Topics in Two-Sample Testing

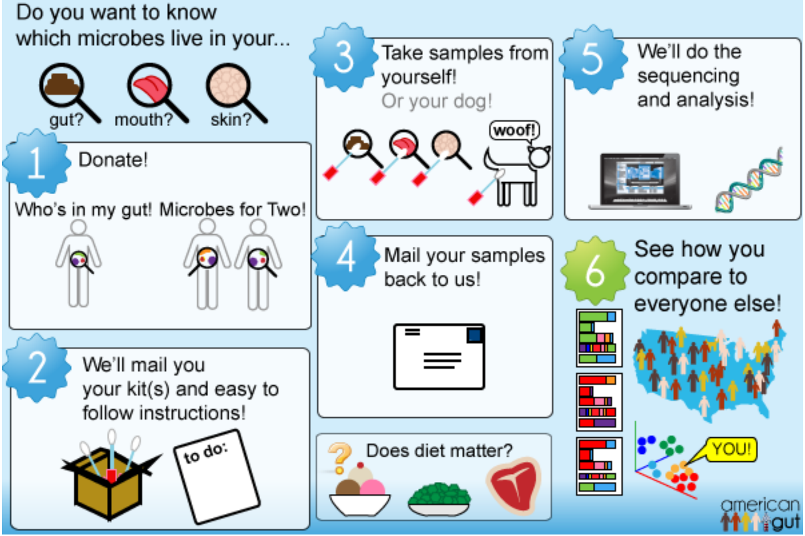
Nelson Ray
(joint work with Susan Holmes)

Stanford University

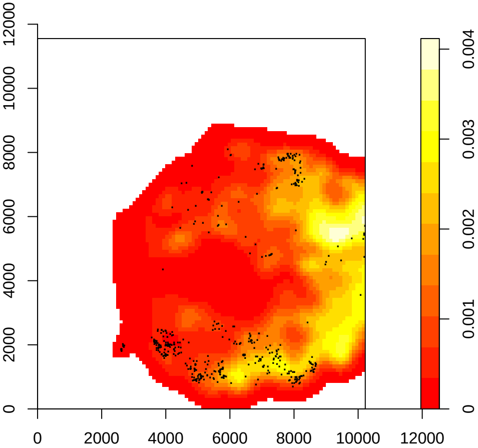
April 1, 2013

Motivation

American Gut Study: `www.indiegogo.com/americangut`



Breast Cancer Data: Spatial



Breast Cancer Data: Survival

Pathology no.	Initial Diagnosis Date	Relapse or Disease Free	RDF (R=relapsed; F=DF)	Recurrence Date	Las
98_17969D	1997-08-25	Disease Free	F	Disease Free	
97_24046C8	1997-08-25	Disease Free	F	Disease Free	
98_8501C1	1998-04-03	Disease Free	F	Disease Free	
98_8501A1	1998-04-03	Disease Free	F	Disease Free	
98_9134D4	1998-04-09	Left in-situ BrCa in 1999 (2nd primary cancer, not a metastasis from the right BrCa in 1997)	F	Disease Free	
98_9134B	1998-04-09	Left in-situ BrCa in 1999 (2nd primary cancer, not a metastasis from the right BrCa in 1997)	F	Disease Free	
98_14783B1	1998-06-10	bone, brain, lymph nodes, pericardium, liver metastasis	R	2004-07-30	
98_14783A	1998-06-10	bone, brain, lymph nodes, pericardium, liver metastasis	R	2004-07-30	
98_16169C2	1998-06-24	Disease Free	F	Disease Free	
98_16169A	1998-06-24	Disease Free	F	Disease Free	
98_16169B	1998-06-24	Disease Free	F	Disease Free	
98_16253C1	1998-06-25	Disease Free	F	Disease Free	
60C1	1998-07-10	Disease Free	F	Disease Free	

Breast Cancer Data: Medical

Pathology no.	Age at time of diagnosis	Gender	SLN tumor status	Diagnosis	ER status	PR status	Her-2 overexpressior
98_17969D	68	F	+	Invasive ductal carcinoma (IDC)	-	-	-
97_24046C8	68	F	+	Invasive ductal carcinoma (IDC)	-	-	-
98_8501C1	51	F	+	IDC & DCIS	+	+	?
98_8501A1	51	F	+	IDC & DCIS	+	+	?
98_9134D4	70	F	+	IDC	+	+	n/a
98_9134B	70	F	+	IDC	+	+	n/a
98_14783B1	67	F	+	IDC & DCIS	+	+	+
98_14783A	67	F	+	IDC & DCIS	+	+	+
98_16169C2	79	F	+mic	IDC	+	+	+
98_16169A	79	F	+mic	IDC	+	+	+
98_16169B	79	F	+mic	IDC	+	+	+
98_16253C1	70	F	+mic	IDC & DCIS	+	-	-
60C1	51	F	- (rare keratin+ cells)	IDC & DCIS	+	+	+
				IDC; DCIS;			

Outline

- Motivation: heterogeneous data are ubiquitous.
- Friedman's two-sample test: leverage regression and classification techniques.
- Kernel methods for non-vectorial and heterogeneous data.
- Generalizes permutation t-test!
- Stein's method of exchangeable pairs for Berry–Esseen-type bound.

Outline

- Motivation: heterogeneous data are ubiquitous.
- Friedman's two-sample test: leverage regression and classification techniques.
- Kernel methods for non-vectorial and heterogeneous data.
- Generalizes permutation t-test!
- Stein's method of exchangeable pairs for Berry–Esseen-type bound.

Outline

- Motivation: heterogeneous data are ubiquitous.
- Friedman's two-sample test: leverage regression and classification techniques.
- Kernel methods for non-vectorial and heterogeneous data.
- Generalizes permutation *t*-test!
- Stein's method of exchangeable pairs for Berry–Esseen-type bound.

Outline

- Motivation: heterogeneous data are ubiquitous.
- Friedman's two-sample test: leverage regression and classification techniques.
- Kernel methods for non-vectorial and heterogeneous data.
- Generalizes permutation t-test!
- Stein's method of exchangeable pairs for Berry–Esseen-type bound.

Outline

- Motivation: heterogeneous data are ubiquitous.
- Friedman's two-sample test: leverage regression and classification techniques.
- Kernel methods for non-vectorial and heterogeneous data.
- Generalizes permutation t-test!
- Stein's method of exchangeable pairs for Berry–Esseen-type bound.

Friedman's Two-Sample Test

Friedman (2003)
$\{\mathbf{x}_i\}_{i=1}^{n}$ from $p(\mathbf{x})$ and $\{\mathbf{x}_i\}_{i=n+1}^{n+m}$ from $q(\mathbf{x})$ testing
$\mathcal{H}_A$: $p \neq q$ against $\mathcal{H}_0$: $p = q$

1. Label the first group $y_i = 1$ and the second group $y_i = -1$.
2. Score the observations $\{s_i := f(\mathbf{x}_i)\}_1^{n+m}$ with a learning machine f.
3. Calculate a univariate two-sample test statistic $T = T(\{s_i\}_1^n, \{s_i\}_{n+1}^{n+m})$.
4. Conduct statistical inference based on the permutation null distribution of the above statistic.

Friedman's Two-Sample Test

Friedman (2003)
$\{\mathbf{x}_i\}_{i=1}^{n}$ from $p(\mathbf{x})$ and $\{\mathbf{x}_i\}_{i=n+1}^{n+m}$ from $q(\mathbf{x})$ testing
$\mathcal{H}_A$: $p \neq q$ against $\mathcal{H}_0$: $p = q$

1. Label the first group $y_i = 1$ and the second group $y_i = -1$.
2. Score the observations $\{s_i := f(\mathbf{x}_i)\}_1^{n+m}$ with a learning machine f.
3. Calculate a univariate two-sample test statistic $T = T(\{s_i\}_1^n, \{s_i\}_{n+1}^{n+m})$.
4. Conduct statistical inference based on the permutation null distribution of the above statistic.

Friedman's Two-Sample Test

Friedman (2003)
$\{\mathbf{x}_i\}_{i=1}^{n}$ from $p(\mathbf{x})$ and $\{\mathbf{x}_i\}_{i=n+1}^{n+m}$ from $q(\mathbf{x})$ testing
$\mathcal{H}_A$: $p \neq q$ against $\mathcal{H}_0$: $p = q$

1. Label the first group $y_i = 1$ and the second group $y_i = -1$.
2. Score the observations $\{s_i := f(\mathbf{x}_i)\}_1^{n+m}$ with a learning machine f.
3. Calculate a univariate two-sample test statistic $T = T(\{s_i\}_1^n, \{s_i\}_{n+1}^{n+m})$.
4. Conduct statistical inference based on the permutation null distribution of the above statistic.

Friedman's Two-Sample Test

Friedman (2003)
$\{\mathbf{x}_i\}_{i=1}^{n}$ from $p(\mathbf{x})$ and $\{\mathbf{x}_i\}_{i=n+1}^{n+m}$ from $q(\mathbf{x})$ testing
$\mathcal{H}_A$: $p \neq q$ against $\mathcal{H}_0$: $p = q$

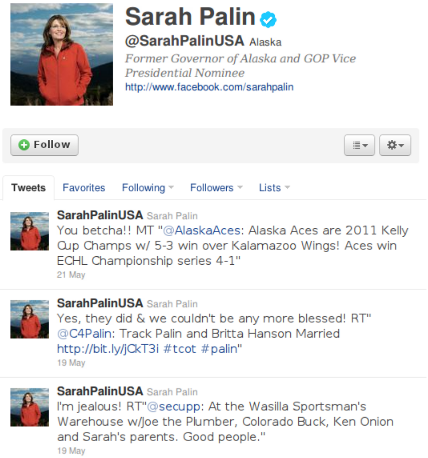
1. Label the first group $y_i = 1$ and the second group $y_i = -1$.
2. Score the observations $\{s_i := f(\mathbf{x}_i)\}_1^{n+m}$ with a learning machine f.
3. Calculate a univariate two-sample test statistic $T = T(\{s_i\}_1^n, \{s_i\}_{n+1}^{n+m})$.
4. Conduct statistical inference based on the permutation null distribution of the above statistic.

Friedman's Two-Sample Test

Friedman (2003)
$\{\mathbf{x}_i\}_{i=1}^{n}$ from $p(\mathbf{x})$ and $\{\mathbf{x}_i\}_{i=n+1}^{n+m}$ from $q(\mathbf{x})$ testing
$\mathcal{H}_A$: $p \neq q$ against $\mathcal{H}_0$: $p = q$

1. Label the first group $y_i = 1$ and the second group $y_i = -1$.
2. Score the observations $\{s_i := f(\mathbf{x}_i)\}_1^{n+m}$ with a learning machine f.
3. Calculate a univariate two-sample test statistic $T = T(\{s_i\}_1^n, \{s_i\}_{n+1}^{n+m})$.
4. Conduct statistical inference based on the permutation null distribution of the above statistic.

Twitter Example



Non-vectorial Data

```
"BarackObama: We need to reward education reforms that are
driven not by Washington, but by principals and teachers and
parents. http://OFA.BO/6p2EMy"
```

$\bar{x} = ?$

$\hat{\sigma}_x = ?$

Kernel methods allow us to lift ourselves up into an inner product space, where we can perform geometric calculations.

Non-vectorial Data

```
"BarackObama: We need to reward education reforms that are
driven not by Washington, but by principals and teachers and
parents. http://OFA.BO/6p2EMy"
```

$\bar{x} = ?$

$\hat{\sigma}_x = ?$

Kernel methods allow us to lift ourselves up into an inner product space, where we can perform geometric calculations.

Non-vectorial Data

`"BarackObama: We need to reward education reforms that are driven not by Washington, but by principals and teachers and parents. http://OFA.BO/6p2EMy"`

$\bar{x} = ?$
$\hat{\sigma}_x = ?$

Kernel methods allow us to lift ourselves up into an inner product space, where we can perform geometric calculations.

Non-vectorial Data

`"BarackObama: We need to reward education reforms that are driven not by Washington, but by principals and teachers and parents. http://OFA.BO/6p2EMy"`

$\bar{x} = ?$
$\hat{\sigma}_x = ?$
Kernel methods allow us to lift ourselves up into an inner product space, where we can perform geometric calculations.

Kernel Methods

The Kernel Trick (Aizerman et al. 1964)

- Data x_i in a general set $\mathcal{X}$.
- Define a feature map $\phi : \mathcal{X} \to \mathcal{H}$, where $\mathcal{H}$ is a Hilbert space.
- $K(\mathbf{x_i}, \mathbf{x_j}) = \langle \phi(\mathbf{x_i}), \phi(\mathbf{x_j}) \rangle_{\mathcal{H}}$
- Use learning algorithms that only require inner products between vectors in $\mathcal{X}$.
- The inner products can be done implicitly, by a kernel function $K : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$.

Kernel Methods

The Kernel Trick (Aizerman et al. 1964)

- Data x_i in a general set $\mathcal{X}$.
- Define a feature map $\phi : \mathcal{X} \to \mathcal{H}$, where $\mathcal{H}$ is a Hilbert space.
- $K(\mathbf{x_i}, \mathbf{x_j}) = \langle \phi(\mathbf{x_i}), \phi(\mathbf{x_j}) \rangle_{\mathcal{H}}$
- Use learning algorithms that only require inner products between vectors in $\mathcal{X}$.
- The inner products can be done implicitly, by a kernel function $K : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$.

Kernel Methods

The Kernel Trick (Aizerman et al. 1964)

- Data x_i in a general set $\mathcal{X}$.
- Define a feature map $\phi : \mathcal{X} \to \mathcal{H}$, where $\mathcal{H}$ is a Hilbert space.
- $K(\mathbf{x_i}, \mathbf{x_j}) = \langle \phi(\mathbf{x_i}), \phi(\mathbf{x_j}) \rangle_{\mathcal{H}}$
- Use learning algorithms that only require inner products between vectors in $\mathcal{X}$.
- The inner products can be done implicitly, by a kernel function $K : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$.

Kernel Methods

The Kernel Trick (Aizerman et al. 1964)

- Data x_i in a general set $\mathcal{X}$.
- Define a feature map $\phi : \mathcal{X} \to \mathcal{H}$, where $\mathcal{H}$ is a Hilbert space.
- $K(\mathbf{x_i}, \mathbf{x_j}) = \langle \phi(\mathbf{x_i}), \phi(\mathbf{x_j}) \rangle_{\mathcal{H}}$
- Use learning algorithms that only require inner products between vectors in $\mathcal{X}$.
- The inner products can be done implicitly, by a kernel function $K : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$.

Kernel Methods

The Kernel Trick (Aizerman et al. 1964)

- Data x_i in a general set $\mathcal{X}$.
- Define a feature map $\phi : \mathcal{X} \to \mathcal{H}$, where $\mathcal{H}$ is a Hilbert space.
- $K(\mathbf{x_i}, \mathbf{x_j}) = \langle \phi(\mathbf{x_i}), \phi(\mathbf{x_j}) \rangle_{\mathcal{H}}$
- Use learning algorithms that only require inner products between vectors in $\mathcal{X}$.
- The inner products can be done implicitly, by a kernel function $K : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$.

Kernel Methods

The Kernel Trick (Aizerman et al. 1964)

- Data x_i in a general set $\mathcal{X}$.
- Define a feature map $\phi : \mathcal{X} \to \mathcal{H}$, where $\mathcal{H}$ is a Hilbert space.
- $K(\mathbf{x_i}, \mathbf{x_j}) = \langle \phi(\mathbf{x_i}), \phi(\mathbf{x_j}) \rangle_{\mathcal{H}}$
- Use learning algorithms that only require inner products between vectors in $\mathcal{X}$.
- The inner products can be done implicitly, by a kernel function $K : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$.

Twitter Data

Raw:

```
"BarackObama: We need to reward education reforms that are
driven not by Washington, but by principals and teachers and
parents. http://OFA.BO/6p2EMy"
"SarahPalinUSA: You betcha!! MT \"@AlaskaAces: Alaska Aces
are 2011 Kelly Cup Champs w/ 5-3 win over Kalamazoo Wings!
Aces win  ECHL Championship series 4-1\""
```

After pre-processing:

```
"we need to reward education reforms that are driven not by
washington but by principals and teachers and parents "
"you betcha mt alaskaaces alaska aces are  kelly cup champs
w  win over kalamazoo wings aces win  echl championship
series "
```

The Spectrum Kernel

The Spectrum Kernel (Leslie 2002)

Compares two strings based on the their length k contiguous subsequences (k-mers).

- $\mathcal{X}$ = set of all finite-length sequences from an alphabet $\mathcal{A}$.
- $\phi_2(\mathbf{x}) = [\#_{\text{aa}}(\mathbf{x}), \#_{\text{ab}}(\mathbf{x}), \#_{\text{ac}}(\mathbf{x}), \ldots]$
- $\mathcal{H} = \mathbb{R}^{|\mathcal{A}|^k}$
- $K_k(\mathbf{x_i}, \mathbf{x_j}) = \langle \phi_k(\mathbf{x_i}), \phi_k(\mathbf{x_j}) \rangle$

The Spectrum Kernel

The Spectrum Kernel (Leslie 2002)
Compares two strings based on the their length k contiguous subsequences (k-mers).

- $\mathcal{X}$ = set of all finite-length sequences from an alphabet $\mathcal{A}$.
- $\phi_2(\mathbf{x}) = [\#_{\mathsf{aa}}(\mathbf{x}), \#_{\mathsf{ab}}(\mathbf{x}), \#_{\mathsf{ac}}(\mathbf{x}), \ldots]$
- $\mathcal{H} = \mathbb{R}^{|\mathcal{A}|^k}$
- $K_k(\mathbf{x_i}, \mathbf{x_j}) = \langle \phi_k(\mathbf{x_i}), \phi_k(\mathbf{x_j}) \rangle$

The Spectrum Kernel

The Spectrum Kernel (Leslie 2002)
Compares two strings based on the their length k contiguous subsequences (k-mers).

- $\mathcal{X}$ = set of all finite-length sequences from an alphabet $\mathcal{A}$.
- $\phi_2(\mathbf{x}) = [\#_{\text{aa}}(\mathbf{x}), \#_{\text{ab}}(\mathbf{x}), \#_{\text{ac}}(\mathbf{x}), \ldots]$
- $\mathcal{H} = \mathbb{R}^{|\mathcal{A}|^k}$
- $K_k(\mathbf{x_i}, \mathbf{x_j}) = \langle \phi_k(\mathbf{x_i}), \phi_k(\mathbf{x_j}) \rangle$

The Spectrum Kernel

The Spectrum Kernel (Leslie 2002)
Compares two strings based on the their length k contiguous subsequences (k-mers).

- $\mathcal{X} =$ set of all finite-length sequences from an alphabet $\mathcal{A}$.
- $\phi_2(\mathbf{x}) = [\#_{\mathrm{aa}}(\mathbf{x}), \#_{\mathrm{ab}}(\mathbf{x}), \#_{\mathrm{ac}}(\mathbf{x}), \ldots]$
- $\mathcal{H} = \mathbb{R}^{|\mathcal{A}|^k}$
- $K_k(\mathbf{x_i}, \mathbf{x_j}) = \langle \phi_k(\mathbf{x_i}), \phi_k(\mathbf{x_j}) \rangle$

The Spectrum Kernel

The Spectrum Kernel (Leslie 2002)
Compares two strings based on the their length k contiguous subsequences (k-mers).

- $\mathcal{X}$ = set of all finite-length sequences from an alphabet $\mathcal{A}$.
- $\phi_2(\mathbf{x}) = [\#_{\mathrm{aa}}(\mathbf{x}), \#_{\mathrm{ab}}(\mathbf{x}), \#_{\mathrm{ac}}(\mathbf{x}), \ldots]$
- $\mathcal{H} = \mathbb{R}^{|\mathcal{A}|^k}$
- $K_k(\mathbf{x_i}, \mathbf{x_j}) = \langle \phi_k(\mathbf{x_i}), \phi_k(\mathbf{x_j}) \rangle$

The Spectrum Kernel

The Spectrum Kernel (Leslie 2002)
Compares two strings based on the their length k contiguous subsequences (k-mers).

- $\mathcal{X}$ = set of all finite-length sequences from an alphabet $\mathcal{A}$.
- $\phi_2(\mathbf{x}) = [\#_{\mathsf{aa}}(\mathbf{x}), \#_{\mathsf{ab}}(\mathbf{x}), \#_{\mathsf{ac}}(\mathbf{x}), \ldots]$
- $\mathcal{H} = \mathbb{R}^{|\mathcal{A}|^k}$
- $K_k(\mathbf{x_i}, \mathbf{x_j}) = \langle \phi_k(\mathbf{x_i}), \phi_k(\mathbf{x_j}) \rangle$

Support Vector Machines

ℓ_1-regularized (soft-margin) support vector classification problem (Vapnik and Cortes, 1995):

$$\begin{aligned} \underset{\mathbf{w}\in\mathcal{H}, b\in\mathbb{R}}{\text{minimize}} \quad & \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{n+m}\xi_i \\ \text{subject to} \quad & y_i(\mathbf{w}^t\mathbf{x}_i + b) \geq 1 - \xi_i \\ & \xi_i \geq 0 \qquad\qquad \text{for all } i = 1, \ldots, n+m. \end{aligned}$$

For the Friedman Test, our scoring function is the margin $f(\mathbf{x}) = \sum_{i=1}^{n+m} y_i\alpha_i K(\mathbf{x}, \mathbf{x}_i) + b$.

Support Vector Machines

ℓ_1-regularized (soft-margin) support vector classification problem (Vapnik and Cortes, 1995):

$$\begin{aligned} \underset{\mathbf{w}\in\mathcal{H}, b\in\mathbb{R}}{\text{minimize}} \quad & \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{n+m}\xi_i \\ \text{subject to} \quad & y_i(\mathbf{w}^t\mathbf{x}_i + b) \geq 1 - \xi_i \\ & \xi_i \geq 0 \qquad\qquad \text{for all } i = 1, \ldots, n+m. \end{aligned}$$

For the Friedman Test, our scoring function is the margin $f(\mathbf{x}) = \sum_{i=1}^{n+m} y_i\alpha_i K(\mathbf{x}, \mathbf{x}_i) + b$.

Support Vector Machines

ℓ_1-regularized (soft-margin) support vector classification problem (Vapnik and Cortes, 1995):

$$\begin{aligned} \underset{\mathbf{w}\in\mathcal{H}, b\in\mathbb{R}}{\text{minimize}} \quad & \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{n+m}\xi_i \\ \text{subject to} \quad & y_i(\mathbf{w}^t\mathbf{x}_i + b) \geq 1 - \xi_i \\ & \xi_i \geq 0 \qquad\qquad \text{for all } i = 1, \ldots, n+m. \end{aligned}$$

For the Friedman Test, our scoring function is the margin $f(\mathbf{x}) = \sum_{i=1}^{n+m} y_i\alpha_i K(\mathbf{x}, \mathbf{x}_i) + b$.

KMMD

Kernel Maximum Mean Discrepancy Test: (Gretton et al. 2006)

$\mathfrak{F}$ a class of functions (unit ball in RKHS), $f : \mathcal{X} \to \mathbb{R}$, p and q probability distributions, and $X \sim p$ and $Z \sim q$ random variables

MMD statistic:

$$\mathrm{MMD}[\mathfrak{F}, p, q] := \sup_{f \in \mathfrak{F}} (\mathbb{E}_{x \sim p}[f(x)] - \mathbb{E}_{z \sim q}[f(z)])$$

Empirical Estimate:

$$\mathrm{MMD}[\mathfrak{F}, X, Z] := \sup_{f \in \mathfrak{F}} \left(\frac{1}{N} \sum_{i=1}^{N} f(x_i) - \frac{1}{M} \sum_{i=1}^{M} f(z_i) \right)$$

KMMD

Kernel Maximum Mean Discrepancy Test: (Gretton et al. 2006)
$\mathfrak{F}$ a class of functions (unit ball in RKHS), $f : \mathcal{X} \to \mathbb{R}$, p and q probability distributions, and $X \sim p$ and $Z \sim q$ random variables

MMD statistic:

$$\mathrm{MMD}[\mathfrak{F}, p, q] := \sup_{f \in \mathfrak{F}} (\mathbb{E}_{x \sim p}[f(x)] - \mathbb{E}_{z \sim q}[f(z)])$$

Empirical Estimate:

$$\mathrm{MMD}[\mathfrak{F}, X, Z] := \sup_{f \in \mathfrak{F}} \left(\frac{1}{N} \sum_{i=1}^{N} f(x_i) - \frac{1}{M} \sum_{i=1}^{M} f(z_i) \right)$$

KMMD

Kernel Maximum Mean Discrepancy Test: (Gretton et al. 2006)
$\mathfrak{F}$ a class of functions (unit ball in RKHS), $f : \mathcal{X} \to \mathbb{R}$, p and q probability distributions, and $X \sim p$ and $Z \sim q$ random variables
MMD statistic:

$$\mathrm{MMD}[\mathfrak{F}, p, q] := \sup_{f \in \mathfrak{F}} (\mathbb{E}_{x \sim p}[f(x)] - \mathbb{E}_{z \sim q}[f(z)])$$

Empirical Estimate:

$$\mathrm{MMD}[\mathfrak{F}, X, Z] := \sup_{f \in \mathfrak{F}} \left(\frac{1}{N} \sum_{i=1}^{N} f(x_i) - \frac{1}{M} \sum_{i=1}^{M} f(z_i) \right)$$

KMMD

Kernel Maximum Mean Discrepancy Test: (Gretton et al. 2006)
$\mathfrak{F}$ a class of functions (unit ball in RKHS), $f : \mathcal{X} \to \mathbb{R}$, p and q probability distributions, and $X \sim p$ and $Z \sim q$ random variables
MMD statistic:

$$\mathrm{MMD}[\mathfrak{F}, p, q] := \sup_{f \in \mathfrak{F}} (\mathbb{E}_{x \sim p}[f(x)] - \mathbb{E}_{z \sim q}[f(z)])$$

Empirical Estimate:

$$\mathrm{MMD}[\mathfrak{F}, X, Z] := \sup_{f \in \mathfrak{F}} \left(\frac{1}{N} \sum_{i=1}^{N} f(x_i) - \frac{1}{M} \sum_{i=1}^{M} f(z_i) \right)$$

Twitter Example

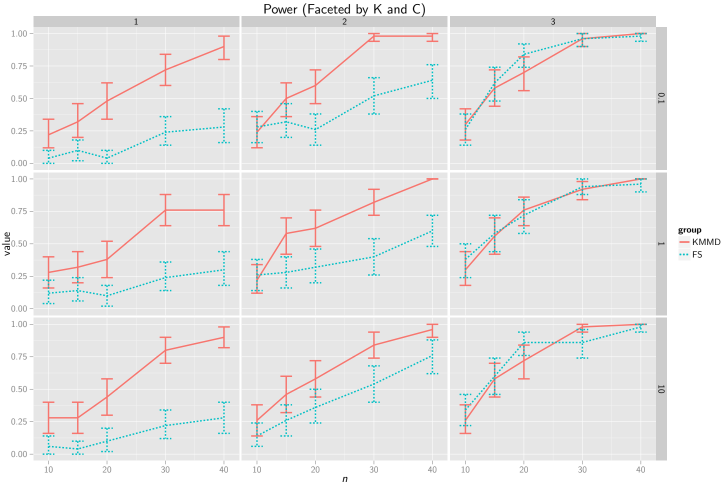


Image Data (Roosters)

Caltech 101 Object Categories (Li et al. 2007) (297×300 grayscale)

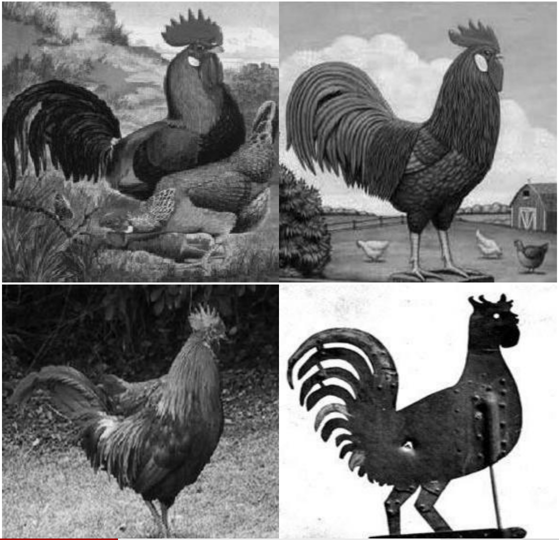
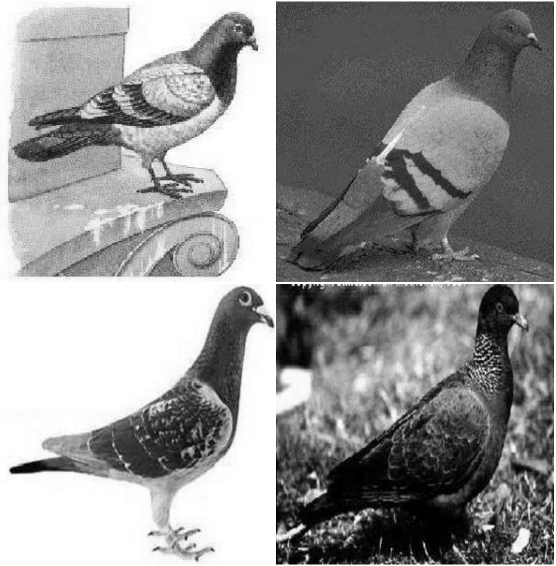


Image Data (Pigeons)



Polynomial Kernel

Compares 2 vectors (images) on products of elements (pixel intensities) up to a certain order.

- $\mathcal{X} = \mathbb{R}^p$
- $\phi_2([x_1, x_2]) = [x_1^2, 2x_1x_2, x_2^2, \sqrt{2c}x_1, \sqrt{2c}x_2, c]$
- $\langle \phi_2(\mathbf{x}), \phi_2(\mathbf{y}) \rangle$ is $\mathcal{O}(n^2)$
- $\mathcal{H} = \mathbb{R}^{d'}$, where $d' = \binom{n+d}{d}$
- $K_d(\mathbf{x}, \mathbf{y}) = (\mathbf{x}^T\mathbf{y} + c)^d$ is $\mathcal{O}(n)$

Polynomial Kernel

Compares 2 vectors (images) on products of elements (pixel intensities) up to a certain order.

- $\mathcal{X} = \mathbb{R}^p$
- $\phi_2([x_1, x_2]) = [x_1^2, 2x_1x_2, x_2^2, \sqrt{2c}x_1, \sqrt{2c}x_2, c]$
- $\langle \phi_2(\mathbf{x}), \phi_2(\mathbf{y}) \rangle$ is $\mathcal{O}(n^2)$
- $\mathcal{H} = \mathbb{R}^{d'}$, where $d' = \binom{n+d}{d}$
- $K_d(\mathbf{x}, \mathbf{y}) = (\mathbf{x}^T\mathbf{y} + c)^d$ is $\mathcal{O}(n)$

Polynomial Kernel

Compares 2 vectors (images) on products of elements (pixel intensities) up to a certain order.

- $\mathcal{X} = \mathbb{R}^p$
- $\phi_2([x_1, x_2]) = [x_1^2, 2x_1x_2, x_2^2, \sqrt{2c}x_1, \sqrt{2c}x_2, c]$
- $\langle \phi_2(\mathbf{x}), \phi_2(\mathbf{y}) \rangle$ is $\mathcal{O}(n^2)$
- $\mathcal{H} = \mathbb{R}^{d'}$, where $d' = \binom{n+d}{d}$
- $K_d(\mathbf{x}, \mathbf{y}) = (\mathbf{x}^T\mathbf{y} + c)^d$ is $\mathcal{O}(n)$

Polynomial Kernel

Compares 2 vectors (images) on products of elements (pixel intensities) up to a certain order.

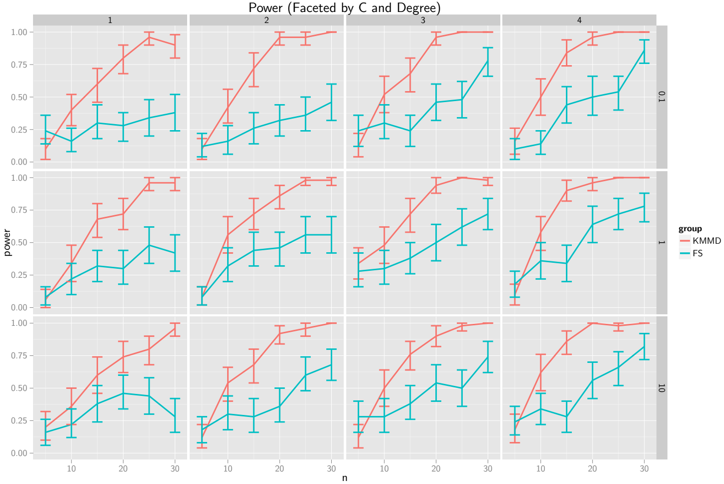
- $\mathcal{X} = \mathbb{R}^p$
- $\phi_2([x_1, x_2]) = [x_1^2, 2x_1x_2, x_2^2, \sqrt{2c}x_1, \sqrt{2c}x_2, c]$
- $\langle \phi_2(\mathbf{x}), \phi_2(\mathbf{y}) \rangle$ is $\mathcal{O}(n^2)$
- $\mathcal{H} = \mathbb{R}^{d'}$, where $d' = \binom{n+d}{d}$
- $K_d(\mathbf{x}, \mathbf{y}) = (\mathbf{x}^T\mathbf{y} + c)^d$ is $\mathcal{O}(n)$

Polynomial Kernel

Compares 2 vectors (images) on products of elements (pixel intensities) up to a certain order.

- $\mathcal{X} = \mathbb{R}^p$
- $\phi_2([x_1, x_2]) = [x_1^2, 2x_1x_2, x_2^2, \sqrt{2c}x_1, \sqrt{2c}x_2, c]$
- $\langle \phi_2(\mathbf{x}), \phi_2(\mathbf{y}) \rangle$ is $\mathcal{O}(n^2)$
- $\mathcal{H} = \mathbb{R}^{d'}$, where $d' = \binom{n+d}{d}$
- $K_d(\mathbf{x}, \mathbf{y}) = (\mathbf{x}^T\mathbf{y} + c)^d$ is $\mathcal{O}(n)$

Rooster/Pigeon Example



Regression and MKL

Regularized regression

- Feature engineering/extraction: $\mathbf{x}_i$
- Feature normalization: $\mathbf{x}_i \leftarrow \frac{\mathbf{x}_i - \hat{\mu}_i}{\hat{\sigma}_i}$
- Regularization/feature selection: $\inf_\beta \sum_{i=1}^{m+n} L(\beta_0 + \tilde{\mathbf{x}}_i^T \beta, y_i)$ s.t. $||\beta||_p \leq t$

MKL

- Feature engineering/extraction: K_i
- Feature normalization: $K_i(\mathbf{x}, \mathbf{x}') \leftarrow \frac{K_i(\mathbf{x}, \mathbf{x}')}{\sqrt{K_i(\mathbf{x}, \mathbf{x})}\sqrt{K_i(\mathbf{x}', \mathbf{x}')}}$
- Regularization/feature selection (Kloft et al. 2011)):

$$\inf_{\mathbf{w}, b, \theta : \theta \succeq 0} C \sum_{i=1}^{m+n} L\left(\sum_{j=1}^{M} \sqrt{\theta_j} \langle \mathbf{w}_j, \phi_j(\mathbf{x_i}) \rangle_{\mathcal{H}_j} + b, y_i\right) + \frac{1}{2} \sum_{j=1}^{M} ||\mathbf{w}_j||_{\mathcal{H}_j}^2 \text{ s.t. } ||\theta||_p \leq 1$$

Regression and MKL

Regularized regression

- Feature engineering/extraction: $\mathbf{x}_i$
- Feature normalization: $\mathbf{x}_i \leftarrow \frac{\mathbf{x}_i - \hat{\mu}_i}{\hat{\sigma}_i}$
- Regularization/feature selection:
  $\inf_\beta \sum_{i=1}^{m+n} L(\beta_0 + \tilde{\mathbf{x}}_i^T \beta, y_i)$ s.t. $||\beta||_p \leq t$

MKL

- Feature engineering/extraction: K_i
- Feature normalization: $K_i(\mathbf{x}, \mathbf{x}') \leftarrow \frac{K_i(\mathbf{x},\mathbf{x}')}{\sqrt{K_i(\mathbf{x},\mathbf{x})}\sqrt{K_i(\mathbf{x}',\mathbf{x}')}}$
- Regularization/feature selection (Kloft et al. 2011)):
  $$\inf_{\mathbf{w},b,\theta:\theta\succeq 0} C \sum_{i=1}^{m+n} L(\sum_{j=1}^{M} \sqrt{\theta_j} \langle \mathbf{w}_j, \phi_j(\mathbf{x_i}) \rangle_{\mathcal{H}_j} + b, y_i) + \frac{1}{2} \sum_{j=1}^{M} ||\mathbf{w}_j||_{\mathcal{H}_j}^2 \text{ s.t. } ||\theta||_p \leq 1$$

Regression and MKL

Regularized regression

- Feature engineering/extraction: $\mathbf{x}_i$
- Feature normalization: $\mathbf{x}_i \leftarrow \frac{\mathbf{x}_i - \hat{\mu}_i}{\hat{\sigma}_i}$
- Regularization/feature selection: $\inf_\beta \sum_{i=1}^{m+n} L(\beta_0 + \tilde{\mathbf{x}}_i^T \beta, y_i)$ s.t. $||\beta||_p \leq t$

MKL

- Feature engineering/extraction: K_i
- Feature normalization: $K_i(\mathbf{x}, \mathbf{x}') \leftarrow \frac{K_i(\mathbf{x},\mathbf{x}')}{\sqrt{K_i(\mathbf{x},\mathbf{x})}\sqrt{K_i(\mathbf{x}',\mathbf{x}')}}$
- Regularization/feature selection (Kloft et al. 2011)):
$$\inf_{\mathbf{w},b,\theta:\theta\succeq 0} C \sum_{i=1}^{m+n} L\left(\sum_{j=1}^{M} \sqrt{\theta_j}\langle \mathbf{w}_j, \phi_j(\mathbf{x_i})\rangle_{\mathcal{H}_j} + b, y_i\right) + \frac{1}{2}\sum_{j=1}^{M} ||\mathbf{w}_j||^2_{\mathcal{H}_j} \text{ s.t. } ||\theta||_p \leq 1$$

Regression and MKL

Regularized regression

- Feature engineering/extraction: $\mathbf{x}_i$
- Feature normalization: $\mathbf{x}_i \leftarrow \frac{\mathbf{x}_i - \hat{\mu}_i}{\hat{\sigma}_i}$
- Regularization/feature selection: $\inf_{\boldsymbol{\beta}} \sum_{i=1}^{m+n} L(\beta_0 + \tilde{\mathbf{x}}_i^T \boldsymbol{\beta}, y_i)$ s.t. $||\boldsymbol{\beta}||_p \leq t$

MKL

- Feature engineering/extraction: K_i
- Feature normalization: $K_i(\mathbf{x}, \mathbf{x}') \leftarrow \frac{K_i(\mathbf{x},\mathbf{x}')}{\sqrt{K_i(\mathbf{x},\mathbf{x})}\sqrt{K_i(\mathbf{x}',\mathbf{x}')}}$
- Regularization/feature selection (Kloft et al. 2011)): $$\inf_{\mathbf{w},b,\boldsymbol{\theta}:\boldsymbol{\theta}\succeq 0} C \sum_{i=1}^{m+n} L\left(\sum_{j=1}^{M} \sqrt{\theta_j}\langle \mathbf{w}_j, \phi_j(\mathbf{x_i})\rangle_{\mathcal{H}_j} + b, y_i\right) + \frac{1}{2}\sum_{j=1}^{M} ||\mathbf{w}_j||^2_{\mathcal{H}_j} \text{ s.t. } ||\boldsymbol{\theta}||_p \leq 1$$

Regression and MKL

Regularized regression

- Feature engineering/extraction: $\mathbf{x}_i$
- Feature normalization: $\mathbf{x}_i \leftarrow \frac{\mathbf{x}_i - \hat{\mu}_i}{\hat{\sigma}_i}$
- Regularization/feature selection: $\inf_{\boldsymbol{\beta}} \sum_{i=1}^{m+n} L(\beta_0 + \tilde{\mathbf{x}}_i^T \boldsymbol{\beta}, y_i)$ s.t. $||\boldsymbol{\beta}||_p \leq t$

MKL

- Feature engineering/extraction: K_i
- Feature normalization: $K_i(\mathbf{x}, \mathbf{x}') \leftarrow \frac{K_i(\mathbf{x},\mathbf{x}')}{\sqrt{K_i(\mathbf{x},\mathbf{x})}\sqrt{K_i(\mathbf{x}',\mathbf{x}')}}$
- Regularization/feature selection (Kloft et al. 2011)): $\inf_{\mathbf{w},b,\boldsymbol{\theta}:\boldsymbol{\theta}\succeq 0} C \sum_{i=1}^{m+n} L(\sum_{j=1}^{M} \sqrt{\theta_j} \langle \mathbf{w}_j, \phi_j(\mathbf{x_i}) \rangle_{\mathcal{H}_j} + b, y_i)$
  $+\frac{1}{2} \sum_{j=1}^{M} ||\mathbf{w}_j||_{\mathcal{H}_j}^2$ s.t. $||\boldsymbol{\theta}||_p \leq 1$

Regression and MKL

Regularized regression

- Feature engineering/extraction: $\mathbf{x}_i$
- Feature normalization: $\mathbf{x}_i \leftarrow \frac{\mathbf{x}_i - \hat{\mu}_i}{\hat{\sigma}_i}$
- Regularization/feature selection: $\inf_{\boldsymbol{\beta}} \sum_{i=1}^{m+n} L(\beta_0 + \tilde{\mathbf{x}}_i^T \boldsymbol{\beta}, y_i)$ s.t. $||\boldsymbol{\beta}||_p \leq t$

MKL

- Feature engineering/extraction: K_i
- Feature normalization: $K_i(\mathbf{x}, \mathbf{x}') \leftarrow \frac{K_i(\mathbf{x}, \mathbf{x}')}{\sqrt{K_i(\mathbf{x}, \mathbf{x})}\sqrt{K_i(\mathbf{x}', \mathbf{x}')}}$
- Regularization/feature selection (Kloft et al. 2011)): $\inf_{\mathbf{w}, b, \boldsymbol{\theta} : \boldsymbol{\theta} \succeq 0} C \sum_{i=1}^{m+n} L(\sum_{j=1}^{M} \sqrt{\theta_j} \langle \mathbf{w}_j, \phi_j(\mathbf{x_i}) \rangle_{\mathcal{H}_j} + b, y_i)$
$+\frac{1}{2} \sum_{j=1}^{M} ||\mathbf{w}_j||_{\mathcal{H}_j}^2$ s.t. $||\boldsymbol{\theta}||_p \leq 1$

Regression and MKL

Regularized regression

- Feature engineering/extraction: $\mathbf{x}_i$
- Feature normalization: $\mathbf{x}_i \leftarrow \frac{\mathbf{x}_i - \hat{\mu}_i}{\hat{\sigma}_i}$
- Regularization/feature selection:
  $\inf_{\boldsymbol{\beta}} \sum_{i=1}^{m+n} L(\beta_0 + \tilde{\mathbf{x}}_i^T \boldsymbol{\beta}, y_i)$ s.t. $||\boldsymbol{\beta}||_p \leq t$

MKL

- Feature engineering/extraction: K_i
- Feature normalization: $K_i(\mathbf{x}, \mathbf{x}') \leftarrow \frac{K_i(\mathbf{x},\mathbf{x}')}{\sqrt{K_i(\mathbf{x},\mathbf{x})}\sqrt{K_i(\mathbf{x}',\mathbf{x}')}}$
- Regularization/feature selection (Kloft et al. 2011)):
  $\inf_{\mathbf{w},b,\boldsymbol{\theta}:\boldsymbol{\theta}\succeq 0} C \sum_{i=1}^{m+n} L(\sum_{j=1}^{M} \sqrt{\theta_j}\langle \mathbf{w}_j, \phi_j(\mathbf{x_i})\rangle_{\mathcal{H}_j} + b, y_i)$
  $+\frac{1}{2}\sum_{j=1}^{M} ||\mathbf{w}_j||^2_{\mathcal{H}_j}$ s.t. $||\boldsymbol{\theta}||_p \leq 1$

Regression and MKL

Regularized regression

- Feature engineering/extraction: $\mathbf{x}_i$
- Feature normalization: $\mathbf{x}_i \leftarrow \frac{\mathbf{x}_i - \hat{\mu}_i}{\hat{\sigma}_i}$
- Regularization/feature selection:
  $\inf_{\boldsymbol{\beta}} \sum_{i=1}^{m+n} L(\beta_0 + \tilde{\mathbf{x}}_i^T \boldsymbol{\beta}, y_i)$ s.t. $||\boldsymbol{\beta}||_p \leq t$

MKL

- Feature engineering/extraction: K_i
- Feature normalization: $K_i(\mathbf{x}, \mathbf{x}') \leftarrow \frac{K_i(\mathbf{x},\mathbf{x}')}{\sqrt{K_i(\mathbf{x},\mathbf{x})}\sqrt{K_i(\mathbf{x}',\mathbf{x}')}}$
- Regularization/feature selection (Kloft et al. 2011)):
  $$\inf_{\mathbf{w},b,\boldsymbol{\theta}:\boldsymbol{\theta}\succeq 0} C \sum_{i=1}^{m+n} L\left(\sum_{j=1}^{M} \sqrt{\theta_j} \langle \mathbf{w}_j, \phi_j(\mathbf{x_i}) \rangle_{\mathcal{H}_j} + b, y_i\right) + \frac{1}{2} \sum_{j=1}^{M} ||\mathbf{w}_j||_{\mathcal{H}_j}^2 \text{ s.t. } ||\boldsymbol{\theta}||_p \leq 1$$

Simulated Data (DNA)

Generate independent DNA sequences of length $N \sim \text{Pois}(100)$ according to the transition matrix

$$M(p^\star) = \begin{array}{c} \\ A \\ C \\ T \\ G \end{array} \begin{array}{c} \begin{array}{cccc} A & C & T & G \end{array} \\ \begin{pmatrix} \frac{1-p^\star}{3} & p^\star & \frac{1-p^\star}{3} & \frac{1-p^\star}{3} \\ \frac{1-p^\star}{3} & \frac{1-p^\star}{3} & p^\star & \frac{1-p^\star}{3} \\ \frac{1-p^\star}{3} & \frac{1-p^\star}{3} & \frac{1-p^\star}{3} & p^\star \\ p^\star & \frac{1-p^\star}{3} & \frac{1-p^\star}{3} & \frac{1-p^\star}{3} \end{pmatrix} \end{array}$$

with stationary distribution [.25, .25, .25, .25].

p takes $p^\star = .25$, and q takes $p^\star > .25$.

p and q generate similar numbers of 1-mers, but q can generate more AC, CT, TG, GA 2-mers.

Simulated Data (DNA)

Generate independent DNA sequences of length $N \sim \text{Pois}(100)$ according to the transition matrix

$$M(p^\star) = \begin{array}{c c} & \begin{array}{cccc} A & C & T & G \end{array} \\ \begin{array}{c} A \\ C \\ T \\ G \end{array} & \begin{pmatrix} \frac{1-p^\star}{3} & p^\star & \frac{1-p^\star}{3} & \frac{1-p^\star}{3} \\ \frac{1-p^\star}{3} & \frac{1-p^\star}{3} & p^\star & \frac{1-p^\star}{3} \\ \frac{1-p^\star}{3} & \frac{1-p^\star}{3} & \frac{1-p^\star}{3} & p^\star \\ p^\star & \frac{1-p^\star}{3} & \frac{1-p^\star}{3} & \frac{1-p^\star}{3} \end{pmatrix} \end{array}$$

with stationary distribution [.25, .25, .25, .25].
p takes $p^\star = .25$, and q takes $p^\star > .25$.
p and q generate similar numbers of 1-mers, but q can generate more AC, CT, TG, GA 2-mers.

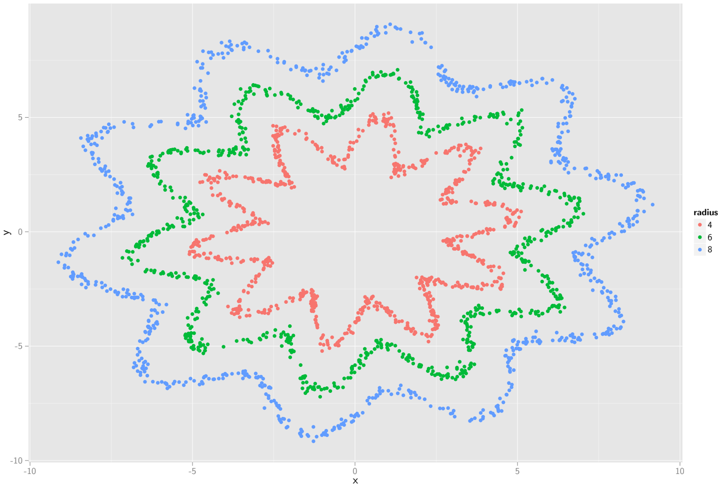
Simulated Data (DNA)

Generate independent DNA sequences of length $N \sim \text{Pois}(100)$ according to the transition matrix

$$M(p^\star) = \begin{array}{c c} & \begin{array}{cccc} A & C & T & G \end{array} \\ \begin{array}{c} A \\ C \\ T \\ G \end{array} & \begin{pmatrix} \frac{1-p^\star}{3} & p^\star & \frac{1-p^\star}{3} & \frac{1-p^\star}{3} \\ \frac{1-p^\star}{3} & \frac{1-p^\star}{3} & p^\star & \frac{1-p^\star}{3} \\ \frac{1-p^\star}{3} & \frac{1-p^\star}{3} & \frac{1-p^\star}{3} & p^\star \\ p^\star & \frac{1-p^\star}{3} & \frac{1-p^\star}{3} & \frac{1-p^\star}{3} \end{pmatrix} \end{array}$$

with stationary distribution $[.25, .25, .25, .25]$.
p takes $p^\star = .25$, and q takes $p^\star > .25$.
p and q generate similar numbers of 1-mers, but q can generate more AC, CT, TG, GA 2-mers.

Simulated Data (Star)



Two-Sample Tests

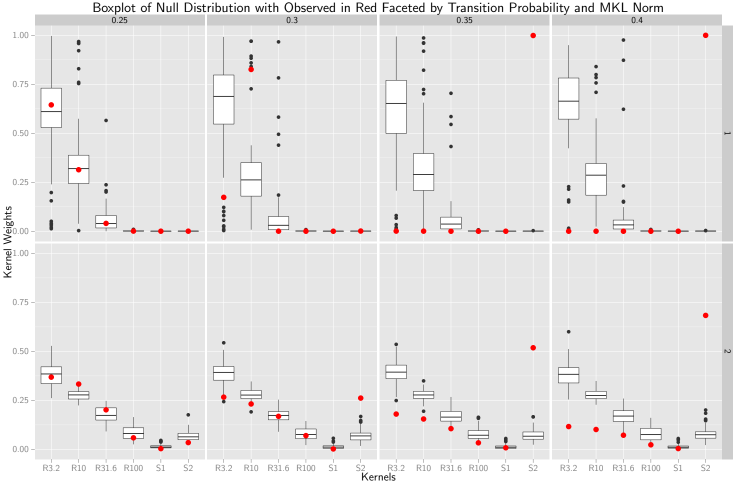
Two-sample tests typically provide 1 bit of information: accept or reject.

The MKL-based two-sample test generates the observed kernel weight vector θ and its permuted values $\theta^{(i)}$.

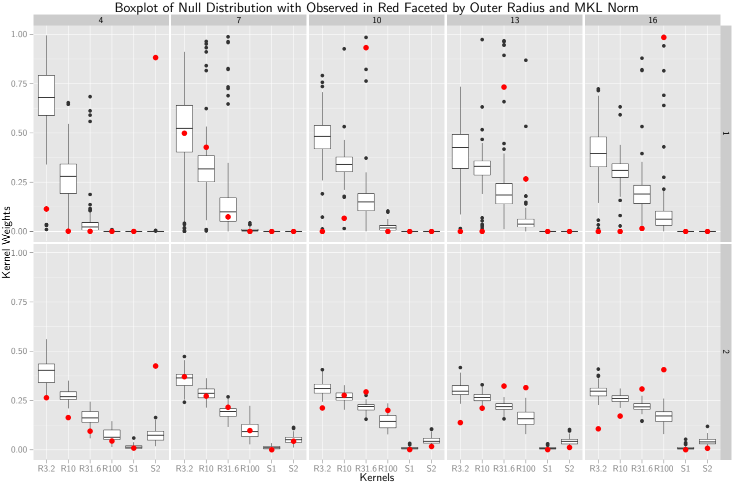
Two-Sample Tests

Two-sample tests typically provide 1 bit of information: accept or reject. The MKL-based two-sample test generates the observed kernel weight vector $\boldsymbol{\theta}$ and its permuted values $\boldsymbol{\theta}^{(i)}$.

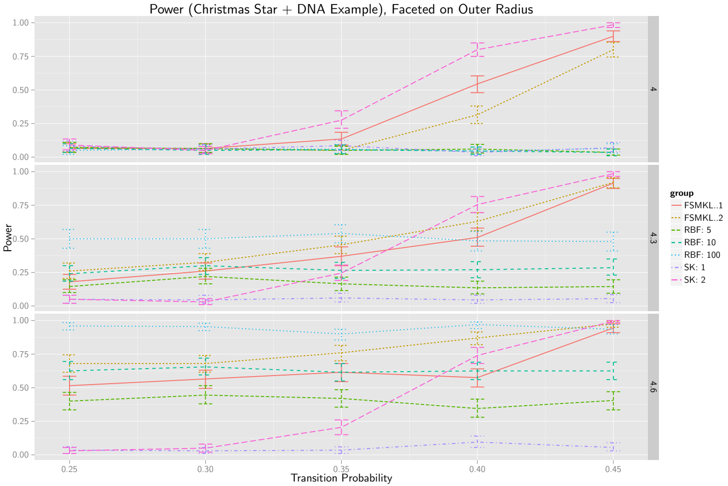
MKL Weights


Boxplot of Null Distribution with Observed in Red Faceted by Transition Probability and MKL Norm
0.25
0.3
0.35
0.4
1
2
Kernel Weights
1.00
0.75
0.50
0.25
0.00
R3.2
R10
R31.6
R100
S1
S2
Kernels

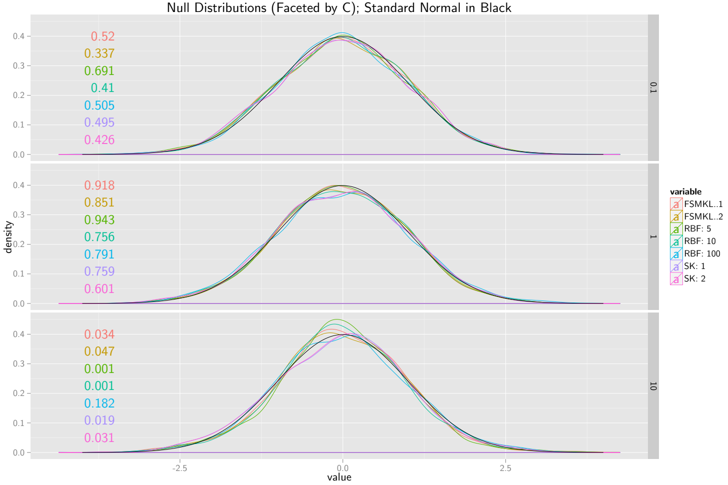
MKL Weights


Boxplot of Null Distribution with Observed in Red Faceted by Outer Radius and MKL Norm
4
7
10
13
16
1
2
Kernel Weights
1.00
0.75
0.50
0.25
0.00
R3.2 R10 R31.6 R100 S1 S2
Kernels

MKL Power


Power (Christmas Star + DNA Example), Faceted on Outer Radius
Power
Transition Probability
1.00
0.75
0.50
0.25
0.00
0.25
0.30
0.35
0.40
0.45
4
4.3
4.6
group
FSMKL..1
FSMKL..2
RBF: 5
RBF: 10
RBF: 100
SK: 1
SK: 2

MKL Null Distribution



Permutation t-test Connection

Want to understand theoretically why the randomization null is normal-like.

And we wish to derive bounds on

$$\sup_{t \in \mathbb{R}} |P(T_\Pi \le t) - \Phi(t)|,$$

where

$$T_\Pi = T_\Pi \left(\{f(\mathbf{x}_{\Pi(i)})\}_{i=1}^{n}, \{f(\mathbf{x}_{\Pi(i)})\}_{i=n+1}^{2n} \right).$$

If $f(\mathbf{x})$ is affine, we recover the permutation t-test.
SVM with linear kernel: $f(x) = \sum_{i=1}^{2n} y_i \alpha_i x x_i + b$.

Permutation t-test Connection

Want to understand theoretically why the randomization null is normal-like.
And we wish to derive bounds on

$$\sup_{t\in\mathbb{R}} \left|P(T_\Pi \leq t) - \Phi(t)\right|,$$

where

$$T_\Pi = T_\Pi \left(\{f(\mathbf{x}_{\Pi(i)})\}_{i=1}^{n}, \{f(\mathbf{x}_{\Pi(i)})\}_{i=n+1}^{2n}\right).$$

If $f(\mathbf{x})$ is affine, we recover the permutation t-test.
SVM with linear kernel: $f(x) = \sum_{i=1}^{2n} y_i \alpha_i x x_i + b$.

Permutation t-test Connection

Want to understand theoretically why the randomization null is normal-like.
And we wish to derive bounds on

$$\sup_{t \in \mathbb{R}} |P(T_\Pi \leq t) - \Phi(t)|,$$

where

$$T_\Pi = T_\Pi \left(\{f(\mathbf{x}_{\Pi(i)})\}_{i=1}^{n}, \{f(\mathbf{x}_{\Pi(i)})\}_{i=n+1}^{2n} \right).$$

If $f(\mathbf{x})$ is affine, we recover the permutation t-test.
SVM with linear kernel: $f(x) = \sum_{i=1}^{2n} y_i \alpha_i x x_i + b$.

Permutation t-test Connection

Want to understand theoretically why the randomization null is normal-like.
And we wish to derive bounds on

$$\sup_{t \in \mathbb{R}} |P(T_\Pi \leq t) - \Phi(t)|,$$

where

$$T_\Pi = T_\Pi \left(\{f(\mathbf{x}_{\Pi(i)})\}_{i=1}^{n}, \{f(\mathbf{x}_{\Pi(i)})\}_{i=n+1}^{2n} \right).$$

If $f(\mathbf{x})$ is affine, we recover the permutation t-test.
SVM with linear kernel: $f(x) = \sum_{i=1}^{2n} y_i \alpha_i x x_i + b$.

Other Work

- Fisher (1935) proposed distribution-free randomization test.
- Lehmann proved a normal convergence result for the randomization distribution.
- Bentkus et al. (1996), Shao (2005) proved Berry–Esseen bounds for Student's t-statistic in independent case.

Other Work

- Fisher (1935) proposed distribution-free randomization test.
- Lehmann proved a normal convergence result for the randomization distribution.
- Bentkus et al. (1996), Shao (2005) proved Berry–Esseen bounds for Student's t-statistic in independent case.

Other Work

- Fisher (1935) proposed distribution-free randomization test.
- Lehmann proved a normal convergence result for the randomization distribution.
- Bentkus et al. (1996), Shao (2005) proved Berry–Esseen bounds for Student's t-statistic in independent case.

Other Results

Theorem (Berry–Esseen 1941, 1942)

Suppose $X_1, \ldots, X_n$ are i.i.d. random variables with $\mathbb{E}X_i = 0$, $\mathbb{E}X_i^2 = \sigma^2 > 0$, and $\mathbb{E}|X_i|^3 = \rho < \infty$. Let $F_n(x)$ denote the CDF of standardized sample mean of the X_i. Then

$$\sup_x |F_n(x) - \Phi(x)| \leq \frac{0.33477(\rho + 0.429\sigma^3)}{\sigma^3\sqrt{n}}$$

$$= \frac{C}{\sqrt{n}} f(\rho, \sigma).$$

Note that ρ and σ are fixed as $n \to \infty$.

Other Results

Theorem (Berry–Esseen 1941, 1942)

Suppose $X_1, \ldots, X_n$ are i.i.d. random variables with $\mathbb{E}X_i = 0$, $\mathbb{E}X_i^2 = \sigma^2 > 0$, and $\mathbb{E}|X_i|^3 = \rho < \infty$. Let $F_n(x)$ denote the CDF of standardized sample mean of the X_i. Then

$$\begin{aligned} \sup_x |F_n(x) - \Phi(x)| &\leq \frac{0.33477(\rho + 0.429\sigma^3)}{\sigma^3\sqrt{n}} \\ &= \frac{C}{\sqrt{n}} f(\rho, \sigma). \end{aligned}$$

Note that ρ and σ are fixed as $n \to \infty$.

Other Results

Theorem (Berry–Esseen 1941, 1942)

Suppose $X_1, \ldots, X_n$ are i.i.d. random variables with $\mathbb{E}X_i = 0$, $\mathbb{E}X_i^2 = \sigma^2 > 0$, and $\mathbb{E}|X_i|^3 = \rho < \infty$. Let $F_n(x)$ denote the CDF of standardized sample mean of the X_i. Then

$$\begin{aligned} \sup_x |F_n(x) - \Phi(x)| &\leq \frac{0.33477(\rho + 0.429\sigma^3)}{\sigma^3\sqrt{n}} \\ &= \frac{C}{\sqrt{n}} f(\rho, \sigma). \end{aligned}$$

Note that ρ and σ are fixed as $n \to \infty$.

Other Results

Theorem (Hoeffding 1951, Stein 1986)

Let $A = \{a_{ij}\}_{i,j\in\{1,\dots,n\}}$ be a square array of numbers such that $\sum_j a_{ij} = 0$ for all i, $\sum_i a_{ij} = 0$ for all j, and $\sum_i \sum_j a_{ij}^2 = n-1$. Then with $F_n(x) = P(\sum_i a_{i\Pi(i)} \le x)$,

$$\sup_x |F_n(x) - \Phi(x)| \le \frac{C}{\sqrt{n}} \left(\sqrt{\sum_{i,j} a_{ij}^4} + \sqrt{\sum_{i,j} |a_{ij}|^3} \right)$$

$$= \frac{C}{\sqrt{n}} f(A).$$

Given a sampling scheme for A, $f(A)$ must be $\mathcal{O}(1)$ to have rate $\mathcal{O}(n^{-1/2})$.

Other Results

Theorem (Hoeffding 1951, Stein 1986)

Let $A = \{a_{ij}\}_{i,j \in \{1,\dots,n\}}$ be a square array of numbers such that $\sum_j a_{ij} = 0$ for all i, $\sum_i a_{ij} = 0$ for all j, and $\sum_i \sum_j a_{ij}^2 = n - 1$. Then with $F_n(x) = P(\sum_i a_{i\Pi(i)} \leq x)$,

$$\begin{aligned} \sup_x |F_n(x) - \Phi(x)| &\leq \frac{C}{\sqrt{n}} \left(\sqrt{\sum_{i,j} a_{ij}^4} + \sqrt{\sum_{i,j} |a_{ij}|^3} \right) \\ &= \frac{C}{\sqrt{n}} f(A). \end{aligned}$$

Given a sampling scheme for A, $f(A)$ must be $\mathcal{O}(1)$ to have rate $\mathcal{O}(n^{-1/2})$.

Other Results

Theorem (Hoeffding 1951, Stein 1986)

Let $A = \{a_{ij}\}_{i,j \in \{1,\ldots,n\}}$ *be a square array of numbers such that* $\sum_j a_{ij} = 0$ *for all* i, $\sum_i a_{ij} = 0$ *for all* j, *and* $\sum_i \sum_j a_{ij}^2 = n - 1$. *Then with* $F_n(x) = P(\sum_i a_{i\Pi(i)} \le x)$,

$$\begin{aligned} \sup_x |F_n(x) - \Phi(x)| &\le \frac{C}{\sqrt{n}} \left(\sqrt{\sum_{i,j} a_{ij}^4} + \sqrt{\sum_{i,j} |a_{ij}|^3} \right) \\ &= \frac{C}{\sqrt{n}} f(A). \end{aligned}$$

Given a sampling scheme for A, $f(A)$ must be $\mathcal{O}(1)$ to have rate $\mathcal{O}(n^{-1/2})$.

Exchangeable Pair

Assume $m = n$. Fix data $\{u_1, \ldots, u_n, u_{n+1}, \ldots, u_{2n}\}$. Π is a uniformly random permutation, and let

$$T_\Pi = T_\Pi\left(\{u_{\Pi(i)}\}_{i=1}^{n}, \{u_{\Pi(i)}\}_{i=n+1}^{2n}\right).$$

Let $(I, J) = (i, j)$ w.p. $\frac{1}{n^2}$ for $1 \leq i \leq n$ and $n+1 \leq j \leq 2n$. Then

$$T' = T\left(\{u_{\Pi\circ(I,J)(i)}\}_{i=1}^{n}, \{u_{\Pi\circ(I,J)(i)}\}_{i=n+1}^{2n}\right).$$

T and T' form an exchangeable pair.

Exchangeable Pair

Assume $m = n$. Fix data $\{u_1, \ldots, u_n, u_{n+1}, \ldots, u_{2n}\}$. Π is a uniformly random permutation, and let

$$T_\Pi = T_\Pi\left(\{u_{\Pi(i)}\}_{i=1}^{n}, \{u_{\Pi(i)}\}_{i=n+1}^{2n}\right).$$

Let $(I, J) = (i, j)$ w.p. $\frac{1}{n^2}$ for $1 \le i \le n$ and $n+1 \le j \le 2n$. Then

$$T' = T\left(\{u_{\Pi\circ(I,J)(i)}\}_{i=1}^{n}, \{u_{\Pi\circ(I,J)(i)}\}_{i=n+1}^{2n}\right).$$

T and T' form an exchangeable pair.

Exchangeable Pair

Assume $m = n$. Fix data $\{u_1, \ldots, u_n, u_{n+1}, \ldots, u_{2n}\}$. Π is a uniformly random permutation, and let

$$T_\Pi = T_\Pi\left(\{u_{\Pi(i)}\}_{i=1}^{n}, \{u_{\Pi(i)}\}_{i=n+1}^{2n}\right).$$

Let $(I, J) = (i, j)$ w.p. $\frac{1}{n^2}$ for $1 \leq i \leq n$ and $n+1 \leq j \leq 2n$. Then

$$T' = T\left(\{u_{\Pi\circ(I,J)(i)}\}_{i=1}^{n}, \{u_{\Pi\circ(I,J)(i)}\}_{i=n+1}^{2n}\right).$$

T and T' form an exchangeable pair.

Main Theorem

Theorem

If T, T' are mean 0, exchangeable random variables with variance $\mathbb{E}[T^2]$ satisfying

$$\mathbb{E}[T' - T|T] = -\lambda(T - R)$$

for some $\lambda \in (0,1)$ and some random variable R, then $\sup_{t\in\mathbb{R}} |P(T \leq t) - \Phi(t)|$ *is bounded by*

$$\underbrace{(2\pi)^{-1/4}\sqrt{\frac{\mathbb{E}|T'-T|^3}{\lambda}}}_{\leq n^{-1/4}f_1(\mathbf{x})} + \underbrace{\frac{1}{2\lambda}\sqrt{\mathrm{var}(\mathbb{E}[(T'-T)^2|T])}}_{\leq n^{-1}f_2(\mathbf{x})}$$

$$\underbrace{|\mathbb{E}T^2 - 1|}_{\leq n^{-1}f_3(\mathbf{x})} + \underbrace{\mathbb{E}|TR|}_{\leq n^{-1/2}f_4(\mathbf{x})} + \underbrace{\mathbb{E}|R|}_{\leq n^{-1/2}f_5(\mathbf{x})} \leq n^{-1/4}f_6(\mathbf{x})$$

Main Theorem

Theorem

If T, T' are mean 0, exchangeable random variables with variance $\mathbb{E}[T^2]$ satisfying

$$\mathbb{E}[T' - T|T] = -\lambda(T - R)$$

for some $\lambda \in (0,1)$ and some random variable R, then $\sup_{t\in\mathbb{R}} |P(T \leq t) - \Phi(t)|$ *is bounded by*

$$\underbrace{(2\pi)^{-1/4}\sqrt{\frac{\mathbb{E}|T' - T|^3}{\lambda}}}_{\leq n^{-1/4} f_1(\mathbf{x})} + \underbrace{\frac{1}{2\lambda}\sqrt{\mathrm{var}(\mathbb{E}[(T' - T)^2|T])}}_{\leq n^{-1} f_2(\mathbf{x})}$$

$$\underbrace{|\mathbb{E}T^2 - 1|}_{\leq n^{-1} f_3(\mathbf{x})} + \underbrace{\mathbb{E}|TR|}_{\leq n^{-1/2} f_4(\mathbf{x})} + \underbrace{\mathbb{E}|R|}_{\leq n^{-1/2} f_5(\mathbf{x})} \leq n^{-1/4} f_6(\mathbf{x})$$

Main Theorem

Theorem

If T, T' are mean 0, exchangeable random variables with variance $\mathbb{E}[T^2]$ satisfying

$$\mathbb{E}[T' - T|T] = -\lambda(T - R)$$

for some $\lambda \in (0, 1)$ and some random variable R, then $\sup_{t\in\mathbb{R}} |P(T \leq t) - \Phi(t)|$ *is bounded by*

$$\underbrace{(2\pi)^{-1/4}\sqrt{\frac{\mathbb{E}|T' - T|^3}{\lambda}}}_{\leq n^{-1/4} f_1(\mathbf{x})} + \underbrace{\frac{1}{2\lambda}\sqrt{\mathrm{var}(\mathbb{E}[(T' - T)^2|T])}}_{\leq n^{-1} f_2(\mathbf{x})}$$

$$\underbrace{|\mathbb{E}T^2 - 1|}_{\leq n^{-1} f_3(\mathbf{x})} + \underbrace{\mathbb{E}|TR|}_{\leq n^{-1/2} f_4(\mathbf{x})} + \underbrace{\mathbb{E}|R|}_{\leq n^{-1/2} f_5(\mathbf{x})} \leq n^{-1/4} f_6(\mathbf{x})$$

Main Theorem (Improved Rate)

Theorem

If in addition $|T' - T| \leq \delta$, $\sup_{t\in\mathbb{R}} |P(T \leq t) - \Phi(t)|$ *is bounded by*

$$\underbrace{\frac{.41\delta^3}{\lambda}}_{\leq n^{-1/2} c_1''^*} + \underbrace{3\delta(\sqrt{\mathbb{E}T^2} + \mathbb{E}|R|)}_{\leq n^{-1} f_1'(\mathbf{x})^*} + \underbrace{\frac{1}{2\lambda}\sqrt{\mathrm{var}(\mathbb{E}[(T'-T)^2|T])}}_{\leq n^{-1} f_2(\mathbf{x})}$$

$$\underbrace{|\mathbb{E}T^2 - 1|}_{\leq n^{-1} f_3(\mathbf{x})} + \underbrace{\mathbb{E}|TR|}_{\leq n^{-1/2} f_4(\mathbf{x})} + \underbrace{\mathbb{E}|R|}_{\leq n^{-1/2} f_5(\mathbf{x})} \leq n^{-1/2} f_6'(\mathbf{x})^*$$

* if $\delta < c_1' n^{-1/2}$

Main Theorem (Improved Rate)

Theorem

If in addition $|T' - T| \leq \delta$, $\sup_{t \in \mathbb{R}} |P(T \leq t) - \Phi(t)|$ *is bounded by*

$$\underbrace{\frac{.41\delta^3}{\lambda}}_{\leq n^{-1/2} c_1''^{*}} + \underbrace{3\delta(\sqrt{\mathbb{E}T^2} + \mathbb{E}|R|)}_{\leq n^{-1} f_1'(\mathbf{x})^*} + \underbrace{\frac{1}{2\lambda}\sqrt{\mathrm{var}(\mathbb{E}[(T'-T)^2|T])}}_{\leq n^{-1} f_2(\mathbf{x})}$$

$$\underbrace{|\mathbb{E}T^2 - 1|}_{\leq n^{-1} f_3(\mathbf{x})} + \underbrace{\mathbb{E}|TR|}_{\leq n^{-1/2} f_4(\mathbf{x})} + \underbrace{\mathbb{E}|R|}_{\leq n^{-1/2} f_5(\mathbf{x})} \leq n^{-1/2} f_6'(\mathbf{x})^*$$

* if $\delta < c_1' n^{-1/2}$

Main Theorem (Improved Rate)

Theorem

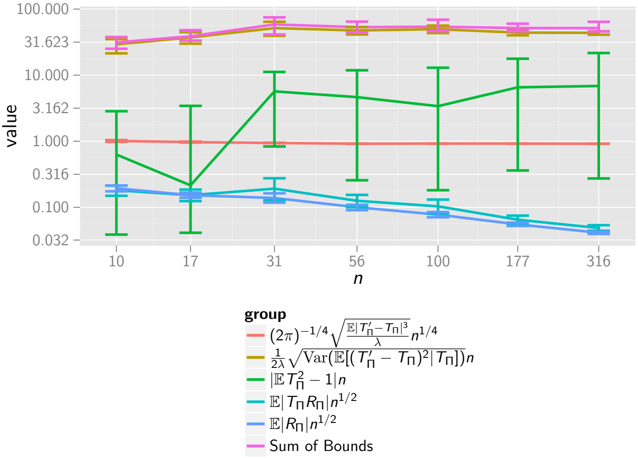
If in addition $|T' - T| \leq \delta$, $\sup_{t\in\mathbb{R}} |P(T \leq t) - \Phi(t)|$ *is bounded by*

$$\underbrace{\frac{.41\delta^3}{\lambda}}_{\leq n^{-1/2} c_1''^{*}} + \underbrace{3\delta(\sqrt{\mathbb{E}T^2} + \mathbb{E}|R|)}_{\leq n^{-1} f_1'(\mathbf{x})^*} + \underbrace{\frac{1}{2\lambda}\sqrt{\mathrm{var}(\mathbb{E}[(T' - T)^2|T])}}_{\leq n^{-1} f_2(\mathbf{x})}$$

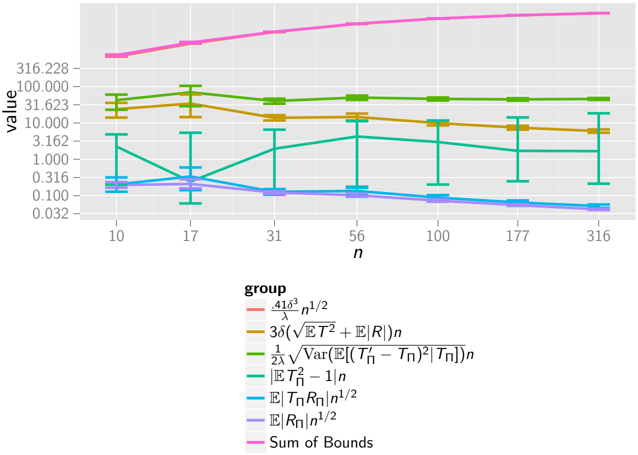
$$\underbrace{|\mathbb{E}T^2 - 1|}_{\leq n^{-1} f_3(\mathbf{x})} + \underbrace{\mathbb{E}|TR|}_{\leq n^{-1/2} f_4(\mathbf{x})} + \underbrace{\mathbb{E}|R|}_{\leq n^{-1/2} f_5(\mathbf{x})} \leq n^{-1/2} f_6'(\mathbf{x})^*$$

* if $\delta < c_1' n^{-1/2}$

Simulated Bounds

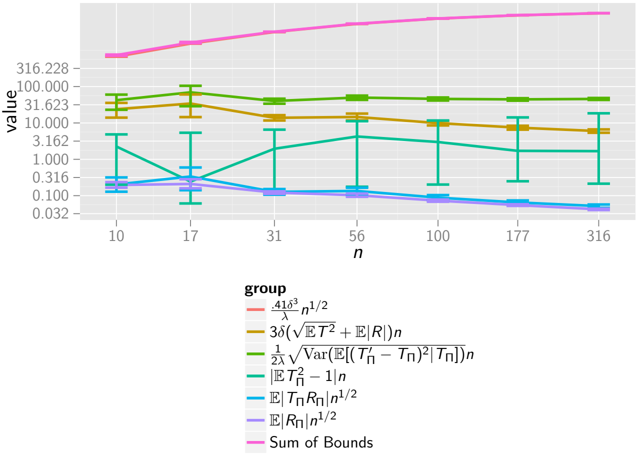


Simulated Bounds (Improved Rate)



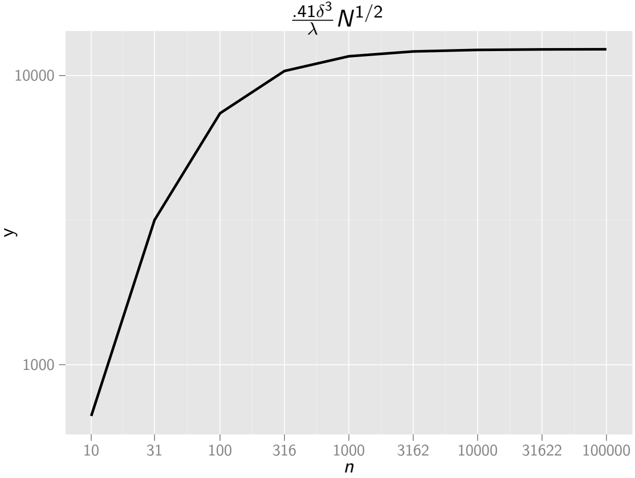
When $\mathbf{u} = \{i\}_{i=1}^{i=2n}$, $\frac{.41\delta^3}{\lambda} n^{1/2} \to .205(16\sqrt{6})^3$

Simulated Bounds (Improved Rate)



When $\mathbf{u} = \{i\}_{i=1}^{i=2n}$, $\frac{.41\delta^3}{\lambda} n^{1/2} \to .205(16\sqrt{6})^3$

Behavior of δ



Conclusion

- Friedman's test for non-vectorial and heterogeneous data.
- MKL can learn structure of data.
- MKL power competitive with best-performing kernel and obviates multiple testing considerations.
- Normal-like null distributions.
- Berry–Esseen-type convergence result via Stein's method of exchangeable pairs.

Conclusion

- Friedman's test for non-vectorial and heterogeneous data.
- MKL can learn structure of data.
- MKL power competitive with best-performing kernel and obviates multiple testing considerations.
- Normal-like null distributions.
- Berry–Esseen-type convergence result via Stein's method of exchangeable pairs.

Conclusion

- Friedman's test for non-vectorial and heterogeneous data.
- MKL can learn structure of data.
- MKL power competitive with best-performing kernel and obviates multiple testing considerations.
- Normal-like null distributions.
- Berry–Esseen-type convergence result via Stein's method of exchangeable pairs.

Conclusion

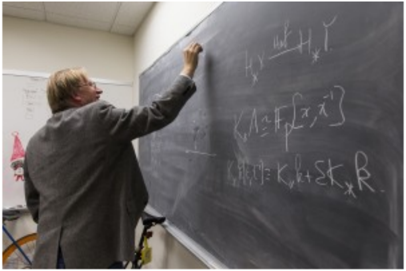
- Friedman's test for non-vectorial and heterogeneous data.
- MKL can learn structure of data.
- MKL power competitive with best-performing kernel and obviates multiple testing considerations.
- Normal-like null distributions.
- Berry–Esseen-type convergence result via Stein's method of exchangeable pairs.

Conclusion

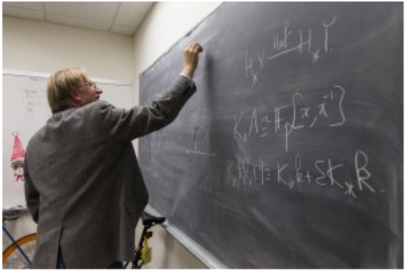
- Friedman's test for non-vectorial and heterogeneous data.
- MKL can learn structure of data.
- MKL power competitive with best-performing kernel and obviates multiple testing considerations.
- Normal-like null distributions.
- Berry–Esseen-type convergence result via Stein's method of exchangeable pairs.

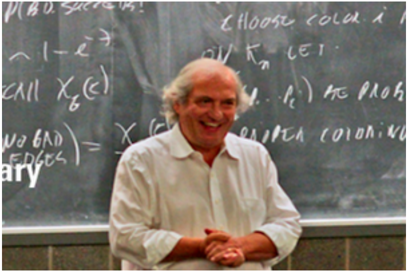
Acknowledgements

Acknowledgements

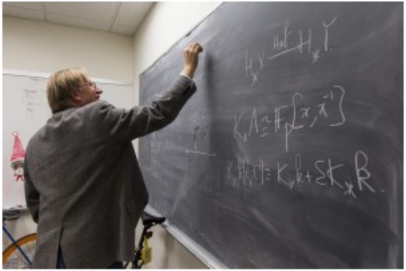


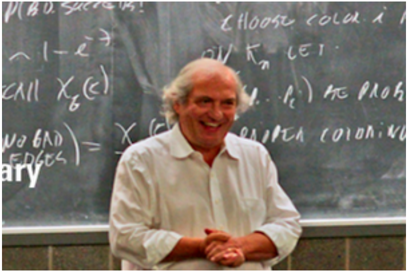
Acknowledgements

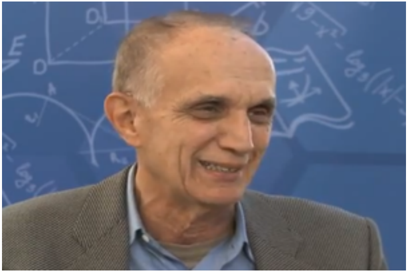




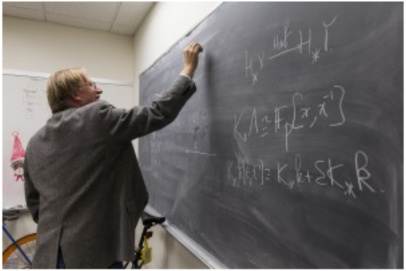
Acknowledgements

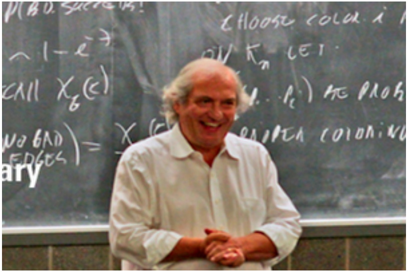


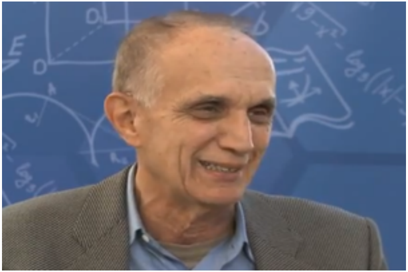


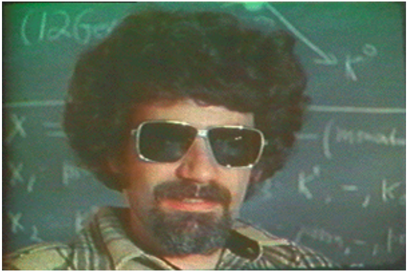


Acknowledgements

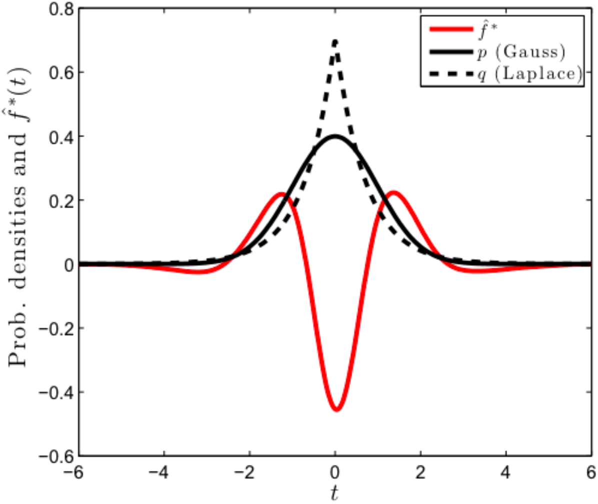




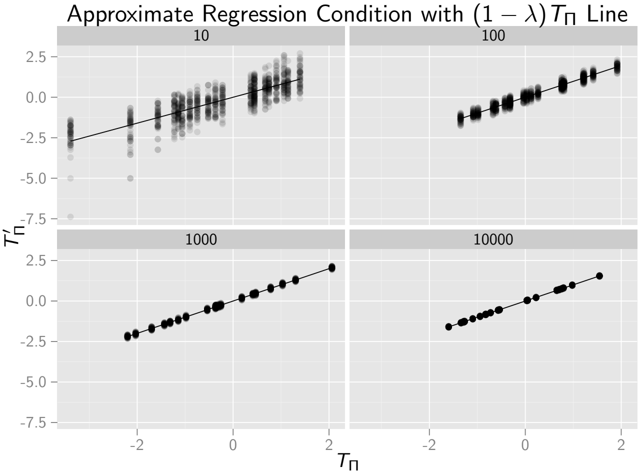




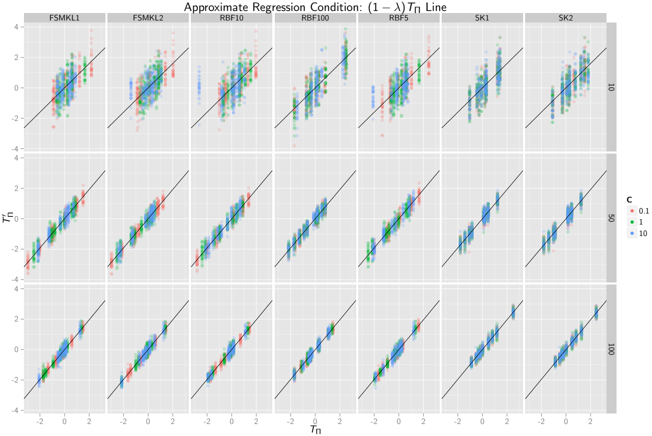
KMMD Function Example



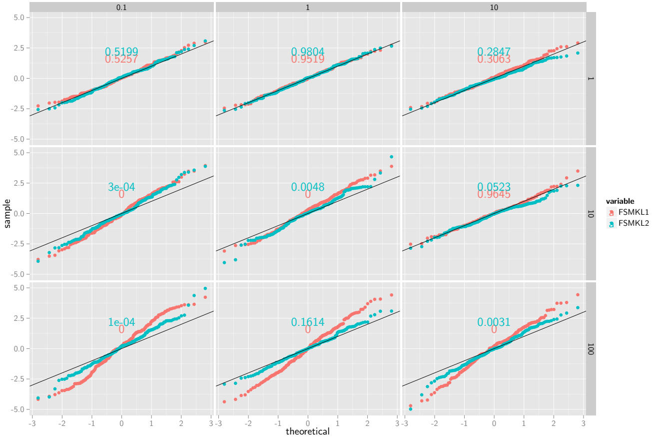
Approximate Regression Condition



ARC (MKL)


Approximate Regression Condition: $(1-\lambda)T_\Pi$ Line
FSMKL1
FSMKL2
RBF10
RBF100
RBF5
SK1
SK2
10
50
100
T'_Π
T_Π
c
0.1
1
10

Overfitting on Kernels



ARC (MKL, Overfit)