TUM 2016
Class 1
Statistical learning theory

Lorenzo Rosasco
UNIGE-MIT-IIT

July 25, 2016
Machine learning applications

Texts

<table>
<thead>
<tr>
<th>Subject</th>
<th>Date</th>
<th>Time</th>
<th>Body</th>
<th>Spam?</th>
</tr>
</thead>
<tbody>
<tr>
<td>I have the viagra</td>
<td>03/12/1992</td>
<td>12:23 pm</td>
<td>Hi! I noticed that you are a software engineer so here’s the pleasure you were looking for...</td>
<td>Yes</td>
</tr>
<tr>
<td>for you</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Important business</td>
<td>05/29/1995</td>
<td>01:24 pm</td>
<td>Give me your account number and you’ll be rich. I’m totally serial</td>
<td>Yes</td>
</tr>
<tr>
<td>Business Plan</td>
<td>05/23/1996</td>
<td>07:19 pm</td>
<td>As per our conversation, here’s the business plan for our new venture Warm regards...</td>
<td>No</td>
</tr>
<tr>
<td>Job Opportunity</td>
<td>02/29/1998</td>
<td>08:19 am</td>
<td>Hi! I am trying to fill a position for a PHP...</td>
<td>Yes</td>
</tr>
<tr>
<td>[A few thousand rows omitted]</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Call mom</td>
<td>05/23/2000</td>
<td>02:14 pm</td>
<td>Call mom. She’s been trying to reach you for a few days now</td>
<td>No</td>
</tr>
</tbody>
</table>

Note: $x_i$’s huge dimensional!

Data: $(x_1, y_1), \ldots, (x_n, y_n)$

Images

L. Rosasco, TUM 2016
All starts with DATA

- **Supervised:** \( \{(x_1, y_1), \ldots, (x_n, y_n)\} \),

- **Unsupervised:** \( \{x_1, \ldots, x_m\} \),

- **Semi-supervised:** \( \{(x_1, y_1), \ldots, (x_n, y_n)\} \cup \{x_1, \ldots, x_m\} \)
Learning from examples

**Problem:** given $S_n = \{(x_1, y_1), \ldots, (x_n, y_n)\}$ find $f(x_{\text{new}}) \sim y_{\text{new}}$
Setting for the supervised learning problem

- $X \times Y$ probability space, with measure $\rho$.

- $S_n = (x_1, y_1), \ldots, (x_n, y_n) \sim \rho^n$, i.e. sampled i.i.d.

- $L : Y \times Y \to [0, \infty)$, measurable loss function.

- Expected risk

  $$\mathcal{E}(f) = \int_{X \times Y} L(y, f(x)) d\rho(x, y).$$

Problem: Solve

$$\min_{f : X \to Y} \mathcal{E}(f),$$

given only $S_n$ ($\rho$ fixed, but unknown).
Data space

\[ f(X) = Y \]

Input space: \( X \)

Output space: \( Y \)

L. Rosasco, TUM 2016
**Input space**

\( X \) input space:

- linear spaces, e. g.
  - vectors,
  - functions,
  - matrices/operators

- “structured” spaces, e. g.
  - strings,
  - probability distributions,
  - graphs
Output space

\( Y \) output space

- linear spaces, e.g.
  - \( Y = \mathbb{R} \), regression,
  - \( Y = \{+1, -1\} \), classification,
  - \( Y = \mathbb{R}^T \), multi-task regression,
  - \( Y = \{1, \ldots, T\} \), multi-label classification

- “structured” spaces
  - strings,
  - probability distributions,
  - graphs
Probability distribution

Reflects *uncertainty* and *stochasticity* of the learning problem

\[ \rho(x, y) = \rho_X(x) \rho(y|x), \]

- \( \rho_X \) marginal distribution on \( X \),
- \( \rho(y|x) \) conditional distribution on \( Y \) given \( x \in X \).
Conditional distribution and noise

Regression

\[ y_i = f_*(x_i) + \epsilon_i, \]

- Let \( f_* : X \to Y \), fixed function
- \( \epsilon_1, \ldots, \epsilon_n \) zero mean random variables
- \( x_1, \ldots, x_n \) random
Conditional distribution and misclassification

Classification

\[ \rho(y|x) = \{ \rho(1|x), \rho(-1|x) \}, \]

Noise in classification: overlap between the classes

\[ \Delta_t = \left\{ x \in X \left| |\rho(1|x) - \rho(-1|x)| \leq t \right. \right\} \]
Marginal distribution and sampling

$\rho_X$ takes into account uneven sampling of the input space
Marginal distribution, densities and manifolds

\[ d\rho(x, y) \quad \text{or} \quad p(x, y) \, dx \, dy? \]

\[
p(x) = \frac{d\rho_x}{dx}(x)
\]

\[
p(x) = \frac{d\rho_x}{d\text{vol}}(x)
\]

L. Rosasco, TUM 2016
Loss functions

\[ L : Y \times Y \rightarrow [0, \infty), \]

- The cost of predicting \( f(x) \) in place of \( y \).
- Part of the problem definition \( \mathcal{E}(f) = \int L(y, f(x)) d\rho(x, y) \)
- Measures the pointwise error,
Loss functions for regression

\[ L(y, y') = L(y - y') \]

- **Square loss** \( L(y, y') = (y - y')^2 \),
- **Absolute loss** \( L(y, y') = |y - y'| \),
- **\( \epsilon \)-insensitive** \( L(y, y') = \max(|y - y'| - \epsilon, 0) \),
Loss functions for classification

\[ L(y, y') = L(-yy') \]

- 0-1 loss \[ L(y, y') = 1_{\{-yy' > 0\}} \]
- Square loss \[ L(y, y') = (1 - yy')^2 \],
- Hinge-loss \[ L(y, y') = \max(1 - yy', 0) \],
- logistic loss \[ L(y, y') = \log(1 + \exp(-yy')) \],
Loss functions for structured prediction

Loss specific for each learning task e. g.

- Multi-class: square loss, weighted square loss, logistic loss, …
- Multi-task: weighted square loss, absolute, …
- …
Expected risk

\[ \mathcal{E}(f) = \mathcal{E}_L(f) = \int_{X \times Y} L(y, f(x)) d\rho(x, y) \]

note that \( f \in \mathcal{F} \) where

\[ \mathcal{F} = \{ f : X \to Y \mid f \text{ measurable} \}. \]

Example \( Y = \{-1, +1\} \), \( L(y, f(x)) = 1_{\{-y f(x) > 0\}} \)

\[ \mathcal{E}(f) = \mathbb{P}(\{(x, y) \in X \times Y \mid f(x) \neq y\}). \]
Target function

\[ f_\rho = \arg \min_{f \in \mathcal{F}} E(f), \]

can be derived for many loss functions...
Target functions in regression

square loss,

\[ f_\rho(x) = \int y d\rho(y|x) \]

absolute loss,

\[ f_\rho(x) = \text{median } \rho(y|x), \]

where

\[ \text{median } p(\cdot) = y \quad \text{s.t.} \quad \int_{-\infty}^{y} tdp(t) = \int_{y}^{+\infty} tdp(t). \]
Target functions in classification

0-1 loss,
\[ f_\rho(x) = \text{sign}(\rho(1|x) - \rho(-1|x)) \]

square loss,
\[ f_\rho(x) = \rho(1|x) - \rho(-1|x) \]

logistic loss,
\[ f_\rho(x) = \log \frac{\rho(1|x)}{\rho(-1|x)} \]

hinge-loss,
\[ f_\rho(x) = \text{sign}(\rho(1|x) - \rho(-1|x)) \]
Learning algorithms

\[ S_n \rightarrow \hat{f}_n = \hat{f}_{S_n} \]

\( f_n \) estimates \( f_\rho \) given the observed examples \( S_n \)

How to measure the error of an estimator?
**Excess risk**

**Excess Risk:**

\[ \mathcal{E}(\hat{f}) - \inf_{f \in \mathcal{F}} \mathcal{E}(f), \]

**Consistency:** For any \( \epsilon > 0 \)

\[ \lim_{n \to \infty} \mathbb{P} \left( \mathcal{E}(\hat{f}) - \inf_{f \in \mathcal{F}} \mathcal{E}(f) > \epsilon \right) = 0, \]
Tail bounds, sample complexity and error bound

- **Tail bounds**: For any $\epsilon > 0, n \in \mathbb{N}$
  
  $$\mathbb{P}\left(\mathcal{E}(\hat{f}) - \inf_{f \in \mathcal{F}} \mathcal{E}(f) > \epsilon\right) \leq \delta(n, \mathcal{F}, \epsilon)$$

- **Sample complexity**: For any $\epsilon > 0, \delta \in (0, 1]$, when $n \geq n_0(\epsilon, \delta, \mathcal{F})$
  
  $$\mathbb{P}\left(\mathcal{E}(\hat{f}) - \inf_{f \in \mathcal{F}} \mathcal{E}(f) > \epsilon\right) \leq \delta,$$

- **Error bounds**: For any $\delta \in (0, 1], n \in \mathbb{N}$, with probability at least $1 - \delta$,
  
  $$\mathcal{E}(\hat{f}) - \inf_{f \in \mathcal{F}} \mathcal{E}(f) \leq \epsilon(n, \mathcal{F}, \delta),$$

L.Rosasco, TUM 2016
Error bounds and no free-lunch theorem

**Theorem** For any $\hat{f}$, there exists a problem for which

$$\mathbb{E}(\mathcal{E}(\hat{f}) - \inf_{f \in F} \mathcal{E}(f)) > 0$$
No free-lunch theorem continued

**Theorem** For any $\hat{f}$, there exists a $\rho$ such that

$$\mathbb{E}(\mathcal{E}(\hat{f}) - \inf_{f \in \mathcal{F}} \mathcal{E}(f)) > 0$$

$\mathcal{F} \rightarrow \mathcal{H}$  Hypothesis space
Hypothesis space

$\mathcal{H} \subset \mathcal{F}$

Example: linear functions $X = \mathbb{R}^d$

$$\mathcal{H} = \{ f(x) = w^\top x = \sum_{j=1}^{d} w_j x_j, \ | \ w \in \mathbb{R}^d, \forall x \in X \}$$

then $\mathcal{H} \cong \mathbb{R}^d$. 
Finite dictionaries

\[ D = \{ \phi_i : X \rightarrow \mathbb{R} \mid i = 1, \ldots, p \} \]

\[ \mathcal{H} = \{ f(x) = \sum_{j=1}^{p} w_j \phi_j(x) \mid w_1, \ldots, w_p \in \mathbb{R}, \forall x \in X \} \]

\[ f(x) = w^\top \Phi(x), \quad \Phi(x) = (\phi_1(x), \ldots, \phi_p(x)) \]
This class

Learning theory ingredients

- Data space/distribution
- Loss function, risks and target functions
- Learning algorithms and error estimates
- Hypothesis space
Next class

- Regularized learning algorithm: penalization
- Statistics and computations
- Nonparametrics and kernels

