Machine Learning:
Assignment Sheet #5

Due on March 15, 2022 at 10:00

Group HB

Henri Sota, Enis Mustafaj

Problem 5.1

In this problem, you will need to remind yourself about the definition of a positive definite matrix. Moreover you will look for local maxima and minima of a function of two variables.

a) Compute for the following matrices, whether they are positive definite.

$$M = \begin{bmatrix} 1 & 2 \\ 2 & 1 \end{bmatrix}, \quad N = \begin{bmatrix} 2 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 2 \end{bmatrix}$$

To compute if a matrix is positive definite, we compute the eigenvalues of the matrix and see if they are positive.

a) Eigenvalues of the first matrix

$$M - \lambda \cdot I = \begin{bmatrix} 1 - \lambda & 2 \\ 2 & 1 - \lambda \end{bmatrix}$$

To calculate the eigenvalues we find the roots of the determinant.

$$\Delta = (1 - \lambda)^2 - 4 = 0$$

From the equation above we find that the eigenvalues are -1 and -3, so the matrix is not positive definite.

b) Eigenvalues of the second matrix For the second matrix we repeat the steps from above.

$$N - \lambda \cdot I = \begin{bmatrix} 2 - \lambda & -1 & 0 \\ -1 & 2 - \lambda & -1 \\ 0 & -1 & 2 - \lambda \end{bmatrix}$$

$$\Delta = (2 - \lambda) \cdot [(2 - \lambda)^2 - 1] - (\lambda - 2) = 0$$
$$\Delta = (2 - \lambda) \cdot [(2 - \lambda)^2 - 2] = 0$$

From the equation above, the eigenvalues are 2, $2 - \sqrt{2}$, $2 + \sqrt{2}$. The eigenvalues are positive, so the matrix is positive definite.

b) You are given the following function:

$$f(x, y) = 3x^2 y + y^3 - 3x^2 - 3y^2 + 2$$

Use the Knowledge 4.1 statement from the lecture notes to find and classify all locations of minima and maxima of this function.

The gradient of the function $f(x, y)$ is

$$\nabla f = \begin{pmatrix} \dfrac{\partial f(x, y)}{\partial x} \\ \dfrac{\partial f(x, y)}{\partial y} \end{pmatrix} = \begin{pmatrix} 6xy - 6x \\ 3x^2 + 3y^2 - 6y \end{pmatrix}$$

The points for which the gradient is equal to $\vec{0}$ are $\{(1, 1), (-1, 1), (0, 2), (0, 0)\}$. The Hessian matrix of the function f is

$$\mathcal{H}_{f(x,y)} = \begin{pmatrix} \dfrac{\partial^2 f(x, y)}{\partial x^2} & \dfrac{\partial^2 f(x, y)}{\partial x \cdot \partial y} \\ \dfrac{\partial^2 f(x, y)}{\partial y \cdot \partial x} & \dfrac{\partial^2 f(x, y)}{\partial y^2} \end{pmatrix} = \begin{pmatrix} 6y - 6 & 6x \\ 6x & 6y - 6 \end{pmatrix}$$

- For point $(1, 1)$ we get the matrix : $\begin{pmatrix} 0 & 6 \\ 6 & 0 \end{pmatrix}$ with eigenvalues 6 and -6. The matrix is not positive or negative definite, so the point is neither local minimum or maximum.

- For point $(-1, 1)$ we get the matrix : $\begin{pmatrix} 0 & -6 \\ -6 & 0 \end{pmatrix}$ with eigenvalues 6 and -6. The matrix is not positive or negative definite, so the point is neither local minimum or maximum.

- For point $(0, 2)$ we get the matrix : $\begin{pmatrix} 6 & 0 \\ 0 & 6 \end{pmatrix}$ with eigenvalues 6 and 6. The matrix is positive definite, so the point is local minimum.

- For point $(0, 0)$ we get the matrix : $\begin{pmatrix} -6 & 0 \\ 0 & -6 \end{pmatrix}$ with eigenvalues -6 and -6. The matrix is negative definite, so the point is local maximum.

Problem 5.2

You are given the training data $\mathcal{T} = \left\{ ((1,1)^\top, 2), ((1,2)^\top, 3), ((2,2)^\top, 3), ((2,4)^\top, 4) \right\}$.

a) Use linear regression by least squares to compute a predictor for the output. Do this by hand, i.e., don't use the computer and show all computation steps.

We create the matrix X by adding the 1 before each training data point:

$$
X = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 2 & 2 \\ 1 & 2 & 4 \end{bmatrix}
\quad
X^T = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 2 & 2 \\ 1 & 2 & 2 & 4 \end{bmatrix}
\quad
y = \begin{pmatrix} 2 \\ 3 \\ 3 \\ 4 \end{pmatrix}
$$

We compute the matrix A by:

$$
A = X^T \cdot X = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 2 & 2 \\ 1 & 2 & 2 & 4 \end{bmatrix} \cdot \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 2 & 2 \\ 1 & 2 & 4 \end{bmatrix} = \begin{bmatrix} 4 & 6 & 9 \\ 6 & 10 & 15 \\ 9 & 15 & 25 \end{bmatrix}
$$

We compute the vector b by:

$$
b = X^T \cdot y = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 2 & 2 \\ 1 & 2 & 2 & 4 \end{bmatrix} \cdot \begin{pmatrix} 2 \\ 3 \\ 3 \\ 4 \end{pmatrix} = \begin{pmatrix} 12 \\ 19 \\ 30 \end{pmatrix}
$$

Now, we need to solve the equation $A \cdot \beta = b$. We use Cholesky decomposition for express the matrix A in the form $L \cdot L^T$, where L is a matrix of the form:

$$
L = \begin{bmatrix} a & 0 & 0 \\ b & c & 0 \\ d & e & f \end{bmatrix}
$$

To compute the entries of the matrix L, we use the following formulas:

$$
l_{kk} = \sqrt{a_{kk} - \sum_{j=1}^{k-1} l_{kj}^2} \quad l_{ki} = \frac{a_{ki} - \sum_{j=1}^{i-1} l_{ij} \cdot l_{kj}}{l_{ii}}
$$

$$l_{11} = \sqrt{a_{11}} = 2 \qquad l_{21} = \frac{a_{21}}{l_{11}} = 3 \qquad l_{22} = \sqrt{a_{22} - l_{21}^2} = 1$$

$$l_{31} = \frac{a_{31}}{l_{11}} = \frac{9}{2} \qquad l_{32} = \frac{a_{32} - l_{21} \cdot l_{31}}{l_{22}} = \frac{3}{2} \qquad l_{33} = \sqrt{a_{33} - l_{31}^2 - l_{32}^2} = \frac{\sqrt{10}}{2}$$

The matrix L is:

$$L = \begin{bmatrix} 2 & 0 & 0 \\ 3 & 1 & 0 \\ \frac{9}{2} & \frac{3}{2} & \frac{\sqrt{10}}{2} \end{bmatrix}$$

The equation now becomes $L \cdot L^T \cdot \vec{\beta} = \vec{b}$. We express $L^T \cdot \vec{\beta} = \vec{x}$. The final equation becomes:

$$L \cdot \vec{x} = \vec{b} \qquad \begin{bmatrix} 2 & 0 & 0 \\ 3 & 1 & 0 \\ \frac{9}{2} & \frac{3}{2} & \frac{\sqrt{10}}{2} \end{bmatrix} \cdot \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{pmatrix} 12 \\ 19 \\ 30 \end{pmatrix}$$

From the above equation we get:

$$\vec{x} = \begin{pmatrix} 6 \\ 1 \\ \frac{3 \cdot \sqrt{10}}{10} \end{pmatrix}$$

The last equation is:

$$L^T \cdot \vec{\beta} = \vec{x}$$

$$\begin{bmatrix} 2 & 3 & \frac{9}{2} \\ 0 & 1 & \frac{3}{2} \\ 0 & 0 & \frac{\sqrt{10}}{2} \end{bmatrix} \cdot \begin{pmatrix} \beta_1 \\ \beta_2 \\ \beta_3 \end{pmatrix} = \begin{pmatrix} 6 \\ 1 \\ \frac{3 \cdot \sqrt{10}}{10} \end{pmatrix}$$

From the equation above we get:

$$\vec{\beta} = \begin{pmatrix} 1.5 \\ 0.1 \\ 0.6 \end{pmatrix}$$

The linear predictor is:

$$f(x_1, x_2) = 1.5 + 0.1 \cdot x_1 + 0.6 \cdot x_2$$

b) Predict the output for $\mathbf{x} = (1.5, 1.5)^\top$.

$$f(1.5, 1.5) = 1.5 + 0.1 \cdot 1.5 + 0.6 \cdot 1.5 = 2.55$$

The predicted value of $(1.5, 1.5)$ is 2.55.

Problem 5.3

Prove Theorem 3.2 from the lecture. If you consider this as too hard, consult the literature from this field to solve the problem. In case you use external literature, make sure that you give a citation for that literature (i.e. say where you took it from) and rephrase the proof appropriately.

Assuming the input, output and mixed joint density from Theorem 3.2. To find the function $f : R^D \to R_G$ that minimizes the expected prediction error $\text{EPE}(f)$ with respect to the 0 - 1 loss, the following steps can

be taken:

$$\arg\min_{g \in R_G} \text{EPE}(f(\mathbf{x})) = \arg\min_{g \in R_G} E[L(g, f(\mathbf{x}))|\mathbf{X} = \mathbf{x}]$$

$$= \arg\min_{g \in R_G} P(g = g_1|\mathbf{X} = \mathbf{x}) \cdot L(g_1, f(\mathbf{x})) + \ldots + P(g = g_r|\mathbf{X} = \mathbf{x}) \cdot L(g_r, f(\mathbf{x}))$$

$$= \arg\min_{g \in R_G} \sum_{i=1}^{r} P(g = g_i|\mathbf{X} = \mathbf{x}) \cdot L(g_i, f(\mathbf{x})$$

Due to the loss function having a range of 0 and 1, the last result can be simplified to a sum of the probabilities for which the predicted class, $f(x)$, does not equal the true class, g.

$$\arg\min_{g \in R_G} \text{EPE}(f(\mathbf{x})) = \arg\min_{g \in R_G} P(g \neq f(\mathbf{x})|\mathbf{X} = \mathbf{x})$$

$$= \arg\min_{g \in R_G} 1 - P(g = f(\mathbf{x})|\mathbf{X} = \mathbf{x})$$

$$= \arg\max_{g \in R_G} P(g = f(\mathbf{x})|\mathbf{X} = \mathbf{x})$$

If $\mathfrak{g}$ is the true output class, such that $p(\mathfrak{g}|\mathbf{x}) = \max_{g \in R_G} p(g|\mathbf{x})$, then the minimized $f(\mathbf{x})$ found above matches the definition of the Bayes classifier.

Programming Problem 5.1

Consider the Examples 4.2 and 4.3 from the lecture, for which you also have access to the source code. Complete the following tasks:

a) (Re-)implement Example 4.3. This time, however you need to implement the linear regression yourself, without a machine learning library. (If you implement in Python, just start from the available Jupyter notebook.) Verify the correctness of your implementation by cross-checking it with Example 4.3.

b) Apply your implementation to the Energy efficiency Data Set from the UCI Machine Learning Repository. Build the predictor for the required heating load on the full data set and predict the load on the first three samples of the data set.

c) Implement Example 4.5 starting from Example 4.2 from the lecture notes.

The implementation of the linear regressor by least squares can be found in the file `programming_exercises.ipynb`. The class offers two methods, `fit` and `predict`, which add the data and train the model, and predict the expected regression value.