# CS281B/Stat241B. Statistical Learning Theory. Lecture 1.

**Peter Bartlett**

1. Organizational issues.
2. Overview.
3. Probabilistic formulation of prediction problems.
4. Game theoretic formulation of prediction problems.

## Organizational Issues

- Lectures: Tue/Thu 12:30–2:00, 334 Evans.
- Peter Bartlett. bartlett@cs.
  Office hours: Mon 11-12 (Sutardja-Dai Hall), Thu 2-3 (Evans 399).
- GSI: Alan Malek. malek@berkeley Office hours: TBA.
- Web site: see http://www.stat.berkeley.edu/∼bartlett/courses
  Check it for details of office hours, the syllabus, assignments, readings, lecture notes, and announcements.
- No text. See website for readings.

## Organizational Issues

- **Assessment:**

  Homework Assignments (50%): posted on the website.
  (approximately one every two weeks)
  Final Project (50%): Proposals due March 13. Report due May 2.

- Required background:

  CS281A/Stat241A/Stat205A/Stat210A.

# Overview

Theoretical analysis of prediction methods.

1. Probabilistic formulation of prediction problems
2. Risk bounds
3. Game theoretic formulation of prediction problems
4. Regret bounds
5. Algorithms:
   (a) Kernel methods
   (b) Boosting algorithms
6. Model selection

# Probabilistic Formulations of Prediction Problems

**Aim:** Predict an outcome $y$ from some set $\mathcal{Y}$ of possible outcomes, on the basis of some observation $x$ from a feature space $\mathcal{X}$. Some examples:

| $x$ | $y$ |
|---|---|
| words in a document | topic (sports, music, tech, ...) |
| image of a digit in a zipcode | the digit |
| email message | spam or ham |
| sentence | correct parse tree |
| patient medical test results | patient disease state |
| gene expression levels of a tissue sample | presence of cancer |

# Probabilistic Formulations of Prediction Problems

| $x$ | $y$ |
|---|---|
| phylogenetic profile of a gene (i.e., relationship to genomes of other species) | gene function |
| image of a signature on a check | identity of the writer |
| web search query | ranked list of pages |

Use *data set* of $n$ pairs:

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

to choose a function $f : \mathcal{X} \to \mathcal{Y}$ so that, for subsequent $(x, y)$ pairs, $f(x)$ is a good prediction of $y$.

## Probabilistic Formulations of Prediction Problems

To define the notion of a 'good prediction,' we can define a **loss function**

$$\ell : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}.$$

So $\ell(\hat{y}, y)$ quantifies the cost of predicting $\hat{y}$ when the true outcome is $y$. Then the aim is to ensure that $\ell(f(x), y)$ is small.

# Probabilistic Formulations of Prediction Problems

**Example:** In *pattern classification* problems, the aim is to classify a pattern $x$ into one of a finite number of classes (that is, the label space $\mathcal{Y}$ is finite). If all mistakes are equally bad, we could define

$$\ell(\hat{y}, y) = 1[\hat{y} \neq y] = \begin{cases} 1 & \text{if } \hat{y} \neq y, \\ 0 & \text{otherwise.} \end{cases}$$

**Example:** In a *regression* problem, with $\mathcal{Y} = \mathbb{R}$, we might choose the quadratic loss function, $\ell(\hat{y}, y) = (\hat{y} - y)^2$.

# Probabilistic Assumptions

Assume:

- There is a probability distribution $P$ on $\mathcal{X} \times \mathcal{Y}$,
- The pairs $(X_1, Y_1), \ldots, (X_n, Y_n), (X, Y)$ are chosen independently according to $P$

The aim is to choose $f$ with small *risk*:

$$R(f) = \mathrm{E}\ell(f(X), Y).$$

For instance, in the pattern classification example, this is the misclassification probability.

$$R(f) = \mathrm{E}1[f(X) \neq Y] = \Pr(f(X) \neq Y).$$

## Probabilistic Assumptions

Some things to notice:

1. Capital letters denote random variables.
2. The distribution $P$ can be viewed as modelling both the relative frequency of different features or covariates $X$, together with the conditional distribution of the outcome $Y$ given $X$.
3. The assumption that the data is i.i.d. is a strong one.
   But we need to assume something about what the information in the data $(x_1, y_1), \ldots, (x_n, y_n)$ tells us about $(X, Y)$.

## Probabilistic Assumptions

4. The function $x \mapsto f_n(x) = f_n(x; X_1, Y_1, \ldots, X_n, Y_n)$ is random, since it depends on the random data $D_n = (X_1, Y_1, \ldots, X_n, Y_n)$. Thus, the risk

$$\begin{aligned} R(f_n) &= \mathrm{E}\left[\ell(f_n(X), Y)|D_n\right] \\ &= \mathrm{E}\left[\ell(f_n(X; X_1, Y_1, \ldots, X_n, Y_n), Y)|D_n\right] \end{aligned}$$

is a random variable. We might aim for $\mathrm{E}R(f_n)$ small, or $R(f_n)$ small with high probability (over the training data).

## Key Questions

We might choose $f_n$ from some class $F$ of functions
(for instance, linear function, sparse linear function, decision tree, neural network, kernel machine).

There are several questions that we are interested in:

1. Can we design algorithms for which $f_n$ is close to the best that we could hope for, given that it was chosen from $F$? (that is, is $R(f_n) - \inf_{f \in F} R(f)$ small?)
2. How does the performance of $f_n$ depend on $n$? On the complexity of $F$? On $P$?
3. Can we ensure that $R(f_n)$ approaches the best possible performance (that is, the infimum over all $f$ of $R(f)$)?

## Statistical Learning Theory vs Classical Statistics

- In this course, we are concerned with results that apply to large classes of distributions $P$, such as the set of *all* joint distributions on $\mathcal{X} \times \mathcal{Y}$. In contrast to parametric problems, we will not (often) assume that $P$ comes from a small (e.g., finite-dimensional) space, $P \in \{P_\theta : \theta \in \Theta\}$.
- Since we make few assumptions on $P$, and we are concerned with high-dimensional data, the goal is typically to ensure that the performance is close to the best we can achieve using prediction rules from some fixed class $F$.

## Key Issues

Several key issues arise in designing a prediction method for these problems:

**Approximation** How good is the best $f$ in the class $F$ that we are using? That is, how close to $\inf_f R(f)$ is $\inf_{f \in F} R(f)$?

**Estimation** How close is our performance to that of the best $f$ in $F$? (Recall that we only have access to the distribution $P$ through observing a finite data set.)

**Computation** We need to use the data to choose $f_n$, typically by solving some kind of optimization problem. How can we do that efficiently?

## Key Issues

- We will not spend much time on the approximation properties, beyond observing some *universality* results (that particular classes can achieve zero approximation error). (But for complex problems and simple—hence statistically feasible—function classes, this is not a very interesting property.)
- We will focus on the *estimation* issue.
- We will take the approach that efficiency of computation is a *constraint*. Indeed, the methods that we spend most of our time studying involve convex optimization problems. (e.g., kernel methods involve solving a quadratic program, and boosting algorithms involve minimizing a convex criterion in a convex set.)

# More General Probabilistic Formulation

We can consider a decision-theoretic formulation: Have

1. Outcome space $\mathcal{Z}$.
2. Prediction strategy $S : \mathcal{Z}^* \to \mathcal{A}$.
3. Loss function $\ell : \mathcal{A} \times \mathcal{Z} \to \mathbb{R}$.

Protocol:

- See outcomes $Z_1, \ldots, Z_n$, i.i.d. from unknown $P$ on $\mathcal{Z}$.
- Choose action $a = S(Z_1, \ldots, Z_n) \in \mathcal{A}$.
- Incur risk $\mathrm{E}\ell(a, Z)$.

Aim is to minimize the excess risk, compared to the best decision:

$$\mathrm{E}\left[\ell(S(Z_1, \ldots, Z_n), Z) | Z_1^n\right] - \inf_{a \in \mathcal{A}} \mathrm{E}\ell(a, Z).$$

# More General Probabilistic Formulation

**Example:** In *pattern classification* problems,

- $\mathcal{Z} = \mathcal{X} \times \mathcal{Y}$,
- $\mathcal{A} \subset \mathcal{Y}^{\mathcal{X}}$.
- $\ell(f, (x, y)) = 1[f(x) \neq y]$.

**Example:** In *density estimation* problems,

- $\mathcal{Z} = \mathbb{R}^d$ (or some measurable space).
- $\mathcal{A} =$ measurable functions on $\mathcal{Z}$ (densities wrt a reference measure on $\mathcal{Z}$).
- $\ell(p, y) = -\log p(z)$.

In this case, if the distribution $P$ has a density in $\mathcal{A}$, the excess risk is the KL-divergence between $a$ and $P$.

# Game Theoretic Formulation

Decision method plays $a_t \in \mathcal{A}$

World reveals $z_t \in \mathcal{Z}$

Incur loss $\ell(a_t, z_t)$

- Cumulative loss: $\hat{L}_n = \sum_{t=1}^{n} \ell(a_t, z_t)$.
- Aim to minimize regret, that is, perform well compared to the best (in retrospect) from some class:

$$\text{regret} = \underbrace{\sum_{t=1}^{n} \ell(a_t, z_t)}_{\hat{L}_n} - \underbrace{\inf_{a \in \mathcal{A}} \sum_{t=1}^{n} \ell(a, z_t)}_{L_n^*} .$$

- Data can be adversarially chosen.

## Game Theoretic Formulation: Motivation

1. Appropriate formulation for online/sequential prediction problems.
2. Adversarial model is often appropriate (e.g., in computer security, computational finance).
3. Adversarial model assumes little:
   It is often straightforward to convert a strategy for an adversarial environment to a method for a probabilistic environment.
4. Studying the adversarial model can reveal the *deterministic core* of a statistical problem: there are strong similarities between the performance guarantees in the two cases.
5. Significant overlaps in the design of methods for the two problems:
   - *Regularization* plays a central role.
   - Often have a natural interpretation as a *Bayesian method*.

# Examples

**Example:** In an online *pattern classification* problem (like spam classification),

- $\mathcal{Z} = \mathcal{X} \times \mathcal{Y}$,
- $\mathcal{A} \subset \mathcal{Y}^{\mathcal{X}}$.
- $\ell(f, (x, y)) = 1[f(x) \neq y]$.

The action is a classification rule, and the regret indicates how close the spam misclassification rate is to the best performance possible in retrospect on the particular email sequence.

# Example: Portfolio Optimization

- Aim to choose a portfolio (distribution over financial instruments) to maximize utility.
- Other market players can profit from making our decisions bad ones. For example, if our trades have a market impact, someone can *front-run* (trade ahead of us).
- The decision method's action $a_t$ is a distribution on the $m$ instruments, $a_t \in \Delta^m = \{a \in [0,1]^m : \sum_i a_i = 1\}$.
- The outcome $z_t$ is the vector of relative price increases, $z_t \in \mathbb{R}^m_+$; the $i$th component is the ratio of the price of instrument $i$ at time $t$ to its price at the previous time.
- The loss $\ell$ might be the negative logarithm of the portfolio's increase,

$$\ell(a_t, z_t) = -\log(a_t \cdot z_t).$$

## Example: Portfolio Optimization

- We might compare our performance to the best stock (distribution is a delta function), or a set of indices (distribution corresponds to Dow Jones Industrial Average, etc), or the set of all distributions.
- The regret is then the log of the ratio of the maximum value the portfolio would have at the end (for the best mixture choice) to the final portfolio value:

$$\sum_{t=1}^{n} \ell(a_t, z_t) - \min_{a \in \mathcal{A}} \sum_{t=1}^{n} \ell(a, z_t) = \max_{a \in \mathcal{A}} \sum_{t=1}^{n} \log(a \cdot z_t) - \sum_{t=1}^{n} \log(a_t \cdot z_t),$$

since $a \cdot z_t$ is the relative increase in capital under action $a$.

## Key Questions

Often interested in minimax regret, which is the value of the game:

$$\min_{a_1} \max_{z_1} \cdots \min_{a_n} \max_{z_n} \left( \sum_{t=1}^{n} \ell(a_t, z_t) - \min_{a \in \mathcal{A}} \sum_{t=1}^{n} \ell(a, z_t) \right).$$

1. How does the performance (minimax regret) depend on $n$? On the complexity of $\mathcal{A}$ (and $\mathcal{Z}$)?
2. Can we design computationally efficient strategies that (almost) achieve the minimax regret?
3. What if the strategy has *limited information*? (e.g., auctions, bandits)

## Overview: probabilistic and game-theoretic formulations

- Decision-theoretic formulation:
  For outcome $Z$, action $a$, incur loss $\ell(a, Z)$.
- Probabilistic:
  - Data $Z_1, \ldots, Z_n, Z$ i.i.d.,
  - Use data to choose $a \in \mathcal{A}$,
  - Aim to minimize excess risk,

$$\mathrm{E}\ell(a, Z) - \inf_{a^* \in \mathcal{A}} \mathrm{E}\ell(a^*, Z).$$

## Overview: probabilistic and game-theoretic formulations

- Online:
  - Arbitrary (even adversarial) choice of data.
  - Sequential game: at round $t$,
    - Choose $a_t$,
    - See $Z_t$,
    - Incur loss $\ell(a_t, Z_t)$.
  - Aim to minimize regret (excess cumulative loss):

$$\sum_t \ell(a_t, Z_t) - \inf_{a^* \in \mathcal{A}} \sum_t \ell(a^*, Z_t).$$