TUM 2016
Class 3
Large scale learning by regularization

Lorenzo Rosasco
UNIGE-MIT-IIT

July 25, 2016

Learning problem

Solve

$$\min_w \mathcal{E}(w), \quad \mathcal{E}(w) = \int d\rho(x,y) L(w^\top x, y)$$

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

Beyond linear models: non-linear features and kernels

Regularization by penalization

Replace

$$\min_w \mathcal{E}(w) \quad \text{by} \quad \min_w \underbrace{\widehat{\mathcal{E}}(w) + \lambda \|w\|^2}_{\widehat{\mathcal{E}}_\lambda(w)}$$

- $\widehat{\mathcal{E}}(w) = \frac{1}{n} \sum_{i=1}^n L(w^\top x_i, y_i)$
- $\lambda > 0$ regularization parameter

Early stopping regularization

Another example of regularization:

Early stopping of an iterative procedure applied to noisy data.

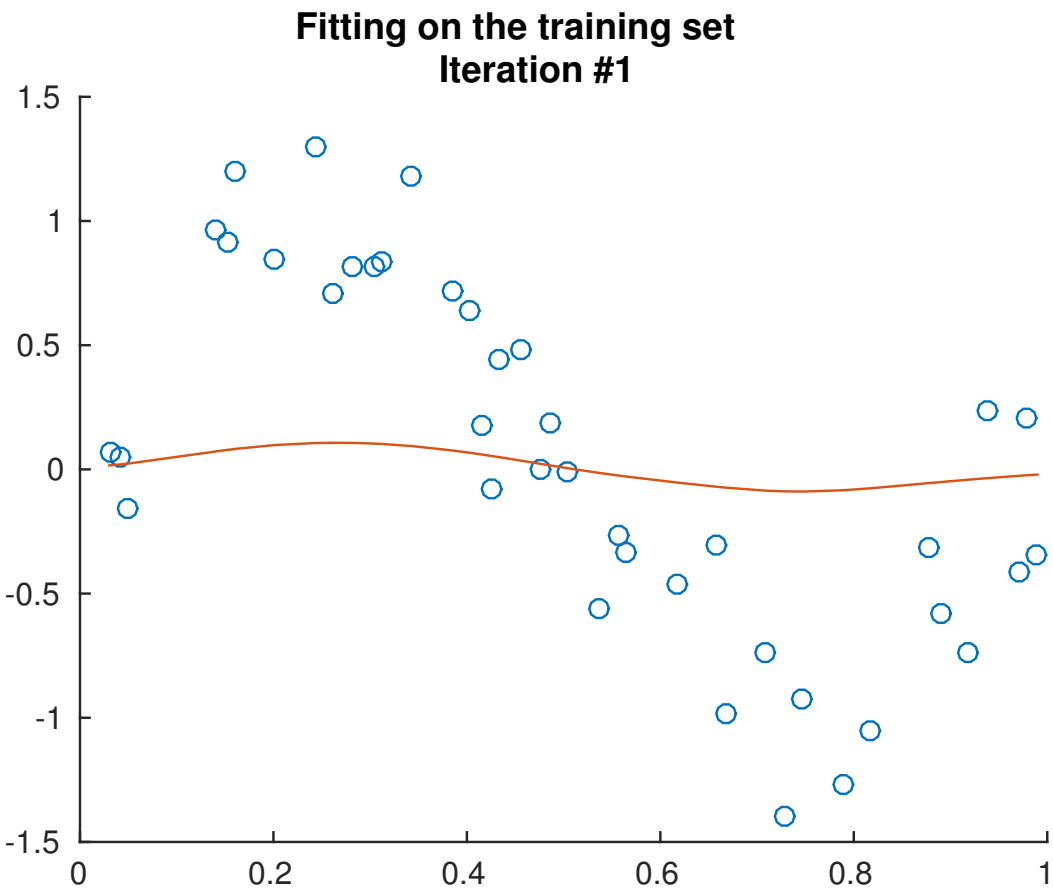
Gradient descent for square loss

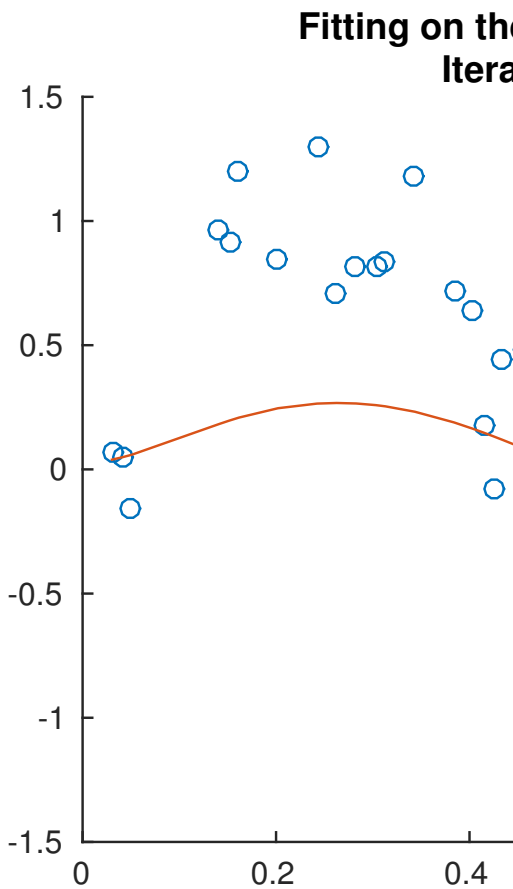
$$w_{t+1} = w_t - \gamma \hat{X}^\top (\hat{X} w_t - \hat{y})$$

$$\sum_{i=1}^{n} (y_i - w^\top x_i)^2 = \|\hat{X} w - \hat{y}\|^2$$

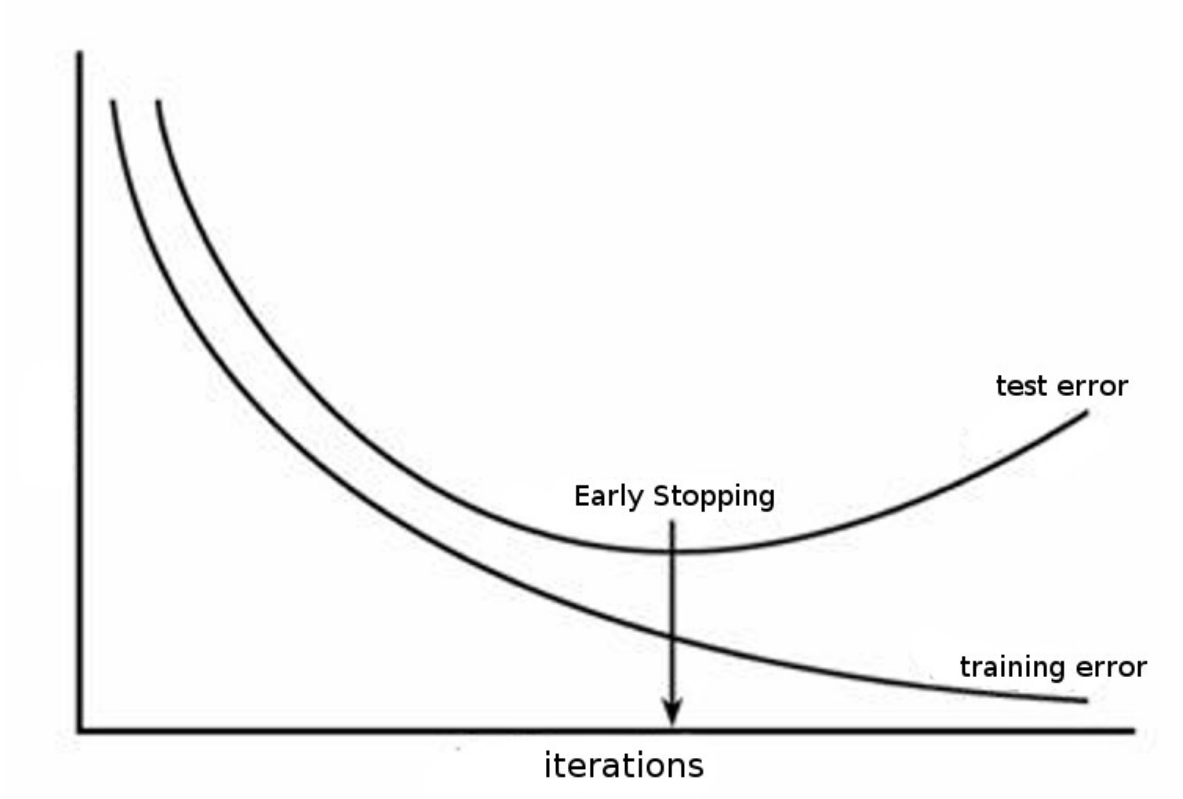
- no penalty
- stepsize chosen a priori $\gamma = \frac{2}{\|\hat{X}^\top \hat{X}\|}$

Early stopping at work





Semi-convergence



$$\min_w \mathcal{E}(w) \quad \text{vs} \quad \min_w \widehat{\mathcal{E}}(w)$$

Connection to Tikhonov

$$
\begin{aligned}
w_{t+1} &= w_t - \gamma \hat{X}^\top (\hat{X} w_t - \hat{y}) \\
&= (I - \gamma \hat{X}^\top \hat{X}) w_t + \gamma \hat{X}^\top \hat{y}
\end{aligned}
$$

by induction

$$
w_t = \underbrace{\gamma \sum_{j=0}^{t-1} (I - \gamma \hat{X}^\top \hat{X})^j}_{\text{Truncated power series}} \hat{X}^\top \hat{y}
$$

Neumann series

$$\gamma \sum_{j=0}^{t-1} (I - \gamma \hat{X}^\top \hat{X})^j$$

- $|a| < 1$

$$(1-a)^{-1} = \sum_{j=0}^{\infty} a^j \quad \Longrightarrow \quad a^{-1} = \sum_{j=0}^{\infty} (1-a)^j$$

- $A \in \mathbb{R}^{d \times d}$, $\|A\| < 1$, invertible

$$A^{-1} = \sum_{j=0}^{\infty} (I - A)^j$$

Stable matrix inversion

Truncated Neumann Series

$$(\hat{X}^\top \hat{X})^{-1} = \gamma \sum_{j=0}^{\infty} (I - \gamma \hat{X}^\top \hat{X})^j \quad \approx \quad \gamma \sum_{j=0}^{t-1} (I - \gamma \hat{X}^\top \hat{X})^j$$

compare to

$$(\hat{X}^\top \hat{X})^{-1} \quad \approx \quad (\hat{X}^\top \hat{X} + \lambda n I)^{-1}$$

Early-stopping: extensions

Early stopping regularization, so far analogous to

$$\min_w \frac{1}{n}\sum_{i=1}^{n}(w^T x_i - y_i)^2 + \lambda\|w\|^2$$

Extensions

Early stopping regularization analogous to

- $$\min_w \frac{1}{n}\sum_{i=1}^{n} V(w^T x_i, y_i) + \lambda\|w\|^2$$

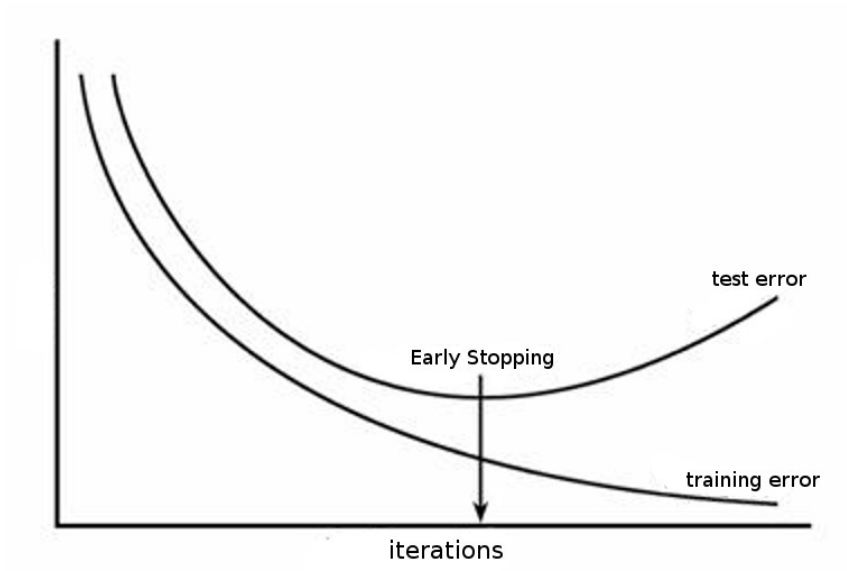
- or

$$\min_w \frac{1}{n}\sum_{i=1}^{n}(w^T x_i - y_i)^2 + \lambda R(w)$$

. . . or both.

Early-stopping why?

- Regularization path
- Warm-restart
- Computational regularization



Beyond Tikhonov: TSVD

$$\hat{X}^\top \hat{X} = V \Sigma V^\top, \qquad w_M = (\hat{X}^\top \hat{X})_M^{-1} \hat{X}^\top \hat{y}$$

- $(\hat{X}^\top \hat{X})_M^{-1} = V \Sigma_M^{-1} V^\top$
- $\Sigma_M^{-1} = \mathrm{diag}(\sigma_1^{-1}, \ldots, \sigma_M{}^{-1}, 0 \ldots, 0)$

Also known as principal component regression (PCR)

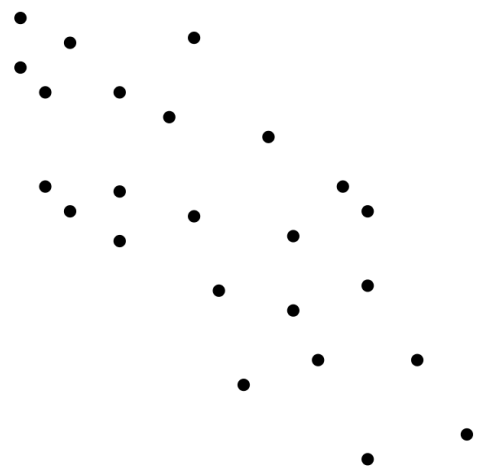
Principal component analysis (PCA)

Dimensionality reduction

$$\hat{X}^\top \hat{X} = V \Sigma V^\top$$

Eigenfunctions are directions, of

- maximum variance
- best reconstruction



TSVD and PCA

$$TSVD \quad \Leftrightarrow \quad PCA + ERM$$

Regularization by projection

TSVD/PCR beyond linearity

Non-linear function

$$f(x) = \sum_{i=1}^{p} w_i \phi_i(x) = \Phi(x)^\top w$$

with

$$w = (\widehat{\Phi}^\top \widehat{\Phi})_M^{-1} \widehat{\Phi}^\top \hat{y}$$

Let $\widehat{\Phi} = (\Phi(x_1), \dots \Phi(x_n))^\top \in \mathbb{R}^{n \times p}$.

$$\widehat{\Phi}^\top \widehat{\Phi} = V \Sigma V^\top, \quad (\widehat{\Phi}^\top \widehat{\Phi})_M^{-1} = V \Sigma_M^{-1} V^\top$$

$$\Sigma = \mathrm{diag}(\sigma_1, \dots, \sigma_p), \quad \Sigma_M^{-1} = \mathrm{diag}(\sigma_1^{-1}, \dots, \sigma_M^{-1}, 0, \dots)$$

TSVD/PCR with kernels

$$f(x) = \sum_{i=1}^{n} K(x, x_i) c_i, \qquad c = (\widehat{K})_M^{-1} \hat{y}$$

$$\widehat{K}_{ij} = K(x_i, x_j), \quad \widehat{K} = U \Sigma U^\top, \quad \Sigma = (\sigma_1, \dots, \sigma_n),$$

$$\widehat{K}_M^{-1} = U \Sigma_M^{-1} U^\top, \quad \Sigma_M^{-1} = (\sigma_1^{-1}, \dots, \sigma_M^{-1}, 0, \dots),$$

Complexity of nonparametric learning

- time: $O(n^3)$ or $O(n^2 T)$ or $O(n^2 M)$
- space: $O(n^2)$

Going big...

Bottleneck of non-linear learning with kernel methods

Memory

$$\widehat{K} \quad \text{is} \quad O(n^2)$$

An intuition

PCR/spectral filtering : first compute then discard.

Since we know we need only part of the information in the data:

Can we compute less?

Approaches to large scale

- (Random) features - find $\widetilde{\Phi} : X \to \mathbb{R}^M$, with $M \ll n$ s.t.

$$K(x, x') \approx \widetilde{\Phi}(x)^\top \widetilde{\Phi}(x')$$

- Subsampling (Nyström) - replace

$$f(x) = \sum_{i=1}^{n} K(x, x_i) c_i \quad \text{by} \quad f(x) = \sum_{i=1}^{M} K(x, \tilde{x}_i) c_i$$

$\tilde{x}_i$ subsampled from training set, M

Random features: Gaussian kernel

It holds (using Fourier transform),

$$K(x,x') = e^{-\|x-x'\|^2\gamma} = \int \underbrace{d\omega e^{-\omega^2 c}}_{dp(\omega)} e^{i\omega^T x} e^{-i\omega^T x'}.$$

Consider,

$$\tilde{K}(x,x') = \underbrace{\frac{1}{m}\sum_{j=1}^{M} e^{i\omega_j^T x} e^{-i\omega_j^T x'}}_{\widetilde{\Phi}(x)^\top \widetilde{\Phi}(x')}$$

with $\omega_1, \ldots, \omega_M$ i.i.d. samples w.r.t. to p.

Random features: Gaussian kernel (cont.)

Then,

$$e^{-\|x-x'\|^2\gamma} \approx \widetilde{\Phi}(x)^\top \widetilde{\Phi}(x')$$

with,

$$\widetilde{\Phi}(x) = (e^{i\omega_1^T x}, \ldots, e^{i\omega_M^T x}).$$

Alternatively consider

$$\widetilde{\Phi}(x) = (\cos(\omega_1^\top x + b_1), \ldots, \cos(\omega_M^\top x + b_M))$$

Other examples of random features

- **translation invariant** kernels $K(x, x') = H(x - x')$,

$$\widetilde{\Phi}(x)^j = e^{i\omega_j^\top x}, \qquad \omega_j \sim \pi = \mathcal{F}(H)$$

- infinite **neural nets** kernels

$$\widetilde{\Phi}(x)^j = |\omega_j^\top x + b_j|_+, \qquad (\omega_j, b_j) \sim \pi = U[\mathbb{S}^{d+1}]$$

- infinite **dot product** kernels
- homogeneous **additive** kernels
- **group invariant** kernels
- ...

Note: Connections with **hashing** and **sketching** techniques.

Learning with random features

Let

$$f(x) = w^\top \widetilde{\Phi}(x)$$

with coefficients solving

$$\min_{w \in \mathbb{R}^M} \frac{1}{n} \left\| \widetilde{\Phi}_n w - \hat{y} \right\|_n^2 + \lambda \|w\|^2 ,$$

$\widetilde{\Phi}_n$ n by M matrix with rows $\widetilde{\Phi}(x_i)$.

Complexity of learning with random features

$$\min_{w \in \mathbb{R}^M} \frac{1}{n} \left\| \widetilde{\Phi}_n w - \hat{y} \right\|_n^2 + \lambda \|w\|^2$$

$$\downarrow$$

$$\underbrace{(\widetilde{\Phi}_n^\top \widetilde{\Phi}_n + \lambda n I)}_{M \times M} w = \widetilde{\Phi}_n^\top \widehat{y}$$

Computations

- Time: ~~$O(n^3)$~~ $O(nM^2)$
- Space: ~~$O(n^2)$~~ $O(nM)$

RF as data independent subsampling

Consider,

$$f(x) = \sum_{j=1}^{d} \cos(\omega_j^\top x + b_j) w_j$$

or more generally,

$$f(x) = \sum_{j=1}^{d} q(x, \omega_j) w_j .$$

with

- w_j optimized
- ω_j randomized **independently of data**

What about data dependent sampling?

Nÿstrom methods

$$f(x) = \sum_{i=1}^{n} K(x, x_i)c_i \quad \mapsto \quad f(x) = \sum_{i=1}^{M} K(x, \tilde{x}_i)c_i$$

$\tilde{x}_i$ *centers* subsampled from training set M

Note: keep all data! (just use fewer to parameterize functions)

Nÿstrom ridge regression

$$\min_{c\in\mathbb{R}^M} \left\| \widehat{K}_{n,M} c - \widehat{y} \right\|_n^2 + \lambda c^\top \widehat{K}_{M,M} c$$

- $(\widehat{K}_{n,M})_{i,j} = K(\tilde{x}_i, x_j)$
- $(\widehat{K}_{M,M})_{i,j} = K(\tilde{x}_i, \tilde{x}_j)$

Complexity of Nÿstrom ridge regression

$$\min_{c\in\mathbb{R}^M} \frac{1}{n}\left\|\widehat{K}_{M,n}c-\hat{y}\right\|_n^2+\lambda c^\top \widehat{K}_{M,M}c$$

$$\downarrow$$

$$\underbrace{(K_{n,M}^\top K_{n,M}+\lambda n K_{M,M})}_{M\times M}c=K_{n,M}^\top \hat{y}$$

Computations

- Time: ~~$O(n^3)$~~ $O(nM^2)$
- Space: ~~$O(n^2)$~~ $O(nM)$

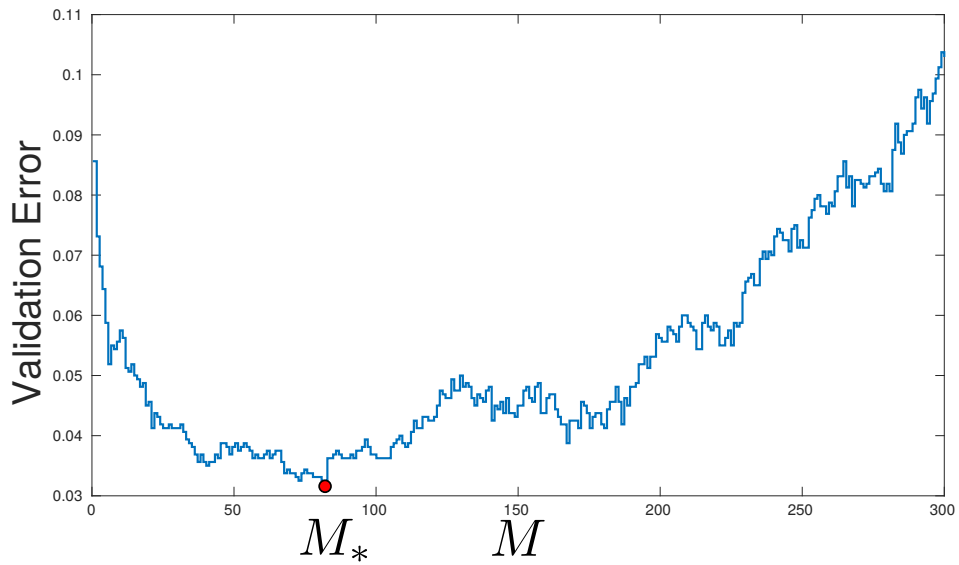
Subsampling and regularization

- Random features

$$f(x) = \sum_{i=1}^{M} q(x, \omega_i) w_i$$

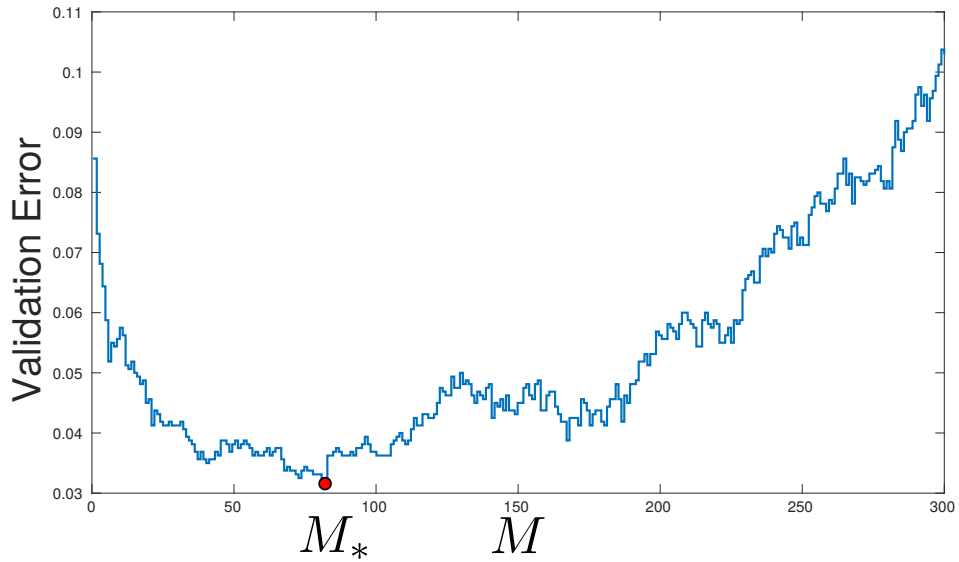
- Nÿstrom

$$f(x) = \sum_{i=1}^{M} K(x, \tilde{x}_i) c_i$$



Subsampling as stochastic regularization

The subsampling level M can be seen as a regularization parameter!



M controls: statistics, space and time complexity!

An incrementation approach

Algorithm

1. *Pick a center + compute solution*
2. *Pick another center +* **rank one update**
3. *Pick another center . . .*

Computational regularization

Computational regularization idea:

use computations to regularize

Iterative and subsampling regularization can be seen as instances.

Approaches to large scale non-linear learning

Consider,

$$f(x) = \sum_{j=1}^{d} Q(x, \omega_j) w_j .$$

with Q feature or kernel and

- w_j optimized,
- ω_j randomized.

Shallow nets

Consider,

$$f(x) = \sum_{j=1}^{d} Q(x, \omega_j) w_j .$$

This is a one layer neural net!

Neural nets

- w_j optimized
- ω_j ~~randomized~~ optimized

From shallow to deep nets

Shallow nets

$$f(x) = w^\top \Phi_W(x), \qquad \Phi_W(x) = Q(W^\top x)$$

Q activation function.

Deep nets

$$f(x) = w^\top \bar{\Phi}(x), \qquad \bar{\Phi} = \Phi_{W_L} \circ \cdots \circ \Phi_{W_1} \qquad \Phi_{W_j} = Q(W_j^\top x)$$

Deep nets

$$f(x) = w^\top \bar{\Phi}(x), \qquad \bar{\Phi} = \Phi_{W_L} \circ \cdots \circ \Phi_{W_1} \qquad \Phi_{W_j} = Q(W_j^\top x)$$

Neural nets

- w_j, W_j optimized,
- learning data representation (?)

This class

- early stopping
- projection regularization
- subsampling & regularization

Next class

- a practical experience


Frank Bauer, Sergei Pereverzev, and Lorenzo Rosasco.
On regularization algorithms in learning theory.
Journal of complexity, 23(1):52–72, 2007.


Raffaello Camoriano, Tomás Angles, Alessandro Rudi, and Lorenzo Rosasco.
Nytro: When subsampling meets early stopping.
In *Proceedings of the 19th International Conference on Artificial Intelligence and Statistics*, pages 1403–1411, 2016.


L Lo Gerfo, Lorenzo Rosasco, Francesca Odone, Ernesto De Vito, and Alessandro Verri.
Spectral algorithms for supervised learning.
Neural Computation, 20(7):1873–1897, 2008.


Junhong Lin, Lorenzo Rosasco, and Ding-Xuan Zhou.
Iterative regularization for learning with convex loss functions.
Journal of Machine Learning Research, 17(77):1–38, 2016.


Sofia Mosci, Lorenzo Rosasco, and Alessandro Verri.
Dimensionality reduction and generalization.
In *Proceedings of the 24th international conference on Machine learning*, pages 657–664. ACM, 2007.


Ali Rahimi and Benjamin Recht.
Random features for large-scale kernel machines.
In *Advances in neural information processing systems*, pages 1177–1184, 2007.


Alessandro Rudi, Raffaello Camoriano, and Lorenzo Rosasco.
Less is more: Nyström computational regularization.
In *Advances in Neural Information Processing Systems*, pages 1657–1665, 2015.


Alessandro Rudi, Raffaello Camoriano, and Lorenzo Rosasco.
Generalization properties of learning with random features.
arXiv preprint arXiv:1602.04474, 2016.


Alex J Smola and Bernhard Schölkopf.
Sparse greedy matrix approximation for machine learning.
2000.