# Corrections for the Classification Exam - Third Year - Speciality Signal and Image Processing

Friday, January 22, 2016

*Lecture notes and slides authorized*

## Exercice 1

We consider a classification problem with two classes $\omega_1$ and $\omega_2$ whose densities are

$$f(x|\omega_1) = 2x\mathbb{I}_{]0,1[}(x) \quad \text{and} \quad f(x|\omega_2) = 2(1-x)\mathbb{I}_{]0,1[}(x) \tag{1}$$

where $\mathbb{I}_{]0,1[}(x)$ is the indicator function on the interval $]0,1[$ (such that $\mathbb{I}(x) = 1$ if $x \in ]0,1[$ and $\mathbb{I}(x) = 0$ if $x \notin ]0,1[$).

1. (3 pts) Derive the Bayesian classification rule associated with this problem when we use the $0-1$ cost function and when the two classes have the prior probabilities $P(\omega_1) = 1/4$ and $P(\omega_2) = 3/4$. How does this rule modify when the two classes are equiprobable? Explain this result. Determine the probability of error in the equiprobable case.
   *Response* : The Bayesian classifier accepts the class $\omega_1$ if

   $$f(x|\omega_1)P(\omega_1) \geq f(x|\omega_2)P(\omega_2)$$

   or equivalently if, for $x \in ]0,1[$

   $$2x \times \frac{1}{4} \geq 2(1-x) \times \frac{3}{4}.$$

   As a consequence, the class $\omega_1$ is accepted if

   $$x \geq \frac{3}{4}.$$

   When the two classes are equiprobable, we have $P(\omega_1) = P(\omega_2) = \frac{1}{2}$. Thus, the Bayesian classifier accepts the class $\omega_1$ if

   $$2x \times \frac{1}{2} \geq 2(1-x) \times \frac{1}{2} \Leftrightarrow x \geq \frac{1}{2}.$$

   In the equiprobable case, the error probability of the Bayesian classifier can be computed as follows

   $$P_e = \int_0^{\frac{1}{2}} f(x|\omega_1)P(\omega_1)dx + \int_{\frac{1}{2}}^1 f(x|\omega_2)P(\omega_2)dx.$$

   i.e.,

   $$P_e = \int_0^{\frac{1}{2}} xdx + \int_{\frac{1}{2}}^1 (1-x)dx = \frac{1}{4}.$$

2. (3 pts) Assume that we have a learning set composed of two elements of class $\omega_1$ denoted as $x_1 = 3/4$ and $x_2 = 7/8$ and two elements of class $\omega_2$ denoted as $x_3 = 1/8$ and $x_4 = 3/8$. What is the classification rule associated with the nearest neighbor rule. The asymptotic error probability of the nearest neighbor rule is known to be

   $$P_1 = \int_{-\infty}^{+\infty} \left[1 - \sum_{i=1}^{2} P^2(\omega_i|x)\right] f(x)dx.$$

Compute this error probability in the equiprobable case. Check that this result is in good agreement with the Cover and Hart inequality.
*Response* : the 1 nearest neighbor rule assigns $x$ to class $\omega_1$ if the nearest neighbor of $x$ belongs to class $\omega_1$. After displaying the different points in the interval $]0,1[$, it can be seen that the class $\omega_1$ is accepted if

$$x \geq x^* = \frac{x_4 + x_1}{2} = \frac{9}{16}.$$

In order to compute the error probability of the 1-nearest neighbor rule, we need to determine $P(\omega_1|x)$, $P(\omega_2|x)$ and $f(x)$. In the equiprobable case, straightforward computations lead to

$$f(x) = f(x|\omega_1)P(\omega_1) + f(x|\omega_2)P(\omega_2) = 1$$

$$P(\omega_1|x) = \frac{f(x|\omega_1)P(\omega_1)}{f(x)} = x$$

$$P(\omega_2|x) = \frac{f(x|\omega_2)P(\omega_2)}{f(x)} = 1 - x.$$

The asymptotic error probability of the nearest neighbor rule can then be computed as follows

$$P_1 = \int_{-\infty}^{+\infty} \left[1 - \sum_{i=1}^{2} P^2(\omega_i|x)\right] f(x)dx = \int_0^1 [1 - x^2 - (1-x)^2]dx = \frac{1}{3}.$$

Since $K = 2$, we have

$$P_e\left(2 - \frac{K}{K-1}P_e\right) = \frac{3}{8}.$$

The double inequality of Cover and Hart

$$P_e = \frac{1}{4} \leq P_1 = \frac{1}{3} \leq P_e\left(2 - \frac{K}{K-1}P_e\right) = \frac{3}{8}$$

is clearly satisfied.

3. (3 pts) We assume now that the probability density function $f(x|\omega_1)$ is unknown and estimate it using the following estimator

$$f_n(x) = \frac{1}{nh_n}\sum_{i=1}^{n} \phi\left(\frac{x - x_i}{h_n}\right)$$

where $x_1, ..., x_n$ are training samples from the class $\omega_1$ (i.e., distributed according to $f(x|\omega_1)$) and

$$\phi(u) = \begin{cases} e^{-u} & \text{if } u > 0 \\ 0 \text{ sinon} \end{cases}$$

Provide some motivations for the estimator $f_n(x)$ introduced above. Determine $E[f_n(x)]$ as a function of $x$ and $h_n$. Determine also

$$\lim_{h_n \to 0} E[f_n(x)].$$

What can we conclude about the estimator $f_n(x)$?
*Response* : the estimator $f_n(x)$ results from a direct application of the theory of Parzen windows. Since the training samples $x_i$ are distributed according to $f(x|\omega_1) = 2x\mathbb{I}_{]0,1[}(x)$, we have

$$E[f_n(x)] = \frac{1}{nh_n}\sum_{i=1}^{n} E\left[\phi\left(\frac{x - x_i}{h_n}\right)\right] = \frac{1}{nh_n}\sum_{i=1}^{n}\int_0^1 2x_i\phi\left(\frac{x - x_i}{h_n}\right)dx_i.$$

Using the definition of $\phi$, we have

$$\int_0^1 2x_i \phi\left(\frac{x - x_i}{h_n}\right) dx_i = \int_0^x 2x_i \exp\left(\frac{x_i - x}{h_n}\right) dx_i = 2\exp\left(\frac{-x}{h_n}\right) I$$

with

$$I = \int_0^x x_i \exp\left(\frac{x_i}{h_n}\right) dx_i.$$

Integrating by parts leads to

$$I = h_n x \exp\left(\frac{x}{h_n}\right)$$

hence

$$E[f_n(x)] = 2x - 2h_n + 2h_n \exp\left(-\frac{x}{h_n}\right).$$

Finally, we obtain for $x \in ]0,1[$

$$\lim_{h_n \to 0} E[f_n(x)] = 2x = f(x|\omega_1)$$

which shows that $f_n(x)$ is an asymptotically unbiased estimator of $f(x|\omega_1)$ (asymptotically meaning $h_n \to 0$).

4. (2 pts) Create by hand a dendrogram for the following 6 points in one dimension: $x_1 = -5.5$, $x_2 = -4.0$, $x_3 = -3.0$, $x_4 = 5.0$, $x_5 = 6.1$ and $x_6 = 7.3$, when the distance between two clusters $X_i$ and $X_j$ is defined as

$$d(X_i, X_j) = \min_{x \in X_i, y \in X_j} d(x, y)$$

*Response*: a similar example was processing during one of the lectures.

**Questions related to the working paper**

*Remark*: please make sure to justify all your responses very carefully.

1. (1 pt) Explain how we can classify a feature vector with the decision tree displayed in Fig. 8.1.
*Response*: for a given signal or image to classify, we first build a feature vector. In the example of Fig. 8.1, this feature vector contains four components related to the taste, the color, the shape and the size of the fruit. This feature vector is then propagated into the tree according to the different decisions made at the leaves of the tree. For instance, the feature vector associated with a banana will be (yellow,thin,medium,sweet). If we present this vector to the tree of Fig. 8, it will go in the middle branch since it is yellow, then it will go in the right branch as it is thin and thus it will be classified as a banana.

2. (0.5 pt) Does CART belong to the class of supervised or un-supervised classification methods?
*Response*: to build the classification tree, we need some labelled feature vectors. Thus CART belongs to the family of supervised classification methods.

3. (1 pt) Can we always build a tree with binary decisions? Justify your response by means of an example.
*Response*: if there are more than two decisions at a node of the tree, we can always replace these decisions by a sub-tree with binary decisions. Consider for instance the node "size" at level 1 of the tree displayed in Fig. 8.1. There are three decisions associated with this node: "big", "medium" and "small". These three decisions might be replaced by a two-level sub-tree asking first whereas the fruit is "big" (decision #1) or "medium or small" (decision #2). Then the second decision can be split into two part "medium" or "small". This two-level sub-tree is clearly defined using binary decisions only.

4. (1 pt) Build a branch of a decision tree which leads to the decision region $R_1$ displayed in the left figure of Fig. 8.3.
*Response*: This branch can be obtained by the following steps

   - Step 1: test whether $x_1$ is larger or smaller than a given threshold (e.g., $s_1 = 1$).
   - Step 2: test whether $x_2$ is larger or smaller than a given threshold (e.g., $s_2 = 1$).
   - Step 3: test whether $x_1$ is larger or smaller than a given threshold (e.g., $s_3 = 2$).
   - Step 4: test whether $x_2$ is larger or smaller than a given threshold (e.g., $s_4 = 1.5$).

5. (1 pt) What is the value of the entropy $i(N)$ defined in (1) for equally likely (equiprobable) classes? for two classes with respective probabilities $P(\omega_1) = 0$ and $P(\omega_2) = 1$?
*Response*: if the two fractions of samples belonging to classes $\omega_1$ and $\omega_2$ are the same, we have $P(\omega_1) = 1/2$ and $P(\omega_2) = 1/2$ leading to the maximum entropy $i(N) = 1$. For $P(\omega_1) = 0$ and $P(\omega_2) = 1$, we obtain $i(N) = 0$.

6. (0.5 pt) Why might we prefer to use the Gini impurity index rather than the misclassification impurity?
*Response*: the tree obtained with the Gini impurity is generally richer (more branches) and thus can anticipate later splits that won't be considered by the misclassification impurity.

7. (1 pt) Explain the term "cross-validation" (appearing page 11 of the paper).
*Response*: Cross-validation means that you divide your set of labelled samples into two parts, a "training set" (containing for instance $90\%$ of the data) and a "test set" (containing $10\%$ of the data). The training set is used to build the decision tree, i.e., to choose the decisions associated with each branch of the tree. The "test set" is used to decide whether a node is a leaf node (terminal node) or not, i.e., if we continue to split a given node into two parts, or not.

8. (2 pts) As explained in the paper, we can use hypothesis testing to decide whether we have to stop the growing of a tree at a given node. Explain how this strategy is working. What is the distribution of $\chi^2$ defined in (9)? Provide a mathematical expression of the test threshold as a function of the confidence level $\alpha$ (probability if false alarm of the test) and the inverse cumulative distribution function of $\chi^2$.
*Response*: Assume that we have samples belonging to classes $\omega_1$ and $\omega_2$ at a given node of the tree ($n_1$ samples in $\omega_1$ and $n_2$ samples in $\omega_2$). Consider a given split $s$, which sends $Pn$ pattern to the left and $(1-P)n$ patterns to the right (with $n = n_1 + n_2$). If this split differs significantly from a random split, which would send (in average) $Pn_1$ patterns to the left and $(1-P)n_2$ patterns to the right, we continue to split this node. In order to measure this kind of similarity measure, we use the value of $\chi^2$ defined in (9). The distribution of $\chi^2$ is a chi-square distribution with one degree of freedom under hypothesis $H_0$ (random split). If $\chi^2$ is less than a given threshold, we accept the hypothesis $H_0$ and do not split the node. Conversely, if $\chi^2$ is larger than a given threshold, we accept the hypothesis $H_1$ and we split the node. If $F$ denotes the cumulative distribution function of the chi-square distribution with one degree of freedom, the test threshold can be determined from the probability of false alarm of the test

$$\alpha = \text{PFA} = P[\chi^2 < S_\alpha | H_0 \text{ true}] = F(S_\alpha)$$

hence

$$S_\alpha = F^{-1}(\alpha).$$

9. (1 pt) Explain how the first threshold 0.6 has been obtained in the top right tree of Example 1.
*Response*: In order to determine this threshold, we have to test all possible values of thresholds $t$ for splitting the data from the two sides of a vertical line (defined by $x_1 = t$) and from the two sides of an horizontal line (defined by $x_2 = t$). The split which provides the largest impurity (here

entropy) is determined. If this threshold does not validate the stopping criterion, the data are split according to this threshold. In example 1, the largest impurity was obtained for a vertical line of equation $x_1 = t$ with $t \in ]0.57, 0.70[$. The value $t = 0.6$ is an example of solution.

10. (1pt) Explain “Preprocessing by principal components can be effective” in Section 3.7.1. In particular, explain how these principal components can be computed.
    *Response*: The aim of principal component analysis is to project the data into a lower dimensional sub-space which represents the data with good accuracy. Thus it makes sense to use these projections (instead of the features) to feed the tree. The principal components are defined by the eigenvectors associated with the largest eigenvalues of the data covariance matrix.