Random Fourier Features for Kernel Ridge Regression: Approximation Bounds and Statistical Guarantees

Haim Avron
Department of Applied Math
Tel Aviv University, Israel

Joint work with: M. Kapralov, C. Musco, C. Musco, A. Velingker, A. Zandieh

Tel Aviv University, Graduate Student Seminar
Dec 10, 2017
Outline

• Tutorial part:
  – The mathematical problem of learning from examples
  – Kernel ridge regression and (kernel methods in general)
  – Scalability issue and Random Fourier Features

• New results part:
  – Statistical guarantees \(\leftrightarrow\) approximation bounds
  – Analysis of random Fourier features
Learning from examples

- Given: input domain $\mathcal{X}$, output domain $\mathcal{Y}$.
- Input: examples $(x_1, y_1), \ldots, (x_n, y_n) \subseteq \mathcal{X} \times \mathcal{Y}$
- Output: some functional dependency between $y$ and $x$.

How do we formalize this problem?
Let’s consider some extremely simplified examples.
Example 1: Learning a physical law by curve fitting

• Assume: unknown \( f : \mathbb{R} \to \mathbb{R} \) with specific form.

• For example, a \( d \) degree polynomial \( f(x) = f_w(x) = \sum_{j=0}^{d} w_j x^j \)

• We want to estimate \( w \). We select \( x_1, \ldots, x_n \) and measure \( f(x_i) \).

• Since measurement is usually noiseless, we model \( y_i = f(x_i) + noise \).

• **Goal:** given \((x_1, y_1), \ldots, (x_n, y_n)\) estimate \( w \).
Example 2: Recognizing hand-written characters

We are given randomly drawn small images of numbers, and noisy labels, and we want to find a way to classify a number image. $\mathcal{Y}$ is multivariate

$$\mathcal{Y} = \{y \in \mathbb{R}^{10} \mid y \text{ convex comb. of } e_1, \ldots, e_{10}\}$$

and should be interpreted as probability vector.
Example 3: Monte Carlo integration

- There is an unknown function $f : [0, 1]^d \rightarrow \mathbb{R}$. We want to estimate

$$I(f) = \int_{x \in [0, 1]^d} f(x) dx.$$  

- We are given random samples (distributed uniformly) $x_1, \ldots, x_n \in [0, 1]^d$ and the corresponding values of $f$: $y_i = f(x_i)$.  

- **Goal:** estimate $I(f)$.  

Example 4: PAC learning

PAC - Probably Approximately Correct

• $T \subseteq \mathbb{R}^n$ target set, $f_T$ indicator functions.

• A set $S$ approximates $T$ is $S \Delta T$ is small.

• Measure the sets using some unknown probability measure $\rho$.

• Input: $x_1, \ldots, x_n$ from $\rho$, $y_j = f_T(x_j)$.

• Goal: Find an approximation $f_S$ of $f_T$.

• In PAC learning: how large should $n$ be to learn a $\epsilon$ approximation of $T$ with probability at least $1 - \delta$?
Common Themes

• "Unknown" $f : \mathcal{X} \rightarrow \mathcal{Y}$.

• Probability measure allowing one to draw from $\mathcal{X} \times \mathcal{Y}$:
  
  – Possibly on $\mathcal{X}$ alone (example 3 and 4).
  – Possibly on $\mathcal{Y}$ varying with $x \in \mathcal{X}$ (example 1).
  – Possibly on the product $\mathcal{X} \times \mathcal{Y}$ (example 2).

• The measure can be known (case 3) or unknown.

• $f(x)$ is the expected value in $\mathcal{Y}$ given $x \in \mathcal{X}$.
Formal Setting

- Input domain: $\mathcal{X} \subseteq \mathbb{R}^d$, Output domain: $\mathcal{Y} \subseteq \mathbb{R}$.
  - Regression: $\mathcal{Y} = \mathbb{R}$, Binary classification: $\mathcal{Y} = \{0, 1\}$.
- Exists a probability measure $\rho$ on $\mathcal{Z} = \mathcal{X} \times \mathcal{Y}$.
  - Marginal distributions: $\rho_x, \rho_y$.
  - $X \sim \rho_x, Y \sim \rho_y$.
- We are given $n$ samples from $\rho$: $D = \{(x_i, y_i)\}_{i=1}^n \subseteq \mathcal{X} \times \mathcal{Y}$.
- Goal: estimate the function that minimizes the error (also: risk, $MSE$)

\[
E(f) = \int_{\mathcal{Z}} (f(x) - y)^2 d\rho(x, y)
\]
Regression Function

• The minimum error is obtained by the *regression function*:

\[ f^*(x) = E(Y|X = x) = \int y \, d\rho(y|x). \]

• Remark: the definition above of the error/risk is based on the square loss

\[ V(y, z) = (y - z)^2. \]

Other loss functions can be used, which leads to alternative definition of risk and a different optimal function.
From Risk to Empirical Risk

- We cannot compute $E(f)$ let alone find the minimizer....

- However, we can compute the empirical error

$$E_D(f) = \sum_{i=1}^{n} (f(x_i) - y_i)^2$$

and optimize it (this is referred to as empirical risk minimization).

- When $E(f) \approx E_D(f)$? Simplified answer: if both $\mathcal{Y}$ and $f$ are bounded, and $n$ is large enough.
Hypothesis Space

• To have something computational, we need to restrict our search.

• Let $\mathcal{H}$ be a class of functions, with a norm $\| \cdot \|_{\mathcal{H}}$.
  
  – We do not require $\mathcal{H}$ to be a Banach space (you’ll see soon why), but we assume it is a subset of a Banach space with the same norm.

• The best estimate in $\mathcal{H}$, and the empirical error minimizers are:

$$f_\mathcal{H} \equiv \arg \min_{f \in \mathcal{H}} E(f)$$

$$\hat{f}_\mathcal{H} \equiv \arg \min_{f \in \mathcal{H}} E_D(f)$$

(assuming that you set up $\mathcal{H}$ so there is a (unique) minimizer)
Bias-Variance Tradeoff

We can write:

\[ E(\hat{f}_H) = E(f_H) + E(\hat{f}_H) - E(f_H) \]

- The first term depends only on \( H \) (not the sample), so acts as a bias. It is generally reduced as \( H \) becomes "bigger".
- For a fixed \( H \), the second terms comes from the sample, and is reduced as \( n \) grows. Thus, it is acts like as a variance. It generally increases as \( H \) becomes "bigger".
Regularized Least-Squares Estimation

- One way to manage this bias-variance tradeoff is to restrict the norm of functions in $\mathcal{H}$:

$$\hat{f}_{\mathcal{H},B} \equiv \arg \min_{\|f\|_{\mathcal{H}} \leq B} E_D(f)$$

($B$ is a parameter.)

- However, it is somewhat easier to use a penalty instead:

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

where $\lambda \geq 0$ is a regularization parameter.
If you lost me, now is the time to RECONNECT!

To summarize, we are given \((x_1, y_1), \ldots, (x_n, y)\) and we wish to compute

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

where \(\mathcal{H}\) is some predefined hypothesis space equipped with a norm, and \(\lambda \geq 0\) is a parameter.
Defining an Hypothesis Space using a Kernel

• Start with a symmetric positive definite kernel function \( k : \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R} \)
  
  – That is: for every \( \{x_1, \ldots, x_k\} \) the matrix \( K \) defined by \( K_{ij} = k(x_i, x_j) \) is positive definite.

  – For example:

  \[
  k(x, z) = \exp(-\|x - z\|_2^2)
  \]
Defining an Hypothesis Space using a Kernel

- Start with a symmetric positive definite kernel function \( k : \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R} \)
- Define a function space and an inner product

\[
\mathcal{H}_k^0 := \left\{ \sum_{i=1}^{m} \alpha_i k(z_i, \cdot) \mid z_i \in \mathcal{X}, \alpha_i \in \mathbb{R}, m \in \mathbb{Z}^+ \right\}
\]

\[
\left( \sum_{i=1}^{n} \alpha_i k(x_i, \cdot), \sum_{j=1}^{m} \beta_j k(z_j, \cdot) \right)_{\mathcal{H}_k} = \sum_{i=1}^{n} \sum_{j=1}^{m} \alpha_i \beta_j k(x_i, z_j)
\]

- Complete the space to form an Hilbert space: \( \mathcal{H}_k = \overline{\mathcal{H}_k^0} \).
- \( \mathcal{H}_k \) is a reproducing kernel Hilbert space (RKHS) with kernel \( k \).
Reproducing kernel Hilbert spaces (RKHSs)

**Definition 1.** A reproducing kernel Hilbert space is an Hilbert space $\mathcal{H}$ for which the point evaluation functional

$$L_x : \mathcal{H} \rightarrow \mathbb{R}, \quad L_x(f) = f(x)$$

is bounded for every $x$ in $\mathcal{H}$’s domain.
Reproducing kernel Hilbert spaces (RKHSs)

**Claim 1.** If $\mathcal{H}$ is a RKHS there exists a $h: \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ such that

1. $h(x, \cdot) \in \mathcal{H}$ for all $x \in \mathcal{X}$.

2. $(f, h(x, \cdot))_{\mathcal{H}} = f(x)$ for all $x \in \mathcal{X}$ and $f \in \mathcal{H}$ (Reproducing property)

**Proof (sketch).** The Riesz representation theorem implies that for every $x$ there exists an $h_x \in \mathcal{H}$ such that $(f, h_x)_{\mathcal{H}} = f(x)$ for all $f \in \mathcal{H}$. The reproducing kernel is $h(x, z) = (h_x, h_z)_H$. □

**Fact 2.** $\mathcal{H}_k$ constructed previously is a RKHS with kernel $k$. 

Tel Aviv University: Haim Avron
Kernel Ridge Regression: In Short

• Pick: $k : \mathcal{X} \times \mathcal{X} \to \mathcal{R}$.

• Defines Hilbert space $\mathcal{H}_k$ with nice features:
  1. $h(x, \cdot) \in \mathcal{H}_k$ for all $x \in \mathcal{X}$.
  2. $(f, h(x, \cdot))_{\mathcal{H}_k} = f(x)$ for all $x \in \mathcal{X}$ and $f \in \mathcal{H}_k$.

  (However, we’ll see: construction of $\mathcal{H}_k$ is actually irrelevant!)

• Use $\mathcal{H}_k$ as the hypothesis space: solve

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

HOW?
Representer Theorem

**Theorem 1.** Let $c : \mathcal{Y} \times \mathcal{Y} \rightarrow \mathbb{R}$ be any loss function, and let $g : [0, \infty) \rightarrow \infty$ by any strict monotonically increasing function. Then

$$f_{c,g,k} := \arg\min_{f \in \mathcal{H}_k} \frac{1}{n} \sum_{i=1}^{n} c(y_i, f(x_i)) + g(\|f\|_{\mathcal{H}_k})$$

has the form

$$f_{c,g,k}(x) = \sum_{i=1}^{n} \alpha_i k(x_i, x)$$
Proof sketch

• Recall the optimization problem:

\[ f_{c,g,k} := \arg\min_{f \in \mathcal{H}_k} \frac{1}{n} \sum_{i=1}^{n} c(y_i, f(x_i)) + g(\|f\|_\mathcal{H}) \]

• Every \( f \in \mathcal{H}_k \) can be written as \( f = \sum_{i=1}^{n} \alpha_i k(x_i, \cdot) + v \) where \((v, k(x_i, \cdot))_{\mathcal{H}_k} = 0 \) for \( i = 1, \ldots, n \).

• Reproducing property implies: \( f(x_j) = \sum_{i=1}^{n} \alpha_i k(x_i, x_j) \).
  So: First part of the objective is unchanged if we remove \( v \).

• Due to orthogonality: \( \|f\|_\mathcal{H} > \| \sum_{i=1}^{n} \alpha_i k(x_i, \cdot) \|_{\mathcal{H}_k} \) if \( v \neq 0 \)
  So: Second part of the objective decreases if we remove \( v \).
Finding \( \alpha = (\alpha_1, \ldots, \alpha_n)^T \)

- Define the Gram matrix: \( K_{ij} := k(x_i, x_j) \) \((K \in \mathbb{R}^{n \times n})\)

- Let \( f_\alpha = \alpha_j k(x_j, \cdot) \). Our goal: find \( \alpha \) that minimizes

\[
L(\alpha) \equiv \frac{1}{n} \sum_{i=1}^{n} (y_i - f_\alpha(x_i))^2 + \lambda \|f_\alpha\|_{\mathcal{H}_k}^2
\]

\[
= \frac{1}{n} \sum_{i=1}^{n} \left( y_i - \sum_{j=1}^{n} \alpha_j k(x_j, x_i) \right)^2 + \lambda \|\alpha_j k(x_j, \cdot)\|_{\mathcal{H}_k}^2
\]

\[
= \frac{1}{n} \|y - K\alpha\|_2^2 + \lambda \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j k(x_i, x_j)
\]

\[
= \frac{1}{n} \|y - K\alpha\|_2^2 + \lambda \alpha^T K\alpha
\]
Finding $\alpha = (\alpha_1, \ldots, \alpha_n)^T$

- From previous slide: want to find minimum of

$$L(\alpha) \equiv \frac{1}{n} \| y - K\alpha \|_2^2 + \lambda \alpha^T K\alpha$$

- So, we need to find $\alpha$ such that $\nabla L(\alpha) = 0$. The minimizing $\alpha$ is

$$\alpha = (K + \lambda n I_n) y$$
If you lost me, now is the time to RECONNECT!

All of this was a lengthy explanation to why what we are interested in solving the linear equation

\[(K + \lambda n I_n)\alpha = y\]

where \(K\) is an \(n \times n\) matrix whose entries are

\[K_{ij} = k(x_i, x_j)\]

for some \(k\) (we’ll assume things on \(k\) later).
Example Kernels (k’s)

- Polynomial Kernel:
  \[ k(x, z) = (x^T z + c)^q \]
  
  \((q \text{ and } c \text{ are parameters})\)
  
  - \(\mathcal{H}_k\) is the space of \(d\)-variate polynomials of degree at most \(q\).
  - \(c\) affects the definition of the inner product.
Example Kernels ($k$’s)

- Gaussian Kernel:

$$k(x, z) = \exp(-\|x - z\|^2_2/\sigma^2)$$

($\sigma$, aka "bandwidth", is a parameter.)

- Very popular in machine learning.
- Justified by it’s *universality*: if $\mathcal{X}$ is compact, $\mathcal{H}_k$ is dense in $C(\mathcal{X})$.
- However, approximation theorists and statisticians explain that this is not a good choice and other kernels (e.g., Matérn) should be used.
Kernel Ridge Regression In Action

\[ f(x) = \sin(6x) + \sin(\exp(60x)) \]
Main Limiting Factor: Scalability

- Kernel ridge regression:

\[(K + \lambda nI)\alpha = y\]

\[O(n^2)\] storage

\[O(n^3 + n^2d)\] training

\[O(nd)\] test speed

- Too expensive if \(n\) is even moderately large!
Low-Rank Approximation Approach

- Storing $Z$ uses $O(ns)$ space and computing $ZZ^T x$ takes $O(ns)$ time. Orthogonalization, eigendecomposition, and pseudo-inversion of $ZZ^T$ all take just $O(ns^2)$ time.
Efficient Low-Rank Approximation?

Low-rank approximation is itself an expensive task.

- Optimal low-rank approximation via a direct eigendecomposition, or even approximation via Krylov subspace methods are expensive since they, at the very least, require fully forming $\mathbf{K}$.

- Many faster methods have been studied: incomplete Cholesky factorization (Fine & Scheinberg ‘02, Bach & Jordan ‘02), entrywise sampling (Achlioptas, McSherry, & Schölkopf ‘01), Nyström approximation (Williams & Seeger ‘01), random Fourier features (Rahimi & Recht ‘07).
Approximation via a Separable Function

- One idea is to find a \( \varphi : \mathbb{R}^d \to \mathbb{C}^s \) such that

\[
k(x, z) \approx \varphi(x)^* \varphi(z)
\]

- Then,

\[
K \approx \begin{pmatrix}
\varphi(x_1)^* \\
\vdots \\
\varphi(x_n)^*
\end{pmatrix}
\begin{pmatrix}
\varphi(x_1) & \cdots & \varphi(x_n)
\end{pmatrix} = ZZ^*
\]

- Equivalent to using \( \tilde{k}(x, z) = \varphi(x)^* \varphi(z) \) as a substitute of \( k \).

- Goal: small \( s \), "good approximation to \( k \)" (How to quantize this?)
Random Fourier Features (Rahimi and Recht 2007)

• Suppose \( k(x, x) = 1 \) and that \( k(\cdot, \cdot) \) is *shift-invariant*, i.e.

\[
k(x, z) = \psi(x - z),
\]

for some *positive definite function* \( \psi \) on \( \mathbb{R}^d \).

• Due to Bochner’s Theorem: there exists a distribution \( \mu = \mu_k \) such that

\[
k(x, z) = \int_{\mathbb{R}^d} e^{-2\pi i \eta^T (x - z)} d\mu(\eta) = \int_{\mathbb{R}^d} e^{-2\pi i \eta^T (x - z)} p(\eta) d\eta
\]

• Gaussian kernel: \( k(x, z) = e^{-\frac{|x - z|^2}{2\sigma^2}} \iff \mu = \mathcal{N}(0, \sigma^{-2} I_d) \).
Random Fourier Features (Rahimi and Recht 2007)

- Main idea: approximate $k(\cdot, \cdot)$ via Monte-Carlo.

- Draw $\eta_1, \ldots, \eta_s \sim \mu$ and approximate:

$$k(x, z) = \int_{\mathbb{R}^d} e^{-2\pi i \eta^T (x-z)} p(\eta) d\eta \approx \frac{1}{s} \sum_{l=1}^{s} e^{-2\pi i \eta_l^T (x-z)}$$

- Define the random feature map:

$$\varphi(x) \equiv \frac{1}{\sqrt{s}} \left[ e^{-2\pi i \eta_1^T x}, \ldots, e^{-2\pi i \eta_s^T x} \right]^* \in \mathbb{C}^s.$$
Very Popular Method...

Scholarly articles for **random features for large-scale kernel machines**

**Random Features for Large-Scale Kernel Machines.** - Rahimi - Cited by 817

[PDF] Random Features for Large-Scale Kernel Machines - EECS at UC ...

[https://papers.nips.cc/paper/3182-random-features-for-large-scale-kernel-machines](https://papers.nips.cc/paper/3182-random-features-for-large-scale-kernel-machines)

by A Rahimi - 2008 - Cited by 810 - Related articles

**Random Features for Large-Scale Kernel Machines.** Part of: Advances in Neural Information Processing Systems 20 (NIPS 2007) · [PDF] [BibTeX] ...
The Success of Random Fourier Features
Example on a Speech Recognition Dataset

Huang, Avron, Sainath, Sindhwani, Ramabhadran, Kernel Methods Match Deep Neural Networks on TIMIT, ICASSP 2014
Avron and Sindhwani, High-Performance Kernel Machines With Implicit Distributed Optimization and Randomization, Technometrics 68 (3) 2016
Scalable Random Features

Intensive recent research on scalable methods for huge random features:

1. Huang et al., Kernel Methods Match Deep Neural Networks on TIMIT.
2. Avron and Sindhwani, High-Performance Kernel Machines With Implicit Distributed Optimization and Randomization.
3. Lu et al., How to Scale Up Kernel Methods to Be As Good As Deep Neural Nets
4. Sparks et al., KeystoneML: Optimizing Pipelines for Large-Scale Advanced Analytics
5. Tu et al., Training Large Scale Kernel Machines with Block Coordinate Descent
6. Dai et al., Scalable Kernel Methods via Doubly Stochastic Gradients
7. and more!

Treating the symptom instead of the problem!
Questions

- Why so many random features are needed?
- Are they really needed? Can the method be improved?
- How well does the approximate estimator ($\hat{f}_k$) compare to the kernel ridge regression estimator ($\hat{f}^*_k$)? How do we measure it?

Recall:
- $\hat{f}_k(\cdot)$ is the "true" kernel ridge regression estimator.
- $\hat{f}_k(\cdot)$ is the estimator with the approximate kernel.
The Original Analysis of Random Fourier Features

**Theorem 3** (Rahimi and Recht, 2007). Suppose $\mathcal{X}$ be a compact subset of $\mathbb{R}^d$ with diameter $\text{diam}(\mathcal{X})$. If $\eta_1, \ldots, \eta_s$ are drawn according to $\mu$, then for $0 < \epsilon < 1$ and $s > 0$ we have

$$\Pr \left( \sup_{x,z \in \mathcal{X}} |k(x, z) - \tilde{k}(x, z)| \geq \epsilon \right) < 2^8 \left( \frac{\sigma_p \text{diam}(\mathcal{X})}{\epsilon} \right)^2 \exp \left( -\frac{s\epsilon^2}{4(d + 2)} \right)$$

where $\tilde{k}(x, z) = \varphi(x)^* \varphi(z)$ and $\sigma_p \equiv \mathbb{E}_{\eta \sim \mu}[|\eta|^2]$.

This analysis is mostly useless – little implications to comparing estimators!
Our Suggestion: Spectral Approximation Guarantees

• Instead of looking point-wise, look at the entire Gram matrix.

• Recall the PSD partial ordering: \( A \preceq B \) iff \( \forall x, x^*Ax \leq x^*Bx \).

• We say that \( \tilde{K} + \lambda nI_n \) is \( \Delta \)-spectral approximation of \( K + \lambda nI_n \) if

\[
(1 - \Delta)(K + \lambda nI_n) \leq \tilde{K} + \lambda nI_n \leq (1 + \Delta)(K + \lambda nI_n)
\]

• ”Claim”: \( \tilde{k}(\cdot, \cdot) \) is a good approximation if \( \tilde{K} + \lambda nI_n \) is a spectral approximation of \( K + \lambda nI_n \).

– \( \Delta \) is a lever for trading quality of approximation with computation.
Fixed Design (aka De-noising) Analysis of KRR

- Assume the data points $x_1, \ldots, x_n$ are deterministic. So, we assess an estimator $\hat{f}$ using the expected in-sample prediction error:

$$E(f) \equiv \mathbb{E}_{y_1, \ldots, y_n} \left[ \frac{1}{n} \sum_{j=1}^{n} (f(x_i) - \mathbb{E}y_i)^2 \right].$$

- Assume a Gaussian model $y_i = f^*(x_i) + \sigma w_i$ and let $f \in \mathbb{R}^n$ for which the $i$th entry is $f^*(x_i)$. Simple algebra leads to the bound:

$$E(\hat{f}_k) \leq \lambda f^T (K + \lambda n I_n)^{-1} f + n^{-1} \sigma^2 s_\lambda(K)$$

where $s_\lambda(K) \equiv \text{Tr} \left( K(K + \lambda n I_n)^{-1} \right)$ (the effective degrees of freedom or statistical dimension).
Fixed Design (aka De-noising) Analysis of KRR

Lemma 4. If

\[(1 - \Delta)(K + \lambda nI_n) \leq \tilde{K} + \lambda nI_n \leq (1 + \Delta)(K + \lambda nI_n)\]

then

\[E(\hat{f}_k) \leq \left(1 - \Delta\right)^{-1} \left(\lambda f^T(K + \lambda nI_n)^{-1}f + n^{-1}\sigma^2 s_{\lambda}(K)\right)\]

\[+ \frac{\Delta}{1 + \Delta} n^{-1}\sigma^2 \text{rank}(\tilde{K})\]

\[\text{additive increase}\]

\[\text{relative increase}\]

\[\text{Interpretation: Small increase in error bound if } \Delta \text{ is small, and } \text{rank}(\tilde{K}) \text{ is not much larger than } s_{\lambda}(K).\]
An Illustration

| Estimator                     | $E(f)$ | $\frac{\|K - \tilde{K}\|^2_F}{\|K\|^2_F}$ | $\kappa (K + \lambda nI, \tilde{K} + \lambda nI)$ |
|-------------------------------|--------|---------------------------------|---------------------------------|
| Kernel Ridge Regression       | 0.016  |                                 |                                 |
| Classical Random Features     | 0.1474 | 0.17                            | 1458.6                          |
| Modified Random Features      | 0.0178 | 0.31                            | 56.2                            |
Additional Graphs

![Graph 1: Risk Comparison](image1)

- KRR
- CRF
- MRF

![Graph 2: Relative Error in K](image2)

- CRF
- MRF

![Graph 3: Generalized Condition Number](image3)

- CRF
- MRF

Tel Aviv University: Haim Avron
Takeaways

1. Kernel function approximation should be analyzed via spectral approximation bounds on the entire kernel matrix (instead of pointwise analysis).
Back to Random Features

- Recall:
  - Parameter $s$ - number of random features.
  - $\varphi : \mathbb{R}^d \rightarrow \mathbb{C}^s$.
  - $\tilde{k}(x, z) = \varphi(x)^* \varphi(z)$.

- Obviously: $\text{rank} (\tilde{K}) = s$.

- How large should $s$ be for

\[(1 - \Delta)(K + \lambda n I_n) \preceq \tilde{K} + \lambda n I_n \preceq (1 + \Delta)(K + \lambda n I_n) \]

to hold? (Optimally, as a function of only $\Delta$ and $s_{\lambda}(K)$.)
Approximation Bounds for Random Fourier Features

Theorem 5. Let $\Delta > 0$ and $\delta \in (0,1)$. If $\tilde{k}(\cdot, \cdot)$ is the random Fourier features approximation to $k(\cdot, \cdot)$ with $s \geq (4/3)\Delta^{-2}\lambda^{-1}\ln(16\lambda^{-1}s\lambda/\delta)$ random features, then with probability of at least $1 - \delta$:

$$(1 - \Delta)(K + \lambda nI_n) \preceq \tilde{K} + \lambda nI_n \preceq (1 + \Delta)(K + \lambda nI_n)$$

Proof is based on tail bounds for sums of random matrices. Outline:

1. Let $K + \lambda nI_n = V\Sigma^2V^T$ be an eigendecomposition of $K + \lambda nI_n$.
2. Spectral approximation is equivalent to $\|\Sigma^{-1}V(K - \tilde{K})V^T\Sigma^{-1}\|_2 \leq \Delta$.
3. Define $s$ i.i.d random matrices $Y_1, \ldots, Y_s$ such that

$$\Sigma^{-1}V\tilde{K}V^T\Sigma^{-1} = \frac{1}{s}\sum_{i=1}^{s} Y_i, \quad \mathbb{E}[Y_i] = \Sigma^{-1}VKV^T\Sigma^{-1}$$

4. Use matrix tail-bounds to prove the claim.
Interpreting the Bound: Value of $\lambda^{-1}$

\[ s \geq (4/3) \Delta^{-2} \lambda^{-1} \ln(16\lambda^{-1}s_{\lambda}/\delta) \]

- Approximation is useful only if $s < n$ (preferably $s \ll n$).
- Raises the question: how $\lambda$ relates to $n$?
- Generally, $\lambda$ is a parameter so naturally ask: what is the optimal $\lambda$?
  - Optimality in the error/generalization, under assumptions on $f^*$, distribution that generated the data,.... Rich ground for mathematical analysis.
- Complex issue; generally optimal $\lambda^{-1}$ grows asymp. slower than $n$ (but at least as fast as $\sqrt{n}$, see Caponnetto and De Vito, FoCM 2007).
Takeaways

1. Kernel function approximation should be analyzed via spectral approximation bounds on the entire kernel matrix (instead of pointwise analysis).

2. Random Fourier Features can be provably useful under reasonable assumptions.
\[ \lambda^{-1} \textbf{vs.} \ s_\lambda(K) \]

\[
E(\hat{f}_k) \leq \left(1 - \Delta\right)^{-1} \left(\lambda f^T(K + \lambda nI_n)^{-1}f + n^{-1}\sigma^2 s_\lambda(K)\right) + \frac{\Delta}{1 + \Delta} n^{-1}\sigma^2 s
\]

- Due to the additive term, we want \( s \) to be proportional to \( s_\lambda(K) \).
  - Always: \( s_\lambda(K) \leq \lambda^{-1} \).
  - There exists bounded datasets for which \( s_\lambda(K) \ll \lambda^{-1} \).
  - It is generally well understood that it is better to avoid polynomial dependence on \( \lambda^{-1} \).

- Can the bound/dependence on \( s \) be improved?
Lower Bound (Gaussian Kernel)

**Theorem 6** (Informal Statement). *Even on one dimensional data, the number of samples required to guarantee*

\[
(1 - \Delta)(K + \lambda nI_n) \leq \tilde{K} + \lambda nI_n \leq (1 + \Delta)(K + \lambda nI_n)
\]

*must either* depend exponentially on the radius of the data **or** linearly on \(\lambda^{-1}\).
Modified Random Fourier Features

- Recall: Random Fourier Features samples $\eta_1, \ldots, \eta_s \sim p$ where $p(\cdot)$ corresponds to $k(\cdot, \cdot)$.

- There is no reason, in principal, that we must sample according to $p(\cdot)$!

- Let $q(\cdot)$ be another density. Sample $\eta_1, \ldots, \eta_s \sim q$ and define

$$
\phi(x) \equiv \frac{1}{\sqrt{s}} \left[ \sqrt{\frac{p(\eta_1)}{q(\eta_1)}} e^{-2\pi i \eta_1^T x}, \ldots, \sqrt{\frac{p(\eta_s)}{q(\eta_s)}} e^{-2\pi i \eta_s^T x} \right]^* \in \mathbb{C}^s .
$$

We still have $k(x, z) = \mathbb{E}[\phi(x)^* \phi(z)]$.

- We call this Modified Random Fourier Features.
Leverage Function Sampling

Definition 7. Let $z : \mathbb{R}^d \to \mathbb{C}^n$ be defined by $z(\eta)_j = e^{-2\pi i x^T_j \eta}$. The ridge leverage function is define by

$$\tau_\lambda(\eta) \equiv p(\eta)z(\eta)^*(K + \lambda nI_n)^{-1}z(\eta).$$

Theorem 8. Suppose we use modified Random Fourier Features with $q(\eta) \propto \tau_\lambda(\eta)$. If $s \geq (4/3) \Delta^{-2} s_\lambda \ln(16s^2_\lambda/\delta)$ random features, then with probability of at least $1 - \delta$:

$$(1 - \Delta)(K + \lambda nI_n) \leq \tilde{K} + \lambda nI_n \leq (1 + \Delta)(K + \lambda nI_n)$$

Problem: there is no easy way to sample proportional to $\tau_\lambda(\eta)$. 

Tel Aviv University: Haim Avron 52
Takeaways

1. Kernel function approximation should be analyzed via spectral approximation bounds on the entire kernel matrix (instead of pointwise analysis).

2. Random Fourier Features can be provably useful under reasonable assumptions.

3. Random Fourier Features are suboptimal.
**Improved Random Fourier Features on 1D**

**Definition 9 (Improved Fourier Features Distribution for the Gaussian Kernel).** Given $\chi > 0$, define the function

$$
\bar{\tau}_\chi(\eta) \equiv \begin{cases} 
25 \max(\chi, 3000 \log^{1.5} \lambda^{-1}) & |\eta| \leq 10 \sqrt{\log(\lambda^{-1})} \\
p(\eta)\lambda^{-1} & \text{o.w.}
\end{cases}
$$

![Graph of distributions](image-url)
**Improved Random Fourier Features on 1D**  
(Gaussian kernel)

**Theorem 10.** Consider the Gaussian kernel. Suppose our data is one dimensional and inside the interval $[−\chi, \chi]$. Suppose we use modified Random Fourier Features with $q(\eta) \propto \pi_\chi(\eta)$. If $s \geq (4/3)\Delta^{-2}s_\pi \ln(16s_\pi^2/\delta)$ random features, then with probability of at least $1 - \delta$:

$$(1 - \Delta)(K + \lambda nI_n) \leq \tilde{K} + \lambda nI_n \leq (1 + \Delta)(K + \lambda nI_n)$$

*In the above: $s_\pi = O(\chi \sqrt{\log \lambda^{-1} + \log^2 \lambda^{-1}}).$*

*In short: If you sample using a uniform distribution, you get logarithmic dependence on $\lambda^{-1}$.*

Can be generalized to higher dimension, but with exponential dependence on the dimension...
Experiment with 2D

True Function

KRR Estimator

CRF Estimator

MRF Estimator
Takeaways

1. Kernel function approximation should be analyzed via spectral approximation bounds on the entire kernel matrix (instead of pointwise analysis).

2. Random Fourier Features can be provably useful under reasonable assumptions.

3. Random Fourier Features are suboptimal.

4. Improved, mostly-optimal, Random Fourier Features are possible on low-dimension.
Main technical observation is the following:

- Define the operator $\Phi : L^2(\mu) \to \mathbb{C}^n$ by
  \[ \Phi y \equiv \int_{\mathbb{R}^d} z(\xi) y(\xi) d\mu(\xi) . \]

- Fact: $K = \Phi \Phi^*.$

- It is useful to think of $\Phi$ as $n$-by-$\infty$ matrix. 
  
  This view connects Random Fourier Features to the problem of approximating matrix multiplication by column sampling.
A Few Slides on the Analysis

The analysis is based on the following primal-dual characterization: **Lemma 11.**

\[ \tau_\lambda(\eta) = \min_{y \in L^2(\mu)} \lambda^{-1} \| \Phi y - \sqrt{p(\eta)}z(\eta) \|_2^2 + \| y \|_{L^2(\mu)}^2 \]
\[ = \max_{\alpha \in \mathbb{C}^n} \frac{p(\eta) \cdot |z(\eta)^* \alpha|^2}{\| \Phi^* \alpha \|_{L^2(\mu)}^2 + \lambda \| \alpha \|_2^2} \]

This lemma is a generalization of known results on "regular" matrices.
A Few Slides on the Analysis

• The leverage function is bounded by building a $y_\eta(\cdot)$ so that

$$\lambda^{-1}\|\Phi y_\eta - \sqrt{p(\eta)}z(\eta)\|_2^2 + \|y_\eta\|_{L^2(\mu)}^2$$

is small.

• Trivial choices do not work:
  
  – A shifted Dirac delta recovers the vector exactly, but is not in $L_2(d\mu)$ (has infinite norm).
  – A zero function just recovers the trivial column-norm bound.

• The idea is to use a ’soft’-spike instead.
A first attempt multiplies the pure frequency with a box in the frequency domain. The norm of the test function is still too large.
A Few Slides on the Analysis

To get a tight bound, we further damp with a Gaussian, centered at the target frequency.

\[ g(\xi) = y(\xi)p(\xi) \]

sinc and gaussian components
dampened test function with lower energy
A Few Slides on The Analysis

- The lower bounds is proved by building a dataset with a large discrepancy between $\tau_\lambda(\cdot)$ and $p(\cdot)$.

- Specifically, we build a dataset on which $\tau_\lambda(\cdot)$ is large on low frequencies, but $p(\cdot)$ is tiny on the edges. The construction is based on the soft-spike function.

- As a consequence, too few frequencies are sampled on the edge.
Conclusions

1. Kernel function approximation should be analyzed via spectral approximation bounds on the entire kernel matrix (instead of pointwise analysis).

2. Random Fourier Features can be provably useful under reasonable assumptions.

3. Random Fourier Features are suboptimal.

4. Improved, mostly-optimal, Random Fourier Features are possible on low-dimension.

5. Open question: are optimal random features possible in higher dimensions?
Thank You!

Acknowledgments:

• Co-authors:
  – Tight analysis of Random Fourier Features: Michael Kapralov, Cameron Musco, Christoper Musco, Ameya Velingker, Amir Zandieh.
  – Other papers mentioned: Po-Sen Huang, Bhuvana Ramabhadran, Tara N. Sainath, Vikas Sindhwani.

• Funding:
  XDATA program of the Defense Advanced Research Projects Agency (DARPA), IBM Faculty Award.