Machine Learning (SS 23)

Assignment 5: Linear Methods for Classification (Solution)

Mojtaba Nayyeri
Mojtaba.Nayyeri@ipvs.uni-stuttgart.de

Akram Sadat Hosseini
Akram.Hosseini@ipvs.uni-stuttgart.de

Nadeen Fathallah
Nadeen.Fathallah@ipvs.uni-stuttgart.de

Rodrigo Lopez Portillo Alcocer
rodrigo.lopez-portillo-alcocer@ipvs.uni-stuttgart.de

Tim Schneider
timphillip.schneider@ipvs.uni-stuttgart.de

Osama Mohammed
osama.mohammed@ipvs.uni-stuttgart.de

Daniel Frank
daniel.frank@ipvs.uni-stuttgart.de

Make sure to list the full names of all participants, matriculation number, study program, and B.Sc. or M.Sc on the first page. Optionally, you can *additionally* upload source files (e.g., PPTX files). If you have any questions, feel free to ask them in the exercise forum in ILIAS.

Submission is open until Monday, 5th of June 2023, 12:00 noon.

Linear Regression with Regularization

We continue with the lake example from `Assignment 4`. Please follow the instructions in the notebook `05_regression_ctd.ipynb` *Task 1 (Assignment 5)*.

From Linear Regression to Classification

1. Research (e.g. see Hastie et al. chapter 4.1) and then explain what a discriminative function is and how it can be used for classification problems using the tools from linear regression.

   Assume we have a classifier $F : \mathbb{R}^n \mapsto Y$ that maps an input $x \in \mathbb{R}^n$ to a discrete class $y \in Y$, then a *discriminative function* maps the input and class to a scalar value such that

   $$F : x \mapsto \arg\max_{y} f(x, y).$$

   That means a discriminate function has a high value if y is a correct answer to the input x and a low value if y is from the wrong class.

2. In the plots on slide 28 ("Quadratic regression to the rescue") of the *Logistic Regeression* slide deck from the lecture you can see the *Pros* and *Cons* mentioned on slide 25 ("Pros and Cons"). Explain the following:

   (a) Why does masking occur for this particular dataset, even though linear decision boundaries can perfectly separate the classes?

   When the classes are represented by an indicator matrix and linear regression is applied then the predicted values $\hat{Y}$ can take values greater than one and smaller than zero. This can lead to masking which is visible on the left plot in slide 28. The sum of the predicted values is always one (summing over the rows of $\hat{Y}$).

   (b) Assume a fourth class would be lined up (plot on slide 26) in the same space. Would quadratic features be enough to avoid masking?

   Quadratic features would probably not go down fast enough. Therefore, higher-order features might be required to avoid masking. As a rule of thumb, if $K \geq 3$, polynomial terms up to degree $K - 1$ might be needed. Where K is the number of classes.

3. Construct a simple example for linear regression of an indicator matrix with two classes $y \in \{0, 1\}$ and calculate the optimal parameters $\hat{B}$. Which class would the optimal parameters predict on your example data ($\arg\max_{y \in \{0,1\}} \hat{Y}$)?

   $$\hat{Y} = \begin{bmatrix} 0.54965227 & 0.45034773 \\ 0.42924838 & 0.57075162 \\ 0.70715667 & 0.29284333 \\ 0.21448008 & 0.78551992 \\ 0.27223365 & 0.72776635 \\ -\mathbf{0.17277106} & \mathbf{1.17277106} \end{bmatrix}$$

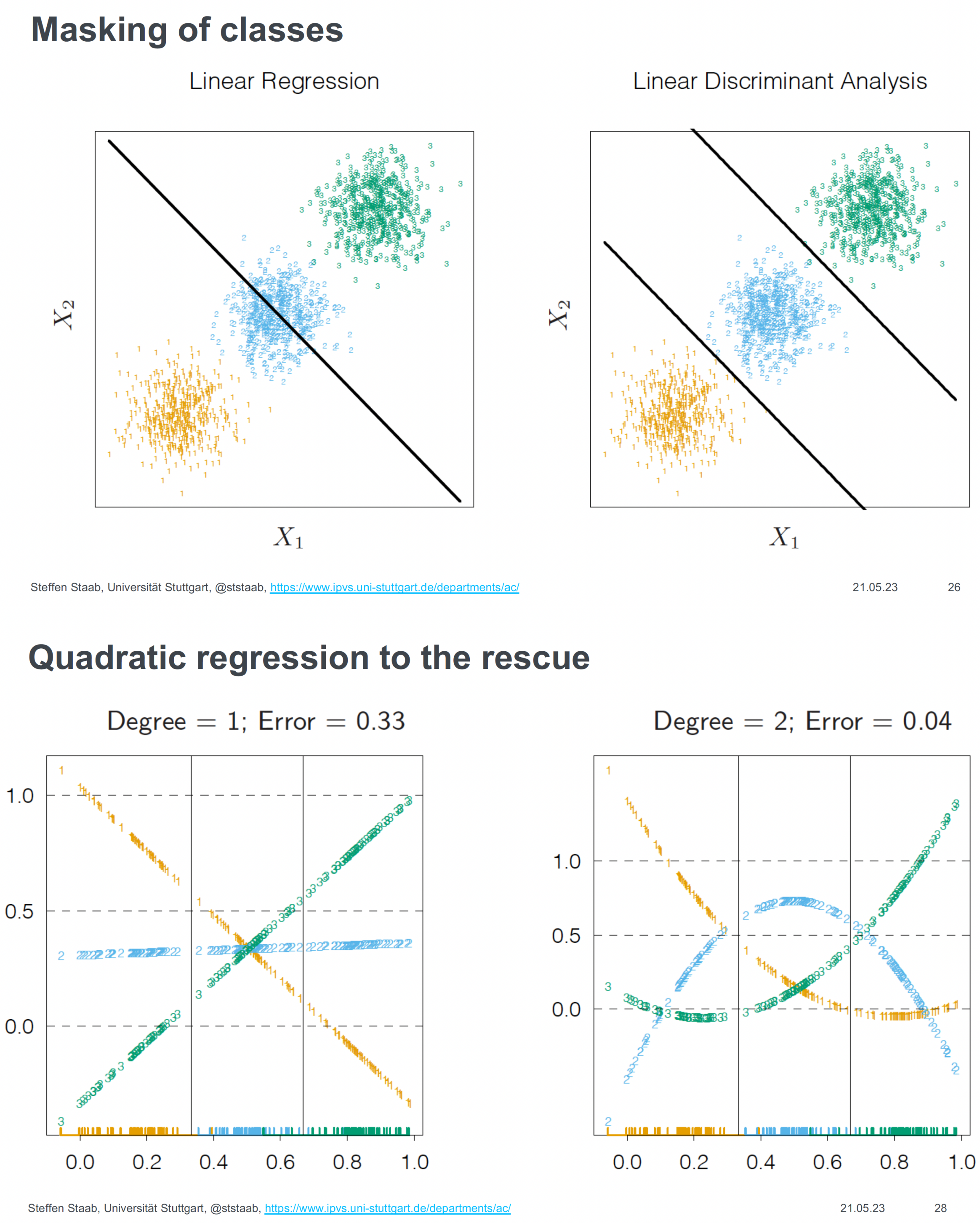
   Each column of $\hat{Y}$ represents one class, and the arg max for the discriminative function is taken row-wise. "The position of the maximum value represents the class". Even though the values look like probabilities, they are not, which can be seen by the bold values in $\hat{Y}$. See Listing 1 for the full example.

Listing 1 Linear regression of an indicator matrix, an example

```python
import numpy as np
import matplotlib.pyplot as plt

X = np.array([[1, 1, 3], [1, 2, 3], [1, 0.3, 1], [1, 5, -1], [1, 3, 4], [1, 7, 3]])
Y = np.array([[0, 1], [1, 0], [1, 0], [0, 1], [0, 1], [0, 1]])

```



Masking of classes

Linear Regression

Linear Discriminant Analysis

Steffen Staab, Universität Stuttgart, @ststaab, https://www.ipvs.uni-stuttgart.de/departments/ac/ 21.05.23 26

Quadratic regression to the rescue

Degree = 1; Error = 0.33

Degree = 2; Error = 0.04

Steffen Staab, Universität Stuttgart, @ststaab, https://www.ipvs.uni-stuttgart.de/departments/ac/ 21.05.23 28

Figure 1 Recap from slide

```python
B_hat = np.linalg.inv(X.T @ X) @ X.T @ Y
Y_hat = X @ B_hat
print(np.argmax(Y_hat, axis=1))
print(Y_hat)

fig, ax = plt.subplots()
ax.plot(X[Y[:, 0] == 1, 1], X[Y[:, 0] == 1, 2], "xb")
ax.plot(X[Y[:, 1] == 1, 1], X[Y[:, 1] == 1, 2], "or")
ax.set_xlabel("$x_1$")
ax.set_ylabel("$x_2$");
```

Log-likelihood gradient and Hessian

Consider a binary classification problem with data $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^n$, $x_i \in \mathbb{R}^d$ and $y_i \in \{0, 1\}$. We define

$$f(x) = \phi(x)^\top \beta \ , \quad p(x) = \sigma(f(x)) \ , \quad \sigma(z) = 1/(1 + e^{-z}).$$

$$L^{\text{nll}}(\beta) = -\sum_{i=1}^n \Big[y_i \log p(x_i) + (1 - y_i) \log[1 - p(x_i)] \Big]$$

where $\beta \in \mathbb{R}^d$ is a vector. (Note: $p(x)$ is a short-hand for $p(y = 1|x)$.)

Note: The gradient and Hessian are needed to compute the optimal parameters for *logistic regerssion* models. Details on how to do this will be covered in the upcoming lecture.

1. Compute the derivative $\frac{\partial}{\partial \beta} L(\beta)$. Tip: Use the fact that $\frac{\partial}{\partial z} \sigma(z) = \sigma(z)(1 - \sigma(z))$.
2. Compute the 2nd derivative $\frac{\partial^2}{\partial \beta^2} L(\beta)$.

Let $p_i \equiv p(x_i)$. We have $\frac{\partial}{\partial \beta} p_i = p_i(1 - p_i)\phi(x_i)^\top$

$$\begin{aligned}
L(\beta) &= -\sum_{i=1}^n \Big[y_i \log p_i + (1 - y_i) \log[1 - p_i] \Big] \\
\frac{\partial}{\partial \beta} L(\beta) &= -\sum_{i=1}^n \Big[y_i \frac{p_i(1 - p_i)}{p_i} \phi(x_i)^\top + (1 - y_i) \frac{-p_i(1 - p_i)}{1 - p_i} \phi(x_i)^\top \Big] \\
&= -\sum_{i=1}^n \Big[y_i(1 - p_i) - (1 - y_i)p_i \Big] \ \phi(x_i)^\top \\
&= \sum_{i=1}^n \Big[p_i - y_i \Big] \ \phi(x_i)^\top = (p - y)^\top X \\
\frac{\partial^2}{\partial \beta^2} L(\beta) &= \frac{\partial}{\partial \beta} \sum_{i=1}^n \phi(x_i) \Big[p_i - y_i \Big] \\
&= \sum_{i=1}^n \phi(x_i) p_i (1 - p_i) \phi(x_i)^\top = X^\top W X \ , \quad W = \text{diag}(p \circ (1 - p))
\end{aligned}$$