Master M2 MSIAM and MoSIG

Fundamentals of Probabilistic Data Mining

Graded lab and homeworks

`http://chamilo.grenoble-inp.fr/courses/ENSIMAGWMM9AM21/`

---

1 General rules

This work is to be achieved by groups of four students. The teams have to be created in `https://teide.ensimag.fr/` (you will receive an automatic invitation to create groups). Depending on the number of students, the last three registered teams may contain three students and then be created manually by the teacher.

The whole work is divided into three topics: Probabilistic graphical models, Independent mixture models and Hidden Markov models. Each topic involves:

- lab work, which may be achieved in a supervised 1h30 session
- mandatory unsupervised lab work
- optional research-like work, which must be addressed to obtain grades above 15/20. Will not be considered if some mandatory questions have not been addressed.

A .zip archive has to be uploaded on `https://teide.ensimag.fr/`, which should contain a report, potential source code and a table with the respective contributions of all authors to topics or questions (download the template on Chamilo).

The report must contain your results, figures (do not forget legends!), comments, conclusions and references.

Every protocol for analysis, simulation and estimation has to be fully described. The description of estimation procedures should include, but not be limited to, the name of the estimation algorithm, the stopping criterion, the number of iterations and the choice of the initial parameter value. Every result has to be included in the report with some comments and analysis.

2 Probabilistic graphical models

2.1 Lab work

This part aims at comparing two procedures to estimate multivariate Gaussian directed probabilistic graphical models (PGMs). It relies on the `R bnlearn` package for structural inference.

Run `R` and test whether the `bnlearn` package is installed by trying the command `library(bnlearn)`. If it fails, run `install.packages("bnlearn")`. Similarly, test whether the `Rgraphviz` package is installed by `library(Rgraphviz)`. If it fails, run `source("https://bioconductor.org/biocLite.R")` and then `biocLite("Rgraphviz")`.

2.1.1 Simulated data

Firstly, simulate a Gaussian model with the perfect map in Figure 2.1.1. To do this, use linear regression models using offsets 0, the coefficients in the figure, and the residual standard deviation $\sigma = 1$. Simulate one sample of size 40 and one sample of size 100.

Compare the `gs` and `hc` procedures (Scutari, 2010) using both sample sizes. What is your conclusion?

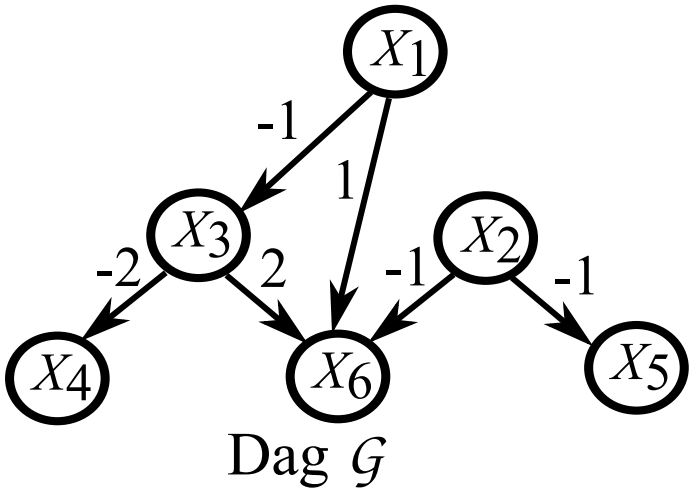

Figure 1: Simulated model

2.1.2 Real data: asset returns

We consider the returns of 8 assets on $n = 5,039$ days. The daily return $X_{t,i}$ of asset i at time t is defined as $(V_{t,i} - V_{t-1,i})/V_{t-1,i}$, where $V_{t,i}$ is the value of asset i at time t.

Here, we consider the assets "AIR.FRANCE.KLM", "ALCATEL.LUCENT" "AXA", "FAURECIA", "GAUMONT", "GEODIS", "PPR" and "UNION.FINC.FRANC." only.

1. Use file "Returns250d.txt" to create a data frame with only the 8 assets listed above.

2. Estimate directed graphs using the `gs` and `hc` procedures (Scutari, 2010) and plot their graphs.

3. Find a marginal independence relationship between two variables found by `gs` but not by `hc`. Use `ci.test` to perform a statistical test of independence. What do you conclude?

4. Find a conditional independence relationship between two variables given another set of variables found by `hc` but not by `gs`. Use `ci.test` to perform a statistical test of (conditional) independence. What do you conclude?

5. Find a conditional independence relationship between two variables given another set of variables found by both `hc` and `gs`. Use `ci.test` to perform a statistical test of (conditional) independence. What do you conclude?

2.2 Mandatory additional questions

We address the issue of consistent directed PGM estimation.

1. Give a formal definition of consistent directed PGM estimation. Write some state-of-the-art on that topic, choose one of the references therein, justifying your choice. Provide a one-page description of the approach developed in that reference.

2. Apply the chosen approach to the asset returns data set. Compare it with the results of `hc` in part 2.1.2. Use `ci.test` to try to evaluate the relevance of proposals for edges where both methods yield different results.

2.3 Optional additional questions

1. Imagine, describe and implement a protocol to evaluate consistency of any arbitrary directed PGM estimation method. Test this protocol on the method chosen in part 2.2 and provide the result. What is your conclusion? What is the effect of the number of variables?

2. What are the assumption of the method chosen in part 2.2? Check as many assumptions as you can on the asset returns data set.

3. Define and simulate some 4-variable model that cannot have a perfect directed map. Estimate a directed PGM using the method chosen in part 2.2. What do you obtain? Why? (How to interpret this result?)

4. If two methods claim to be consistent but yield different DAGs on a same data set, how would you used both graph to propose a new graph? What would the expected properties of that graph be?

3 Independent mixture models

This is a preparatory work for topic 4: Hidden Markov models, which is related to gesture recognition.

The Unistroke alphabet, closely related to Graffiti[1], is an essentially single-stroke shorthand handwriting recognition system used in PDAs. The data set is composed of 50×6 time-trajectories representing the drawing of letters A, E, H, L, O and Q in a plane.

[1] `http://en.wikipedia.org/wiki/Graffiti_(Palm_OS)`

Here you will focus on modelling letter A (actually drawn as a Λ), you may ignore the other five letters if you want. After some pre-processing, we obtain the data set "Amerge.txt", which is composed of every stroke of every trial for the gestures associated with that letter (the temporal aspect of sequences and the separations between sequences were lost here).

3.1 Lab work

3.1.1 Modelling

A priori (before reading the data), do you think a two-state Gaussian model could be appropriate? Why?

3.1.2 Data analysis: Gaussian model

1. Plot the data set. Estimate a bivariate Gaussian mixture model.
2. Label the data using the estimated model.
3. Propose and implement a graphical (visual) method to validate the assumption of bivariate Gaussian emission distributions. What to think about this assumption?
4. Define von Mises and mixtures of von Mises distributions.
5. A priori, would a mixture of von Mises distributions be more or less adequate than Gaussian mixtures on the real data set of part 3.1? Why?

3.2 Mandatory additional questions

The aim of this part is to compare mixture of von Mises distributions with Gaussian mixtures.

1. Extend the scikit-learn mixture library by implementing mixtures of von Mises distributions. Justify the E-step of the EM algorithm with equations.
2. What is a consistent estimator of mixture parameters? Imagine, describe and implement a protocol to evaluate consistency of any arbitrary estimator. Test this protocol on the algorithm developed in the previous question.
3. Use a 2-state mixture of von Mises distributions on the real data set of part 3.1. Transform the data to angular data. Provide a quantitative and a graphical way to compare the fitted mixture of von Mises distributions with the Gaussian mixture obtained in part 3.1.

3.3 Optional additional questions

1. Give a formal definition of consistent estimator of the number of states in a mixture model. Write some state-of-the-art on that topic, choose one of the references therein, justifying your choice. Provide a one-page description of the approach developed in that reference.

2. Imagine, describe and implement a protocol to evaluate consistency of any arbitrary estimator of the number of states. Test this protocol on mixtures of your choice, among mixtures of von Mises distributions and Gaussian mixtures.

3. Apply the chosen approach to estimate the number of states in the real data set of part 3.1 (with mixture of von Mises distributions)? Does it yield the expected number of states?

4 Hidden Markov models

The Unistroke alphabet, closely related to Graffiti[2], is an essentially single-stroke shorthand handwriting recognition system used in PDAs. The data set is composed of 50×6 time-trajectories representing the drawing of letters A, E, H, L, O and Q in a plane.

4.1 Lab work

Here you will focus on discriminating letters A and L, you may ignore the other four letters if you want.

1. Define an HMC model (number of states, every parameter, ...) and simulate some trajectories, until it resembles letter Λ up to some Gaussian noise.
   Hint: you may use a differential representation of the signal. If you want to output $x_1, \dots, x_n$, simulate $y_1 = x_1 - 0$, $y_2 = x_2 - x_1, \dots, y_2 = x_n - x_{n-1}$.

2. Develop some procedure to plot a real trajectory $x_1^n = (x_1, \dots, x_n) \in (\mathbb{R}^2)^n$. Connect successive points with segments, but do not care about potential loss of temporal information in the produced figures. The trajectory will be a reverse letter but you may ignore this too. Include such figure in your report.

3. Normalize the trajectories by computing $y_t = \dfrac{x_t - x_{t-1}}{\|x_t - x_{t-1}\|_2}$ for $t = 2, \dots, n$. Estimate an HMC model with all the normalized trajectories for letter A, using bivariate Gaussian emission distribution. Justify your choice for the number of states. Provide the estimates and comment them.

4. Same question as above for letter L. You may use the python `hmmlearn` library[3] or another library of your choice.

5. Use the Viterbi algorithm on `A1.txt` and plot the unnormalized sequence using different colours for different states.

6. Propose and implement a graphical (visual) method to validate the assumption of bivariate Gaussian emission distribution. What to think about this assumption?

[2] `http://en.wikipedia.org/wiki/Graffiti_(Palm_OS)`
[3] `http://github.com/hmmlearn/hmmlearn`

4.2 Mandatory additional questions

The aim of this part is to compare von Mises and Gaussian emission distributions.

1. Transform the data to angular data. Implement von Mises emission distributions (including formal computations into the report) and compare the results with bivariate Gaussians.

2. Compute a 5-fold cross-validated classification error with both families of emission distributions (for the moment, only consider classes A and L.)

4.3 Optional additional questions

1. Give a formal definition of consistent estimators of the number of states in a HMM. Write some state-of-the-art on that topic, choose one of the references therein, justifying your choice. Provide a one-page description of the approach developed in that reference.

2. Imagine, describe and implement a protocol to evaluate consistency of any arbitrary estimator of the number of states. Test this protocol on HMM with von Mises distributions, using the estimator chosen in the previous question.

3. Apply the chosen approach to estimate the number of states in the real data set of part 4.1 (with mixture of von Mises distributions), considering now the 6 letters A, E, H, L, O and Q.

4. Compute a 5-fold cross-validated classification error using the 6 estimated models. Provide the parameter estimates for each of the 6 models. Provide some confusion matrix using the selected models and comment the errors.

References

[1] Scutari, M. Learning Bayesian Networks with the bnlearn R Package. *Journal of Statistical Software* **31**(3), 1–22, 2010.