1 Lineární klasifikátory, Perceptronový algoritmus

Bayesovské rozhodování používalo ztrátovou funkci, apriorní a aposteriorní pravděpodobnosti, nebayesovské si vystačilo s aposteriorními pravděpodobnostmi. Lineární klasifikátory nepotřebují ani to. Jejich klasifikace je založena pouze na dodané trénovací množině $T = \{(x_1, k_1), ..., (x_l, k_l)\}$. Výhody a nevýhody tohoto přístupu v Figure 1.

− : There is no direct relationship between known properties of estimated $\hat{p}(x, k)$ and the properties (typically the risk) of the obtained classifier $q'(x)$

− : If the true $p(x, k)$ is not of the assumed form, $q'(x)$ may be arbitrarily bad, even if the size of training set L approaches infinity!

+ : Implementation is often straightforward, especially if parameters Θ_k for each class are assumed independent.

+ : Performance on training data can be predicted by crossvalidation.

Figure 1: Výhody a nevýhody lin. klasifikátorů

1.1 Perceptron learning

Algorimus očekává na vstupu trénovací množinu pozorování $T = \{x'_1, ..., x'_l\}$ kde $x'_j = k_j \cdot [x_j\ 1]$ (tzn. množina souřadnic pozorování kde ke každému pozorování přidáme 1 a vynásobíme klasifikací $k_j \in \{-1, 1\}$). Výstup algoritmu je vektor vah $w' = [w\ b]$ takových, že $\langle w', x'_j \rangle \geq 0, \forall j \in \{1, ..., l\}$. Vlastní algoritmus je popsán ve Figure 2.

1.1.1 Neseparovatelná data

Input: $T = \{x_1, \dots x_L\}$
Output: a weight vector w

Perceptron algorithm, (Rosenblat 1962):

1. $w_1 = 0$.
2. A wrongly classified observation x_j is sought, i.e., $\langle w_t, x_j \rangle < 0$, $j \in \{1..L\}$.
3. If there is no misclassified observation then the algorithm terminates otherwise
$$w_{t+1} = w_t + x_j \ .$$
4. Goto 2.

Figure 2: Perceptron popis

Perceptron algorithm, batch version, handling non-separability:

Input: $T = \{x_1, \dots x_L\}$
Output: a weight vector w^*

1. $w_1 = 0$, $E = |T| = L$, $w^* = 0$.
2. Find all mis-classified observations $X^- = \{x \in X : \langle w_t, x \rangle < 0\}$.
3. if $|X^-| < E$ then $E = |X^-|; w^* = w_t$
4. if $tc(w^*, t, t_{lu})$ then terminatate else $w_{t+1} = w_t + \eta_t \sum_{x \in X^-} x$
5. Goto 2.

- The algorithm converges with probability 1 to the optimal solution.
- Convergence rate not known (to me).
- Termination condition $tc(.)$ is a complex function of the quality of the best solution, time since last update $t - t_{lu}$ and requirements on the solution.

Figure 3: Perceptron popis s neseparovatelnými daty