Cognitive Algorithms
Exercise
Kernel Methods

Hannah Marienwald

hannah.marienwald@campus.tu-berlin.de

Linear Regression

Given a data set $\{x_n, y_n\}_{n=1}^{N}$, the goal of linear regression is to estimate the function relationship between the independent variable $x_n \in \mathbb{R}^D$ and the dependent variable $y_n \in \mathbb{R}$, i.e.

$$y_n = w^T x_n + \varepsilon_n$$

as

$$\widetilde{y_n} = w^T x_n$$

such that

$$\min_{w} \sum_{n=1}^{N} (y_n - \widetilde{y_n})^2$$

Task 1.1

Consider a data set with two data points, $x_1 = -1, x_2 = 1$ with respective labels $y_1 = 1, y_2 = 1$.

1. We want to fit a simple linear model $f(x) = \omega \cdot x$ to the data using Ordinary Least Squares (OLS). Recall the OLS solution is obtained as

$$\omega = \underset{\omega}{\operatorname{argmin}} \sum_{n=1}^{N} (y_n - f(x_n))^2 = (XX^\top)^{-1} X y^\top$$

where $N = 2$ is the number of data points, $X = [x_1, x_2]$ and $y = [y_1, y_2]$. Compute ω.

Task 1.1

Leads to $w = 0$

$$\Rightarrow \widetilde{y_1} = f(x_1) = 0$$
$$\Rightarrow \widetilde{y_2} = f(x_2) = 0$$

$$\Rightarrow f(x) = 0$$

Task 1.2

Consider a data set with two data points, $x_1 = -1, x_2 = 1$ with respective labels $y_1 = 1, y_2 = 1$.

2. We obtain a better fit using the model $g(x) = w_1 + w_2 \cdot x = \mathbf{w}^T \cdot \phi(x)$ where we have defined a mapping $\phi : \mathbb{R} \ni x \mapsto \begin{bmatrix} 1 \\ x \end{bmatrix} \in \mathbb{R}^2$ and a weight vector $\mathbf{w} = \begin{bmatrix} w_1 \\ w_2 \end{bmatrix} \in \mathbb{R}^2$. Recall the OLS solution is obtained as

$$\mathbf{w} = \underset{\mathbf{w}}{\operatorname{argmin}} \sum_{n=1}^{N} (y_n - g(x_n))^2 = (XX^\top)^{-1} X y^\top$$

where N and y are defined as above and $X = [\phi(x_1), \phi(x_2)] = \begin{bmatrix} 1 & 1 \\ x_1 & x_2 \end{bmatrix}$. Compute $\mathbf{w}$ and the corresponding function $g(x)$.

Task 1.2

Leads to $w^T = [1, 0]$

$$\Rightarrow \widetilde{y_1} = g(x_1) = 1$$
$$\Rightarrow \widetilde{y_2} = g(x_2) = 1$$

$$\Rightarrow g(x) = 1$$

Kernel Trick

When the functional relationship is non-linear, linear regression fails to give proper estimates.

Kernel Trick

In that case we can try to

1. Map the data into a (high dimensional) kernel space, i.e. $x_n \mapsto \varphi(x_n)$
2. Look for linear relations in that kernel space. Here: Apply linear regression on $\{\varphi(x_n), y_n\}_{n=1}^{N}$.

Kernel Trick

Any algorithm for vectorial data that can be expressed only in terms of scalar products between vectors can be performed implicitly in the kernel space associated with any kernel, by replacing each scalar product by a kernel evaluation.

$$k(x_i, x_j) = \varphi(x_i)^T \varphi(x_j)$$

where k is called the kernel function.

Task 1.3

Consider a data set with two data points, $x_1 = -1, x_2 = 1$ with respective labels $y_1 = 1, y_2 = 1$.

3. Now, we want to obtain the same solution as above, but by solving the dual representation. Instead of learning $\mathbf{w}$ directly, we learn a linear combination $\alpha = \begin{bmatrix} \alpha_1 \\ \alpha_2 \end{bmatrix} \in \mathbb{R}^N$ of the data points, i.e. $\mathbf{w} = \alpha_1 \phi(x_1) + \alpha_2 \phi(x_2)$. In the lecture, we derived the following formula:

$$\alpha = K^{-1} y^\top \tag{1}$$

where $k : \mathbb{R} \times \mathbb{R} \to \mathbb{R}$ is the kernel function and $K \in \mathbb{R}^{N \times N}$ is the kernel matrix, with $K_{ij} = k(x_i, x_j)$. Here, the Kernel function is given as $k(x_i, x_j) = \phi(x_i)^\top \phi(x_j) = \begin{bmatrix} 1 & x_i \end{bmatrix} \begin{bmatrix} 1 \\ x_j \end{bmatrix} = 1 + x_i x_j$.

Using only α and the Kernel function, the predictions for a new data point x are given as

$$g_2(x) = \alpha_1 k(x, x_1) + \alpha_2 k(x, x_2)$$

Compute α and show that we obtain the same solution as before, i.e. show that $g_2(x) = g(x)$.

Task 1.3

Leads to $\alpha^T = [{}^1/_2, {}^1/_2]$

$$\Rightarrow \widetilde{y_1} = g_2(x_1) = 1$$
$$\Rightarrow \widetilde{y_2} = g_2(x_2) = 1$$

$$\Rightarrow g_2(x) = g(x) = 1$$

Task 1.4

Consider a data set with two data points, $x_1 = -1, x_2 = 1$ with respective labels $y_1 = 1, y_2 = 1$.

4. Now we want to use the Gaussian Kernel defined as

$$k(x_i, x_j) = e^{-\frac{(x_i - x_j)^2}{2\sigma^2}}$$

with kernel width $\sigma = 0.5$. Compute the Kernel matrix K and the Kernel Ridge Regression coefficients α using Equation (1) . To simplify calculations, approximate $e^{-8} = 0.000335 \approx 0$.

Sketch the obtained function $h(x) = \alpha_1 k(x, x_1) + \alpha_2 k(x, x_2)$ as a 2D plot (use e.g. $e^{-2} = 0.14$).

Task 1.4

Leads to $\alpha^T = [1, 1]$

$$\Rightarrow \widetilde{y_1} = h(x_1) = 1$$
$$\Rightarrow \widetilde{y_2} = h(x_2) = 1$$

$$\Rightarrow h(x) = e^{-2\,(x+1)^2} + e^{-2\,(x-1)^2}$$

Task 1.5

Consider a data set with two data points, $x_1 = -1, x_2 = 1$ with respective labels $y_1 = 1, y_2 = 1$.

5. Suppose we use a linear Kernel function, $k(x_i, x_j) = x_i x_j$. What problem do we get when computing the Kernel Ridge Regression coefficients α using Equation (1)?

Task 1.5

Kernel matrix is not invertible.

$\Rightarrow$ use Kernel Ridge Regression (as seen in the homework)

Cross-Validation

Cross-validation gives you an estimation of …

1) … values for parameters, that do not over- or underfit

   Example: weight w for linear regression

2) … the generalization error

   Example: How good linear regression (with fixed weight w) will perform on new/unseen test data.

If you want to estimate the generalization error when parameters are not known, perform *Nested Cross-Validation*!

Task 2.1

1. You are a reviewer for the International Mega-Conference on Machine Learning of Outrageous Stuff, and you read a paper that selected a small number of features out of a large number of features for a given classification problem. The paper argues as follows:

   (a) We used all our available data to select a subset of "good" features that had fairly strong correlation with the class labels.

   (b) Our final model contained only those features. We evaluate the prediction error of the final model by 10-fold crossvaldiation on all the available data.

   (c) We obtained a low cross-validation error. Thus, we have achieved high classification accurarcy with only few meaningful features. (This is novel and amazing.)

   Would you accept or reject the paper? Why?

Task 2.1

- How are the features selected? Manually?
- How can we select the features for other data sets?

⇒Cross-Validation automatically estimates relevant features!

In regression scenario: weights w of irrelevant features will be (close to) 0.

Task 2.2

2. Suppose you are testing a new algorithm on a data set consisting of 100 positive and 100 negative examples. You plan to use leave-one-out cross-validation (i.e. 200-fold cross-validation) and compare your algorithm to a baseline function, a simple majority classifier. Given a set of training data, the majority classifier always outputs the class that is in the majority in the training set, regardless of the input. You expect the majority classifier to achieve about 50% classification accuracy, but to your surprise, it scores zero every time. Why?

Task 2.2

We have 100 positive labeled samples and 100 negative labeled samples.

When we select e.g. one positive labeled sample as test sample, the majority of the training samples will have a negative label (99 positive labeled samples vs. 100 negative labeled samples). Thus, the classifier will decide for a negative label.

Same holds, when a negative labeled samples is chosen as test sample.

Majority classifier fails to classify correctly in all cases.