# NaiveBayes Classifier and k-fold Cross Evaluation

You are provided with a dataset containing 800 articles from newsgroups. In particular, each article contains a discussion related to one of the following 8 topics:

1. automobiles (cars): all files containing the prefix "auto";
2. baseball: all files containing the prefix "baseball";
3. electronics (general topics related to electronics): all files containing the prefix "electronics";
4. hockey: all files containing the prefix "hockey";
5. IBM hardware: all files containing the prefix "ibm-hw";
6. motorbikes: all files containing the prefix "moto-hw";
7. use of guns: all files containing the prefix "pol-guns";
8. Mac hardware: all files containing the prefix "mac-hw".

Some basic data cleaning and filtering have been performed on such a collection of documents. We are going to train and evaluate a text classifier using the naive Bayes classifier. Given a text document (whose topic or file name is not known), the classifier should classify the document as belonging to one of the aforementioned topics. We shall see that such a simple classifier performs quite well for this task. This exercise consists of the following steps:

1. turn each document into a vector in the Euclidean space. Each dimension of the vector corresponds to a word, with its value specifying the number of occurrences of the corresponding word in the corresponding document. This can be done by using *CountVectorizer* in scikit-learn (see tutorial). It helps to remove so called "stopwords", which are words that occur often in the English language such as articles (e.g. 'the','a',), adjectives (e.g. 'my','yours) etc. Stopwords usually do not carry much information about the topic of a document and should be removed.
2. train a naive Bayes classifier on the collection of documents. In Python, we are going to consider the multinomial naive Bayes classifier and the Gaussian naive Bayes classifier. See tutorial.

3. perform a 10-fold cross validation to determine which classifier performs best. More informations on how to do that are provided in the tutorial.

You should perform the following tasks and submit your Jupyter notebook with all the answers and the code.

(a) perform a 10-fold cross validation and report the average accuracy (i.e. average number of documents classified correctly) for both the multinomial and Gaussian naive Bayes classifier.

(b) consider the best classifier according to your evaluation. How does such a classifier compare with a random classifier which classifies the documents randomly? (i.e. it assigns to each document one of the eight topics, chosen randomly). To answer this question, compute the accuracy of the random classifier (i.e. the probability that a document is classified correctly) and include it in your Jupyter notebook.

(c) report the accuracy of the best classifier when the "stopwords" are not removed. Does the accuracy improve or worsen? Try to explain why the accuracy improves or worsens.

(d) Consider the following two classification tasks. In the first task, you should consider only the documents related to the following topics: "use of guns", "hockey","Mac hardware", while in the second task you should consider "Mac hardware","IBM hardware" and "electronics". Perform a 10-fold cross validation for each of the two tasks and report the average accuracy. Try to explain why one task might be easier than the other one.

**What to submit:** You should submit a Jupyter notebook containing:

- the answers to the question in a markdown cell of your Jupyter notebook;
- the code in Python;