Statistical Learning Theory and Applications
9.520/6.860 in Fall 2016

Class Times:
Monday and Wednesday 1pm-2:30pm in 46-3310 Units: 3-0-9 H,G


Email Contact:
9.520@mit.edu

Instructors: Tomaso Poggio, Lorenzo Rosasco
Guest lectures: Charlie Frogner, Carlo Ciliberto, Alessandro Verri
TAs: Hongyi Zhang, Max Kleiman-Weiner, Brando Miranda, Georgios Evangelopoulos

Web: http://www.mit.edu/~9.520/
Office Hours: Friday 2-3 pm, 46-5156 (Poggio Lab lounge)

Further Info: 9.520/6.860 is currently NOT using the Stellar system.
Registration: Fill online registration form.
Mailing list: Registered students will be added in the course mailing list (9520students)
Class
http://www.mit.edu/~9.520/

Class 2: Mathcamps

- Functional analysis (~45mins)

Linear Algebra
Basic notion and definitions: matrix and vectors norms, positive, symmetric, invertible matrices, linear systems, condition number.

Functional Analysis:
Linear and Euclidean spaces
scalar product, orthogonality
orthonormal bases, norms and semi-norms,
Cauchy sequence and complete spaces
Hilbert spaces, function spaces
and linear functional, Riesz representation theorem, convex functions, functional calculus.

- Probability (~45mins)

Probability Theory:
Random Variables (and related concepts), Law of Large Numbers,
Probabilistic Convergence, Concentration Inequalities.
• Course focuses on regularization techniques for supervised learning.
• Support Vector Machines, manifold learning, sparsity, batch and online supervised learning, feature selection, structured prediction, multitask learning.
• Optimization theory critical for machine learning (first order methods, proximal/splitting techniques).
• In the final part focus on emerging deep learning theory

The goal of this class is to provide the theoretical knowledge and the basic intuitions needed to use and develop effective machine learning solutions to a variety of problems.
Rules of the game:

- Problem sets: 4
- Final project: 2 weeks effort, you have to give us title + abstract before November 23
- Participation: check-in/sign in every class
- Grading: Psets (60%) + Final Project (30%) + Participation (10.0%)

Slides on the Web site (most classes on blackboard)

Staff mailing list is 9.520@mit.edu

Student list will be 9.520students@mit.edu

Please fill form (independent of MIT/Harvard registration)!!

send email to us if you want to be added

to mailing list
Material:
Most classes on blackboard.

Book draft:

Office hours: Friday 2-3 pm in 46-5156, Poggio Lab lounge

Tentative dates
Problem Sets (due dates will be 11 days)
Problem Set 1: 26 Sep. (due: 10/05)
Problem Set 2: 12 Oct. (due: 10/24)
Problem Set 3: 26 Oct. (due: 11/07)
Problem Set 4: 14 Nov. (due: 11/23)

Final projects:
Announcement/projects are open: Nov. 16
Deadline to suggest/pick suggestions (title/abstract): Nov. 23
Submission: Dec. xx
Final Project

The course project can be:

- **Research project** (suggested by you): Review, theory and/or application (~4 page report in NIPS format).
- **Wikipedia articles** (suggested list by us): Editing or creating new Wikipedia entries on a topic from the course syllabus.
- **Coding** (suggested by you or us): Implementation of one of the course algorithms and integration on the open-source library GURLS (Grand Unified Regularized Least Squares) https://github.com/LCSL/GURLS

- Research project reports will be archived online (on a dedicated page on our web)
- Wikipedia entries links will be archived (on a dedicated page on our web), https://docs.google.com/document/d/1RpLDfy1yMBNaSGqsdnl7w1GgznN4lb-wPaLwRJJ44mA/edit

- Classes 3-9 are the core: foundations + regularization
- Classes 10-22 are state-of-the-art topics for research in — and applications of — ML
- Classes 23-25 are partly unpublished theory on multilayer networks (DCLNs)
Today is big picture day...

Be ready for quite a bit of material

If you need a complete renovation of your Fourier analysis or linear algebra background...you should not be in this class.

Class
http://www.mit.edu/~9.520/
Summary of today’s overview

- Motivations for this course: a golden age for new AI, the key role of Machine Learning, CBMM
- A bit of history: Statistical Learning Theory, Neuroscience
- A bit of history: applications
- Now:
  - why depth works
  - why is neuroscience important
  - the challenge of sampling complexity
The problem of (human) intelligence is one of the great problems in science, probably the greatest.

Research on intelligence:
• a great intellectual mission: understand the brain, reproduce it in machines
• will help develop intelligent machines

These advances will be critical to our society’s
• future prosperity
• education, health, security
• solve all other great problems in science
CBMM’s **main** goal is to *make progress in the science of intelligence* which enables better *engineering* of intelligence.
Interdisciplinary

Cognitive Science

Machine Learning
Computer Science

Neuroscience
Computational Neuroscience

Science + Technology of Intelligence
Centerness: collaborations across different disciplines and labs

**MIT**
Boyden, Desimone, Kaelbling, Kanwisher, Katz, Poggio, Sassanfar, Saxe, Schulz, Tenenbaum, Ullman, Wilson, Rosasco, Winston

**Harvard**
Blum, Kreiman, Mahadevan, Nakayama, Sompolinsky, Spelke, Valiant

**Rockefeller**
Freiwald

**Allen Institute**
Koch

**UCLA**
Yuille

**Stanford**
Goodman

**Cornell**
Hirsh

**Hunter**
Epstein, Sakas, Chodorow

**Wellesley**
Hildreth, Conway, Wiest

**Puerto Rico**
Bykhovaskaia, Ordonez, Arce Nazario

**Howard**
Manaye, Chouikha, Rwebargira
Recent Stats and Activities

- IIT Metta, Shashua
- A*star Tan
- Hebrew U. Shashua
- MPI Buelthoff
- Genoa U. Verri
- Weizmann Ullman
- City U. HK Smale
- Google
- IBM
- Microsoft
- Schlumberger
- GE
- DeepMind
- Honda
- Boston Dynamics
- Orca
- Nvidia
- Rethink Robotics
- MobilEye
Recent Stats and Activities

Summer school at Woods Hole:
Our flagship initiative, very good!

Brains, Minds & Machines Summer Course
An intensive three-week course will give advanced students a “deep end” introduction to the problem of intelligence
Intelligence in games: the beginning
Recent progress in AI
The 2 best examples of the success of new ML

- AlphaGo
- Mobileye
Demis Hassabis, master of the new machine age

Murad Ahmed

The creator of the AI game-playing program makes all the right moves, writes Murad Ahmed

The victories have a human mastermind in Demis Hassabis, co-founder and chief executive of DeepMind. He describes Mr Lee as the “Roger Federer of Go”, and for some the computer program’s achievement is akin to a robot taking to the lawns of Wimbledon and beating the legendary tennis champion.

“I think it is pretty huge but, ultimately, it will be for historians in 200 years’ time to judge.”
Real Engineering: Mobileye
History
History: same hierarchical architectures in the cortex, in models of vision and in deep networks

Desimone & Ungerleider 1989; vanEssen+Movshon
The Science of Intelligence

The science of intelligence was at the roots of today’s engineering success

We need to make another basic effort on it

• for the sake of basic science
• for the engineering of tomorrow
Summary of today’s overview

• Motivations for this course: a golden age for new AI, the key role of Machine Learning, CBMM

• A bit of history: Statistical Learning Theory, Neuroscience

• A bit of history: applications

• Now:
  - why depth works
  - why is neuroscience important
  - the challenge of sampling complexity
Given a set of \( l \) examples (data)

\[
\{(x_1, y_1), (x_2, y_2), \ldots, (x_l, y_l)\}
\]

Question: find function \( f \) such that

\[ f(x) = \hat{y} \]

is a good predictor of \( y \) for a future input \( x \) (fitting the data is not enough!)
Generalization:
estimating value of function where there are no data (good generalization means predicting the function well; important is for empirical or validation error to be a good proxy of the prediction error)
Statistical Learning Theory: supervised learning
Statistical Learning Theory: part of mainstream math not just statistics (Valiant, Vapnik, Smale, Devore...)

ON THE MATHEMATICAL FOUNDATIONS OF LEARNING

FELIPE CUCKER AND STEVE SMALE

The problem of learning is arguably at the very core of the problem of intelligence, both bi

T. Poggio and C.R. Shelton

INTRODUCTION

(1) A main theme of this report is the relationship of approximation to learning and the primary role of sampling (inductive inference). We try to emphasize relations of the theory of learning to the mainstream of mathematics. In particular, there are large roles for probability theory, for algorithms such as least squares, and for tools and ideas from linear algebra and linear analysis. An advantage of doing this is that communication is facilitated and the power of core mathematics is more easily brought to bear.
There is an unknown **probability distribution** on the product space $Z = X \times Y$, written $\mu(z) = \mu(x, y)$. We assume that $X$ is a compact domain in Euclidean space and $Y$ a bounded subset of $\mathbb{R}$. The **training set** $S = \{(x_1, y_1), \ldots, (x_n, y_n)\} = \{z_1, \ldots, z_n\}$ consists of $n$ samples drawn i.i.d. from $\mu$.

$\mathcal{H}$ is the **hypothesis space**, a space of functions $f : X \rightarrow Y$.

A **learning algorithm** is a map $L : Z^n \rightarrow \mathcal{H}$ that looks at $S$ and selects from $\mathcal{H}$ a function $f_S : x \rightarrow y$ such that $f_S(x) \approx y$ in a **predictive way**.
Statistical Learning Theory

Given a function \( f \), a loss function \( V \), and a probability distribution \( \mu \) over \( Z \), the expected or true error of \( f \) is:

\[
l[f] = \mathbb{E}_Z V[f, z] = \int_Z V(f, z) d\mu(z)
\]

which is the expected loss on a new example drawn at random from \( \mu \).

The empirical error of \( f \) is:

\[
l_S[f] = \frac{1}{n} \sum V(f, z_i)
\]

A very natural requirement for \( f_S \) is distribution independent generalization

\[
\forall \mu, \lim_{n \to \infty} |l_S[f_S] - l[f_S]| = 0 \quad \text{in probability}
\]

In other words, the training error for the solution must converge to the expected error and thus be a “proxy” for it. Otherwise the solution would not be “predictive”. 
The ERM problem does not have a *predictive* solution in general (just fitting the data does not work).

Choosing an appropriate hypothesis space $H$ (for instance a compact set of continuous functions) can guarantee generalization. A necessary and sufficient condition for generalization is that $H$ *is uGC*.

Related concept, measuring complexity of the hypothesis space, are:

- VC dimension,
- $V_{\gamma}$ dimension,
- Rademacher numbers.
Statistical Learning Theory: the learning problem should be well-posed

A problem is well-posed if its solution exists, unique and is stable, eg depends continuously on the data (here examples)

J. S. Hadamard, 1865-1963
This is an example of foundational results in learning theory...
Conditions for generalization in learning theory have deep, almost philosophical, implications:

- they can be regarded as equivalent conditions that guarantee a
  theory to be predictive (that is scientific)

- theory must be chosen from a small hypothesis set

- theory should not change much with new data...most of the time (stability)
Equation includes splines, Radial Basis Functions and SVMs (depending on choice of $K$ and $V$).

Classical algorithm:
Regularization in RKHS (eg. kernel machines)

$$\min_{f \in H} \left[ \frac{1}{n} \sum_{i=1}^{n} V(f(x_i) - y_i) + \lambda \|f\|_K^2 \right]$$

implies

$$f(x) = \sum_{i}^{n} \alpha_i K(x, x_i)$$

For a review, see Poggio and Smale, 2003; see also Schoelkopf and Smola, 2002; Bousquet, O., S. Boucheron and G. Lugosi; Cucker and Smale; Zhou and Smale...
Classical algorithm:
Regularization in RKHS (eg. kernel machines)

\[
\min_{f \in H} \left[ \frac{1}{n} \sum_{i=1}^{n} V(f(x_i) - y_i) + \lambda \left\| f \right\|_K^2 \right]
\]

implies

\[
f(x) = \sum_{i=1}^{n} \alpha_i K(x, x_i)
\]

Remark (for later use):

Classical kernel machines correspond to shallow networks
Summary of today’s overview

- Motivations for this course: a golden age for new AI, the key role of Machine Learning, CBMM

- A bit of history: Statistical Learning Theory, Neuroscience

- A bit of history: applications

- Now:
  - why depth works
  - why is neuroscience important
  - the challenge of sampling complexity
Learning

LEARNING THEORY + ALGORITHMS

Theorems on foundations of learning
Predictive algorithms

COMPUTATIONAL NEUROSCIENCE: models+experiments

How visual cortex works

Sung & Poggio 1995, also Kanade& Baluja....
LEARNING THEORY + ALGORITHMS

Theorems on foundations of learning
Predictive algorithms

Sung & Poggio 1995

How visual cortex works

Computational neuroscience: models + experiments
Engineering of Learning

**LEARNING THEORY + ALGORITHMS**
- Theorems on foundations of learning
- Predictive algorithms

*Face detection* has been available in digital cameras for a few years now

**COMPUTATIONAL NEUROSCIENCE: models+experiments**
- How visual cortex works

\[
\min_{f \in H} \left[ \frac{1}{l} \sum_{i=1}^{l} V(y_i, f(x_i)) + \mu \|f\|_2^2 \right]
\]

\[
f(x) = \sum_{i=1}^{l} c_i K(x_i, x)
\]
Engineering of Learning

LEARNING THEORY + ALGORITHMS

Theorems on foundations of learning
Predictive algorithms

People detection
Papageorgiou & Poggio, 1997, 2000
also Kanade & Scheiderman

How visual cortex works

COMPUTATIONAL NEUROSCIENCE: models + experiments
Engineering of Learning

LEARNING THEORY + ALGORITHMS

Theorems on foundations of learning
Predictive algorithms

Pedestrian detection
Papageorgiou&Poggio, 1997, 2000
also Kanade&Scheiderman

COMPUTATIONAL NEUROSCIENCE:
models+experiments

How visual cortex works
Some other examples of past ML applications from my lab

Computer Vision
- Face detection
- Pedestrian detection
- Scene understanding
- Video categorization
- Video compression
- Pose estimation

Graphics
Speech recognition
Speech synthesis
Decoding the Neural Code
Bioinformatics
Text Classification
Artificial Markets
Stock option pricing
Decoding the neural code: Matrix-like read-out from the brain
The end station of the ventral stream in visual cortex is IT.
Recording at each recording site during passive viewing

- 77 visual objects
- 10 presentation repetitions per object
- Presentation order randomized and counter-balanced
Decoding the neural code ... using a classifier

Learning from \((x, y)\) pairs

\(x\)

\(y \in \{1, \ldots, 8\}\)
We can decode the brain’s code and read-out from neuronal populations: reliable object categorization (>90% correct) using ~200 arbitrary AIT “neurons”

Hung, Kreiman, Poggio, DiCarlo. *Science* 2005
We can decode the brain’s code and read-out from neuronal populations:

reliable object categorization using ~100 arbitrary AIT sites

- [100-300 ms] interval
- 50 ms bin size
Learning: image analysis

⇒ Bear (0° view)

⇒ Bear (45° view)
Learning: image synthesis

UNCONVENTIONAL GRAPHICS

\[ \Theta = 0^\circ \text{ view} \implies \]

\[ \Theta = 45^\circ \text{ view} \implies \]
Mary101

A- more in a moment
1. **Learning**

System learns from 4 mins of video face appearance (Morphable Model) and speech dynamics of the person.

2. **Run Time**

For any speech input the system provides as output a synthetic video stream.

```
[Diagram]
Phone Stream

<table>
<thead>
<tr>
<th>Trajectory Synthesis</th>
</tr>
</thead>
<tbody>
<tr>
<td>MMM</td>
</tr>
</tbody>
</table>

Phonetic Models
Image Prototypes
```

**Image**

[Image of a person]
B-Dido
A Turing test: what is real and what is synthetic?
Summary of today’s overview

• Motivations for this course: a golden age for new AI, the key role of Machine Learning, CBMM

• A bit of history: Statistical Learning Theory, Neuroscience

• A bit of history: applications

• Now:
  - why depth works
  - why is neuroscience important
  - the challenge of sampling complexity
Classical learning algorithms: “high” sample complexity and shallow architectures

How do the learning machines described by classical learning theory -- such as kernel machines -- compare with brains?

- One of the most obvious differences is the ability of people and animals to learn from very few examples (“poverty of stimulus” problem).

- A comparison with real brains offers another, related, challenge to learning theory. Classical “learning algorithms” correspond to one-layer architectures. The cortex suggests a hierarchical architecture.

Thus...are hierarchical architectures with more layers important perhaps for the sample complexity issue?

Tomaso Poggio and Steve Smale

The Mathematics of Learning: Dealing with Data
Tomaso Poggio and Steve Smale
Computation in a neural net

\[ f(x) = f_L(\ldots f_2(f_1(x))) \]
Computation in a neural net

Rectified linear unit (ReLU)

\[ g(y) = \max(0, y) \]
Classical kernel machines are equivalent to shallow networks:

Kernel machines...

\[ f(x) = \sum_{i}^{l} c_i K(x, x_i) + b \]

can be “written” as shallow networks: the value of K corresponds to the “activity” of the “unit” for the input and the correspond to “weights”
Deep and shallow networks: universality

**Theorem** Shallow, one-hidden layer networks with a nonlinear $\phi(x)$ which is not a polynomial are universal. Arbitrarily deep networks with a nonlinear $\phi(x)$ (including polynomials) are universal.

$$g(x) = \sum_{i=1}^{r} c_i \langle w_i, x \rangle + b_i$$
Theorem:
why and when are deep networks better than shallow network?

\[ f(x_1, x_2, ..., x_8) = g_3(g_{21}(g_{11}(x_1, x_2), g_{12}(x_3, x_4))g_{22}(g_{11}(x_5, x_6), g_{12}(x_7, x_8))) \]

Theorem (informal statement)
Suppose that a function of \( d \) variables is compositional. Both shallow and deep network can approximate \( f \) equally well. The number of parameters of the shallow network depends exponentially on \( d \) as \( O(\varepsilon^{-d}) \) with the dimension whereas for the deep network depends linearly on \( d \) that is \( O(d\varepsilon^{-2}) \).
The curse of dimensionality, the blessing of compositionality

For compositional functions deep networks — but not shallow ones — can avoid the curse of dimensionality, that is the exponential dependence on the dimension of the network complexity and of its sample complexity.
Summary of today’s overview

- Motivations for this course: a golden age for new AI, the key role of Machine Learning, CBMM

- A bit of history: Statistical Learning Theory, Neuroscience

- A bit of history: applications

- Now:
  - why depth works
  - why is neuroscience important
  - to the brain from physics via depth?
  - the challenge of sampling complexity
CBMM: motivations

Key recent advances in the engineering of intelligence have their roots in basic science of the brain.
Recognition in Visual Cortex


- As a biological model of object recognition in the ventral stream – from V1 to PFC -- it is perhaps the most quantitatively faithful to known neuroscience data.
Hierarchical **feedforward** models of the ventral stream do “work”
Using goal-driven deep learning models to understand sensory cortex

Daniel L K Yamins\textsuperscript{1,2} & James J DiCarlo\textsuperscript{1,2}
Using goal-driven deep learning models to understand sensory cortex

Daniel L K Yamins\textsuperscript{1,2} & James J DiCarlo\textsuperscript{1,2}

\begin{itemize}
  \item \textbf{a}: IT single-site neural predictivity (\% explained variance) vs. Categorization performance (balanced accuracy). Points indicate model predictions.\textsuperscript{1} The \textit{IT} single-site neural predictivity is derived from IT responses to individual test images. The categorization performance is derived from a logistic regression of the IT neural responses to each image category.
  \item \textbf{b}: HCNN top hidden layer response prediction. The response to a test image is predicted by an HCNN and compared to the actual neural response of the IT cortex to that image.\textsuperscript{2}
  \item \textbf{c}: Single-site neural predictivity for Monkey V4 (\textit{n} = 128) and Monkey IT (\textit{n} = 166). The ideal observer and control models are compared to the HCNN layers.
  \item \textbf{d}: Human IT (fMRI) vs. HCNN model. Artificial and natural stimuli are classified into categories of animal, non-animal, face, and body.
  \item \textbf{e}: Human V1–V3 and Human IT. Visual voxel correlation (Kendall's $\tau_b$) for each layer.
\end{itemize}
Summary of today’s overview

• Motivations for this course: a golden age for new AI, the key role of Machine Learning, CBMM

• A bit of history: Statistical Learning Theory, Neuroscience

• A bit of history: applications

• Now:
  - why depth works
  - why is neuroscience important
  - to the brain from physics via depth?
  - the challenge of sampling complexity
How do the learning machines described by classical learning theory -- such as kernel machines -- compare with brains?

- One of the most obvious differences is the ability of people and animals to learn from very few examples ("poverty of stimulus" problem).

- A comparison with real brains offers another, related, challenge to learning theory. Classical "learning algorithms" correspond to one-layer architectures. The cortex suggests a hierarchical architecture.

Thus...are hierarchical architectures with more layers the answer to the sample complexity issue?


*Tomaso Poggio and Steve Smale*
Today’s science, tomorrow’s engineering: learn like children learn

The first phase (and successes) of ML: supervised learning, big data: $n \to \infty$

from programmers…
…to labelers…
…to computers that learn like children…

The next phase of ML: implicitly supervised learning, learning like children do, small data: $n \to 1$
Summary of today’s overview

- Motivations for this course: a golden age for new AI, the key role of Machine Learning, CBMM

- A bit of history: Statistical Learning Theory, Neuroscience

- A bit of history: applications

- Now:
  - why depth works
  - why is neuroscience important
  - to the brain from physics via depth?
  - the challenge of sampling complexity