Mini-Project Overview

Lecture: Prof. Aude Billard (aude.billard@epfl.ch)

Teaching Assistants:
Nadia Figueroa,
Ilaria Lauzana, Brice Platerrier
Deadlines for projects / surveys

Sign up for lit. survey and mini-project must be done by **March 10 2017**.

Literature surveys and mini-project reports must be handed out by **May 19 2017**.

Oral presentations will take place on **May 26 2017**.

Webpage dedicated to mini-projects:
[http://lasa.epfl.ch/teaching/lectures/ML_MSc_Advanced/miniprojects.html](http://lasa.epfl.ch/teaching/lectures/ML_MSc_Advanced/miniprojects.html)
Topics for literature surveys

Here is a list of proposed topics for survey / review papers:

- Methods for learning the kernels
- Methods for active learning
- Data mining methods for crawling mailboxes
- Data mining methods for crawling git-hub
- Classification methods for spam/no-spam
- Pros and cons of crowdsourcing
- Recent trends and open problems in speech recognition
- Ethical issues on data mining

Sign up on doodle for the project with your team partner!

Instructions:
Survey of the literature / review papers must be written by teams of two people. The document should be 8 pages long double column format, see example on mini-project webpage.

Caveats: Do not paraphrase the papers you read, i.e. avoid saying “Andrew et al did A. Suzie et al. did B, etc.” but make a synthesis of what the field is about. While you may read up to 100 papers total, but you should report on those that are most relevant.
Topics for Mini-Projects

Topics for mini-project will entail implementing either of these:

• Manifold learning/Non-linear Dimensionality Reduction
  • Isomap and Laplacian Eigenmaps
  • LLE and variants
  • SNE and variant
• Non-linear Regression
  • Relevance Vector Machine
  • Non-Parametric Approximations Techniques for Mixture Models
Mini-Projects Requirements

Coding:
Self-contained piece of code in:
• Matlab
• Python
• C/C++

Including:
• Demo scripts
• Datasets
• Systematic assessment.

Report:
Algorithm analysis, including but not limited to:
• Number/sensitivity to hyper-parameters
• Computational costs train/test
• Growth of computation cost wrt. dataset dimension
• Sensitivity to non-uniformity/non-convexity in data.
• Precision of regression
• Benefits/disadvantages of algorithm wrt. to different types of data/applications.
• …

<table>
<thead>
<tr>
<th>Project Name</th>
<th>Coding</th>
<th>Analysis</th>
</tr>
</thead>
<tbody>
<tr>
<td>Isomap and Laplacian Eigenmaps</td>
<td>20%</td>
<td>80%</td>
</tr>
<tr>
<td>LLE, MLLE and HLLE</td>
<td>30%</td>
<td>70%</td>
</tr>
<tr>
<td>SNE and t-SNE</td>
<td>30%</td>
<td>70%</td>
</tr>
<tr>
<td>RVR vs SVR</td>
<td>30%</td>
<td>70%</td>
</tr>
<tr>
<td>GMM vs DP-GMM for GMR</td>
<td>50%</td>
<td>50%</td>
</tr>
</tbody>
</table>
## Useful ML Toolboxes

### Matlab

<table>
<thead>
<tr>
<th>Toolbox</th>
<th>URL</th>
</tr>
</thead>
<tbody>
<tr>
<td>Matlab Toolbox for Dimensionality Reduction</td>
<td><img src="https://lvdmaaten.github.io/drtoolbox/" alt="URL" /></td>
</tr>
<tr>
<td>Least Squares - Support Vector Machine</td>
<td><img src="http://www.esat.kuleuven.be/sista/lssvmlab/" alt="URL" /></td>
</tr>
<tr>
<td>LIBSVM</td>
<td><img src="www.csie.ntu.edu.tw/~cjlin/libsvm/" alt="URL" /></td>
</tr>
<tr>
<td>GMM/GMR v2.0</td>
<td><img src="http://lasa.epfl.ch/sourcecode/?showComments=14#GMM" alt="URL" /></td>
</tr>
<tr>
<td>Probabilistic Modeling Toolkit for Matlab/Octave</td>
<td><img src="https://github.com/probml/pmtk3" alt="URL" /></td>
</tr>
<tr>
<td>Gaussian Dirichlet Process Mixture Models (DPMMs)</td>
<td><img src="https://github.com/jacobelstein/DPMM" alt="URL" /></td>
</tr>
<tr>
<td>Dirichlet Process Mixture Modeling</td>
<td><img src="http://www.gatsby.ucl.ac.uk/~fwood/code.html" alt="URL" /></td>
</tr>
</tbody>
</table>

### Python

<table>
<thead>
<tr>
<th>Toolbox</th>
<th>URL</th>
</tr>
</thead>
<tbody>
<tr>
<td>scikit-learn. Machine Learning in Python</td>
<td><img src="http://scikit-learn.org/stable/" alt="URL" /></td>
</tr>
<tr>
<td>bnp. Bayesian NonParametric Machine Learning for Python</td>
<td><img src="https://bitbucket.org/michaelchughes/bnp/%C2%AE" alt="URL" /></td>
</tr>
</tbody>
</table>
Topics for Mini-Projects

Topics for mini-project will entail implementing either of these:

- Manifold learning/Non-linear Dimensionality Reduction
  - Isomap and Laplacian Eigenmaps
  - LLE and variants
  - SNE and variant
- Non-linear Regression
  - Relevance Vector Machine
  - Non-Parametric Approximations Techniques for Mixture Models
Isomaps and Laplacian Eigenmaps

- **ISOMAP (Isometric Mapping)**: Can be viewed as an extension of multi-dimensional Scaling or Kernel PCA, as it seeks a lower-dimensional embedding which maintains geodesic distances between all points.

- **LAPLACIAN EIGENMAPS (also known as Spectral Embedding)**: It finds a low dimensional representation of the data using a spectral decomposition of the graph Laplacian. The graph generated can be considered as a discrete approximation of the low dimensional manifold in the high dimensional space.
Locally Linear Embedding (LLE) and its Modified (MLLE) and Hessian (HLLE) variants

- **LLE**: LLE seeks a lower-dimensional projection of the data which preserves distances within local neighborhoods. It can be thought of as a series of local PCA which are globally compared to find the best non-linear embedding.

- **MLLE**: Solves the regularization problem of LLE by using multiple weight vectors in each neighborhood.

- **HLLE**: Solves the regularization problem of LLE by using a hessian-based quadratic form in each neighborhood.
Stochastic Neighbor Embedding (SNE) ans its t-distributed (t-SNE) variant

• **SNE** : First, SNE constructs a Gaussian distribution over pairs of high-dimensional objects. Second, SNE defines a similar probability distribution over the points in the low-dimensional map, and it minimizes the Kullback–Leibler divergence (using gradient descent) between the two distributions with respect to the locations of the points in the map.

• **t-SNE** : A variant of SNE, which represents the similarities in the high-dimensional space by Gaussian joint probabilities and the similarities in the embedded space by Student's t-distributions, making it more sensitive to local structure.
Comparison aspects

- Preservation of the geometry
- Handling holes in a dataset (non-convexity)
- Behaviour with high-curvature
- Behaviour with non-uniform sampling
- Preservation of clusters
- Algorithmic/theoretical differences
- Usefullness for different types of datasets
Toolboxes

- Matlab Toolbox:
  - *Matlab Toolbox for Dimensionality Reduction*

- Python Library:
  - *Sci-kit learn for Python*
Perspectives of comparison

• In addition to answering the general assessment questions for these topics the team should generate or test high-dimensional datasets.

• Apply standard clustering or classification algorithms of their choosing and evaluate their performance with F-measure, BIC, AIC, Precision, Recall, etc.
Repositories for High-Dimensional Real-World Datasets

UCI Machine Learning Repository:
http://archive.ics.uci.edu/ml/

Kaggle:
https://www.kaggle.com/datasets
Topics for Mini-Projects

Topics for mini-project will entail implementing either of these:

- Manifold learning/Non-linear Dimensionality Reduction
  - Isomap and Laplacian Eigenmaps
  - LLE and variants
  - SNE and variant
- Non-linear Regression
  - Relevance Vector Machine
  - Non-Parametric Approximations Techniques for Mixture Models
RVR vs SVR

• **Relevance Vector Machine (RVM)** is a machine learning technique that uses Bayesian inference to obtain solutions for probabilistic regression and classification.

• The RVM applies the Bayesian 'Automatic Relevance Determination' (ARD) methodology to linear kernel models, which have a very similar formulation to the SVM, hence, it is considered as sparse SVM.

*Sparse Bayesian learning and the relevance vector machine ; Tipping, M. E. ; Journal of Machine Learning Research 1, 211-244 (2001)*
Perspectives of comparison for different datasets

- Computational cost for training and testing
- Precision of the regression
- Evolution with the size of the dataset
- Memory cost
- Choice of hyper-parameters
- Choice of Kernel
- …
Toolboxes

- Support Vector Machine for regression in :
  - *The Statistics and Machine Learning Toolbox* of *Matlab*
  - *Scikit-learn* for *Python*
  - *LibSVM* for *C++/MATLAB*

- Relevance Vector Machine for regression in :
  - *Matlab SparseBayes*
  - *sklearn_bayes* for *Python*
GMM vs DP-GMM for Regression

• **Gaussian Mixture Model (GMM)**: Parametric approach to learn GMM consists in fitting several models with parametrizations via the EM algorithm and use model selection approaches, like Bayesian Information Criterion, to find the best model.

• **Dirichlet Process – GMM**: DP is a stochastic process which produces a probability distribution whose domain is itself a probability distribution. It enables to add a prior on the number of models in the mixture. Variational and Sampling-based inference approaches are used to approximate the optimal parameters.
Perspectives of comparison

- Computational cost for training
- Advantage of automatic determination of parameter vs cross-validation
- Sensitivity to hyper-parameters
Toolboxes

• GMM for regression in:
  – GMM/GMR v2.0 for Matlab
  – ML_Toolbox for Matlab
  – Scikit-learn for Python

• DP-GMM in:
  – Dirichlet Process – Gaussian Mixture Models for Matlab
  – bnpy for Python
Examples of Self-Contained Code

Follow examples in Sci-kit Learn package:
http://scikit-learn.org/stable/auto_examples/

– Ideal Classification Comparison Example:
Code Submission/Organization

My ML Mini-Project
- Datasets
- Figures
- My Functions
- 3rd Party Toolboxes
demo_script.m
comparison_script.m
highd_results_scripts.m
README.txt

Submit! (Moodle)
- My_ML_MiniProject.zip
- My_ML_MiniProject.pdf
Examples of Well-Documented Code

Matlab/C++ package for SVM + Derivative Evaluation:
https://github.com/nbfigueroa/SVMGrad

Python/C++ package for Locally Weighted Regression:
https://github.com/gpldecha/non-parametric-regression