theory2

by Kondrashov Anton

January 26, 2017

1 Task

1.1

Let's introduce grid coordinate system over the usual grid coordinate system used for display of data (inputs x, outputs y) so that horizontal axis contains both points of 2 given classes. Name that axis a and the other one b. That means that in new coordinate system given points have some coordinates $p_1(a_1, 0)$ and $p_2(a_2, 0)$. Equidistant point is $p(a, b)$

In KNN with decision boundary is a locus equidistant from two points which means:

$$\sqrt{(a-a_1)^2+b^2} = \sqrt{(a-a_2)^2+b^2}$$
$$(a-a_1)^2 = (a-a_2)^2$$
$$a = \frac{a_1+a_2}{2}$$

That formula shows that the boundary is linear and perpendicular to . The same can be proven for multiple features(dimensions) by analogy.

1.2

With $k = 1$ number of classes C doesn't affect the boundaries because the nearest neighbour is always one and the same.

For multiple points(objects) the fact that each two of them have linear boundary is proved above. Suppose that for m points(objects) the piecewise linearity is proved. Then add to m another point(object). It will become NN for the set of points(objects) on the closest part to it from contour formed by intersecting linear decision boundaries. The contour is piecewise linear by definition and it didn't change the boundaries which it didn't intersect and intersected ones can only be replaced by the contour which means that the new boundary is also piecewise linear.

2 Task

Calculate all distances for queried object to all training objects takes $O(nd)$. Doesn't matter wether feature is nonzero, because classic algorithm calculates distances for all of them.

3 Task

The frequent values should affect the similarity value less, because they don't differentiate more objects. That's why I propose to use the same calculation method with only difference: the per-feature similarity should be multiplied by $\frac{1}{frequency}$ where frequency is ratio of value encounters to objects number.