Abteilung Maschinelles Lernen
Institut für Softwaretechnik und theoretische Informatik
Fakultät IV, Technische Universität Berlin
Prof. Dr. Klaus-Robert Müller
Email: klaus-robert.mueller@tu-berlin.de

Exercise Sheet 11

Exercise 1: Activation Maximization (20 P)

Consider the linear model $f(\boldsymbol{x}) = \boldsymbol{w}^\top \boldsymbol{x} + b$ mapping some input $\boldsymbol{x}$ to an output $f(\boldsymbol{x})$. We would like to interpret the function f by building a prototype $\boldsymbol{x}^\star$ in the input domain which produces a large value f. Activation maximization produces such interpretation by optimizing

$$\max_{\boldsymbol{x}} \big[f(\boldsymbol{x}) - \Omega(\boldsymbol{x}) \big].$$

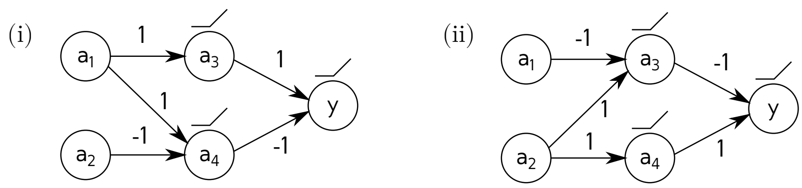
(a) *Find* the prototype $\boldsymbol{x}^\star$ obtained by activation maximization subject to the penalty $\Omega(\boldsymbol{x}) = \lambda \|\boldsymbol{x}\|^2$.

(b) *Find* the prototype $\boldsymbol{x}^\star$ obtained by activation maximization subject to the penalty $\Omega(\boldsymbol{x}) = -\log p(\boldsymbol{x})$ with $\boldsymbol{x} \sim \mathcal{N}(\boldsymbol{\mu}, \Sigma)$ where $\boldsymbol{\mu}$ and Σ are the mean and covariance.

(c) *Find* the prototype $\boldsymbol{x}^\star$ obtained when the data is generated as (i) $\boldsymbol{z} \sim \mathcal{N}(0, I)$ and (ii) $\boldsymbol{x} = A\boldsymbol{z} + \boldsymbol{c}$, with A and $\boldsymbol{c}$ the parameters of the generator. Here, we optimize f w.r.t. the code $\boldsymbol{z}$ subject to the penalty $\Omega(\boldsymbol{z}) = \lambda \|\boldsymbol{z}\|^2$.

Exercise 2: Layer-Wise Relevance Propagation (30 P)

We would like to test the dependence of layer-wise relevance propagation (LRP) on the structure of the neural network. For this, we consider the function $y = \min(a_1, a_2)$, where $a_1, a_2 \in \mathbb{R}^+$ are the input activations. This function can be implemented as a ReLU network in multiple ways. Two examples are given below.



(a) *Show* that these two networks implement the 'min' function on the relevant domain.

(b) We consider the LRP-γ propagation rule:

$$R_j = \sum_k \frac{a_j \cdot (w_{jk} + \gamma w_{jk}^+)}{\sum_j a_j \cdot (w_{jk} + \gamma w_{jk}^+)} R_k$$

where $()^+$ denotes the positive part. For each network, *give* for the case $a_1 = a_2$ an analytic solution for the scores R_1 obtained by application this propagation rule at each layer. More specifically, express R_1 as a function of the input activations.

Exercise 3: Neuralization (20 P)

Consider the one-class SVM that predicts for every new data point $\boldsymbol{x}$ the 'inlierness' score:

$$f(\boldsymbol{x}) = \sum_{i=1}^{M} \alpha_i k(\boldsymbol{x}, \boldsymbol{u}_i)$$

where $(\boldsymbol{u}_i)_{i=1}^M$ is the collection of support vectors, and $\alpha_i > 0$ are their weightings. We use the Gaussian kernel $k(\boldsymbol{x}, \boldsymbol{x}') = \exp(-\gamma \|\boldsymbol{x} - \boldsymbol{x}'\|^2)$.

Because we are typically interested in the degree of anomaly of a particular data point, we can also define the score $o(\boldsymbol{x}) = -\frac{1}{\gamma} \log f(\boldsymbol{x})$ which grows with the degree of anomaly of the data point.

(a) Show that the outlier score $o(\boldsymbol{x})$ can be rewritten as a two-layer neural network:

$$h_i = \|\boldsymbol{x} - \boldsymbol{u}_i\|^2 - \gamma^{-1} \log \alpha_i \qquad \text{(layer 1)}$$

$$o(\boldsymbol{x}) = -\tfrac{1}{\gamma} \log \textstyle\sum_{i=1}^{M} \exp(-\gamma h_i) \qquad \text{(layer 2)}$$

(b) Show that the layer 2 converges to a min-pooling (i.e. $o(\boldsymbol{x}) = \min_{i=1}^{N}\{h_i\}$) in the limit of $\gamma \to \infty$.

Exercise 4: Programming (30 P)

Download the programming files on ISIS and follow the instructions.

Exercise Sheet 11

1 Activation Maximization

a) $\frac{\partial}{\partial x} \max_x \; w^T x + b - \lambda \|x\|^2 \overset{!}{=} 0$

$\Leftrightarrow \quad w^T - 2\lambda x \overset{!}{=} 0 \qquad \Leftrightarrow \quad x^* = \frac{w^T}{2\lambda}$

b) $\max_x \; w^T x + b + \log(p(x))$

$= \max_x \; w^T x + b + \log\left(\frac{\exp(-\frac{1}{2}(x-\mu)^T \Sigma^{-1}(x-\mu))}{(2\pi)^{\frac{d}{2}} |\Sigma|}\right)$

$= \max_x \; w^T x + b + (\log(\exp(-\frac{1}{2}(x-\mu)^T \Sigma^{-1}(x-\mu)) - \log((2\pi)^{\frac{d}{2}} |\Sigma|)$

$\Rightarrow \quad \frac{\partial}{\partial x} w^T x - \frac{1}{2}(x-\mu)^T \Sigma^{-1}(x-\mu) \overset{!}{=} 0$

$\Leftrightarrow \quad w^T - \Sigma^{-1}(x-\mu) = 0$

$\Leftrightarrow \quad w^T - \Sigma^{-1} x + \Sigma^{-1}\mu = 0$

$\Leftrightarrow \quad x = \Sigma(\Sigma^{-1}\mu + w^T)$

$\Leftrightarrow \quad x^* = \mu + \Sigma w^T$

c) $\max_z \; w^T(Az + c) + b - \lambda \|z\|^2$

$\Rightarrow \quad \frac{\partial}{\partial z} w^T(Az+c) + b - \lambda\|z\|^2 \overset{!}{=} 0$

$\Leftrightarrow \quad w^T A - 2\lambda z = 0$

$\Leftrightarrow \quad z = \frac{w^T A}{2\lambda} \qquad | \; x = Az + c \;\Leftrightarrow\; z = \frac{x-c}{A}$

$\Leftrightarrow \quad \frac{x-c}{A} = \frac{w^T A}{2\lambda}$

$\Leftrightarrow \quad x = \frac{w^T A A}{2\lambda} + c$

2 Layer-Wise Relevance Propagation

a) $y = \underbrace{a_3}_{a_1} - \underbrace{a_4}_{\max(0,\, a_1 - a_2)}$

ReLU has no effect
bc. a_1 is positive

if $a_1 \geq a_2$: $y = a_1 - (a_1 - a_2) = a_2$

if $a_1 \leq a_2$: $y = a_1 - 0 = a_1$

b) i) $R_y = a$ ii) $R_y = a$

$R_3 = a$ $R_3 = 0$

$R_1 = a$ $R_1 = 0$

3 Neuralization

a) $$o(x) = -\frac{1}{\beta} \log\left(\sum_{i=1}^{M} \alpha_i e^{-\beta \| x - \mu_i \|^2}\right)$$

$$= -\frac{1}{\beta} \log\left(\sum_{i=1}^{M} e^{-\beta \| x - \mu_i \|^2 - \frac{1}{\beta} \log(\alpha_i)}\right)$$

b) $\min_{i=1}^{N} \{h_i\} = m$

$$o(x) - m = -\frac{1}{\beta} \log\left(\sum_{i=1}^{M} e^{-\beta h_i}\right) + \frac{1}{\beta} \log e^{-\beta m}$$

$$= -\frac{1}{\beta} \log\left(\sum_{i=1}^{M} e^{-\beta h_i} e^{\beta m}\right)$$

$$= -\frac{1}{\beta} \log\left(\sum_{i=1}^{M} e^{-\beta (h_i - m)}\right)$$

$$= -\frac{1}{\beta} \log e^{-\beta \cdot 0} + \sum_{i} e^{-\beta (h_i - m)}$$

$$= -\frac{1}{\beta} \log\left(1 + \sum_{i} e^{-\beta (h_i - m)}\right)$$

$o(x) - m = 0$ for $\beta \to \infty$

$\Leftrightarrow$ $o(x) = m$ $\rightarrow$ min-pooling