Machine Learning 2017/2018

Home Assignment 1

Yevgeny Seldin, Christian Igel
Department of Computer Science, University of Copenhagen

The deadline for this assignment is **28 November 2017**. You must submit your *individual* solution electronically via the Absalon home page.

A solution consists of:

- A PDF file with detailed answers to the questions, which may include graphs and tables if needed. Do *not* include your source code in the PDF file.
- A .zip file with all your solution source code (Matlab / R / Python scripts / C / C++ / Java / etc.) with comments about the major steps involved in each question (see below). Source code must be submitted in the original file format, not as PDF.
- IMPORTANT: Do NOT zip the PDF file, since zipped files cannot be opened in the speed grader. Zipped pdf submissions will not be graded.
- Your PDF report should be self-sufficient. I.e., it should be possible to grade it without opening the .zip file. We do not guarantee opening the .zip file when grading.
- Your code should be structured such that there is one main file (or one main file per question) that we can run to reproduce all the results presented in your report. This main file can, if you like, call other files with functions, classes, etc.
- Your code should include a README text file describing how to compile and run your program, as well as a list of all relevant libraries needed for compiling or using your code.

Math and ML

You need basic linear algebra and calculus for understanding machine learning. To recall some of your math knowledge, answer the following questions. Do the calculations by hand.

If you feel unsure about what a question means and how to answer it, this indicates that you are not fully comfortable with mathematical skills that are assumed in this course. No worries. In this case, just take your time and to go back to your notes from school or your first study years – or grab one of the numerous textbooks that are around.

Feel free to ask questions about the mathematical background in the exercise classes!

1 Vectors and Matrices [Optional]

This is an optional question and it is not counted in grading. If you feel comfortable with the material you can skip it. Otherwise, you are strongly encouraged to solve it. If you submit your solution, you will get feedback from the TAs.

Consider the two vectors

$$\boldsymbol{a} = \begin{pmatrix} 1 \\ 2 \\ 2 \end{pmatrix} \text{ and } \boldsymbol{b} = \begin{pmatrix} 3 \\ 2 \\ 1 \end{pmatrix}$$

and the matrix

$$\boldsymbol{M} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 4 & 0 \\ 0 & 0 & 2 \end{pmatrix} \quad .$$

Question 1.1. Calculate the *inner product* (also known as *scalar product* or *dot product*) denoted by $\langle \boldsymbol{a}, \boldsymbol{b} \rangle$, $\boldsymbol{a}^{\mathrm{T}}\boldsymbol{b}$, or $\boldsymbol{a} \cdot \boldsymbol{b}$.

Question 1.2. Calculate the length (also known as Euclidean norm) $\|\boldsymbol{a}\|$ of the vector $\boldsymbol{a}$.

Question 1.3. Calculate the *outer product* $\boldsymbol{a}\boldsymbol{b}^{\mathrm{T}}$. Is it equal to the inner product $\boldsymbol{a}^{\mathrm{T}}\boldsymbol{b}$ you computed in Question 1.1?

Question 1.4. Calculate the four quantities, $\boldsymbol{a}\boldsymbol{b}^{\mathrm{T}}$, $\boldsymbol{a}^{\mathrm{T}}\boldsymbol{b}$, $\boldsymbol{b}\boldsymbol{a}^{\mathrm{T}}$, $\boldsymbol{b}^{\mathrm{T}}\boldsymbol{a}$. Which of them are equal and which are not? (Test yourself: there is one pair that is equal and all the remaining quantities are different.)

Question 1.5. Calculate the inverse of matrix $\boldsymbol{M}$, denoted by $\boldsymbol{M}^{-1}$. We remind that you should get that $\boldsymbol{M}\boldsymbol{M}^{-1} = \boldsymbol{I}$, where $\boldsymbol{I}$ is the identity matrix.

Question 1.6. Calculate the matrix-vector product $\boldsymbol{M}\boldsymbol{a}$.

Question 1.7. Let $\boldsymbol{A} = \boldsymbol{a}\boldsymbol{b}^{\mathrm{T}}$. Calculate the transpose of $\boldsymbol{A}$, denoted by $\boldsymbol{A}^{\mathrm{T}}$. Is $\boldsymbol{A}$ symmetric? (A matrix is called symmetric if $\boldsymbol{A} = \boldsymbol{A}^{\mathrm{T}}$.)

Question 1.8. What is the rank of $\boldsymbol{A}$? (The rank is the number of linearly independent columns.) Give a short explanation.

Question 1.9. What should be the relation between the number of columns and the rank of a square matrix in order for it to be invertible? Is $\boldsymbol{A} = \boldsymbol{a}\boldsymbol{b}^{\mathrm{T}}$ invertible?

2 Derivatives [Optional]

This is an optional question and it is not counted in grading. If you feel comfortable with the material you can skip it. Otherwise, you are strongly encouraged to solve it. If you submit your solution, you will get feedback from the TAs.

We denote the derivative of a univariate function $f(x)$ with respect to the variable x by $\frac{df(x)}{dx}$. We denote the partial derivative of a multivariate function $f(x_1, \ldots, x_n)$ with respect to the variable x_i, where $1 \leq i \leq n$, by $\frac{\partial f(x_1,\ldots,x_n)}{\partial x_i}$. The partial derivative $\frac{\partial f(x_1,\ldots,x_n)}{\partial x_i}$ is the derivative of $f(x_1, \ldots, x_n)$ with respect to x_i when we treat all other variables x_j for $j \neq i$ in f as constants.

Please recall the basic rules for derivatives, namely the sum rule, the chain rule, and the product rule, see the lecture notes available on Absalon.

Question 2.1. What is the derivative of $f(x) = \frac{1}{1+\exp(-x)}$ with respect to x?

Question 2.2. What is the partial derivative of $f(w, x) = 2(wx+5)^2$ with respect to w?

3 Probability Theory: Sample Space [Optional]

This is an optional question and it is not counted in grading. If you feel comfortable with the material you can skip it. Otherwise, you are strongly encouraged to solve it. If you submit your solution, you will get feedback from the TAs.

An urn contains five red, three orange, and one blue ball. Two balls are randomly selected (without replacement).

1. What is the sample space of this experiment?
2. What is the probability of each point in the sample space?
3. Let X represent the number of orange balls selected. What are the possible values of X?
4. Calculate $\mathbb{P}\{X = 0\}$.
5. Calculate $\mathbb{E}[X]$.

4 Probability Theory: Properties of Expectation

Let X and Y be two discrete random variables taking values in $\mathcal{X}$ and $\mathcal{Y}$, respectively. Starting from the definitions, prove the following identities:

1. $\mathbb{E}[X+Y] = \mathbb{E}[X] + \mathbb{E}[Y]$.

2. If X and Y are independent then $\mathbb{E}[XY] = \mathbb{E}[X]\,\mathbb{E}[Y]$. (Mark the step where you are using the independence assumption. Note that this assumption was not required in point 1.)

3. Provide an example of two random variables X and Y for which $\mathbb{E}[XY] \neq \mathbb{E}[X]\,\mathbb{E}[Y]$. (Describe how you define the random variables, provide a joint probability distribution table [see comment below], and calculate $\mathbb{E}[XY]$ and $\mathbb{E}[X]\,\mathbb{E}[Y]$.)

4. $\mathbb{E}[\mathbb{E}[X]] = \mathbb{E}[X]$.

5. Variance of a random variable is defined as $\mathbb{V}[X] = \mathbb{E}\left[(X - \mathbb{E}[X])^2\right]$. Show that $\mathbb{E}\left[(X - \mathbb{E}[X])^2\right] = \mathbb{E}[X^2] - (\mathbb{E}[X])^2$.

Comment: A convenient way to represent a joint probability distribution of two discrete random variables is a table. For example, if X and Y are Bernoulli random variables with bias $\frac{1}{2}$ (fair coins), then the joint distribution table looks like this:

$X\backslash Y$	0	1
0	1/4	1/4
1	1/4	1/4

.

And if Z_1 and Z_2 are Bernoulli random variables with bias $\frac{1}{2}$ and we define $X = Z_1 + Z_2$ and $Y = Z_1 \times Z_2$, then the joint distribution of X and Y is:

$X\backslash Y$	0	1
0	1/4	0
1	1/2	0
2	0	1/4

.

5 Probability Theory: Complements of Events

1. The complement of event A is denoted by $\bar{A}$ and defined by $\bar{A} = \Omega \setminus A$. Starting from probability axioms prove that $\mathbb{P}\{A\} = 1 - \mathbb{P}\{\bar{A}\}$.

2. In many cases it is easier to calculate the probability of a complement of an event than to calculate the probability of the event itself. Use this to solve the following question. We flip a fair coin 10 times.

- What is the probability to observe at least one tail?
- What is the probability to observe at least two tails?

6 Digits Classification with Nearest Neighbors

In this question you will implement and apply Nearest Neighbors learning algorithm to classify handwritten digits.

Preparation

- Download `MNIST-cropped-txt.zip` file from Absalon. The file contains `MNIST-Train-cropped.txt`, `MNIST-Test-cropped.txt`, `MNIST-Train-Labels-cropped.txt`, and `MNIST-Test-Labels-cropped.txt` files.
- `MNIST-Train-cropped.txt` is a space-separated file of real numbers. It contains a 784 × 10000 matrix, written column-by-column (the first 784 numbers in the file correspond to the first column; the next 784 numbers are the second column, and so on).
- Each column in the matrix above is a 28 × 28 grayscale image of a digit, stored column-by-column (the first 28 out of 784 values correspond to the first column of the 28 × 28 image, the next 28 values correspond to the second column, and so on). Test yourself: reshape the first column into a 28 × 28 matrix and display it as an image - did you get an image of digit "5"?
- `MNIST-Train-Labels-cropped.txt` is a space-separated file of 10000 integers. The numbers label the images in `MNIST-Train-cropped.txt` file: the first number ("5") is the number drawn in the image corresponding to the first column; the second number corresponds to the second column, and so on.
- `MNIST-Test-cropped.txt` is a space-separated file of real numbers containing 784 × 2000 matrix of test images.
- `MNIST-Test-Labels-cropped.txt` is a space-separated file of 2000 integers labeling the images in `MNIST-Test-cropped.txt` file.

Tasks

1. Compare K-NN algorithms for $K = 1, 3, 5, 7, \dots, 33$ in their ability to distinguish between digits "0" and "1".
2. Repeat the same for "0" and "8".
3. Repeat the same for "5" and "6".

Detailed Instructions

- For each of the tasks above pick the images of the corresponding digit from `MNIST-Train-cropped.txt` file for training.
- Pick the images of the corresponding digit from `MNIST-Test-cropped.txt` for the test set.
- **Validation**: use the first 80% of the training images for training and the last 20% of the images for calculating the validation error. Plot the validation error as a function of K and compare it with the test error. Test error should be calculated on the test set (as described above).
  - How well does the validation error match the test error?
  - How closely does the best value of K according to the validation error match the best value of K according to the test error?
  - How the validation and test errors behave as a function of K?
  - Does the best value of K depend on the difficulty of the task and how? (It is easier to tell apart "0" and "1" than "5" and "6"; the difficulty of separating "0" and "8" should be somewhere in between.)

Practical Details and Some Practical Advice

- Use square Euclidean distance to calculate the distance between the images. If $\mathbf{x}_1$ and $\mathbf{x}_2$ are two 784-long vectors representing two images, then the distance is $\|\mathbf{x}_1 - \mathbf{x}_2\|^2 = (\mathbf{x}_1 - \mathbf{x}_2)^T(\mathbf{x}_1 - \mathbf{x}_2)$.
- If you work with an interpreted programming language, such as Python or MATLAB, do your best to use vector operations and avoid for-loops as much as you can. This will make your code orders of magnitude faster.
- Assume that $\mathbf{X} = \left(\begin{pmatrix} | \\ \mathbf{x}_1 \\ | \end{pmatrix}, \cdots, \begin{pmatrix} | \\ \mathbf{x}_n \\ | \end{pmatrix}\right)$ is a matrix holding data vectors $\mathbf{x}_1, \dots, \mathbf{x}_n$ and you want to calculate distances between all these points and a test point $\mathbf{x}$. *Do your best to avoid a for-loop!* One way of doing so

is to create another matrix $\mathbf{X}' = \left(\begin{pmatrix} | \\ \mathbf{x} \\ | \end{pmatrix}, \cdots, \begin{pmatrix} | \\ \mathbf{x} \\ | \end{pmatrix}\right)$ and calculate all n distances in one shot using matrix and vector operations.

- Note that you can compute the output of K-NN for all K in one shot using vector operations.
- It is possible to write the whole exercise with a single for-loop over the test/validation set.
- You may find the following functions useful:
  - Built-in sorting functions for sorting the distances.
  - Built-in functions for computing a cumulative sum of elements of a vector `v` (for computing the predictions of K-NN for all K at once).
- It may be a good idea to debug your code with a small subset of the data.

Deliverables For each of the three tasks you should include in your `.pdf` report:

- A plot of validation and test error.
- Do not forget to add titles, axis labels, and legends to your plots! We will deduct points if they are missing!

Your `.pdf` report should also include:

- Discussion of the three plots according to the guidelines provided in the exercise.

Include all your code in the accompanying `.zip` file.

As a general rule, you must not use the test data in the model building process at all (neither for training, data normalization, nor hyperparameter selection), because otherwise you may get a biased estimate of the generalization performance of the model (see [1, Example 5.3]).

7 Linear regression

Consider the data in the file `DanWood.dt`, which contains measurements from an experiment originally described by Keeping [3] and used in several textbooks [2, 4]. The data is one of the statistical reference data sets of the US American *National Institute of Standards and Technology.*

The experiment studies a carbon filament lamp. For a given absolute temperature of the carbon filament the energy radiated from the filament was recorded. Temperature was measured in units of 1000 degrees Kelvin and energy radiation per cm^2 per second.

Each line in `DanWood.dt` is one training pattern. The first number is the input (x), the absolute temperature, and the second is the corresponding output / target / label (y), the radiated energy.

Tasks

1. Implement the linear regression algorithm as described in the lecture. For vector and matrix operations such as computing the inverse of a matrix you can use high-level (library) functions.
2. Build an *affine* linear model $h : \mathbb{R} \to \mathbb{R}$ of the data described above using linear regression. Report the two parameters of the model as well as the mean-squared-error of the model computed over the complete data set.
3. Plot the data and the regression line. The plot must have proper axes labels and a legend.
4. Compute and report the variance of the labels in the training data set (i.e., the variance of the y values). Now compare the variance to the mean-squared-error of your model. Discuss the quotient of the two quantities, the mean-squared-error over the variance. What does it mean if the quotient is larger or smaller than one?

Deliverables: source code, plot of the data and the regression line, mean-squared error, parameters of the regression model, variance of y values, discussion of mean-squared-error over the variance

References

[1] Y. S. Abu-Mostafa, M. Magdon-Ismail, and H.-T. Lin. *Learning from Data.* AMLbook, 2012.

[2] C. Daniel and F. S. Wood. *Fitting Equations to Data*, pages 428–431. John Wiley and Sons, 1980.

[3] E. S. Keeping. *Introduction to Statistical Inference*, page 354. Van Nostrand Company, Princeton, NJ, 1962.

[4] A. B. Shiflet and G. W. Shiflet. *Introduction to Computational Science: Modeling and Simulation for the Sciences*. Princeton University, 2006.