FN}}$), versus (on the X-axis) the false positive rate ($\texttt{FPR} = \frac{\texttt{FP}}{\texttt{FP}+\texttt{TN}}$), at various threshold settings. $\texttt{TP} + \texttt{FN} = N_+$ number of real positive. $\texttt{FP} + \texttt{TN} = N_-$ number of real negative
- The ROC curve is the polygonal curve plotted by connecting all pairs of consecutive points
- The ROC space is a unit square because $0 \leq \texttt{FPR} \leq 1$ and $0 \leq \texttt{TPR} \leq 1$

$$\begin{array}{ccc} & & \phi(\mathbf{x}) \\ & & \begin{array}{cc} + & \qquad - \end{array} \\ C & \begin{array}{c} + \\ - \end{array} & \begin{pmatrix} \texttt{TP} & \qquad \texttt{FN} \\ \texttt{FP} & \qquad \texttt{TN} \end{pmatrix} \end{array}$$

Binary classification

- Point $(0, 0)$, with both $\texttt{FPR}$ and $\texttt{TPR}$ equal to zero, denotes the model that classifies all instances as negative
- Point $(1, 1)$, with both $\texttt{FPR}$ and $\texttt{TPR}$ equal to one, represents the classifier labeling all instances as positive
- The diagonal of the ROC space, that is, the line connecting points $(0, 0)$ and $(1, 1)$, verifies $\texttt{FPR} = \texttt{TPR}$ at all points. The classifiers represented with points along this diagonal are regarded as random classifiers. The random classifier at point (a, a) means that, for a positive labelled case, $C = +$, the probability that the classifier, ϕ, classifies it as positive, $\phi = +$, equals a. In mathematical notation, $p(\phi = +|C = +) = a$. For a negative labelled case, $p(\phi = +|C = -) = a$.
- The classifiers represented by points above (or below) the diagonal perform better (or worse) than random
- $(\texttt{FPR}_1, \texttt{TPR}_1)$ represents a better classifier than $(\texttt{FPR}_2, \texttt{TPR}_2)$ if $(\texttt{FPR}_1, \texttt{TPR}_1)$ is on the left and higher up than $(\texttt{FPR}_2, \texttt{TPR}_2)$, because these positions signify that $\texttt{FPR}_1 < \texttt{FPR}_2$ and $\texttt{TPR}_1 > \texttt{TPR}_2$
- For point $(1, 0)$, $\texttt{FPR} = 1$ and $\texttt{TPR} = 0$. It denotes a classifier that gets all its predictions wrong
- Point $(0, 1)$ represents the best classifier, which gets all the positive cases right and none the negative ones wrong

Binary classification

Notation used by Algorithm 1 for building a ROC curve

ROC analysis in binary classification

- $\mathcal{D}$: the set of cases
- $\phi(\mathbf{x}^i)$: the continuous output of the classifier for instance $\mathbf{x}^i$
- *min* and *max*: the smallest and largest values returned by $\phi(\mathbf{x})$, respectively,
- *incr*: the smallest difference between any two output values
- N_+ and N_-: the number of real positive and negative cases, respectively
- The range of the threshold values t is $min, min + incr, min + 2 \cdot incr, ..., max$
- TP and FP are initialized as 0 (lines 2 and 3)
- For each case whose classification output exceeds threshold t (line 5), the TP counter is incremented by one if the case is positive (lines 6-7); for negative cases (lines 8-9) the FP counter is incremented by one
- TPR and FPR are respectively computed (lines 12 and 13) and the associated (FPR, TPR) is added to the ROC curve (line 14)

Binary classification

ROC analysis in binary classification

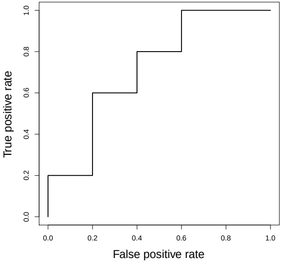
Algorithm 1: A simple algorithm for building a ROC curve (Fawcett, 2006)

Input : A classifier ϕ, and constants *min*, *max*, *incr*, N_+, N_-
Output: A ROC curve

```
for $t = min$ to $max$ by $incr$ do
    TP = 0
    FP = 0
    for $\mathbf{x}^i \in \mathcal{D}$ do
        if $\phi(\mathbf{x}^i) \geq t$ then
            if $\mathbf{x}^i$ is a positive case then
                TP = TP +1
            else
                FP = FP +1
        endif
    endfor
    TPR = TP/$N_+$
    FPR = FP/$N_-$
    Add point (FPR, TPR) to ROC curve
endfor
```

Binary classification

Instances	$\mathbf{x}^i$	1	2	3	4	5	6	7	8	9	10
Output	$p(+\|\mathbf{x}^i)$	0.97	0.91	0.84	0.80	0.68	0.67	0.66	0.61	0.49	0.46
True class	c^i	$+$	$-$	$+$	$+$	$-$	$+$	$-$	$+$	$-$	$-$



- $\texttt{FPR} = \frac{\texttt{FP}}{N_-}$ and $\texttt{TPR} = \frac{\texttt{TP}}{N_+}$
- First threshold at 0.46. At it, the five positive instances are well classified, whereas the five negative instances are misclassified. We get $\texttt{FPR} = \texttt{TPR} = 1$
- All the thresholds output by increments of 0.01 (value of `incr`) up to 0.49 yield the same results
- At 0.49, the instance $\mathbf{x}^{10}$ is correctly classified as $-$, and we get $\texttt{FPR} = 0.80$, and $\texttt{TPR} = 1$
- The next significant threshold is 0.61, where we get the third point, $(0.60, 1)$
- The other points are output in a similar fashion

Binary classification

The area under the ROC curve (AUC)

- The AUC is the most popular summary statistic for the ROC curve: $\text{AUC}(\phi) \in [0, 1]$
- A perfect classifier, ($\texttt{FPR} = 0$, $\texttt{TPR} = 1$): $\text{AUC}(\phi_{\text{perfect}}) = 1$
- A random classifier: $\text{AUC}(\phi_{\text{random}}) = 0.5$
- The AUC can be computed as: $\mathrm{AUC}(\phi) = 1 - \frac{\sum_{i=1}^{N_+}(i - \mathrm{rank}_i)}{N_+ \cdot N_-}$
  - rank_i is the rank (according to the posterior probability of $C = +$) of the i-th case in the subset of positive labels given by classifier ϕ
  - N_+ and N_- denote the number of real positive and negative cases in $\mathcal{D}$, respectively
- AUC can be interpreted as a measurement indicator of whether a classifier is able to rank a randomly chosen positive instance higher than a negative one

$$\mathrm{AUC}(\phi) = 1 - \frac{(1-1)+(3-2)+(4-3)+(6-4)+(8-5)}{5 \cdot 5} = 0.72$$

The AUC directly from the Figure:

$$\mathrm{AUC}(\phi) = 0.20 \cdot 0.20 + 0.20 \cdot 0.60 + 0.20 \cdot 0.80 + 0.40 \cdot 1 = 0.72$$

Multi-class classification

ROC analysis in multi-class problems

- For multi-class problems the AUC can be generalized as the volume under the ROC surface (Ferri et al., 2003)
- Alternatively, as an average AUC of all possible two-class ROC curves that can be generated from the multi-class problem (Hand and Till, 2001)

$$\mathrm{AUC}_{\text{multi-class}}(\phi) = \frac{2}{R(R-1)} \sum_{\substack{c_i, c_j \in \Omega_C \\ c_i \neq c_j}} \mathrm{AUC}_{c_i, c_j}(\phi)$$

  - $\mathrm{AUC}_{\text{multi-class}}(\phi)$ is the total AUC of the multi-class ROC for classifier ϕ
  - $\mathrm{AUC}_{c_i, c_j}(\phi)$ is the AUC of the two-class ROC curve of ϕ for classes c_i and c_j

Outline

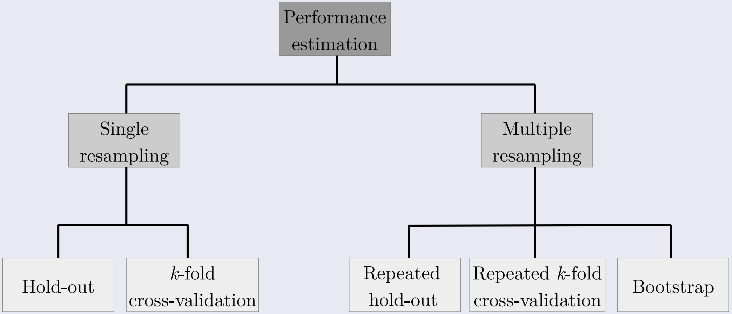
1. The supervised classification learning problem

2. Performance measures

3. Performance estimation

Honest estimation methods

Honest estimation methods according to their sampling characteristics


Performance estimation

Single resampling

Multiple resampling

Hold-out

k-fold cross-validation

Repeated hold-out

Repeated k-fold cross-validation

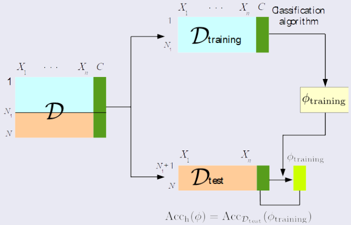
Bootstrap

Single resampling-based estimation methods

$\mathcal{D} = \{(\mathbf{x}^1, c^1), \ldots, (\mathbf{x}^N, c^N)\}$ is partitioned into two disjoint data subsets:

- The training data set: $\mathcal{D}_{\text{training}} = \{(\mathbf{x}^1, c^1), \ldots, (\mathbf{x}^{N_1}, c^{N_1})\}$ with N_1 cases and
- The test data set: $\mathcal{D}_{\text{test}} = \mathcal{D} \setminus \mathcal{D}_{\text{training}} = \{(\mathbf{x}^{N_1+1}, c^{N_1+1}), \ldots, (\mathbf{x}^N, c^N)\}$ with $N - N_1$ cases

Hold-out estimation



A general empirical risk function is estimated as follows: $R_{\mathcal{D}_{\text{test}}}(\phi_{\text{training}}) = \frac{1}{N-N_1} \sum_{(\mathbf{x}^i, c^i) \in \mathcal{D}_{\text{test}}} L(c^i, \phi_{\text{training}}(\mathbf{x}^i))$

The hold-out estimation of classification accuracy:

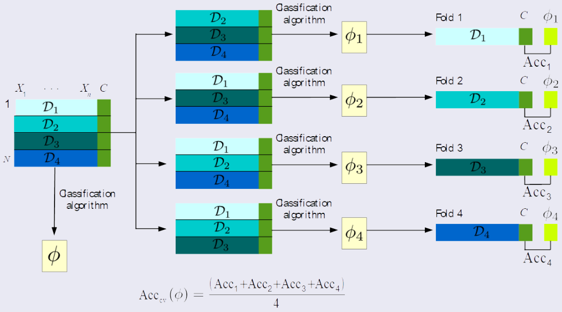
$$\text{Acc}_{\text{h}}(\phi) = \text{Acc}_{\mathcal{D}_{\text{test}}}(\phi_{\text{training}}) = \frac{1}{N - N_1} \sum_{(\mathbf{x}^i, c^i) \in \mathcal{D}_{\text{test}}} \mathbb{I}(c^i = \phi_{\text{training}}(\mathbf{x}^i))$$

where $\mathbb{I}(a)$ is the indicator function

Single resampling-based estimation methods

$\mathcal{D} = \{(\mathbf{x}^1, c^1), \ldots, (\mathbf{x}^N, c^N)\}$, is partitioned into k folds: $\mathcal{D}_1, \ldots, \mathcal{D}_k$, verifying $\mathcal{D} = \bigcup_{l=1}^{k} \mathcal{D}_l$ with $\mathcal{D}_w \bigcap \mathcal{D}_t = \varnothing$

k-fold cross-validation (Kurtz, 1948)



Accuracy of ϕ estimated as $\mathrm{Acc}_{\mathrm{cv}}(\phi) = \frac{1}{k}\sum_{l=1}^{k} \mathrm{Acc}_l$ with $\mathrm{Acc}_l = \frac{1}{|\mathcal{D}_l|} \sum_{(\mathbf{x}^i, c^i)\in\mathcal{D}_l} \mathbb{I}(c^i = \phi_l(\mathbf{x}^i))$

- The k-fold cross-validation estimator is very nearly unbiased, but its variance can be large (Stone, 1977)
- Leave-one-out cross-validation when $k = N$
- Stratified k-fold cross-validation for unbalanced data sets

Multiple resampling-based estimation methods

Repeated hold-out

- $\mathcal{D} = \{(\mathbf{x}^1, c^1), ..., (\mathbf{x}^N, c^N)\}$, is randomly partitioned B times as training data sets, $\mathcal{D}^l_{\text{training}}$, and test data sets, $\mathcal{D}^l_{\text{test}}$. For each partition $l \in \{1, ..., B\}$: $\mathcal{D} = \mathcal{D}^l_{\text{training}} \bigcup \mathcal{D}^l_{\text{test}}$ and $\mathcal{D}^l_{\text{training}} \bigcap \mathcal{D}^l_{\text{test}} = \varnothing$
- The final model, ϕ, is learned from $\mathcal{D}$, and its accuracy is estimated as

$$\text{Acc}_{\text{rh}}(\phi) = \frac{1}{B} \sum_{l=1}^{B} \text{Acc}^l$$

where Acc^l denotes the estimation of the accuracy of model ϕ^l_{training} , learned from $\mathcal{D}^l_{\text{training}}$, and tested over $\mathcal{D}^l_{\text{test}}$

- Repeated hold-out extends the main idea of the hold-out scheme to a multiple resampling scenario. The partition in the hold-out scheme is repeated several times, each with a random assignment of training and test cases
- Advantage: stability of the estimates (variance is low), resulting from a large number of sampling repetitions
- Drawback: there is no control of how many times each case is used in the training data sets or in the test data sets

Multiple resampling-based estimation methods

Repeated k-fold cross-validation

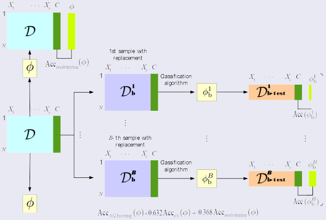
Repeated k-fold cross-validation reduces the variability of the estimator by multiple rounds of k-fold cross-validation performed using different partitions

- The 5×2 cross-validation (Dietterich, 1998) performs five repetitions of two-fold cross-validation
- The 10×10 cross-validation (Bouckaert, 2003) based on 10 repetitions of 10-fold cross-validation

Multiple resampling-based estimation methods

- Bootstrap sampling method consists of sampling with replacement N cases from $\mathcal{D} = \{(\mathbf{x}^1, c^1), \ldots, (\mathbf{x}^N, c^N)\}$
- Repeated B times: $\mathcal{D}_{\mathrm{b}}^l$, with $l \in \{1, \ldots, B\}$, all of size N
- The probability of a case not being chosen after N selections is $(1 - \frac{1}{N})^N \approx \frac{1}{e} \approx 0.368$
- The expected number of distinct cases in each of the B data sets $\mathcal{D}_{\mathrm{b}}^l$ used for training the classifier is $0.632N$
- $\mathcal{D}_{\mathrm{b-test}}^l = \mathcal{D} \setminus \mathcal{D}_{\mathrm{b}}^l$ and $\mathrm{Acc}(\phi_{\mathrm{b}}^l) = \frac{1}{|\mathcal{D}_{\mathrm{b-test}}^l|} \sum_{(\mathbf{x}^i, c^i) \in \mathcal{D}_{\mathrm{b-test}}^l} \mathbb{I}(c^i = \phi_{\mathrm{b}}^l(\mathbf{x}^i))$
- The **e0** bootstrap estimate, $\mathrm{Acc}_{\mathrm{e0}}(\phi) = \frac{1}{B} \sum_{l=1}^{B} \mathrm{Acc}(\phi_{\mathrm{b}}^l)$ can be pessimistic

0.632 bootstrap method (Efron, 1979)


X_1 ··· X_n C ϕ
1
$\mathcal{D}$
N
ϕ
$\mathrm{Acc}_{\mathrm{resubstitution}}(\phi)$
X_1 ··· X_n C
1
$\mathcal{D}$
N
ϕ
1st sample with replacement
X_1 ··· X_n C
$\mathcal{D}_{\mathrm{b}}^1$
Classification algorithm
ϕ_{b}^1
X_1 ··· X_n C ϕ_{b}^1
$\mathcal{D}_{\mathrm{b-test}}^1$
$\mathrm{Acc}(\phi_{\mathrm{b}}^1)$
B- th sample with replacement
X_1 ··· X_n C
$\mathcal{D}_{\mathrm{b}}^B$
Classification algorithm
ϕ_{b}^B
X_1 ··· X_n C ϕ_{b}^B
$\mathcal{D}_{\mathrm{b-test}}^B$
$\mathrm{Acc}(\phi_{\mathrm{b}}^B)$
$\mathrm{Acc}_{.632\,\mathrm{bootstrap}}(\phi) = 0.632\mathrm{Acc}_{\mathrm{e0}}(\phi) + 0.368\mathrm{Acc}_{\mathrm{resubstitution}}(\phi)$

$$\mathrm{Acc}_{.632\mathrm{bootstrap}}(\phi) = 0.632\mathrm{Acc}_{\mathrm{e0}}(\phi) + 0.368\mathrm{Acc}_{\mathrm{resubstitution}}(\phi)$$

where the resubstitution estimation is: $\mathrm{Acc}_{\mathrm{resubstitution}}(\phi) = \frac{1}{N} \sum_{(\mathbf{x}^i, c^i) \in \mathcal{D}} \mathbb{I}(c^i = \phi(\mathbf{x}^i))$

- Bootstrap estimation is asymptotically (large values of B) unbiased and its variance is small. These are interesting properties when working with small data sets.

References

- G. W. Brier (1950). Verification of forecasts expressed in terms of probability. *Monthly Weather Review*, 78, 1-3
- J. Cohen (1960). A coefficient of agreement for nominal scales. *Educational and Psychological Measurements*, 20, 37-46
- T.M. Cover, J.A. Thomas (1991). *Elements of Information Theory*. Wiley
- A. Edwards (1948). Note on the “correction for continuity” in testing the significance of the difference between correlated proportions. *Psychometrika*, 13, 185-187
- B. Efron (1979). Bootstrap methods: Another look at the jackknife. *Annals of Statistics*, 7, 1-26
- T. Fawcett (2006). An introduction to ROC analysis. *Pattern Recognition Letters*, 27, 861-874
- C. Ferri, J. Hernández-Orallo, M.A. Salido (2003). Volume under the ROC surface for multi-class problems. *Proceedings of the 14th European Conference on Machine Learning*, 108-120
- D.J. Hand, R.J. Till (2001). A simple generalization of the area under the ROC curve for multiple class classification problems. *Machine Learning*, 45, 171-186
- S. Kullback, R.A. Leibler (1951). On information and sufficiency. *Annals of Mathematical Statistics*, 22(1), 79-86
- A.K. Kurtz (1948). A research test of Rorschach test. *Personnel Psychology*, 1, 41-53
- L.B. Lusted (1960). Logical analysis in roentgen diagnosis. *Radiology*, 74, 178-193
- C.E. Shannon (1948). A mathematical theory of communication. *The Bell System Technical Journal*, 27, 3, 379-423
- Z. Šidák (1967). Rectangular confidence regions for the means of multivariate normal distributions. *Journal of the American Statistical Association*, 62(31), 626-633
- M. Stone (1977). Asymptotics for and against cross-validation. *Biometrika*, 64(1), 29-35
- C.J. van Rijsbergen (1979). *Information Retrieval*. Butterworth