Stars & Fraud

Exam Assignment

Christian Igel and Kim Steenstrup Pedersen
Department of Computer Science
University of Copenhagen

This is the final exam assignment on the course *Statistical Methods for Machine Learning*, block 3, 2015, at the University of Copenhagen. It is based on the full course curriculum. The assignment is centered around real world pattern recognition tasks.

This assignment must be made and submitted *individually*. It is acceptable (and we encourage) to use bits and pieces of your solutions and the hand-out code from the previous assignments.

Your solution to this assignment will be graded using the 7-point scale and will be your final grade for the course. To obtain the best grade of 12, you must fulfil all the course learning objectives (see course description) at an excellent level. In terms of the questions in this assignment, this means that you have to answer all questions with no or only a few mistakes or parts missing. To obtain the passing grade of 02 you need to fulfil the learning objectives at a minimum level, which means you have to make a serious attempt at solving the central questions in the assignment (but not necessarily all) with some mistakes allowed.

The deadline for this assignment is **April 7, 2015**. You must submit your solution electronically via the Absalon home page. Go to the assignments list and choose this assignment and upload your solution prior to the deadline.

Solution format

The deliverables for each question are listed at the end of each question. The deliverable "description of software used" means that you should hand in the source code you have written to solve the problem. If you have used a software library to solve the problem, this library should be described and reasons for the particular choice should be given.

Thus, a solution should contain:

- A PDF document showing your results and giving detailed answers to the

questions. If relevant, this may include graphs and tables with comments (**max. 10 page of text including figures and tables**). Use meaningful labels, captions, and legends. Do **not** include your source code in this PDF file. You will be graded mainly on the basis of this report.

- Your solution code (Matlab / R / Python scripts or C / C++ / Java code). The code must be submitted in its original format (e.g., in `.m`, `.R` or `.py` file format – not as PDF files). Use meaningful names for files, constants, variables, functions and procedures etc. Add comments to the code to make it more readable. Your code should be structured such that there is one main file 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 also include a README text file describing how to compile (if relevant) and run your program, as well as a list of all relevant libraries needed for compiling or using your code. If we cannot make your code run we will consider your submission incomplete.

Photometric Redshift Estimation

Astronomy is rich with data. The advent of wide-area digital cameras on large aperture telescopes has led to ever more ambitious surveys of the sky. The data volume of an entire survey of a decade ago can now be acquired in a single night. Automatic data analysis methods are required to fully exploit this wealth of data.

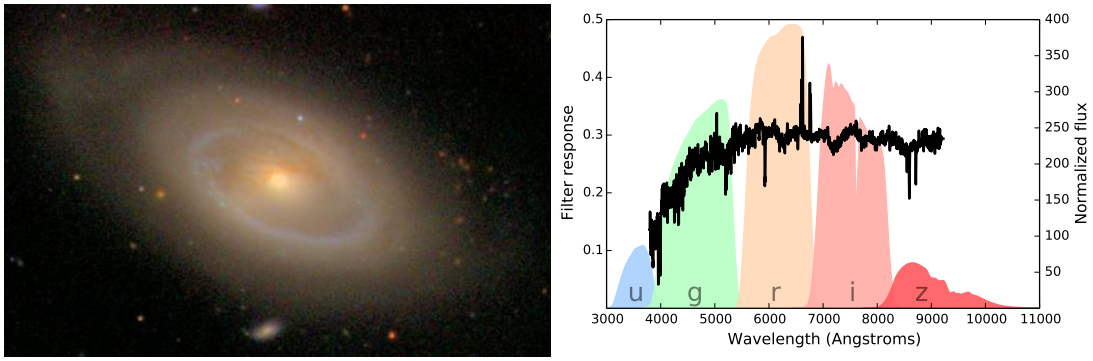
Here we consider photometric redshift estimation of galaxies. The redshift phenomenon is caused by the Doppler effect, which shifts the spectrum of an object towards longer wavelengths if it is moving away from the observer. Because the universe is expanding uniformly, we can infer a galaxy's velocity by its redshift and, thus, its distance to Earth. Hence, redshift estimation is a useful tool for determining the geometry of the universe. A photometric observation contains the intensities of an object (in our case, galaxies) in 5 different bands (u,g,r,i,z), ranging from ultraviolet to infrared, see Figure 1. Spectroscopy, in contrast, measures the photon count at certain wavelengths. The resulting spectrum allows for identifying the chemical components of the observed object and thus, enables determining many interesting properties, including the redshift. Spectroscopy, however, is much more time-consuming than photometric observation and therefore, costs could be greatly reduced if we could predict suitable candidates for follow-up spectroscopy from low-quality low-cost photometry.

Your task is to train and evaluate a photometric redshift estimator using astronomical data [Sheldon et al., 2012]. For each of the 5 bands a model point spread function (*model*) and a composite model (*cmodel*) are fit to the photo-

Feature #	Feature name
0	model magnitude (*model*) difference between bands *u*-*g*
1	model magnitude (*model*) difference between bands *g*-*r*
2	model magnitude (*model*) difference between bands *r*-*i*
3	model magnitude (*model*) difference between bands *i*-*z*
4	model magnitude (*model*) in band *r*
5	composite model magnitude (*cmodel*) difference between bands *u*-*g*
6	composite model magnitude (*cmodel*) difference between bands *g*-*r*
7	composite model magnitude (*cmodel*) difference between bands *r*-*i*
8	composite model magnitude (*cmodel*) difference between bands *i*-*z*
9	composite model magnitude (*cmodel*) in band *r*
target	redshift (determined by spectroscopy)

Table 1: Relevant features for photometric redshift estimation.

metric observation. We take the 4 magnitude differences between adjacent bands and the magnitude in the red band for *model* and *cmodel*. Thus, we arrive at $2 \times (4 + 1) = 10$ variables for each galaxy. The target variable *redshift* is the ground-truth as determined by spectroscopic observation.



(a) (b)

Figure 1: An example from the *Sloan Digital Sky Survey* (SDSS) [Aihara et al., 2011]. (a) An image of the spiral galaxy NGC 5750. (b) Its associated spectrum overlapping the five photometric intensity band filters *u,g,r,i,z*.

The dataset contains a sample of 5000 galaxies whose redshift has been determined by spectroscopy. It is split in training and test set, each consisting of 2500 galaxies, which are described by the features listed in Table 1.

Question 1 (linear regression)**.** The goal of our modeling is to find a mapping $f : \mathbb{R}^{10} \to \mathbb{R}$ for predicting the redshift.

Build an affine linear model of the data using linear regression and the training data in `redshiftTrain.csv` only. Report the parameters of the model (do not forget the offset/bias parameter).

Determine the training error by computing the mean-squared-error of the model over the complete *training* data set. Compute the mean-squared-error on the test data set `redshiftTest.csv`. Comment very briefly on the result.

Deliverables: Description of software used; parameters of the regression model; mean-squared error on the training and test data set; short discussion of results

Question 2 (non-linear regression)**.** The question arises if one can improve the results using a non-linear model. Can you?

This is actually not easy. Try to find an algorithm that fits a non-linear regression function to the data. Use the same split in training and test data as above, and use the mean-squared error to evaluate your model.

Either

- describe an approach that performs better than the linear model; *or*
- present the results of two non-linear regression methods covered in the lectures and argue why you think that you used them correctly.

In any case, describe the measures you have taken to ensure good generalization.

Deliverables: Description of software used; mean-squared error on the training and test data set; detailed description of approaches; describe what you did to achieve good generalization performance; discussion of results

Cybercrime Detection

According to the PWC "2011 Global Economics Crime Survey", cybercrime ranks among the top four economic crimes, not only leading to financial loss, but also bearing the risk of reputational damage for financial institutions and other online service providers.

In this part of the assignment, you are supposed to develop a system for detecting fraudulent logins. The idea is to identify a criminal who has stolen a password by the way s-/he types the password. That is, the goal is to identify the user that produced a given keyboard input based on the keystroke timings. We consider the dataset from Killourhy and Maxion [2009], which consists of 400 keystroke samples from 51 subjects, all typing the same strong password "`.tie5Roanl`". The keystrokes are collected into a vector of flight and dwell times that contains 21 features [Moskovitch et al., 2009]. This assignment is divided into two parts,

a binary and a multi-class classification task. Both tasks consist on separating samples of different users.

Two classes

First, we consider the data in the files `keystrokesTrainTwoClasses.csv` and `keystrokesTestTwoClasses.csv`, containing the training and test data, respectively. Each row corresponds to a sample, with 21 features plus the label in the last column. All features are floating point numbers and the labels are integers.

Question 3 (binary classification)**.** The task is to perform binary classification using a linear and a non-linear method. Only use the training data in `keystrokesTrainTwoClasses.csv` in the model building process. Use grid-search for model selection.

In this application, it is reasonable to distinguish between *sensitivity* (also called true positive rate) and *specificity* (also called true negative rate). Given a data set, the sensitivity is computed as the number of patterns correctly classified as belonging to the positive class divided by the number of patterns in the data set that belong to the positive class. Specificity is the number of patterns correctly classified as belonging to the negative class divided by the number of patterns in the data set that belong to the negative class. Do a literature study to understand what specificity and sensitivity mean. Compute classification accuracy (one minus classification error), specificity, and sensitivity of your model on the training and test set `keystrokesTestTwoClasses.csv`. Assume that a class label "0" indicates the negative class (an imposter typing a stolen password) and a class label "1" indicates the positive class (the righteous user, who correctly types his password). Briefly explain what specificity and sensitivity mean in this application scenario.

Deliverables: Description of software used; a short description of the model selection process; classification accuracy on training and test data; specificity and sensitivity on training and test data; explanation of specificity and sensitivity in the context of the application; discussion of the results

Question 4 (principal component analysis)**.** In this exercise, we look more closely at the two users in `keystrokesTrainTwoClasses.csv`. Perform a principal component analysis of the input patterns. Plot the eigenspectrum (see Figure 12.4 by Bishop [2006] for an example). Visualize the data by a scatter plot of the data projected on the first two principal components.

Deliverables: Description of software used; plot of the eigenspectrum; scatter plot of the data projected on the first two principal components

Question 5 (clustering)**.** Perform 2-means clustering of the input patterns in `keystrokesTrainTwoClasses.csv` and report the 21-dimensional cluster centers

(i.e., the centroids). *After that*, project the cluster centers to the first two principal components of the training data. Then visualize the clusters by adding the cluster centers to the plot from the previous exercise 4. Briefly discuss the results: Did you get meaningful clusters?

Deliverables: Description of software used; cluster centers; one plot with cluster centers and data points; short discussion of results

Multiple classes

Now we look at several users typing the same password. Use the data in the files `keystrokesTrainMulti.csv` and `keystrokesTestMulti.csv` containing the training and test samples, respectively. Again, each line are 21 features plus the corresponding label in the last column.

Question 6 (multi-class classification)**.** Use a linear and a non-linear classification method (picking from the methods presented in the course) for classifying the classes, for example linear discriminant analysis (LDA) and k-nearest neighbor. Describe how you performed model selection. Only use the training data in `keystrokesTrainMulti.csv` in the model building process.

After you trained a model, use the test data in `keystrokesTestMulti.csv` to evaluate it. Report the classification error on both training and test set.

Deliverables: Description of software used; arguments for your choice of classification methods; a short description of how you proceeded and what training and test results you achieved

General questions

Question 7 (overfitting)**.** John Langford, who is "Doctor of Learning at Microsoft Research", maintains a very interesting blog (web log). Read the very true blog entry: "Clever methods of overfitting," `http://hunch.net/?p=22`, 2005, reposted at:

`http://www.kdnuggets.com/2015/01/clever-methods-overfitting-avoid.html`

Choose three of the different types of overfitting and discuss if and how they can occur when applying machine learning techniques to the photometric redshift estimation task. Ignore the last type of overfitting and issues related to reviewing of scientific papers (still, it is good to keep them in mind).

Deliverables: Short discussion addressing three "methods of overfitting" listed in the blog entry

Acknowledgment

Funding for the SDSS and SDSS-II has been provided by the Alfred P. Sloan Foundation, the Participating Institutions, the National Science Foundation, the U.S. Department of Energy, the National Aeronautics and Space Administration, the Japanese Monbukagakusho, the Max Planck Society, and the Higher Education Funding Council for England. The SDSS Web Site is `http://www.sdss.org/`.

The SDSS is managed by the Astrophysical Research Consortium for the Participating Institutions. The Participating Institutions are the American Museum of Natural History, Astrophysical Institute Potsdam, University of Basel, University of Cambridge, Case Western Reserve University, University of Chicago, Drexel University, Fermilab, the Institute for Advanced Study, the Japan Participation Group, Johns Hopkins University, the Joint Institute for Nuclear Astrophysics, the Kavli Institute for Particle Astrophysics and Cosmology, the Korean Scientist Group, the Chinese Academy of Sciences (LAMOST), Los Alamos National Laboratory, the Max-Planck-Institute for Astronomy (MPIA), the Max-Planck-Institute for Astrophysics (MPA), New Mexico State University, Ohio State University, University of Pittsburgh, University of Portsmouth, Princeton University, the United States Naval Observatory, and the University of Washington.

References

H. Aihara, C. A. Prieto, D. An, S. F. Anderson, É. Aubourg, E. Balbinot, T. C. Beers, A. A. Berlind, S. J. Bickerton, D. Bizyaev, et al. The eighth data release of the Sloan digital sky survey: first data from SDSS-III. *The Astrophysical Journal Supplement Series*, 193(2):29, 2011.

C. M. Bishop. *Pattern Recognition and Machine Learning*. Springer, 2006.

K. S. Killourhy and R. A. Maxion. Comparing anomaly-detection algorithms for keystroke dynamics. In *Dependable Systems & Networks, 2009. DSN'09. IEEE/IFIP International Conference on*, pages 125–134. IEEE, 2009.

R. Moskovitch, C. Feher, A. Messerman, N. Kirschnick, T. Mustafic, A. Camtepe, B. Lohlein, U. Heister, S. Moller, L. Rokach, et al. Identity theft, computers and behavioral biometrics. In *IEEE International Conference on Intelligence and Security Informatics (ISI'09)*, pages 155–160. IEEE, 2009.

E. S. Sheldon, C. E. Cunha, R. Mandelbaum, J. Brinkmann, and B. A. Weaver. Photometric redshift probability distributions for galaxies in the SDSS DR8. *The Astrophysical Journal Supplement Series*, 201(2):32, 2012.