Exercise Sheet 3

Due: 10.11.25 14h00

Christof Gehrig, Gerrit Gerhartz

General Regulations.

- Please hand in your solutions in groups of two (preferably from the same tutorial group).
- Your solutions to theoretical exercises can be either handwritten notes (scanned), or typeset using LaTeX. For scanned handwritten notes, please ensure they are legible and not blurry.
- For the practical exercises, always provide the (commented) code as well as the output, and don't forget to explain/interpret the latter.
- Please hand in a **single PDF** that includes both the exported notebook and your solutions to the theoretical exercises. Submit the PDF to the Übungsgruppenverwaltung once per group, making sure to include the names of both group members in the submission.
- You can find all the data in the GitHub Repository.

1 Regularization and Intercept

Consider a regression problem with two explanatory variables x_1, x_2. As introduced in the lecture, the intercept β_0 can be incorporate by considering $y = \boldsymbol{\beta}^T \mathbf{x}$ with $\boldsymbol{\beta} = (\beta_0, \beta_1, \beta_2)^T$ and $\mathbf{x} = (1, x_1, x_2)^T$.

(a) In this setting, write down the loss function for ridge regression, penalizing the L^2-norm of $\boldsymbol{\beta}$, in components. What is the influence of the regularization strength on the intercept β_0? (1 pt)

(b) Oftentimes, a regularization of the intercept term is unwanted. How would you modify the loss function to account for this? (1 pt)

(c) Which shapes in $\mathbb{R}^3$ do the regularization contours (i.e. sets of parameters with equal regularization penalty) of versions (a) and (b) have? (1 pt)

2 Visualize Regularization Contours

For two-dimensional parameter vectors β, we can visualize the error/loss surface of linear regression using contour plots. In this exercise, you will create a set of such plots in order to familiarize yourself further with the influence of regularization. For example, you can visualize the contours via `plt.contour` or `plt.contourf`.[1]

(a) Plot the Ridge regression regularization term as well as the Lasso[2] regularization term for $\beta_1, \beta_2 \in [-1, 3]$. (2 pts)

(b) For the data set `linreg.npz`, plot the sum of squares (SSQ) of a linear regression as a function of $\boldsymbol{\beta}$ over the same range as in **(a)**, i.e., over the grid $[-1, 3] \times [-1, 3]$. (2 pts)

(c) Plot the ridge and Lasso loss functions, i.e., $\text{SSQ}(\boldsymbol{\beta}) + \lambda ||\boldsymbol{\beta}||_2^2$ and $\text{SSQ}(\boldsymbol{\beta}) + \lambda ||\boldsymbol{\beta}||_1$, for $\lambda \in \{0, 10, 50, 100, 200, 300\}$ in the same $\boldsymbol{\beta}$ grid as before and *discuss your observations!* (2 pts)

[1] See https://matplotlib.org/stable/gallery/images_contours_and_fields/contour_demo.html for an example.
[2] The abbreviation comes from *least absolute shrinkage and selection operator*.

3 CT Reconstruction

Linear regression plays an important role in reconstructing computer tomography (CT) scans. In this task, you will do this on simulated data in the 2D case. You are given a sinogram $Y \in \mathbb{R}^{ar}$, i.e., a matrix where each row corresponds to a (1D) projection of the image consisting of r detector readouts along one of a distinct, evenly spaced angles. Additionally, you are given the design matrix $\mathbf{X} \in \mathbb{R}^{p \times ar}$. Excluding noise, one has $Y = I\mathbf{X}$, with the image $I \in \mathbb{R}^p$ which should be reconstructed.

(a) What is the interpretation of a column of $\mathbf{X}$? Visualize a choice of four columns as images. (2 pts)

(b) What is the interpretation of a row of $\mathbf{X}$? Visualize a choice of four rows as images. (2 pts)

(c) Solve the reconstruction problem with linear regression without any regularization and with ridge regression. What do you observe? (3 pts)

4 Bayes Classifiers

In the lecture, we derived the Bayes classifier for the 0-1 loss. In this exercise, you will find the optimal classifier for another loss function.

Consider classification with k classes in the ground-truth $y \in \{1, \dots, k\}$, but adding 0 as an additional "reject class" to the prediction $\hat{y} \in 0, 1, \dots, k$. For a fixed $\alpha \in (0, 1)$, consider a loss function L with

$$\begin{aligned} L(y, \hat{y}) &= 1 - \delta_{y\hat{y}} = \begin{cases} 0, \text{ for } y = \hat{y} \\ 1, \text{ else} \end{cases} && \text{for} \quad y, \hat{y} = 1, \dots, k \\ L(y, 0) &= \alpha && \text{for} \quad y = 1, \dots, k\,. \end{aligned}$$

Derive the optimal Bayes Classifier in this setting. What is the influence of α? When would you prefer this classifier over the one discussed in the lecture? (4 pts)

(5)

a) $L_\lambda^{ridge}(\vec{\beta}) = (\vec{y} - \underline{X}\vec{\beta})^T(\vec{y} - \underline{X}\vec{\beta}) + \lambda \|\vec{\beta}\|_2^2$

$= \sum_{i=1}^{N} (y_i - \vec{x}_i^T \vec{\beta})^2 + \lambda \sum_{j=0}^{2} \beta_j^2$

It can be seen here that the intercept β_0 grows linearly with the regularisation strength λ.

b) We would like the regularisation term to have no β_0 anymore. Change:

$\lambda \|\vec{\beta}\|_2^2 \rightarrow \lambda \|\underline{D}\vec{\beta}\|_2^2$

where

$$\underline{D} = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}.$$

c) Ridge regularisation has a spherical contour, as $\beta_0 + \beta_1 + \beta_2 = \text{const.}$ for specific λ.
When β_0 is excluded, the contour becomes circular, inside the β_1, β_2 plane.

(4)

A BAYES classifier has chooses with

$$f^*(\vec{x}) = \underset{C_j}{\operatorname{argmin}}\, R(C_j|\vec{x}) = \underset{C_j}{\operatorname{argmin}} \sum_k L(C_k, C_j)\, p(C_k, \vec{x})$$

We have

$$L(C_k, C_j) = \begin{cases} 0, \text{ for } k = j \\ \alpha, \text{ for } j = 0 \\ 1, \text{ else} \end{cases}$$

So

$$R(C_j|\vec{x}) = \sum_{k \neq j} L(C_k, C_j)\, p(C_k, \vec{x})$$

$$\Rightarrow R(0|\vec{x}) = \sum_k \alpha\, p(C_k, \vec{x}) = \alpha$$

$$\Rightarrow R(C_j|\vec{x})^{j \neq 0} = \sum_{k \neq j} \alpha\, p(C_k, \vec{x}) = 1 - p(C_k, \vec{x})$$

Then

$$f^*(\vec{x}) = \begin{cases} \underset{C_k}{\operatorname{argmax}}\, p(C_k, \vec{x}), \text{ if } 1 - \max_{C_k} p(C_k, \vec{x}) < \alpha \\ 0, \text{ else} \end{cases}$$

α makes it possible to add general "reject class"

when one prefers not to assign a class at all, rather than misclassifying. We can also write:

$$f^*(\vec{x}) = \begin{cases} \underset{C_k}{\operatorname{argmax}}\, p(C_k, \vec{x}), & \text{if } \max_{C_k} p(C_k, \vec{x}) > 1-\alpha \\ 0 & \text{if } \max_{C_k} p(C_k, \vec{x}) \leq 1-\alpha \end{cases}$$