Lecture 1: Introduction

Isabelle Guyon
guyoni@inf.ethz.ch

Class Organization

http://clopinet.com/isabelle/Projects/ETH/
### Class Schedule

- **Thursday 10:00AM-12:00AM, CAB G 59:** Lecture.
- **Thursday 12:00AM-13:00PM, CAB G 59:** Exercises.
- **Tuesday 10:00AM-12:00AM, CAB G 82.2:** Office hours.
- **Tuesday afternoon:** turn in the homework (homework turned in late will not be checked). Teams of 2 permitted.

### Class Organization

- **Every week:** list of questions.
- **Every week:** homework.
- **Alternating:**
  - lectures by the instructors (I. Guyon primarily and A. Eliseeff guest instructor)
  - seminars given by the students on selected papers.
**Textbook**


login: fextract
password: ws0506

---

**Requirements and Grading**

The class is worth 5 units:

- Submit one valid entry in the feature selection challenge [http://www.nipsfsc.ecs.soton.ac.uk/](http://www.nipsfsc.ecs.soton.ac.uk/) meeting some criteria of quality.
- Present a seminar on one of the papers proposed.
- Final oral exam *(Monday, February 27)*: Present a poster with your results. Questions.

Teams of 2 permitted.
Choose your paper

- Look at: http://clopinet.com/isabelle/Projects/ETH/
- Email to: guyoni@inf.ethz.ch
- First come first serve!

Course Overview

- Fundamentals:
  - Learning machines
  - Applied statistics
  - Signal/image processing and filtering.
- Feature extraction: Feature extraction = feature construction + feature selection
- Applications:
  - Biology and medicine
  - Text and image processing.
Homework Overview

• Make entries in the feature selection challenge (5 datasets.)

• In the process, learn how to:
  - write a proposal
  - write a conference paper or a report
  - write claims for a patent
  - write a paper review
  - make a presentation (of a paper of the book)
  - make a poster
  - present a poster.

Feature Extraction Applications
Feature Extraction

Feature extraction = feature construction + feature selection

- Methods for training learning machines with millions of low level features.
- Identifying relevant features leads to better, faster, and easier to understand learning machines.

Applications
**Leukemia Diagnosis**

38 training ex. (27 ALL, 11 AML)
34 test ex. (20 ALL, 14 AML).

Golub et al, Science
Vol 286:15 Oct. 1999

\[
\{y_k\}, \ k=1\ldots\text{num}
\]
\[
\{x_k\}, \ k=1\ldots\text{num}
\]

Top 25 positively correlated features (genes)

Top 25 negatively correlated features (genes)

**Prostate Cancer Genes**

Elisseeff-Weston, 2001
- EVMS prostate cancer data: 326 samples (167 cancer, 159 control).
- Preprocessing including m/z 200-10000, baseline removal.
- Split in 3 equal parts and make 3 experiments 2/3 train 1/3 test.
- Non-linear feature selection methods win: 5% error with 100 features, 8% with 7 features.

Instrument characterization and data quality control
In collaboration with Predicant Biosciences
**QSAR: Drug Screening**

**Binding to Thrombin (DuPont Pharmaceuticals)**

- 2543 compounds tested for their ability to bind to a target site on thrombin, a key receptor in blood clotting; 192 “active” (bind well); the rest “inactive”. Training set (1909 compounds) more depleted in active compounds.

- 139,351 binary features, which describe three-dimensional properties of the molecule.

*Weston et al, Bioinformatics, 2002*

---

**Text Filtering**

**Reuters**: 21578 news wire, 114 semantic categories.

**20 newsgroups**: 19997 articles, 20 categories.

**WebKB**: 8282 web pages, 7 categories.

**Bag-of-words**: >100000 features.

Top 3 words of some categories:

- **Alt.atheism**: atheism, atheists, morality
- **Comp.graphics**: image, jpeg, graphics
- **Sci.space**: space, nasa, orbit
- **Soc.religion.christian**: god, church, sin
- **Talk.politics.mideast**: israel, armenian, turkish
- **Talk.religion.misc**: jesus, god, jehovah

*Bekkerman et al, JMLR, 2003*
Face Recognition

- Male/female classification
- 1450 images (1000 train, 450 test), 5100 features (images 60x85 pixels)

Relief:

Simba:

Navot-Bachrach-Tishby, ICML 2004

Pattern Specific features
**Risk Minimization**

- **Learning problem**: find the best function \( f(x; \alpha) \) minimizing the risk functional

\[
R[f] = \int L(f(x; \alpha), y) \, dP(x, y)
\]

- **Examples are given**: 
  \((x_1, y_1), (x_2, y_2), \ldots, (x_m, y_m)\)
**Loss Functions**

\[ L(y, f(x)) \]

- Decision boundary
- Margin
- Well classified
- Missclassified

\[ z = y f(x) \]

- 0/1 loss
- Square loss: \( (1-z)^2 \)
- SVC loss: \( \max(0, 1-z) \)
- Logistic loss: \( \log(1+e^{-z}) \)
- Adaboost loss: \( e^{-z} \)

**Approximations of \( R[f] \)**

\[ R[f] = \int L(f(x; \alpha), y) \ dP(x, y) \]

- Empirical risk: \( R_{emp}[f] = \sum_i L(f(x_i; \alpha), y_i) \)
- Guaranteed risk:
  \[ \text{Proba}( R[f] > R_{emp}[f] + \varepsilon ) < \delta \]
- Penalized/regularized risk:
  \[ R_{reg}[f] = R_{emp}[f] + \Omega[f] \]
Bayesian Decision Making

• Bayes formula: 
  \[ P(a,b) = P(a|b) P(b) = P(b|a) P(a) \]

• Bayes Optimum Classifier (BOC):
  
  Class + if \( P(y=1|x) > P(y=-1|x) \)
  
  Class - otherwise

• Equivalent formulations:
  
  \( P(x|y=1)P(y=1) > P(x|y=-1)P(y=-1) \)
  
  \( P(x, y=1) > P(x, y=-1) \)

Approximations of BOC

• Discriminant function:
  
  Class + if \( f(x) > 0 \)
  
  Class - otherwise

• Linear discriminant:
  
  \( f(x) = w \cdot x + b \)

• \( f(x) \) may approximate
  
  \( P(y=1|x) - P(y=-1|x) \) \( \rightarrow \) square loss
  
  \( \log( P(y=1|x) / P(y=-1|x) ) \) \( \rightarrow \) logistic loss
**Maximum Likelihood**

- **Likelihood**: probability of the data given the model.

\[
P( D \mid f ) = P( \{(x_i, y_i)\} \mid f )
\]

- **Maximum Likelihood (ML)**: find the model that fits best the data.

**ML = ERM**

- **ML**: 
  \[f = \text{argmax} \; P( D \mid f )\]
  \[= \text{argmin} \; -\log P( D \mid f )\]
  Negative log likelihood

\[- \log P( D \mid f ) = - \log P( \{(x_i, y_i)\} \mid f )\]

\[= \sum_i - \log P((x_i, y_i) \mid f )\]

\[= \sum_i L(f(x_i), y_i) \quad \text{loss function}\]

\[= R[f] \quad \text{Empirical risk}\]

- **ERM**: 
  \[f = \text{argmin} \; R[f]\]
Example: Logistic Loss

- Functional margin:
  \[ z = y_i f(x_i) \]
- Link function:
  \[ P((x_i, y_i) \mid f) = \psi(z) \]
- Logistic link:
  \[ \psi(z) = 1/(1+e^{-z}) \]
- Logistic loss:
  \[
  L(f(x_i), y_i) = -\log P((x_i, y_i) \mid f) \\
  = \log (1+e^{-z})
  \]

Priors and Bayesian Learning

- Double random process:
  - Draw a target function \( f \) in a family of functions \( \{f\} \)
  - Draw the data pairs \((x_i, y_i = f(x_i) + \text{noise})\)
- The distribution of \( f \) is called the “prior” \( P(f) \).
- Our revised opinion about \( f \) once we see the data is the “posterior” \( P(f|D) \).
- Bayesian “learning”:
  \[
  P(y|x,D) \propto \int P(y|x,D,f) \, dP(f|D)
  \]
- MAP:
  \[
  f = \text{argmax } P(f|D) \\
  = \text{argmax } P(D|f) \, P(f)
  \]
MAP = RRM

• Maximum A Posteriori (MAP):
  \[ f = \text{argmax} \ P(D|f) \ P(f) \]
  \[ = \text{argmin} -\log P(D|f) - \log P(f) \]

  Negative log likelihood = Empirical risk \( R[f] \)

  Negative log prior = Regularizer \( \Omega[f] \)

• Regularized Risk Minimization (RRM):
  \[ f = \text{argmin} \ R[f] + \Omega[f] \]

Example: Gaussian Prior

• Linear model:
  \[ f(x) = w \cdot x \]

• Gaussian prior:
  \[ P(f) = \exp -\frac{||w||^2}{\sigma^2} \]

• Regularizer:
  \[ \Omega[f] = -\log P(f) = \lambda \ ||w||^2 \]
**Structural Risk Minimization**

- Nested subsets of models, increasing complexity/capacity:
  \[ S_1 \subset S_2 \subset \ldots S_N \]
- Example, rank with \( ||w||^2 \)
  \[ S_k = \{ w \mid ||w||^2 < A_k \}, A_1 < A_2 < \ldots < A_k \]
- Minimization under constraint:
  \[ \min R_{\text{emp}}[f] \quad \text{s.t.} \quad ||w||^2 < A_k \]
- Lagrangian:
  \[ R_{\text{reg}}[f] = R_{\text{emp}}[f] + \lambda ||w||^2 \]
- LOO\(_{\text{SVM}}\) < \( 4 \rho^2 ||w||^2 \)

**Minimum Description Length**

- MDL: minimize the length of the “message”.
- Two part code: transmit the model and the residual.
- \( f = \arg\min -\log_2 P(D|f) - \log_2 P(f) \)

Residual: length of the shortest code to encode the data given the model

Length of the shortest code to encode the model (model complexity)
**Bias-variance tradeoff**

- For the square loss:
  \[
  E_D(f(x) - y)^2 = (E_D f(x) - y)^2 + E_D (f(x) - E_D f(x))^2
  \]

  - **Expected value of the empirical risk over datasets of the same size**
  - **Bias\(^2\)**
  - **Variance**

**Some Learning Machines**

**Next week...**
- Linear discriminant (Naïve Bayes, least square)
- Neural networks
- Kernel methods (Support Vector Machines, kernel least square)
- Tree classifiers
Practical Work

Homework 1: Data and Code

1) Download the 5 datasets of the feature selection challenge from: http://clopinet.com/isabelle/Projects/NIPS2003/ or http://www.nipsfsc.ecs.soton.ac.uk/datasets/ and put all 5 subdirectories ARCENE, DEXTER, DOROTHEA, GISETTE, and MADELON in one directory <data_dir>.

2) Download the Matlab package CLOP from http://www.modelselect.inf.ethz.ch/models.php or http://clopinet.com/isabelle/Projects/modelselect/Clop.zip We will call the directory where it ends up <code_dir>.
Homework 1: Installation

3) Windows users: nothing special.
   Linux users: build libSVM => see instructions in
   the directory
   <code_dir>/challenge_objects/packages/libsvm-mat-2.8-1.

4) Download the sample code from:
   http://clopinet.com/isabelle/Projects/ETH/homework1.zip
   Run the sample code main.m. Troubleshooting: try
   'Prepro+naiveBayes'; try 'zarbi'.

5) Create your own chain object.

Homework 1: Exercise

6) Write your own preprocessing learning object, imitating
   the examples in
   <my_root>/<code_dir>/challenge_objects/prepro.
   Suggestions:
   - sqrt, or a power law;
   - products of original features;
   - binarizing;
   - replacing the feature values by their rank.

7) Email your preprocessing learning object to:
   guyoni@inf.ethz.ch with subject "Homework1" no later
   than Tuesday November 1st.
The Datasets

- **Arcene**: cancer vs. normal with mass-spectrometry analysis of blood serum.
- **Dexter**: filter texts about corporate acquisition from Reuters collection.
- **Dorothea**: predict which compounds bind to Thrombin from KDD cup 2001.
- **Gisette**: OCR digit “4” vs. digit “9” from NIST.
- **Madelon**: artificial data.


Data Preparation

- **Preprocessing** and scaling to numerical range 0 to 999 for continuous data and 0/1 for binary data.
- **Probes**: Addition of “random” features distributed similarly to the real features.
- **Shuffling**: Randomization of the order of the patterns and the features.
- **Baseline error rates (errate)**: Training and testing on various data splits with simple methods.
- **Test set size**: Number of test examples needed using rule-of-thumb $n_{test} = 100/errate$. 

Data Statistics

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Size</th>
<th>Type</th>
<th>Features</th>
<th>Training Examples</th>
<th>Validation Examples</th>
<th>Test Examples</th>
</tr>
</thead>
<tbody>
<tr>
<td>Arcene</td>
<td>8.7 MB</td>
<td>Dense</td>
<td>10000</td>
<td>100</td>
<td>100</td>
<td>700</td>
</tr>
<tr>
<td>Gisette</td>
<td>22.5 MB</td>
<td>Dense</td>
<td>5000</td>
<td>6000</td>
<td>1000</td>
<td>6500</td>
</tr>
<tr>
<td>Dexter</td>
<td>0.9 MB</td>
<td>Sparse integer</td>
<td>20000</td>
<td>300</td>
<td>300</td>
<td>2000</td>
</tr>
<tr>
<td>Dorothea</td>
<td>4.7 MB</td>
<td>Sparse binary</td>
<td>100000</td>
<td>800</td>
<td>350</td>
<td>800</td>
</tr>
<tr>
<td>Madelon</td>
<td>2.9 MB</td>
<td>Dense</td>
<td>500</td>
<td>2000</td>
<td>600</td>
<td>1800</td>
</tr>
</tbody>
</table>

ARCENE

ARCENE is the cancer dataset

- **Sources**: National Cancer Institute (NCI) and Eastern Virginia Medical School (EVMS).
- **Three datasets**: 1 ovarian cancer, 2 prostate cancer, all preprocessed similarly.
- **Task**: Separate cancer vs. normal.
DEXTER

DEXTER filters texts

NEW YORK, October 2, 2001 - Instinet Group Incorporated (Nasdaq: INET), the world's largest electronic agency securities broker, today announced that it has completed the acquisition of ProTrader Group, LP, a provider of advanced trading technologies and electronic brokerage services primarily for retail active traders and hedge funds. The acquisition excludes ProTrader's proprietary trading business. ProTrader's 2000 annual revenues exceeded $83 million.

- **Sources**: Carnegie Group, Inc. and Reuters, Ltd.
- **Preprocessing**: Thorsten Joachims.
- **Task**: Filter “corporate acquisition” texts.

DOROTHEA

DOROTHEA is the Thrombin dataset

- **Sources**: DuPont Pharmaceuticals Research Laboratories and KDD Cup 2001.
- **Task**: Predict compounds that bind to Thrombin.
**GISETTE**

GISETTE contains handwritten digits

- **Source**: National Institute of Standards and Technologies (NIST).
- **Preprocessing**: Yann LeCun and collaborators.
- **Task**: Separate digits “4” and “9”.

**MADELON**

MADELON is random data

- **Source**: Isabelle Guyon, inspired by Simon Perkins et al.
- **Type of data**: Clusters on the summits of a hypercube.
Performance Measures

### Confusion matrix

<table>
<thead>
<tr>
<th>Truth</th>
<th>Class -1</th>
<th>Class +1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Class -1</td>
<td>a</td>
<td>b</td>
</tr>
<tr>
<td>Class +1</td>
<td>c</td>
<td>d</td>
</tr>
</tbody>
</table>

- **Balanced Error Rate (BER):** the average of the error rates for each class: \( BER = 0.5 \times (b/(a+b) + c/(c+d)) \).
- **Area Under Curve (AUC):** the area under the ROC curve obtained by plotting \( a/(a+b) \) against \( d/(c+d) \) for each confidence value, starting at \((0,1)\) and ending at \((1,0)\).
- **Fraction of Features (FF):** the ratio of the num. of features selected to the total num. of features in the dataset.
- **Fraction of Probes (FP):** the ratio of the num. of “garbage features” (probes) selected to the total num. of feat. select.

### BER distribution

- **ARCENE**
- **DEXTER**
- **DOROTHEA**
- **GISETTE**
- **MADELON**
**Power of Feature Selection**

<table>
<thead>
<tr>
<th></th>
<th>Best frac. feat</th>
<th>Actual frac. probes</th>
</tr>
</thead>
<tbody>
<tr>
<td>ARCENE</td>
<td>5%</td>
<td>30%</td>
</tr>
<tr>
<td>DEXTER</td>
<td>1.5%</td>
<td>50%</td>
</tr>
<tr>
<td>DOROTHEA</td>
<td>0.3%</td>
<td>50%</td>
</tr>
<tr>
<td>GISETTE</td>
<td>18%</td>
<td>50%</td>
</tr>
<tr>
<td>MADELON</td>
<td>1.6%</td>
<td>96%</td>
</tr>
</tbody>
</table>

**CLOP Tutorial**

- CLOP=Challenge Learning Object Package.
- Based on the Spider developed at the Max Planck Institute.
- Two basic abstractions:
  - Data object
  - Model object

http://clopinet.com/isabelle/Projects/modelselect/MFAQ.html
**CLOP Data Objects**

At the Matlab prompt:
- `cd <code_dir>`
- `use_spider_clop;`
- `X=rand(10,8);`
- `Y=[1 1 1 1 -1 -1 -1 -1 -1 -1]';`
- `D=data(X,Y); % constructor`
- `[p,n]=get_dim(D)`
- `get_x(D)`
- `get_y(D)`

**CLOP Model Objects**

D is a data object previously defined.
- `model = kridge; % constructor`
- `[resu, model] = train(model, D);`
- `resu, model.W, model.b0`
- `Yhat = D.X*model.W' + model.b0`
- `testD = data(rand(3,8), [-1 -1 1]');`
- `tresu = test(model, testD);`
- `balanced_errate(tresu.X, tresu.Y)`
**Hyperparameters and Chains**

A model often has hyperparameters:

- `default(kridge)`
- `hyper = {'degree=3', 'shrinkage=0.1'};`
- `model = kridge(hyper);`

Models can be chained:

- `model = chain({'standardize',kridge(hyper)});`
- `[resu, model] = train(model, D);`
- `tresu = test(model, testD);`
- `balanced_errate(tresu.X, tresu.Y)`

---

**This week homework...**

- Write your own preprocessing object.
- Get inspired by the modules in `challenge_objects/prepro`:
  - `standardize`
  - `normalize`
  - `shift_n_scale`
  - `pc_extract`
- Chain your module with “zarbi” to test it.