

PREDICTIVE LEARNING

y = “response/output” variable (unknown)

$\mathbf{x} = (x_1, x_2, \dots, x_n)$ = “input/predictor” variables

Prediction: $\hat{y} = F(\mathbf{x})$:

$L(y, F)$ = loss criterion:

regression: $L(y, F) = (y - F)^2, |y - F|$

classification: $y, F \in \{c_1, c_2, \dots, c_K\}$

$L(y, F) = L_{y,F} = K \times K$ matrix

Lack of accuracy (“risk”): $R(F) = E_{\mathbf{x}y} L(y, F(\mathbf{x}))$

Optimal (“target”) function: $F^* = \arg \min_F R(F)$

Learning: $T = \{\mathbf{x}_i, y_i\}_1^N$ “training” sample

$F(\mathbf{x}) = \text{learning procedure}(T) \simeq F^*(\mathbf{x})$

$F^*(\underline{x}) = \text{"target function"}$

Goal: find good approx
to $F^*(\underline{x})$ using the data

$\hat{F}(\underline{x}) = \text{learning procedure}(\{y_i, \underline{x}_i\}_1^N)$

This course:

Modern learning procedures, many
of which were developed in
other fields.

- (1) Machine learning (AI & CS)
 - Decision trees & rule based
- (2) Psychology - neural nets
- (3) Engineering - pattern recog.
 - near neighbor methods
 - clustering
- (4) Biology - clustering

"Theory" very thin for these methods

(1) Most theory developed for linear procedures

$$\hat{F}(\underline{x}) = \sum_{i=1}^N H(\underline{x}, \underline{x}_i) y_i$$

↑ indep of y 's

(2) Nearly all theory for fixed $\{\underline{x}_i\}_1^N$
(even nonlinear theory)

(3) Much intuition about random \underline{x}
extrapolated from this theory
(fixed \underline{x} , nonlinear) Feller \rightarrow Stein

(4) Reality can be quite different
for both
e.g. bias - variance trade-off

Most techniques motivated from
heuristics

Dispite claims to the contrary

(1) No method = "magic bullet"

(2) No method is universally better than any other (reasonable) one

David Wolpert: No free lunch th^ms

(3) Each method has class of target functions $F^*(\underline{x})$, sample size N , signal / noise ratio

$$y = F^*(\underline{x}) + \varepsilon$$

↑ "noise"

for which it's best

(4) How to choose ?

Methods differ on what they assume about $F^*(\underline{x})$

Match method to what is known about the problem at hand

Try several - estimate best or use committee

Comparison caution:

Reported (empirical) performance comparisons between methods must be interpreted with care

No method dominates others over all situations

For every method, there are at least some situations (target function, S/N, sample size) for which it is especially appropriate (and conversely)

Performance rankings on particular examples should not be extrapolated beyond those specific examples.

(^{also,} performance = random variable)

this is especially true if the author's (new) method is one of the competitors.

Selection biases :

Examples are intended to validate authors' method

those for which it does best are more likely to be presented

(none => no paper)

No idea of how many were tried to get chosen presented subset.

Paper selection effect

(no paper if not the best)

Hidden tuning (on "test" data)

Expert bias :

In any data analysis, degree of success depends on the analyst and his/her skill with the method.

Author is more expert in applying their own method than competitors

- more skill in tuning
- more motivated to do it
- untuned competitors represent targets to be beaten

Conclusions:

Comparisons most useful:

Only purpose of the paper
author(s) have no vested
interests,

nor differential skills
among the competitors

(Santa Fe competition)

When not the case:

Comparisons among other
methods most interesting

(second → last on equal
footing)

"Contests" useful since each
entrant uses their best method
Still not reality

Components of Learning Algorithms

(Hand, Mannila & Smyth)

I. Model or pattern structure:
underlying functional form
sought from data

$\hat{F}(x) \in \mathcal{F}$ class of functions
indexed by parameters

Examples: OLR

\mathcal{F} = all linear functions

$$\hat{F}(x) = a_0 + \sum_{j=1}^m a_j x_j \quad \forall (a_0, a_j)$$

Additive modeling:

\mathcal{F} = all smooth additive fun's

$$\hat{F}(x) = \sum_{j=1}^m \overset{\leftarrow \text{smooth}}{f_j(x_j)}$$

SVM: \mathcal{F} = all polynomials to
fixed order (determined
by kernel)

II. Score function : judges (lack of) quality of fitted model to data

Population : $E_{y,x} L(y, F(x))$
(ultimate)

Data : sometimes (most "natural")

$$\frac{1}{N} \sum_{i=1}^N L(y_i, F(x_i))$$

Often different.

OLR :

$$\text{pop: } E_{y,x} (y - a_0 - \sum_{j=1}^m a_j x_j)^2$$

$$\text{data: } \frac{1}{N} \sum_{i=1}^N (y_i - a_0 - \sum_{j=1}^m a_j x_{ij})^2$$

RR / LASSO

$$\text{data: OLR} + \begin{cases} \lambda \sum_{j=1}^m a_j^2 \\ \lambda \sum_{j=1}^m |a_j| \end{cases}$$

SVM :

$$\text{pop: } E_{y,x} I(y_i \neq \text{sign } \hat{F}(x_i))$$

$$\text{data: } \frac{1}{N} \sum_{i=1}^N [1 - y_i \hat{F}(x_i)]_+ + \lambda \sum_{j=1}^m a_j^2$$

III Optimization search method:

Minimize data criterion on sample

$$\hat{\underline{a}} = \underset{\underline{a}}{\operatorname{argmin}} \sum_{i \in \mathcal{I}(\underline{a})} \mathcal{L}(y_i, f(x_i; \underline{a}))$$

Examples:

OLR - direct matrix algebra

RR - "

LASSO - quadratic program

SVM - "

all convex problems

unique minimum 1 data

More flexible procedures

(Trees, neural nets, clustering)

direct sol'n not possible

non CONVEX criteria

Heuristic search strategies
(iterative algorithms)

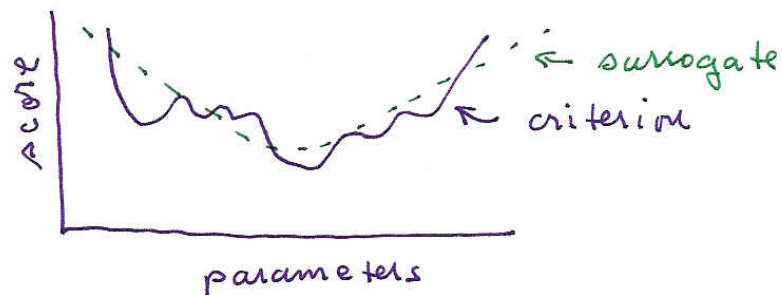
Greedy, steepest descent
EM algorithm
Numerical optimization

Multiple minima

algorithmic + statistical problems

sol'n depends on start (q_0)

Statistical properties depend on
search strategy as well as I. & II.



Can't be separated as is often
done in theory

Often surrogate (smoother) criterion
works better with greedy search

Summary - learning procedures:

I. Model or pattern structure

$$\hat{F}(\underline{x}) = \hat{F}(\underline{x}; \underline{a}) \in \mathcal{F}(\underline{a})$$

II. Score criterion

$$\mathcal{S}^I(\underline{a}) = E_{y, \underline{x}} L(y, \hat{F}(\underline{x}; \underline{a})) \quad \text{pop.}$$

$$\hat{\mathcal{S}}^I(\underline{a}) = \frac{1}{N} \sum_{i=1}^N \Delta(y_i, \hat{F}(\underline{x}_i; \underline{a})) \quad \text{data}$$
$$+ \lambda P(\underline{a})$$

sometimes $\Delta = L$, often not

III. Search strategy

$$\hat{\underline{a}} = \underset{\underline{a}}{\operatorname{argmin}} \hat{\mathcal{S}}^I(\underline{a})$$

"Different" procedures can differ in any or all of above