IST Austria: Statistical Machine Learning 2015/16

Christoph Lampert <chl@ist.ac.at>
TA: Alex Zimin <azimin@ist.ac.at>

Exercise Sheet 3/5

1 Implicit and explicit bias term in maximum margin classification

In the lecture, we used the *augmentation* trick, $x \mapsto \tilde{x} = (x, 1)$, turn an affine decision function $\tilde{f}(x) = \langle w, x \rangle + b$ into a linear function $f(x) = \langle \tilde{w}, \tilde{x} \rangle$, with $\tilde{w} = (b, w)$. For SVM training, both formulations are similar, but not exactly equivalent. In this exercise, we study the differences.
a) Write down the learning problem for the maximum-margin classifier for the augmented data $\tilde{x}^i$ and weight vector $\tilde{w}$, and convert it to a form using x^i, w, and b. What's the difference to an SVM with explicit bias term?
b) Which of both versions make more sense (geometrically?)
c) In practice, one sometimes augments $\tilde{x} = (x, c)$ with a large $c > 0$ (e.g. 1000) instead of $c = 1$. Why?

We now know three variants of the maximum margin classifier: 1) linear with no bias term, 2) affine with explicit bias b, 3) affine with implicit bias (by augmentation). Answer the following questions for each of them:
d) What happens if you take the x-part of the training set and scale every vector by a fixed constant (isotropic scaling). Can you express the optimal weight vector (and bias, if present) for the scaled data in terms of the value(s) for the original data?
e) What happens if you take the x-part of the training set and shift every vector by a fixed vector (translation). Can you express the optimal weight vector (and bias, if present) for the translated data in terms of the value(s) for the original data?

2 Kernels

Which of the following functions are positive definite kernel functions? For positive answers, prove your results (you're allowed to use the properties from slide "Constructing Kernels" of the lecture). For negative answers, give an example for which the conditions are violated.
a) $k(x, x') = (1 + x^\top x')^d$ for any $d \in \mathbb{N}$

b) $k(x, x') = g(x)\hat{k}(x, x')g(x')$, where $\hat{k}$ is a kernel and g is an arbitrary function

c) $k(x, x') = \exp(\ \gamma \|x - x'\|^2\ )$ for some $\gamma > 0$

d) $k(x, x') = \exp(-\gamma \|x - x'\|^2)$ for $\gamma > 0$

e) $k(x, x') = \sum_{j=1}^{d} \min(x_j, x'_j)$ for vectors with non-negative entries

f) $k(x, x') = \sum_{j=1}^{d} \max(x_j, x'_j)$ for vectors with non-negative entries

3 Error-Correcting Output Codes

Many multi-class classification methods can (almost) be expressed as error-correcting output codes.

a) what would the codebook for the *one-versus-rest* approach? what is the decoding rule?

b) construct a codebook that needs only $\lceil \log_2 M \rceil$ classifiers for M classes.

c) what's the smallest code you can find that *detects* if exactly one of the classifiers makes a wrong prediction (use $k = 4$ if the general case is too hard)

d) one can generalize ECOCs by allowing 0 codebook entries, i.e.values that are ignored when comparing the predicted code to the codebook. Which codebook of this kind corresponds to the *one-versus-one* approach?

e) construct a codebook, possibly including 0s, for the hierarchical multi-class classifier

f) *Bonus:* for $k = 5$, what's the smallest code you can find that still *makes the right prediction* even if one classifiers makes a wrong prediction?

4 Statistical Significance

NIPS is one of the top international machine learning conferences. Go to its proceedings website
`https://papers.nips.cc/book/advances-in-neural-information-processing-systems-28-2015`
and find a paper that compares the prediction quality of a new method against one or multiple earlier methods on multiple datasets (or a similar setup). For each of the reported previous methods perform a statstical test if the reported results for the new method are statistically signficantly different. What do you find?

5 Practical Experiments V

- Pick one more training methods from the first sheet and implement it.
- In addition, implement a *linear support vector machine (SVM)* with training by Stochastic Coordinate Dual Ascent.
- What error rates do both methods achieve on the datasets from the previous sheet?
- Plot the number of steps against the values of the objective function and the error rate for the stochastic gradient method for different stepsizes, η, and for SCDA. What do you observe?