Less is More:
Computational Regularization by Subsampling

Lorenzo Rosasco
University of Genova - Istituto Italiano di Tecnologia
Massachusetts Institute of Technology
lcsl.mit.edu

joint work with Alessandro Rudi, Raffaello Camoriano

Jan 13th, 2016
UCL, London
Classically:
Statistics and optimization distinct steps in algorithm design
A Starting Point

Classically:
Statistics and optimization **distinct steps** in algorithm design

Large Scale:
Consider **interplay** between statistics and optimization!
(Bottou, Bousquet ’08)
A Starting Point

Classically:
Statistics and optimization **distinct steps** in algorithm design

Large Scale:
Consider **interplay** between statistics and optimization!
(Bottou, Bousquet '08)

Computational Regularization:
Computation “tricks” = regularization
Supervised Learning

**Problem:** Estimate $f^*$
Supervised Learning

**Problem:** Estimate $f^*$ given $S_n = \{(x_1, y_1), \ldots, (x_n, y_n)\}$
Supervised Learning

**Problem:** Estimate $f^*$ given $S_n = \{(x_1, y_1), \ldots, (x_n, y_n)\}$

**The Setting**

$$y_i = f^*(x_i) + \varepsilon_i \quad i \in \{1, \ldots, n\}$$

- $\varepsilon_i \in \mathbb{R}, x_i \in \mathbb{R}^d$ random (with unknown distribution)
- $f^*$ unknown
Outline

Learning with dictionaries and kernels

Data Dependent Subsampling

Data Independent Subsampling
Non-linear/non-parametric learning

$$\hat{f}(x) = \sum_{i=1}^{M} c_i q(x, w_i)$$
Non-linear/non-parametric learning

\[ \hat{f}(x) = \sum_{i=1}^{M} c_i q(x, w_i) \]

▶ *q* non linear function
Non-linear/non-parametric learning

\[ \hat{f}(x) = \sum_{i=1}^{M} c_i q(x, w_i) \]

- \( q \) non linear function
- \( w_i \in \mathbb{R}^d \) centers

Question: How to choose \( w_i, c_i \) and \( M \) given \( S_n \)?
Non-linear/non-parametric learning

\[ \hat{f}(x) = \sum_{i=1}^{M} c_i q(x, w_i) \]

- \( q \) non linear function
- \( w_i \in \mathbb{R}^d \) centers
- \( c_i \in \mathbb{R} \) coefficients
Non-linear/non-parametric learning

\[ \hat{f}(x) = \sum_{i=1}^{M} c_i q(x, w_i) \]

- \( q \) non linear function
- \( w_i \in \mathbb{R}^d \) centers
- \( c_i \in \mathbb{R} \) coefficients
- \( M = M_n \) could/should grow with \( n \)
Non-linear/non-parametric learning

\[ \hat{f}(x) = \sum_{i=1}^{M} c_i q(x, w_i) \]

- \( q \) non linear function
- \( w_i \in \mathbb{R}^d \) centers
- \( c_i \in \mathbb{R} \) coefficients
- \( M = M_n \) could/should grow with \( n \)

**Question:** How to choose \( w_i, c_i \) and \( M \) given \( S_n \)?
There is an *elegant* answer if:

- $q$ is **symmetric**
- *all* the matrices $\hat{Q}_{ij} = q(x_i, x_j)$ are **positive semi-definite**\(^1\)

\(^1\)They have non-negative eigenvalues
There is an *elegant* answer if:

- $q$ is **symmetric**
- *all* the matrices $\hat{Q}_{ij} = q(x_i, x_j)$ are **positive semi-definite**\(^1\)

**Representer Theorem** (Kimeldorf, Wahba '70; Schölkopf et al. '01)

- $M = n,$
- $w_i = x_i,$
- $c_i$ by **convex** optimization!

---

\(^1\)They have non-negative eigenvalues
Kernel Ridge Regression (KRR)

a.k.a. Penalized Least Squares

\[ \hat{f}_\lambda = \operatorname{argmin}_{f \in \mathcal{H}} \frac{1}{n} \sum_{i=1}^{n} (y_i - f(x_i))^2 + \lambda \|f\|^2 \]
Kernel Ridge Regression (KRR)

a.k.a. Penalized Least Squares

\[ \hat{f}_\lambda = \arg\min_{f \in \mathcal{H}} \frac{1}{n} \sum_{i=1}^{n} (y_i - f(x_i))^2 + \lambda \|f\|^2 \]

where

\[ \mathcal{H} = \{ f \mid f(x) = \sum_{i=1}^{M} c_i q(x, w_i), \ c_i \in \mathbb{R}, w_i \in \mathbb{R}^d, \ M \in \mathbb{N} \} \]

any center! any length!
Kernel Ridge Regression (KRR)

a.k.a. Penalized Least Squares

\[ \hat{f}_\lambda = \arg\min_{f \in \mathcal{H}} \frac{1}{n} \sum_{i=1}^{n} (y_i - f(x_i))^2 + \lambda \|f\|^2 \]

where

\[ \mathcal{H} = \{ f \mid f(x) = \sum_{i=1}^{M} c_i q(x, w_i), c_i \in \mathbb{R}, w_i \in \mathbb{R}^d, \quad M \in \mathbb{N} \} \]

any center! any length!

Solution

\[ \hat{f}_\lambda = \sum_{i=1}^{n} c_i q(x, x_i) \quad \text{with} \quad c = (\hat{Q} + \lambda n I)^{-1} \hat{y} \]
KRR: Statistics

Well understood statistical properties:

Classical Theorem

If \( f^* \in H \), then

\[
\lambda^* = \frac{1}{\sqrt{n}} E \left( \hat{f}_{\lambda^*}(x) - f^*(x) \right)^2 \lesssim \frac{1}{\sqrt{n}}
\]

Remarks

1. Optimal nonparametric bound
2. Results for general kernels (e.g. splines/Sobolev etc.)

\[
\lambda^* = \frac{n - 1}{2}s + 1
\]

\[
E \left( \hat{f}_{\lambda^*}(x) - f^*(x) \right)^2 \lesssim \frac{n - 2}{s^2s + 1}
\]

3. Adaptive tuning via cross validation
4. Proofs: analysis/linear algebra + random matrices (Smale and Zhou + Caponnetto, De Vito, R.+ Steinwart)
KRR: Statistics

Well understood statistical properties:

Classical Theorem
If $f^* \in \mathcal{H}$, then

$$\lambda_* = \frac{1}{\sqrt{n}}$$

$$\mathbb{E} (\hat{f}_{\lambda_*}(x) - f^*(x))^2 \lesssim \frac{1}{\sqrt{n}}$$
Well understood statistical properties:

**Classical Theorem**
If \( f^* \in \mathcal{H} \), then

\[
\lambda_* = \frac{1}{\sqrt{n}} \quad \mathbb{E} \left( \hat{f}_{\lambda_*}(x) - f^*(x) \right)^2 \preccurlyeq \frac{1}{\sqrt{n}}
\]

**Remarks**
1. Optimal nonparametric bound
2. Results for general kernels (e.g. splines/Sobolev etc.)
3. Adaptive tuning via cross validation
4. Proofs: analysis/linear algebra + random matrices (Smale and Zhou + Caponnetto, De Vito, R. + Steinwart)
Well understood statistical properties:

**Classical Theorem**
If \( f^* \in H \), then

\[
\lambda_* = \frac{1}{\sqrt{n}} \quad \mathbb{E} \left( \hat{f}_{\lambda_*}(x) - f^*(x) \right)^2 \lesssim \frac{1}{\sqrt{n}}
\]

Remarks

1. Optimal nonparametric bound
KRR: Statistics

Well understood statistical properties:

Classical Theorem
If $f^* \in \mathcal{H}$, then

$$\lambda_* = \frac{1}{\sqrt{n}} \quad \mathbb{E} (\hat{f}_{\lambda_*}(x) - f^*(x))^2 \lesssim \frac{1}{\sqrt{n}}$$

Remarks

1. Optimal nonparametric bound
2. Results for general kernels (e.g. splines/Sobolev etc.)

$$\lambda_* = n^{-\frac{1}{2s+1}}, \quad \mathbb{E} (\hat{f}_{\lambda_*}(x) - f^*(x))^2 \lesssim n^{-\frac{2s}{2s+1}}$$
KRR: Statistics

Well understood statistical properties:

Classical Theorem
If \( f^* \in \mathcal{H} \), then

\[
\lambda_* = \frac{1}{\sqrt{n}} \quad \mathbb{E} \left( \hat{f}_{\lambda_*}(x) - f^*(x) \right)^2 \lesssim \frac{1}{\sqrt{n}}
\]

Remarks

1. **Optimal nonparametric bound**
2. Results for **general** kernels (e.g. splines/Sobolev etc.)

\[
\lambda_* = n^{\frac{1}{2s+1}} \quad \mathbb{E} \left( \hat{f}_{\lambda_*}(x) - f^*(x) \right)^2 \lesssim n^{-\frac{2s}{2s+1}}
\]

3. **Adaptive** tuning via cross validation
4. **Proofs**: analysis/linear algebra + random matrices
   (Smale and Zhou + Caponnetto, De Vito, R. + Steinwart)
KRR: Optimization

\[ \hat{f}_\lambda = \sum_{i=1}^{n} c_i q(x, x_i) \quad \text{with} \quad c = (\hat{Q} + \lambda n I)^{-1} \hat{y} \]

Linear System

\[ \hat{Q} \quad C = \hat{y} \]

Complexity

- **Space** \( O(n^2) \)
- **Time** \( O(n^3) \)
KRR: Optimization

\[ \hat{f}_\lambda = \sum_{i=1}^{n} c_i q(x, x_i) \quad \text{with} \quad c = (\hat{Q} + \lambda n I)^{-1} \hat{y} \]

Linear System

- \[ \hat{Q} \]
- \[ C \]
- \[ \hat{y} \]

Complexity
- **Space** \( O(n^2) \)
- **Time** \( O(n^3) \)

BIG DATA?
Running out of time and space ...  

Can this be fixed?
Beyond Tikhonov: Spectral Filtering

\((\hat{Q} + \lambda I)^{-1}\) approximation of \(\hat{Q}^\dagger\) controlled by \(\lambda\)
Beyond Tikhonov: Spectral Filtering

\[(\hat{Q} + \lambda I)^{-1}\] approximation of \(\hat{Q}^\dagger\) controlled by \(\lambda\)

Can we approximate \(\hat{Q}^\dagger\) by saving computations?
Beyond Tikhonov: Spectral Filtering

\[(\hat{Q} + \lambda I)^{-1}\] approximation of \(\hat{Q}^\dagger\) controlled by \(\lambda\)

Can we approximate \(\hat{Q}^\dagger\) by saving computations?

Yes!
Beyond Tikhonov: Spectral Filtering

\[(\hat{Q} + \lambda I)^{-1}\] approximation of \(\hat{Q}^\dagger\) controlled by \(\lambda\)

Can we approximate \(\hat{Q}^\dagger\) by saving computations?

Yes!

Spectral filtering (Engl ’96- inverse problems, Rosasco et al. 05- ML)

\[g_\lambda(\hat{Q}) \sim \hat{Q}^\dagger\]

The filter function \(g_\lambda\) defines the form of the approximation
Spectral filtering

Examples

- Tikhonov- ridge regression
- Truncated SVD– principal component regression
- Landweber iteration– GD/ $L_2$-boosting
- nu-method– accelerated GD/Chebyshev method
- ...

Landweber iteration (truncated power series).

$c_t = g_t(\hat{Q}) = 1 - \gamma (\hat{Q}_t) := c_t_{t-1} - \gamma (\hat{Q}_{t-1} - \hat{y})$, $c_0 = 0$.
Spectral filtering

Examples

- Tikhonov-ridge regression
- Truncated SVD– principal component regression
- Landweber iteration– GD/$L_2$-boosting
- nu-method– accelerated GD/Chebyshev method
- ... 

Landweber iteration (truncated power series)...

\[ c_t = g_t(Q) = \gamma \sum_{r=0}^{t-1} (I - \gamma Q)^r \hat{y} \]
Spectral filtering

Examples

- Tikhonov-ridge regression
- Truncated SVD–principal component regression
- Landweber iteration–GD/$L_2$-boosting
- nu-method–accelerated GD/Chebyshev method
- ...

Landweber iteration (truncated power series)...

$$c_t = g_t(\hat{Q}) = \gamma \sum_{r=0}^{t-1} (I - \gamma \hat{Q})^r \hat{y}$$

...it's GD for ERM!!

$$r = 1 \ldots t \quad c_r = c_{r-1} - \gamma (\hat{Q} c_{r-1} - \hat{y}), \quad c_0 = 0$$
Early Stopping at Work

Fitting on the training set
Iteration #2
Early Stopping at Work

Fitting on the training set
Iteration #7
Early Stopping at Work

Fitting on the training set
Iteration #5000
The different filters achieve *essentially the same* optimal statistical error!
Statistics and computations with spectral filtering

The different filters achieve essentially the same optimal statistical error!

Difference is in computations

<table>
<thead>
<tr>
<th>Filter</th>
<th>Time</th>
<th>Space</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tikhonov</td>
<td>$n^3$</td>
<td>$n^2$</td>
</tr>
<tr>
<td>GD</td>
<td>$n^2 \lambda_*^{-1}$</td>
<td>$n^2$</td>
</tr>
<tr>
<td>Accelerated GD</td>
<td>$n^2 \lambda_*^{-1/2}$</td>
<td>$n^2$</td>
</tr>
<tr>
<td>Truncated SVD</td>
<td>$n^2 \lambda_*^{-\gamma}$</td>
<td>$n^2$</td>
</tr>
</tbody>
</table>
Computational Regularization Arises

- Computational regularization: iterations control statistics and time complexity
Computational Regularization Arises

- **Computational regularization**: iterations control statistics and time complexity

- Built-in **regularization path**
Computational Regularization Arises

- **Computational regularization**: iterations control statistics and time complexity

- **Built-in regularization path**

- Is there an advantage going for on-line learning?
Computational Regularization Arises

- **Computational regularization**: iterations control statistics **and** time complexity

- **Built-in regularization path**

- Is there an advantage going for **on-line learning**? Not much, maybe $n^2 \log n$? (Bach Dieleveet ’15 – R. Villa ’15)
Computational Regularization Arises

- **Computational regularization**: iterations control statistics and time complexity

- **Built-in regularization path**

- Is there an advantage going for on-line learning? Not much, maybe $n^2 \log n$? (Bach Dieluevet ’15 – R. Villa ’15)

- **Computational regularization**: principles to control statistics, time and space complexity
Outline

Learning with dictionaries and kernels

Data Dependent Subsampling

Data Independent Subsampling
Subsampling

1. pick \( w_i \) at random...
Subsampling

1. pick $w_i$ at random... from training set
   (Smola, Scholköpf '00)

   $\tilde{w}_1, \ldots, \tilde{w}_M \subset x_1, \ldots x_n$  \quad M \ll n
Subsampling

1. pick $w_i$ at random... from training set
   (Smola, Scholköpf '00)

   $\tilde{w}_1, \ldots, \tilde{w}_M \subset x_1, \ldots x_n \quad M \ll n$

2. perform KRR on

   $\mathcal{H}_M = \{ f \mid f(x) = \sum_{i=1}^{M} c_i q(x, \tilde{w}_i), \ c_i \in \mathbb{R}, \ w_i \in \mathbb{R}^d, \ M \in \mathbb{N} \}$.
Subsampling

1. pick \( w_i \) at random... from training set
   (Smola, Schölkopf '00)

\[ \tilde{w}_1, \ldots, \tilde{w}_M \subset x_1, \ldots x_n \quad M \ll n \]

2. perform KRR on

\[ \mathcal{H}_M = \{ f \mid f(x) = \sum_{i=1}^{M} c_i q(x, \tilde{w}_i), \ c_i \in \mathbb{R}, \ w_i \in \mathbb{R}^d, \ M \in \mathbb{N} \} \]

Linear System

\[ \hat{Q}_M \begin{bmatrix} c \end{bmatrix} = \hat{y} \]

Complexity

- **Space** \( O(n^2) \rightarrow O(nM) \)
- **Time** \( O(n^3) \rightarrow O(nM^2) \)
Subsampling

1. pick \( w_i \) at random... from training set
   (Smola, Scholköpf '00)

\[
\tilde{w}_1, \ldots, \tilde{w}_M \subset x_1, \ldots x_n \quad M \ll n
\]

2. perform KRR on

\[
\mathcal{H}_M = \{ f \mid f(x) = \sum_{i=1}^{M} c_i q(x, \tilde{w}_i), \; c_i \in \mathbb{R}, \; w_i \in \mathbb{R}^d, \; M \in \mathbb{N} \}.
\]

Linear System

\[
\hat{Q}_M \begin{bmatrix} C \end{bmatrix} = \hat{y}
\]

Complexity

- **Space** \( O(n^2) \rightarrow O(nM) \)
- **Time** \( O(n^3) \rightarrow O(nM^2) \)

What about statistics? What's the price for efficient computations?
Putting our Result in Context

- *Many* different subsampling schemes
  (Smola, Scholkopf '00; Williams, Seeger '01; ...20+)

Theoretical guarantees mainly on matrix approximation
\[ \| \hat{Q} - \hat{Q}_M \| \lesssim \frac{1}{\sqrt{M}} \]

Few prediction guarantees either suboptimal or in restricted setting
(Cortes et al. '10; Jin et al. '11, Bach '13, Alaoui, Mahoney '14)
Putting our Result in Context

- *Many* different subsampling schemes
  (Smola, Scholkopf '00; Williams, Seeger '01; ... 20+)

- Theoretical guarantees mainly on matrix approximation
  (Mahoney and Drineas '09; Cortes et al. '10, Kumar et al.'12 ... 10+)

\[ \| \hat{Q} - \hat{Q}_M \| \lesssim \frac{1}{\sqrt{M}} \]
Putting our Result in Context

▶ *Many* **different subsampling** schemes  
(Smola, Scholkopf '00; Williams, Seeger '01; ... 20+)

▶ **Theoretical guarantees** mainly on **matrix approximation**  
(Mahoney and Drineas '09; Cortes et al '10, Kumar et al.'12 ... 10+)

\[ \| \hat{Q} - \hat{Q}_M \| \lesssim \frac{1}{\sqrt{M}} \]

▶ Few prediction guarantees either **suboptimal** or in **restricted setting**  
(Cortes et al. '10; Jin et al. '11, Bach '13, Alaoui, Mahoney '14)
Main Result

(Rudi, Camoriano, Rosasco, '15)

Theorem

If \( f^* \in \mathcal{H} \), then

\[
\lambda_* = \frac{1}{\sqrt{n}}, \quad M_* = \frac{1}{\lambda_*}, \quad \mathbb{E} \left( \hat{f}_{\lambda_*, M_*}(x) - f^*(x) \right)^2 \lesssim \frac{1}{\sqrt{n}}
\]
Main Result

(Rudi, Camoriano, Rosasco, '15)

Theorem

If $f^* \in \mathcal{H}$, then

$$
\lambda_* = \frac{1}{\sqrt{n}}, \quad M_* = \frac{1}{\lambda_*}, \quad \mathbb{E} \left( \hat{f}_{\lambda_*, M_*}(x) - f^*(x) \right)^2 \lesssim \frac{1}{\sqrt{n}}
$$

Remarks

1. Subsampling achieves optimal bound.
2. . . . with $M_* \sim \sqrt{n}$
3. More generally, $\lambda_* = \frac{n-1}{2s+1}, \quad M_* = \frac{1}{\lambda_*}, \quad \mathbb{E} \left( \hat{f}_{\lambda_*, M_*}(x) - f^*(x) \right)^2 \lesssim \frac{1}{n-2s}$
Main Result

(Rudi, Camoriano, Rosasco, ’15)

Theorem
If $f^* \in \mathcal{H}$, then

$$\lambda_* = \frac{1}{\sqrt{n}}, \quad M_* = \frac{1}{\lambda_*}, \quad \mathbb{E} \left( \hat{f}_{\lambda_*,M_*}(x) - f^*(x) \right)^2 \lesssim \frac{1}{\sqrt{n}}$$

Remarks
1. Subsampling achieves **optimal** bound...
Main Result

(Rudi, Camoriano, Rosasco, ’15)

Theorem

If $f^* \in \mathcal{H}$, then

$$\lambda_* = \frac{1}{\sqrt{n}}, \quad M_* = \frac{1}{\lambda_*}, \quad \mathbb{E} (\hat{f}_{\lambda_*} M_* (x) - f^*(x))^2 \lesssim \frac{1}{\sqrt{n}}$$

Remarks

1. Subsampling achives **optimal** bound.
2. . . . with $M_* \sim \sqrt{n}$ !!
Main Result

(Rudi, Camoriano, Rosasco, '15)

Theorem

If \( f^* \in \mathcal{H} \), then

\[
\lambda_* = \frac{1}{\sqrt{n}} \quad , \quad M_* = \frac{1}{\lambda_*} \quad , \quad \mathbb{E} \left( \hat{f}_{\lambda_*,M_*}(x) - f^*(x) \right)^2 \lesssim \frac{1}{\sqrt{n}}
\]

Remarks

1. Subsampling achieves optimal bound...
2. ... with \( M_* \sim \sqrt{n} \) !!
3. More generally,

\[
\lambda_* = n^{-\frac{1}{2s+1}} \quad , \quad M_* = \frac{1}{\lambda_*} \quad , \quad \mathbb{E}_x \left( \hat{f}_{\lambda_*,M_*}(x) - f^*(x) \right)^2 \lesssim n^{-\frac{2s}{2s+1}}
\]

Note: An interesting insight is obtained rewriting the result...
Main Result

(Rudi, Camoriano, Rosasco, '15)

**Theorem**

*If* \( f^* \in \mathcal{H} \), *then*

\[
\lambda_* = \frac{1}{\sqrt{n}} , \quad M_* = \frac{1}{\lambda_*}, \quad \mathbb{E} (\hat{f}_{\lambda_*M_*}(x) - f^*(x))^2 \lesssim \frac{1}{\sqrt{n}}
\]

**Remarks**

1. Subsampling achieves **optimal** bound
2. . . . with \( M_* \sim \sqrt{n} \) !!
3. More generally,

\[
\lambda_* = n^{-\frac{1}{2s+1}} , \quad M_* = \frac{1}{\lambda_*}, \quad \mathbb{E}_x (\hat{f}_{\lambda_*M_*}(x) - f^*(x))^2 \lesssim n^{-\frac{2s}{2s+1}}
\]

**Note:** An interesting insight is obtained rewriting the result
A simple idea: “swap” the role of $\lambda$ and $M$...
Computational Regularization by Subsampling

(Rudi, Camoriano, Rosasco, '15)

A simple idea: “swap” the role of $\lambda$ and $M$.

Theorem
If $f^* \in \mathcal{H}$, then

$$M_* = n^{\frac{1}{2s+1}}, \quad \lambda_* = \frac{1}{M_*}, \quad \mathbb{E}_x (\hat{f}_{\lambda_*,M_*}(x) - f^*(x))^2 \lesssim n^{-\frac{2s}{2s+1}}$$
Computational Regularization by Subsampling

(Rudi, Camoriano, Rosasco, '15)

A simple idea: “swap” the role of $\lambda$ and $M$...

Theorem
If $f^* \in \mathcal{H}$, then

$$M_* = n^{\frac{1}{2s+1}}, \quad \lambda_* = \frac{1}{M_*}, \quad \mathbb{E}_x \left( \hat{f}_{\lambda_*,M_*}(x) - f^*(x) \right)^2 \lesssim n^{-\frac{2s}{2s+1}}$$

- $\lambda$ and $M$ play the same role...
  ...new interpretation: **subsampling regularizes**!
Computational Regularization by Subsampling
(Rudi, Camoriano, Rosasco, '15)

A simple idea: “swap” the role of $\lambda$ and $M$ . . .

Theorem
If $f^* \in \mathcal{H}$, then

$$M_* = n \frac{1}{2s+1}, \quad \lambda_* = \frac{1}{M_*}, \quad \mathbb{E}_x (f_{\lambda_*, M_*}(x) - f^*(x))^2 \lesssim n^{-\frac{2s}{2s+1}}$$

- $\lambda$ and $M$ play the same role . . .
  - . . . new interpretation: subsampling regularizes!
- New natural incremental algorithm...

Algorithm
Computational Regularization by Subsampling

(Rudi, Camoriano, Rosasco, '15)

A simple idea: “swap” the role of $\lambda$ and $M$.

**Theorem**

If $f^* \in \mathcal{H}$, then

$$M^*_* = n^{\frac{1}{2s+1}}, \quad \lambda^*_* = \frac{1}{M^*_*}, \quad \mathbb{E}_{x} (\hat{f}_{\lambda^*_*,M^*_*}(x) - f^*(x))^2 \lesssim n^{-\frac{2s}{2s+1}}$$

- $\lambda$ and $M$ play the same role.
- New interpretation: **subsampling regularizes**!
- New natural **incremental** algorithm...

**Algorithm**

1. *Pick a center + compute solution*
Computational Regularization by Subsampling  
(Rudi, Camoriano, Rosasco, ’15)

A simple idea: “swap” the role of $\lambda$ and $M$ . . .

Theorem
If $f^* \in \mathcal{H}$, then

$$M_* = n^{\frac{1}{2s+1}}, \quad \lambda_* = \frac{1}{M_*}, \quad \mathbb{E}_x (\hat{f}_{\lambda_*,M_*}(x) - f^*(x))^2 \lesssim n^{-\frac{2s}{2s+1}}$$

- $\lambda$ and $M$ play the same role . . .
  - . . . new interpretation: subsampling regularizes!
- New natural incremental algorithm . . .

Algorithm
1. Pick a center + compute solution
2. Pick another center + rank one update
Computational Regularization by Subsampling

(Rudi, Camoriano, Rosasco, '15)

A simple idea: “swap” the role of $\lambda$ and $M$.

Theorem
If $f^* \in \mathcal{H}$, then

$$M_* = n^{\frac{1}{2s+1}}, \quad \lambda_* = \frac{1}{M_*}, \quad \mathbb{E}_x \left( \hat{f}_{\lambda_*,M_*}(x) - f^*(x) \right)^2 \lesssim n^{-\frac{2s}{2s+1}}$$

- $\lambda$ and $M$ play the same role.
- New interpretation: subsampling regularizes!
- New natural incremental algorithm...

Algorithm
1. Pick a center + compute solution
2. Pick another center + rank one update
3. Pick another center ...
CoRe Illustrated

\[ n, \lambda \text{ are fixed} \]

Computation controls stability!

Time/space requirement tailored to \textit{generalization}
## Experiments

comparable/better w.r.t. the state of the art

<table>
<thead>
<tr>
<th>Dataset</th>
<th>$n_{tr}$</th>
<th>$d$</th>
<th>Incremental CoRe $\pm 10^{-5}$</th>
<th>Standard KRLS</th>
<th>Standard Nyström</th>
<th>Random Features</th>
<th>Fastfood RF</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ins. Co.</td>
<td>5822</td>
<td>85</td>
<td>$0.23180 \pm 4 \times 10^{-5}$</td>
<td>0.231</td>
<td>0.232</td>
<td>0.266</td>
<td>0.264</td>
</tr>
<tr>
<td>CPU</td>
<td>6554</td>
<td>21</td>
<td>$2.8466 \pm 0.0497$</td>
<td>7.271</td>
<td>6.758</td>
<td>7.103</td>
<td>7.366</td>
</tr>
<tr>
<td>CT slices</td>
<td>42800</td>
<td>384</td>
<td>$7.1106 \pm 0.0772$</td>
<td>NA</td>
<td>60.683</td>
<td>49.491</td>
<td>43.858</td>
</tr>
<tr>
<td>Year Pred.</td>
<td>463715</td>
<td>90</td>
<td>$0.10470 \pm 5 \times 10^{-5}$</td>
<td>NA</td>
<td>0.113</td>
<td>0.123</td>
<td>0.115</td>
</tr>
<tr>
<td>Forest</td>
<td>522910</td>
<td>54</td>
<td>$0.9638 \pm 0.0186$</td>
<td>NA</td>
<td>0.837</td>
<td>0.840</td>
<td>0.840</td>
</tr>
</tbody>
</table>

- Random Features (Rahimi, Recht '07)
- Fastfood (Le et al. '13)
Summary so far

- **Optimal** learning with data dependent subsampling
- Computational regularization: subsampling *regularizes*!
Summary so far

- **Optimal** learning with data dependent subsampling
- Computational regularization: subsampling *regularizes*

Few more questions:

- Can one do better than *uniform* sampling?
- What about *data independent* sampling?
(Approximate) Leverage scores

Leverage scores

\[ l_i(t) = (\hat{Q}(\hat{Q} + tnI)^{-1})_{ii} \]
(Approximate) Leverage scores

Leverage scores

\[ l_i(t) = (\hat{Q}(\hat{Q} + tnI)^{-1})_{ii} \]

ALS

With probability at least \( 1 - \delta \),

\[ \frac{1}{T}l_i(t) \leq \tilde{l}_i(t) \leq Tl_i(t) \]
(Approximate) Leverage scores

Leverage scores

\[ l_i(t) = \left( \hat{Q}(\hat{Q} + tnI)^{-1} \right)_{ii} \]

ALS

With probability at least \( 1 - \delta \),

\[ \frac{1}{T} l_i(t) \leq \tilde{l}_i(t) \leq T l_i(t) \]

ALS subsampling

Pick \( \tilde{w}_1, \ldots, \tilde{w}_M \subset x_1, \ldots x_n \) with replacement, and probability

\[ P_t(i) = \tilde{l}_i(t) / \sum_j \tilde{l}_j(t). \]
Theorem

If $f^* \in \mathcal{H}$ and the integral operator associated to $q$ has eigenvalue decay $i^{-\frac{1}{\gamma}}$, $\gamma \in (0, 1)$ then

$$\lambda_* = n^{-\frac{1}{1+\gamma}}, \quad M_* = \frac{1}{\lambda_*^{\gamma}}, \quad \mathbb{E} (\hat{f}_{\lambda_*,M_*}(x) - f^*(x))^2 \lesssim n^{-\frac{1}{1+\gamma}}$$
Theorem
If $f_\star \in \mathcal{H}$ and the integral operator associated to $q$ has eigenvalue decay $i^{-\frac{1}{\gamma}}$, $\gamma \in (0, 1)$ then

$$\lambda_\star = n^{-\frac{1}{1+\gamma}}, \quad M_\star = \frac{1}{\lambda_\star^\gamma}, \quad \mathbb{E} \left( \hat{f}_{\lambda_\star, M_\star}(x) - f_\star(x) \right)^2 \lesssim n^{-\frac{1}{1+\gamma}}$$

- Non uniform subsampling achieves optimal bound
Theorem

If $f_* \in \mathcal{H}$ and the integral operator associated to $q$ has eigenvalue decay $\tilde{i}^{-\frac{1}{\gamma}}$, $\gamma \in (0, 1)$ then

$$\lambda_* = n^{-\frac{1}{1+\gamma}}, \quad M_* = \frac{1}{\lambda_*^\gamma}, \quad \mathbb{E} (\hat{f}_{\lambda_*, M_*}(x) - f^*(x))^2 \lesssim n^{-\frac{1}{1+\gamma}}$$

- Non uniform subsampling achieves \textbf{optimal} bound
- Most importantly: potentially \textbf{much fewer} samples needed
Learning with ALS Subsampling

(Rudi, Camoriano, Rosasco, ’15)

Theorem

If \( f_* \in \mathcal{H} \) and the integral operator associated to \( q \) has eigenvalue decay \( i^{\frac{-1}{\gamma}} \), \( \gamma \in (0, 1) \) then

\[
\lambda_* = n^{-\frac{1}{1+\gamma}}, \quad M_* = \frac{1}{\lambda_*^\gamma}, \quad \mathbb{E} (\hat{f}_{\lambda_* M_*}(x) - f^*(x))^2 \lesssim n^{-\frac{1}{1+\gamma}}
\]

- Non uniform subsampling achieves **optimal** bound
- Most importantly: potentially **much fewer** samples needed
- Need **efficient** ALS computation (…)
Learning with ALS Subsampling

(Rudi, Camoriano, Rosasco, '15)

Theorem

If $f_* \in \mathcal{H}$ and the integral operator associated to $q$ has eigenvalue decay $i^{-\frac{1}{\gamma}}$, $\gamma \in (0, 1)$ then

$$\lambda_* = n^{-\frac{1}{1+\gamma}}, \quad M_* = \frac{1}{\lambda_*^{\gamma}}, \quad \mathbb{E} \left( \hat{f}_{\lambda_* M_*}(x) - f^*(x) \right)^2 \lesssim n^{-\frac{1}{1+\gamma}}$$

- Non uniform subsampling achieves **optimal** bound
- Most importantly: potentially **much fewer** samples needed
- Need **efficient** ALS computation (...)
- **Extensions** to more general sampling schemes are possible
Outline

Learning with dictionaries and kernels

Data Dependent Subsampling

Data Independent Subsampling
Random Features

\[ \hat{f}(x) = \sum_{i=1}^{M} c_i q(x, \tilde{w}_i) \]

\( q \) general non-linear function

\( \tilde{w}_i \) random at random according to a distribution \( \mu \)

Perform KRR on \( H_M = \{ f | f(x) = \sum_{i=1}^{M} c_i q(x, \tilde{w}_i), c_i \in \mathbb{R} \} \).
Random Features

\[ \hat{f}(x) = \sum_{i=1}^{M} c_i q(x, w_i) \]

- \( q \) general non linear function
Random Features

\[ \hat{f}(x) = \sum_{i=1}^{M} c_i q(x, \tilde{w}_i) \]

- \( q \) general non linear function

- pick \( \tilde{w}_i \) at random according to a distribution \( \mu \)

\[ \tilde{w}_1, \ldots, \tilde{w}_M \sim \mu \]
Random Features

\[ \hat{f}(x) = \sum_{i=1}^{M} c_i q(x, \tilde{w}_i) \]

- \( q \) general non linear function
- pick \( \tilde{w}_i \) at random according to a distribution \( \mu \)
  \[ \tilde{w}_1, \ldots, \tilde{w}_M \sim \mu \]
- perform KRR on

\[ \mathcal{H}_M = \{ f \mid f(x) = \sum_{i=1}^{M} c_i q(x, \tilde{w}_i), c_i \in \mathbb{R} \} \]
Consider

\[ q(x, w) = e^{i w^T x}, \]

(Rahimi, Recht '07)
Consider

\[ q(x, w) = e^{i w^T x}, \quad w \sim \mu(w) = \mathcal{N}(0, I) \]
Random Fourier Features

(Rahimi, Recht ’07)

Consider

\[ q(x, w) = e^{iw^T x}, \quad w \sim \mu(w) = \mathcal{N}(0, I) \]

Then

\[ \mathbb{E}_w [q(x, w)q(x', w)] = e^{-\|x-x'\|^2 \gamma} = K(x, x') \]
Random Fourier Features

(Rahimi, Recht '07)

Consider

\[ q(x, w) = e^{iw^T x}, \quad w \sim \mu(w) = \mathcal{N}(0, I) \]

Then

\[ \mathbb{E}_w [q(x, w)q(x', w)] = e^{-\|x-x'\|^2\gamma} = K(x, x') \]

By sampling \( \tilde{w}_1, \ldots, \tilde{w}_M \) we are considering the approximating kernel

\[ \frac{1}{M} \sum_{i=1}^{M} [q(x, \tilde{w}_i)q(x', \tilde{w}_i)] = \tilde{K}(x, x') \]
More Random Features

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

Note: Connections with hashing and sketching techniques.
More Random Features

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

\[ q(x, w) = e^{i w^T x}, \]
More Random Features

- **translation invariant** kernels $K(x, x') = H(x - x')$, 
  
  $$q(x, w) = e^{iw^T x}, \quad w \sim \mu = \mathcal{F}(H)$$
More Random Features

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

  $$q(x, w) = e^{\imath w^T x}, \quad w \sim \mu = \mathcal{F}(H)$$

- **infinite neural nets** kernels

Note: Connections with hashing and sketching techniques.
More Random Features

- **translation invariant** kernels $K(x, x') = H(x - x')$,
  $$q(x, w) = e^{iw^T x}, \quad w \sim \mu = \mathcal{F}(H)$$

- **infinite neural nets** kernels
  $$q(x, w) = |w^T x + b|_+,$$
More Random Features

- **translation invariant** kernels $K(x, x') = H(x - x')$,
  
  $$q(x, w) = e^{i w^T x}, \quad w \sim \mu = \mathcal{F}(H)$$

- infinite **neural nets** kernels
  
  $$q(x, w) = |w^T x + b|_+, \quad (w, b) \sim \mu = U[\mathbb{S}^d]$$
More Random Features

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

  \[
  q(x, w) = e^{i w^T x}, \quad w \sim \mu = \mathcal{F}(H)
  \]

- **infinite neural nets** kernels

  \[
  q(x, w) = |w^T x + b|_+, \quad (w, b) \sim \mu = U[\mathbb{S}^d]
  \]

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

**Note:** Connections with **hashing** and **sketching** techniques.
Properties of Random Features

Optimization

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

Statistics

As before: do we pay a price for efficient computations?
Properties of Random Features

Optimization

- Time: $O(n^3) \quad O(nM^2)$
- Space: $O(n^2) \quad O(nM)$
Properties of Random Features

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

Statistics
As before: **do we pay a price for efficient computations?**
Previous works

Many different random features for different kernels (Rahimi, Recht '07, Vedaldi, Zisserman, . . . 10+)

Theoretical guarantees: mainly kernel approximation (Rahimi, Recht '07, . . . , Sriperumbudur and Szabo '15)

\[ |K(x,x') - \tilde{K}(x,x')| \lesssim \sqrt{M} \]

Statistical guarantees: suboptimal or in restricted setting (Rahimi, Recht '09, Yang et al. '13 . . . , Bach '15)
Previous works

- **Many** different random features for different kernels
  (Rahimi, Recht ’07, Vedaldi, Zisserman, ... 10+)

*Theoretical guarantees: mainly kernel approximation (Rahimi, Recht ’07, ... , Sriperumbudur and Szabo ’15)

*Statistical guarantees: suboptimal or in restricted setting (Rahimi, Recht ’09, Yang et al. ’13 ... , Bach ’15)*
Previous works

- *Many* different random features for different kernels
  (Rahimi, Recht '07, Vedaldi, Zisserman, . . . 10+)

- Theoretical guarantees: mainly kernel approximation
  (Rahimi, Recht '07, . . . , Sriperumbudur and Szabo '15)

\[ |K(x, x') - \tilde{K}(x, x')| \lesssim \frac{1}{\sqrt{M}} \]
Previous works

- **Many** different random features for different kernels
  (Rahimi, Recht ’07, Vedaldi, Zisserman, . . . 10+)

- Theoretical guarantees: mainly kernel approximation
  (Rahimi, Recht ’07, . . ., Sriperumbudur and Szabo ’15)

\[ |K(x, x') - \tilde{K}(x, x')| \lesssim \frac{1}{\sqrt{M}} \]

- Statistical guarantees **suboptimal or in restricted setting**
  (Rahimi, Recht ’09, Yang et al. ’13 . . ., Bach ’15)
Main Result

Let

\[ q(x, w) = e^{iw^T x}, \]

Theorem

If \( f^* \in H_s \) Sobolev space, then

\[ \lambda^* = n^{1/2} s + 1, \quad M^* = 1, \quad E(\hat{f}_{\lambda^*}^{M^*}(x) - f^*(x))^2 \lesssim n^{-2} s^2 s + 1 \]

Random feature achieves optimal bound!

Efficient worst case subsampling \( M^* \sim \sqrt{n} \), but cannot exploit smoothness.
Main Result

Let

\[ q(x, w) = e^{iw^T x}, \quad w \sim \mu(w) = c_d \left( \frac{1}{1 + \|w\|^2} \right)^{\frac{d+1}{2}} \]
Main Result

Let

\[ q(x, w) = e^{iw^T x}, \quad w \sim \mu(w) = cd \left( \frac{1}{1 + \|w\|^2} \right)^{\frac{d+1}{2}} \]

Theorem

If \( f_* \in \mathcal{H}_s \) Sobolev space, then

\[ \lambda_* = n^{-\frac{1}{2s+1}}, \quad M_* = \frac{1}{\lambda_*^{2s}}, \quad \mathbb{E} \left( \hat{f}_{\lambda_* M_*}(x) - f^*(x) \right)^2 \lesssim n^{-\frac{2s}{2s+1}} \]
Main Result

Let
\[ q(x, w) = e^{iw^T x}, \quad w \sim \mu(w) = c_d \left( \frac{1}{1 + \|w\|^2} \right)^{\frac{d+1}{2}} \]

Theorem

If \( f_\ast \in \mathcal{H}_s \) Sobolev space, then
\[ \lambda_\ast = n^{-\frac{1}{2s+1}}, \quad M_\ast = \frac{1}{\lambda_\ast^{2s}}, \quad \mathbb{E} \left( \hat{f}_{\lambda_\ast, M_\ast}(x) - f_\ast(x) \right)^2 \lesssim n^{-\frac{2s}{2s+1}} \]

- Random feature achieves **optimal** bound!
Main Result

Let

\[ q(x, w) = e^{iw^T x}, \]
\[ w \sim \mu(w) = cd \left( \frac{1}{1 + \|w\|^2} \right)^{\frac{d+1}{2}} \]

**Theorem**

If \( f_\ast \in \mathcal{H}_s \) Sobolev space, then

\[ \lambda_\ast = n^{-\frac{1}{2s+1}}, \quad M_\ast = \frac{1}{\lambda_\ast^{2s}}, \quad \mathbb{E} (f^{\ast, M_\ast}(x) - f_\ast(x))^2 \lesssim n^{-\frac{2s}{2s+1}} \]

- Random feature achieves **optimal** bound!
- Efficient worst case subsampling \( M_\ast \sim \sqrt{n} \), but cannot exploit smoothness.
Remarks & Extensions

N"ystrom vs Random features

- Both achieve optimal rates
- N"ystrom seems to need fewer samples (random centers)
Remarks & Extensions

Nystrom vs Random features

- Both achieve optimal rates
- Nystrom seems to need fewer samples (random centers)

How tight are the results?
Remarks & Extensions

Nýstrom vs Random features

- Both achieve optimal rates
- Nýstrom seems to need fewer samples (random centers)

How tight are the results?

\[
\log \lambda \quad \text{Test Error} \quad \log M
\]

\[
\begin{array}{c}
\text{log } \lambda \\
0.2201n^{-0.6197}
\end{array}
\quad
\begin{array}{c}
\text{Test Error} \\
0.8826n^{-0.3480}
\end{array}
\quad
\begin{array}{c}
\text{log } M \\
0.3989n^{0.6710}
\end{array}
\]
Contributions

- **Optimal bounds** for data dependent/independent subsampling
- Subsampling: Nystrom vs Random features
- Beyond ridge regression: *early stopping* and multiple passes SGD (coming up in AISTATS!)

Some questions:

- Quest for the **best** sampling
- Regularization by projection: inverse problems and preconditioning
- Beyond randomization: *non convex optimization*?

Some perspectives:

- Computational regularization: subsampling regularizes
- Algorithm design: Control stability with computations
Contributions

- **Optimal bounds** for data dependent/independent subsampling
- Subsampling: Nystrom vs Random features
- Beyond ridge regression: *early stopping* and multiple passes SGD (coming up in AISTATS!)

Some questions:

- Quest for the **best** sampling
- **Regularization by projection:** inverse problems and preconditioning
- Beyond randomization: **non convex optimization**?
Contributions

- **Optimal bounds** for data dependent/independent subsampling
- Subsampling: Nýstrom vs Random features
- Beyond ridge regression: *early stopping* and multiple passes SGD (coming up in AISTATS!)

Some questions:
- Quest for the **best** sampling
- **Regularization by projection**: inverse problems and preconditioning
- Beyond randomization: **non convex optimization**?

Some perspectives:
- **Computational regularization**: subsampling regularizes
- **Algorithm design**: Control stability with computations