F20/21DL. Data Mining and Machine Learning

Lab 9. Linear and Logistic Regression

Covering Practical work to be done by students in Week 9

The purpose of this lab is:

1. to practice what we have learned so far:
   - Linear Regression
   - Logistic Regression
2. understand methods applied in Supervised learning; common pitfalls in supervised learning, and especially the problem of linear separability of data;
3. understand practical issues arising in linear and logistic regression, and differences between the two;
4. prepare for the electronic test;
5. to help you to make progress with Python tutorial and your DM & ML portfolio.

1 Linear Classifiers: Understanding the algorithm for Linear Regression

1.1 Linear Regression

Prepare for Test on algorithm for Linear Regression.

1. Take again the data set with 10 smiley/sad faces (from Lab 5).
2. Convert it to numeric form: Black → 1, White → 0, Happy → 1, Sad → 0. We will in fact need only three first rows, so lets take:

Picture	Cell 33	Cell 42	Cell 48	Cell 58	Face expression
P1	0	1	0	0	1
P2	1	1	0	0	1
P3	0	0	0	1	0
...	...	...	...	...	...

3. Manually execute the Linear Regression algorithm for it (as given in the Lecture), taking first three examples in turn. Do just one iteration over each example. The settings are:
   - We learn the linear function:
     $pval(e, Emotion) = w0 + w1 * Cell33 + w2 * Cell42 + w3 * Cell48 + w4 * Cell58$
   - Random weight initialisation: $w_0 = 1$, $w_1 = 2$, $w_2 = 1$, $w_3 = -2$, $w_4 = -1$.
   - Learning rate $\eta = 1$
4. Record your results, as well as intermediate values in the computation, be ready to answer questions.

5. Check what the resulting linear classifier predicts for our test set (converted to numbers):
    - Test 1: `1, 0, 0, 0, ???`
    - Test 2: `1, 1, 0, 1, ???`
6. Submit your answers on Canvas, check correct solutions.

1.2 Logitic Regression

1. Read the lecture slides, make sure you understand them, ask questions on the Forum on Canvas.
    - (*** Optional) In the lectures, we computed the derivative of the squared Euclidean distance error function for linear classifiers. Do a similar derivation for logistic classifiers. That is, show why, for each given w_i, the following holds:

      $$((val(e,Y) - pval(e,Y))^2)' = 2 \times \delta \times pval(e,Y) \times [1 - pval(e,Y)] \times val(e,X_i)$$

      if $pval(e,Y) = \sigma(\sum_i w_i \times val(e,X_i))$ and $\delta = val(e,Y) - pval^{\overline{w}}(e,Y)$.
2. Check relevant chapters on Linear and Logistic Regression in the recommended textbook: Data Mining, by Witten et al. (2011) §4.6, pp.124-129; §11.4, 459-469. In 2017 edition: §4.6 (pp. 128-133).

2 DM & ML Portfolio

This part is to be completed in groups, and will be assessed during the labs. Marking scheme: this lab will bring you up to 2 points. 1 point for completing the task, 1 additional point for any non-trivial analytical work with the material.

2.1 Python Tutorial and Programming Practice (Prior to the lab)

This part is for your individual programming practice during the week.

- Watch recordings, and run the Python code accompanying tutorial **P5. Decision Trees, Linear Regression and Logistic Regression (week 8)**.
- Make sure you can run this code using **your chosen data set**. In case you have any issues, contact your lab tutor and ask for help.
- Make sure that you obtained or created a test set. Make sure that your class feature is converted to numeric.
- Run a Linear classifier on the training data set, mark the mean squared error (MSE). What hypothesis can you make about this data set being linearly separable or not?

  Note also its MSE on the test set. How well does the linear classifier generalize to new data?
- Use Logistic regression on your training set. Then measure the error on the training set. Record all your findings and explain them.
- *(optional for BSc but recommended for higher marks, mandatory for MSc)* Experiment with various regression parameters that control the learning. For example: the learning rate, the number of iterations and batch size.
- *(optional for BSc but recommended for higher marks, mandatory for MSc)* Put all your results in a suitable form: it can be a table or a series of graphs, that visualise the variations of performance between different settings of the regression algorithm. (Lab 5 gave an example of how machine learning experiments may be assembled into a comparative table. You can use it as a starting point. But let it not limit your creativity.)

2.2 During the lab:

- Firstly, using the results obtained for the *linear classifier*, make conclusions: is your data set linearly separable?
- Secondly, make conclusions about your experiments with tuning parameters of the *logistic classifier*. Make conclusions: what was the influence of various parameters on the classifier's performance? Hypothesise why.
- **The tutors will mark:** quality of your code, completeness of your tables/graphs that summarise the results of your group experiments and your analysis of the tables/graphs, i.e. what sort of conclusions you make, how well the conclusions reflect your understanding of the algorithms.

2.3 After the lab:

- *Group rep:* Make sure all group members have tasks for the week
- *Everyone:* Incorporate the discussion during the lab into your Python code
- *Everyone:* Incorporate all code used in the lab into your Portfolio repository.