University of Southern Denmark
IMADA
Arthur Zimek

DM868/DM870/DS804: Data Mining and Machine Learning

Spring term 2023

Exercise 6: Classification Evaluation, k-Nearest Neighbor Classification, Introduction to R

Exercise 6-1 Measure for Evaluation of Classifiers (1 point)

Given a data set with known class labels ($f(o)$) of the objects. In order to evaluate the quality of a classifier h, each object is additionally classified using h. The results are given in the table (all three columns) below.

ID	$f(o)$	$h(o)$
O_1	A	A
O_2	B	A
O_3	A	C
O_4	C	C
O_5	C	B

ID	$f(o)$	$h(o)$
O_6	B	B
O_7	A	A
O_8	A	A
O_9	A	A
O_{10}	B	C

ID	$f(o)$	$h(o)$
O_{11}	B	A
O_{12}	C	A
O_{13}	C	C
O_{14}	C	C
O_{15}	B	B

- Rewrite the definitions for precision and recall given in the lecture by using TP, TN, FP, and FN.
- Using the table (all three columns) above, compute precision and recall for each class.
- To get a complete measure for the quality of the classification with respect to a single class, the F_1-measure (the harmonic mean of precision and recall) is commonly used. It is defined as follows:

  $$F_1(h,i) = \frac{2 \cdot \text{Recall}(h,i) \cdot \text{Precision}(h,i)}{\text{Recall}(h,i) + \text{Precision}(h,i)}$$

  Compute the F_1-measure for all classes.
- So far, the F_1-measure is only defined for classes and not yet useful to get an overview of the overall performance of the classifiers. To achieve such an overall assessment, one commonly takes the average over all classes using one of the following two approaches:
  - Micro Average F_1-Measure: The values of TP, FP and FN are added up over all classes. Then precision, recall and F_1-measure are computed using these sums.
  - Macro Average F_1-Measure: Precision and recall are computed for each class individually, afterwards the average precision and average recall are used to compute the F_1-measure.

  Compute the Micro- and Macro-Average F_1-measures for the example above. What do you observe?

Exercise 6-2 Procedures for Evaluation of Classifiers (1 point)

Given a data set D with objects from classes A and B ($D = A \cup B$) where the class assignments are *random* (not related to the attribute values). Furthermore, let the two classes have the same size $|A| = |B|$.

- What is the best some classifier could do on such data? What *true error rate* is to be expected for such an *optimal* (for this data set) classifier?
- What error rates are to be expected when training and evaluating an optimal classifier on the given dataset using a leave-one-out test?
- Remember that in Bootstrap we produce the training and test data by sampling with replacement. An object is with a probability of

  $$\left(1 - \frac{1}{n}\right)^n \approx 0.368$$

  not part of the n training objects, i.e. only about 63.2% of the objects are used for training. (Compare this to 10-fold cross validation, where 90% of the data are used for training.)

  This implies that the error estimation is pessimistic, as the training set has size n, but actually only contains $0.632 \cdot n$ *different* examples.

  To make up for this, when evaluating bootstrap it is a common practice to also include the apparent classification error (error on the training data) during evaluation:

  $$\text{error rate} = 0.632 \cdot \text{Error on test set} + 0.368 \cdot \text{Error on training set}$$

  This will be repeated multiple times (with different samples) and averaged.

  What error rates are to be expected when evaluating an optimal classifier on the given dataset using the 0.632 Bootstrap method? Interpret these results.

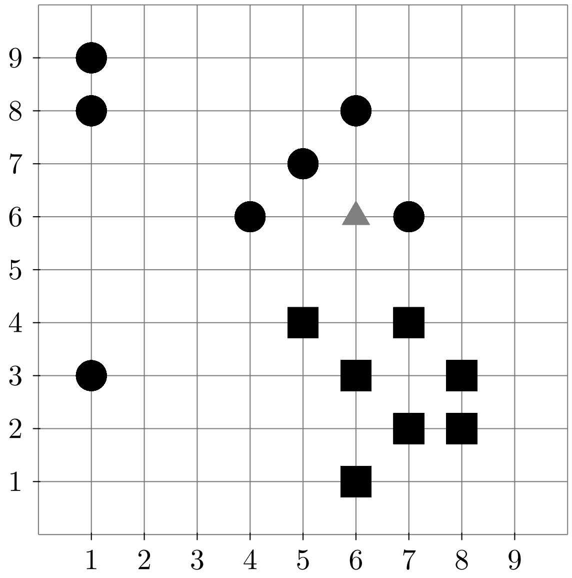
Exercise 6-3 Nearest neighbor classification (1 point)

The 2D feature vectors in the figure below belong to two different classes (circles and rectangles). Classify the object at $(6, 6)$ — in the image represented using a triangle — using k nearest neighbor classification. Use Manhattan distance (L_1 norm) as distance function, and use the non-weighted class counts in the k-nearest-neighbor set, i.e. the object is assigned to the majority class within the k nearest neighbors. Perform kNN classification for the following values of k and compare the results with your own "intuitive" result.

(a) $k = 4$

(b) $k = 7$

(c) $k = 10$



Exercise 6-4 Nearest Neighbor classification (1 point)

Find a scenario where we have a set of at least four points in 2 dimensions, such that the Nearest Neighbor classification ($k = 1$) only gives incorrect classification results when using any of these points as query point and the rest as training examples. Use Euclidean distance as distance function.

Exercise 6-5 Get started with R

(a) Download and install R-Studio: `https://www.rstudio.com/products/rstudio/download/`.

You can try the following exercise suggestions yourself (and explore more of R as much as you want).

(b) Start a new R script containing your solutions. Save the script for later reference.

You can use #### Section With Name #### To define a section within your script with name "Section With Name" in your R code, that you can easily navigate to.

(c) Some important commands for learning R, are `help()`, `class()`, and `mode()`.

You can use these commands on variables, functions, objects, and datasets to obtain information on them.

Exercise 6-6 Vectors in R

(a) Create a vector of length 5 containing both positive and negative numbers, using the concatenate (`c()`) command.

(b) Find the `mean()`, `max()`, `min()` of the vector. Then compute the mean of the absolute values.

(c) Taking a subset of the vector can be done using the following notation:

`vector[1:2]` will take the first two elements of the vector (*R* starts indexing with 1).

Insert 42 on the third position of the vector you created earlier.

(d) Create a new vector and build the sum of the two vectors.

(e) Create a random vector using the `rnorm()` function with no additional arguments.

- Calculate the mean — what do you observe?
- Take the last 5 elements of the vector using the indexing described above.

Exercise 6-7 Matrices in R

(a) Create a 2×2 matrix A by row binding vectors using the `rbind()` command.

(b) Nullify matrix A by adding another matrix that you define.

(c) Double all the values in the original matrix A by multiplication with another matrix that you define.

Exercise 6-8 Exploration of Datasets in R

(a) Use the help command to get information on the built-in dataset AirPassengers.

  (i) Plot the dataset using the `plot()` command. What do you see? Describe the resulting plot.

  (ii) Create a histogram using the built-in function `hist()`
  What do you observe?

  (iii) Check the `class()` and `mode()` of the dataset. Are these as expected? If you are not sure what the `mode` and `class` functions do use the `help()` function.

(b) *R* comes with many historical data sets. One of them is the Titanic dataset. Use the `help()` function to read about the data set. Then make a mosaic plot using the `mosaicplot()` command. What do you observe?

Exercise 6-9 Clustering and Classification on the Iris Data (1 point)

(a) Load the Iris dataset in R, remove the class attribute. Cluster it using k-means with a reasonable choice of k.

(b) Use the clustering result to label the data.

(c) Create some artificial flower data, that could potentially be Iris flowers.

Think about how you will do this, and what you expect the resulting flowers to be.

(d) Try the $k$nn-classifier with different values for k, and use your generated labeled Iris dataset to classify the artificial query points.

(e) Try using the original labeled Iris dataset. Does this yield the same result?

(f) Explain your findings.