# Data Analysis PI
# Theoretical assignment #1

Kupriyanov Kirill

# Task 1.

*Problem:* Consider classification with 1-nearest neighbor using euclidean distance.

**(a)** Prove that the decision boundary for two training objects of different classes is linear.

$\square$ $A = (x_1^0, x_2^0, \ldots, x_n^0)$; $B = (x_1^1, x_2^1, \ldots, x_n^1)$; $A \in \mathbb{R}^n$; $B \in \mathbb{R}^n$. The orthogonal line for AB consists of those points $X = (x_1^x, x_2^x, \ldots, x_n^x)$, for which

$$||A - X|| = ||B - X||$$
$$||(x_1^0 - x_1^x, x_2^0 - x_2^x, \ldots, x_n^0 - x_n^x)|| = ||(x_1^1 - x_1^x, x_2^1 - x_2^x, \ldots, x_n^1 - x_n^x)||$$

$$\sqrt{(x_1^0 - x_1^x)^2 + (x_2^0 - x_2^x)^2 + \cdots + (x_n^0 - x_n^x)^2} =$$
$$\sqrt{(x_1^1 - x_1^x)^2 + (x_2^1 - x_2^x)^2 + \cdots + (x_n^1 - x_n^x)^2}$$

Square both parts

$$(x_1^0 - x_1^x)^2 + (x_2^0 - x_2^x)^2 + \cdots + (x_n^0 - x_n^x)^2 =$$
$$(x_1^1 - x_1^x)^2 + (x_2^1 - x_2^x)^2 + \cdots + (x_n^1 - x_n^x)^2$$
$$(x_1^0)^2 - 2(x_1^0 x_1^x) + \underline{(x_1^x)^2} + \cdots + (x_n^0)^2 - 2(x_n^0 x_n^x) + \underline{(x_n^x)^2} =$$
$$(x_1^1)^2 - 2(x_1^1 x_1^x) + \underline{(x_1^x)^2} + \cdots + (x_n^1)^2 - 2(x_n^1 x_n^x) + \underline{(x_n^x)^2}$$
$$(x_1^0)^2 + \cdots + (x_n^0)^2 - (x_1^1)^2 - \cdots - (x_n^1)^2 = 2(x_1^0 x_1^x) + 2(x_2^0 x_2^x) - 2(x_1^1 x_1^x) - 2(x_2^1 x_2^x) + \cdots$$

$$||A||^2 - ||B||^2 = 2(A - B)^T X$$

As A and B are distinct, this is the equation of a linear hyperplane. $\blacksquare$

**(b)** Explain why decision boundaries separating classes in case of 1-nearest neighbor classifier are piecewise linear for N training objects and C classes.

For every point in the set it's closest neighbor should be found. If multiple points are close to one at the same time, a random one can be picked (or choose another method of selecting).

Knowing that for 2 points, the decision boundary is a linear hyperplane, adding next point to the selected ones would add one or more linear hyperplanes. Adding so forth, the connected piecewise linear line will be built.

# Task 2.

*Problem:* Consider a training set of $N$ objects with $D$ features. Assume that each object has only $s < D$ nonzero features. Find computational complexity of classification of a new object with 1-nearest neighbor classifier with euclidean distance. (Remark: different objects may have different nonzero features but we explicitly know which ones).

*Solution:* If a new object comes, the distance should be count. Assuming that there are $s$ features, the computation of distance between 2 objects is $\boldsymbol{O(s)}$, since (euclidean, manhattan) distance iterates over all features. Computation of distances to the rest objects will take $\boldsymbol{O(ns)}$, since there are $n$ objects. Because $k = 1$, finding 1 closest neighbor will take $\boldsymbol{O(kns)} \overset{k=1}{=} \boldsymbol{O(ns)}$.

# Task 3.

*Problem:* Consider objects with categorical features. The simplest similarity measure for two objects with such features is overlap measure. It counts the number of features that match in both objects. The range of per-features similarity for the overlap measure is $[0, 1]$, with a value of 0 when there is no match, and a value of 1 when the feature values match. Let's say that feature $f$ takes $P$ possible values and some of them are more frequently occurring in the data than the others. For example, if $f$ is a city of residence then *Moscow* is much more frequent value of $f$ than *Bobrov*. Modify the overlap similarity measure in such way that it uses the information about frequency differences.

*Solution:* The most obvious way is to use Goodall2 measure instead of Overlap measure. This measure assigns higher similarity if the matching values are infrequent, and at the same time there are other values that are even less frequent, i.e., the similarity is higher if there are many values with approximately equal frequencies, and lower if the frequency distribution is skewed. The formula looks like that:

$$S_k(X_k, Y_k) = \begin{cases} 1 - \sum\limits_{q \in Q} p_k^2(q), & \text{if } X_k = Y_k \\ 0, & \text{else} \end{cases}$$

where

$$p_k^2(q) = \frac{f_k(q)(f_k(q) - 1)}{N(N - 1)}$$

$f_k(x)$ - number of times feature $A_k$ takes the value $x$ in a dataset.