MACHINE LEARNING

Overview
Exam Format

The exam lasts a total of 3 hours:

- Upon entering the room, you must leave your bag, cell phone, etc., in a corner of the room; you are allowed to keep a couple of pen/pencil/eraser and a few blank sheets of paper.

- The exam will be graded anonymously; make sure to have your camipro card with you to write your sciper number on your exam sheet. Present your card on the table, as we will check your card.

Exam is closed book but you can bring one A4 page with personal notes written recto-verso.
What to know for the exam

Formalism / Taxonomy:

• Be capable of giving formal definitions of a pdf, marginal, likelihood

• Know the difference between clustering, classification, regression and be able to give examples of algorithms in each case

Principles of evaluation:

• Know the basic principles of evaluation of ML techniques: training vs. testing sets, cross-validation, ground truth.

• Be able to discuss the notion of over-fitting and curse of dimensionality

• Know the principle of each method of evaluation seen in class and know which method of evaluation to apply where (F-measure in clustering vs. classification, BIC, etc).
What to know for the exam

• For each algorithm, be able to explain:

  – what it can do: classification, regression, structure discovery / reduction of dimensionality

  – what one should be careful about (limitations of the algorithm, choice of hyperparameters) and how does this choice influence the results.

  – the key steps of the algorithm, its hyperparameters, the variables it takes as input and the variables it outputs
What to know for the exam

• For each algorithm, be able to explain:
  
  SVM
  – what it can do: classification, regression, structure discovery / reduction of dimensionality
  Performs *binary* classification; can be extended to multi-class classification; can be extended to regression (SVR)

  – what one should be careful about (limitations of the algorithm, choice of hyperparameters)
  e.g. choice of kernel; too small kernel width in Gaussian kernels may lead to over-fitting;

  – the key steps of the algorithm, its hyperparameters, the variables it takes as input and the variables it outputs
This overview is meant to highlight similarities and differences across the different methods presented in class.

To be well prepared to the exam, read carefully the slides and the exercises.
Class Overview

This class has presented groups of methods for structure discovery, classification and non-linear regression.

Structure Discovery
- PCA

Clustering Techniques
- K-Means,
- Soft K-means,
- GMM

Classification
- SVM,
- GMM + Bayes

Regression
- SVR
- GMR
Overview: Finding Structure in Data

Techniques for finding structure in data proceed by **projecting or grouping** the data from the original **space into another space of lower dimension**.

The projected space is chosen so as to **highlight particular features** common to subsets of datapoints.

**Pre-processing step:** The found structure may be exploited in a second stage by another algorithm for regression, classification, etc.
Overview: Finding Structure in Data

Principal Component Analysis (PCA)

- Determines what is most common across datapoints.
- Projects onto axes that maximize correlation (eigenvectors of covariance matrix)
- Lower dimensions allow to discriminate across subgroups of datapoints!
- Discard dimensions with the smallest eigenvalues.

$x \in \mathbb{R}^N$

$Y = AX$

$y \in \mathbb{R}^q, \quad q \leq N$
Overview: Finding Structure in Data

Clustering Methods

Except for hierarchical clustering (not exam material), all three methods for clustering we have seen in class (K-means, soft K-means, GMM) are all solved through E-M (expectation-maximization).

You should be able to spell out the similarities and differences across K-means, soft K-means and GMM.

- They are similar in their representation of the problem and optimization method, etc.

- They differ in the number of parameters to estimate and number of hyper-parameters, etc.
Overview: Finding Structure in Data

Clustering Methods and Metric of Similarity

All clustering methods depend on choosing well a metric of similarity to measure how similar subgroup of data-points are.

You should be able to list which metric of similarity can be used in each case and how this choice may impact the clustering.

- Lp-norm in K-means
- Exponential decreasing function in soft K-means
- $\sim$ isotropic rbf (unnormalized Gauss) function
- Isotropic/diagonal & full Covariance Gauss function in GMM
Clustering versus Classification

Fundamental difference between clustering and classification:

- Clustering is *unsupervised* classification
- Classification is *supervised* classification

Both use the F-measure but not in the same way.

The clustering F-measure (F1-measure in MLDemos) assumes a semi-supervised model, in which only a subset of the points are labelled.
Overview: Classification

GMM + Bayes

1 Gauss fct per class
But full covariance matrix

Original two-classes

SVM

7 support vectors

Compare computational costs at training and testing
Computational Costs SVM vs GMM+Bayes

**Training**

**SVM**
- Convex optimization (SMO solver)
- Parameters grow $O(M \times N)$

**GMM + Bayes**
- EM, iterative technique, needs several runs
- Parameters grow $O(K \times N^2)$

**Testing**

**GMM + Bayes**
- Grows $O(K)$
  - $K << M$

**SVM**
- Grows $O(SV) \sim O(M)$
- SV are a fraction of number of datapoints

M: number of datapoints; N: Dimension of data; K: Number of Gauss Functions in GMM model
Overview: Classification Techniques

SVM and GMM + Bayes can both perform binary classification. However, they differ in their objective function.

• SVM:
  • maximizes margin across the two classes through convex optimization
  • finds datapoints (support vectors) that are most similar in each class (the closest to the other class) and use these to increase the discriminative power of SVM.

• GMM:
  • is not meant for classification – one trains a GMM for each class separately (it has no information about the other class during training!) and then uses Bayes to discriminate the likelihood that a new datapoints belong to either model;
  • places Gaussians so as to represent best all members of the class (and not those that are closest to the other class)
Overview: Regression Techniques

SVR and GMR lead to a regressive model that computes a weighted combination of local predictors.

For a query point $x$, predict the associated output $y$:

**SVR Solution**

$$y = \sum_{i=1}^{M} (\alpha_i^* - \alpha_i) k(x^i, x) + b$$

**GMR Solution**

$$y = \sum_{i=1}^{K} \beta_i(x) \cdot \tilde{u}^i(x)$$

In SVR, the computation is reduced to summing only over the support vectors (a subset of datapoints).

In GMR, the sum is over the set of Gaussians. The centers of the Gaussians are usually not located on any particular datapoint. The models are local $\mu(x)$!
Overview: Regression Techniques

SVR and GMR lead to the following regressive model:

8 Gauss functions full covariance matrix

$$y = \sum_{i=1}^{K} \beta_i(x) \cdot \tilde{u}^i(x)$$
Overview: Regression Techniques

SVR and GMR lead to the following regressive model:

$$y = \sum_{i=1}^{M} (\alpha_i^* - \alpha_i) k(x^i, x) + b$$

SVR Solution

27 support vectors
Overview: Regression Techniques

SVR and GMR are based on the same probabilistic regressive model, but do not optimize the same objective function.

- SVR:
  - minimizes reconstruction error through convex optimization
  - finds a number of models ≤ number of datapoints (support vectors)

- GMR:
  - learns $p(x,y)$ through likelihood maximization and then compute $p(y|x)$
  - starts with a low number of models << number of datapoints
Comparison Computational Costs GMR vs SVR

Training

SVR
Convex optimization (SMO solver)
Parameters grow O(M*N)

GMR
EM, iterative technique, needs several runs
Parameters grow O(K*N²)

Testing

GMR
Grows O(K)
K<<M

SVR
Grows O(SV)~O(M)
SV are a fraction of number of datapoints

M: number of datapoints; N: Dimension of data; K: Number of Gauss Functions in GMM model
Conclusion

This class was only a brief overview of main classes of ML algorithms.

Today, ML encompasses hundreds of algorithms.

Check resources given at beginning for latest developments and software.

Merry Christmas

&

Happy New Year