Two-Sample Kernel Based Tests

Nelson Ray (joint work with Susan Holmes)

Stanford University

February 24, 2012

Outline

Outline

1. The two-sample problem
2. Friedman's two-sample test [1]: leverage regression and classification techniques

Outline

1. The two-sample problem
2. Friedman's two-sample test [1]: leverage regression and classification techniques
3. Univariate data and linear scoring functions: permutation t-test

Outline

1. The two-sample problem
2. Friedman's two-sample test [1]: leverage regression and classification techniques
3. Univariate data and linear scoring functions: permutation t-test
4. Twitter example for text data

Outline

1. The two-sample problem
2. Friedman's two-sample test [1]: leverage regression and classification techniques
3. Univariate data and linear scoring functions: permutation t-test
4. Twitter example for text data
5. Image data: airplanes and cars / pigeons and roosters

The Two-Sample Problem

Problem

$\{\mathbf{x}_i\}_1^N$ from $p(\mathbf{x})$ and $\{\mathbf{z}_i\}_1^M$ from $q(\mathbf{z})$ testing $\mathcal{H}_A$: $p \neq q$ against $\mathcal{H}_0$: $p = q$

The Two-Sample Problem

Problem

$\{\mathbf{x}_i\}_1^N$ from $p(\mathbf{x})$ and $\{\mathbf{z}_i\}_1^M$ from $q(\mathbf{z})$ testing $\mathcal{H}_A$: $p \neq q$ against $\mathcal{H}_0$: $p = q$

Solutions

1D/parametric/shift t-test

The Two-Sample Problem

Problem

$\{\mathbf{x}_i\}_1^N$ from $p(\mathbf{x})$ and $\{\mathbf{z}_i\}_1^M$ from $q(\mathbf{z})$ testing $\mathcal{H}_A$: $p \neq q$ against $\mathcal{H}_0$: $p = q$

Solutions

1D/parametric/shift t-test

1D/non-parametric/shift randomization test

The Two-Sample Problem

Problem

$\{\mathbf{x}_i\}_1^N$ from $p(\mathbf{x})$ and $\{\mathbf{z}_i\}_1^M$ from $q(\mathbf{z})$ testing $\mathcal{H}_A$: $p \neq q$ against $\mathcal{H}_0$: $p = q$

Solutions

1D/parametric/shift t-test

1D/non-parametric/shift randomization test

pD/parametric/shift Hotelling's T^2-test

The Two-Sample Problem

Problem

$\{\mathbf{x}_i\}_1^N$ from $p(\mathbf{x})$ and $\{\mathbf{z}_i\}_1^M$ from $q(\mathbf{z})$ testing $\mathcal{H}_A$: $p \neq q$ against $\mathcal{H}_0$: $p = q$

Solutions

1D/parametric/shift *t*-test

1D/non-parametric/shift randomization test

pD/parametric/shift Hotelling's T^2-test

1D/non-parametric/omnibus Mann-Whitney U/Wilcoxon rank-sum test

The Two-Sample Problem

Problem

$\{\mathbf{x}_i\}_1^N$ from $p(\mathbf{x})$ and $\{\mathbf{z}_i\}_1^M$ from $q(\mathbf{z})$ testing $\mathcal{H}_A$: $p \neq q$ against $\mathcal{H}_0$: $p = q$

Solutions

1D/parametric/shift t-test

1D/non-parametric/shift randomization test

pD/parametric/shift Hotelling's T^2-test

1D/non-parametric/omnibus Mann-Whitney U/Wilcoxon rank-sum test

pD/non-parametric/omnibus Friedman-Rafsky test

The Two-Sample Problem

Problem

$\{\mathbf{x}_i\}_1^N$ from $p(\mathbf{x})$ and $\{\mathbf{z}_i\}_1^M$ from $q(\mathbf{z})$ testing $\mathcal{H}_A$: $p \neq q$ against $\mathcal{H}_0$: $p = q$

Solutions

1D/parametric/shift t-test

1D/non-parametric/shift randomization test

pD/parametric/shift Hotelling's T^2-test

1D/non-parametric/omnibus Mann-Whitney U/Wilcoxon rank-sum test

pD/non-parametric/omnibus Friedman-Rafsky test

ker/non-parametric/omnibus KMMD test

Friedman's Two-Sample Test

1. Pool the two samples $\{\mathbf{u}_i\}_1^{N+M} = \{\mathbf{x}_i\}_1^N \cup \{\mathbf{z}_i\}_1^M$.

Friedman's Two-Sample Test

1. Pool the two samples $\{\mathbf{u}_i\}_1^{N+M} = \{\mathbf{x}_i\}_1^N \cup \{\mathbf{z}_i\}_1^M$.
2. Assign label $y_i = 1$ to the first group and $y_i = -1$ to the second group.

Friedman's Two-Sample Test

1. Pool the two samples $\{\mathbf{u}_i\}_1^{N+M} = \{\mathbf{x}_i\}_1^N \cup \{\mathbf{z}_i\}_1^M$.
2. Assign label $y_i = 1$ to the first group and $y_i = -1$ to the second group.
3. Apply a binary classification learning machine f to the training data to score the observations $\{s_i = f(\mathbf{u}_i)\}_1^{N+M}$.

Friedman's Two-Sample Test

1. Pool the two samples $\{\mathbf{u}_i\}_1^{N+M} = \{\mathbf{x}_i\}_1^N \cup \{\mathbf{z}_i\}_1^M$.
2. Assign label $y_i = 1$ to the first group and $y_i = -1$ to the second group.
3. Apply a binary classification learning machine f to the training data to score the observations $\{s_i = f(\mathbf{u}_i)\}_1^{N+M}$.
4. Calculate a univariate two-sample test statistic $T = T(\{s_i\}_1^N, \{s_i\}_{N+1}^{N+M})$.

Friedman's Two-Sample Test

1. Pool the two samples $\{\mathbf{u}_i\}_1^{N+M} = \{\mathbf{x}_i\}_1^N \cup \{\mathbf{z}_i\}_1^M$.
2. Assign label $y_i = 1$ to the first group and $y_i = -1$ to the second group.
3. Apply a binary classification learning machine f to the training data to score the observations $\{s_i = f(\mathbf{u}_i)\}_1^{N+M}$.
4. Calculate a univariate two-sample test statistic $T = T(\{s_i\}_1^N, \{s_i\}_{N+1}^{N+M})$.
5. Determine the permutation null distribution of the above statistic to yield a p-value.

Permutation t-test Connection

With univariate data and linear scoring functions, Friedman's test reduces to the permutation t-test.

Permutation t-test Connection

With univariate data and linear scoring functions, Friedman's test reduces to the permutation *t*-test.

With multivariate data, the test is close to Hotelling's T^2-test.

Warm-up (1D)

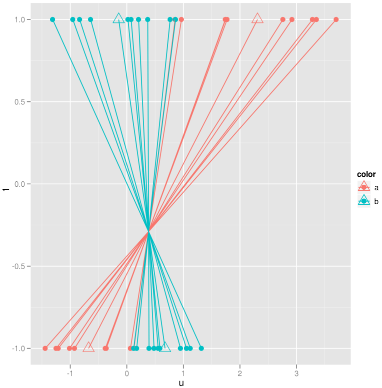
$x_i \sim \mathcal{N}(0,1), z_i \sim \mathcal{N}(3,1), \{u_i\}_{i=1}^{20} = \{x_i\}_1^{10} \cup \{z_i\}_1^{10}$

Warm-up (1D)

$x_i \sim \mathcal{N}(0,1), z_i \sim \mathcal{N}(3,1), \{u_i\}_{i=1}^{20} = \{x_i\}_1^{10} \cup \{z_i\}_1^{10}$
$\hat{f}(u_i) = \hat{\beta}_0 + \hat{\beta}_1 u_i$

Warm-up (1D)

$$x_i \sim \mathcal{N}(0,1), z_i \sim \mathcal{N}(3,1), \{u_i\}_{i=1}^{20} = \{x_i\}_1^{10} \cup \{z_i\}_1^{10}$$
$$\hat{f}(u_i) = \hat{\beta}_0 + \hat{\beta}_1 u_i$$
$$|T(\{u_i\}_1^N, \{u_i\}_{N+1}^{N+M})| = 6.12 = |T(\{s_i\}_1^N, \{s_i\}_{N+1}^{N+M})|$$



Twitter Example

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

Compares two strings based on the their length k contiguous subsequences.

The Spectrum Kernel

Compares two strings based on the their length k contiguous subsequences.

- $\mathcal{X}$ is our input space, built up from an alphabet $\mathcal{A} = \{a, b, \ldots, z, \ \}$ with $|\mathcal{A}| = 27$.

The Spectrum Kernel

Compares two strings based on the their length k contiguous subsequences.

- $\mathcal{X}$ is our input space, built up from an alphabet $\mathcal{A} = \{a, b, \ldots, z, \ \}$ with $|\mathcal{A}| = 27$.
- The k-spectrum ($k \geq 1$) of an input sequence is the set of all length k contiguous subsequences it contains.

The Spectrum Kernel

Compares two strings based on the their length k contiguous subsequences.

- $\mathcal{X}$ is our input space, built up from an alphabet $\mathcal{A} = \{a, b, \ldots, z, \ \}$ with $|\mathcal{A}| = 27$.
- The k-spectrum $(k \geq 1)$ of an input sequence is the set of all length k contiguous subsequences it contains.
- Define the feature map from $\mathcal{X}$ to $\mathbb{R}^{|\mathcal{A}|^k}$ by $\Phi_k(x) = (\phi_a(x))_{a \in \mathcal{A}^k}$ where $\phi_a(x)$ is the number of times a occurs in x: $\{\#aaa, \#aab, \#aac, \ldots, \}$.

The Spectrum Kernel

Compares two strings based on the their length k contiguous subsequences.

- $\mathcal{X}$ is our input space, built up from an alphabet $\mathcal{A} = \{a, b, \ldots, z, \ \}$ with $|\mathcal{A}| = 27$.
- The k-spectrum ($k \geq 1$) of an input sequence is the set of all length k contiguous subsequences it contains.
- Define the feature map from $\mathcal{X}$ to $\mathbb{R}^{|\mathcal{A}|^k}$ by $\Phi_k(x) = (\phi_a(x))_{a \in \mathcal{A}^k}$ where $\phi_a(x)$ is the number of times a occurs in x: $\{\#aaa, \#aab, \#aac, \ldots, \}$.
- $K_k(x, y) = \langle \Phi_k(x), \Phi_k(y) \rangle$.

Support Vector Machines for Regression

$$f(x) = \sum_{m=1}^{M} \beta_m h_m(x) + \beta_0, \quad h_m(x) \text{ basis functions}$$

Support Vector Machines for Regression

$$f(x) = \sum_{m=1}^{M} \beta_m h_m(x) + \beta_0, \quad h_m(x) \text{ basis functions}$$

To estimate β and β_0, minimize

$$H(\beta, \beta_0) = \sum_{i=1}^{N} V(y_i - f(x_i)) + \frac{\lambda}{2} \sum_{m=1}^{M} \beta_m^2.$$

Support Vector Machines for Regression

$$f(x) = \sum_{m=1}^{M} \beta_m h_m(x) + \beta_0, \quad h_m(x) \text{ basis functions}$$

To estimate β and β_0, minimize

$$H(\beta, \beta_0) = \sum_{i=1}^{N} V(y_i - f(x_i)) + \frac{\lambda}{2} \sum_{m=1}^{M} \beta_m^2.$$

V is taken to be ϵ-insensitive loss:

$$V_\epsilon(r) = \begin{cases} 0 & \text{if } |r| < \epsilon, \\ |r| - \epsilon & \text{otherwise.} \end{cases}$$

Support Vector Machines for Regression

$$f(x) = \sum_{m=1}^{M} \beta_m h_m(x) + \beta_0, \quad h_m(x) \text{ basis functions}$$

To estimate β and β_0, minimize

$$H(\beta, \beta_0) = \sum_{i=1}^{N} V(y_i - f(x_i)) + \frac{\lambda}{2} \sum_{m=1}^{M} \beta_m^2.$$

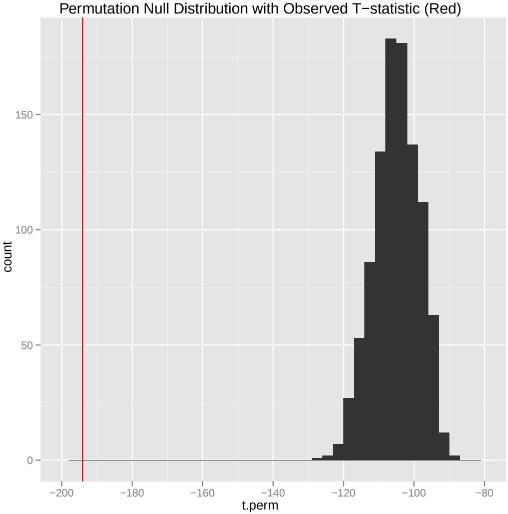
V is taken to be ϵ-insensitive loss:

$$V_\epsilon(r) = \begin{cases} 0 & \text{if } |r| < \epsilon, \\ |r| - \epsilon & \text{otherwise.} \end{cases}$$

The solution has the form $\hat{f}(x) = \sum_{i=1}^{N} \hat{\alpha}_i K(x, x_i)$, where $K(x, y) = \langle h(x), h(y) \rangle$.

Twitter Example

$p < .001$:



Power Simulations at .05 Level

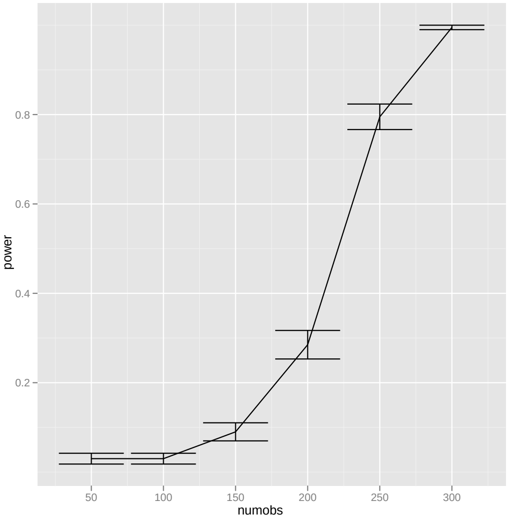


Image Data (Cars)

Caltech 101 Object Categories [2]
The cars are 300×197 grayscale.

Planes Before

The planes aren't.

Planes After

Polynomial Kernel

Each $m \times n$ grayscale image is converted to a vector of length $p = mn$.

Polynomial Kernel

Each $m \times n$ grayscale image is converted to a vector of length $p = mn$. Given $X \in \mathbb{R}^{n \times p}$, the linear kernel is given by

$$K(x, x') = \langle x, x' \rangle = \langle \Phi(x), \Phi(x') \rangle.$$

The kernel matrix is given simply by $XX^T \succeq 0$. This corresponds to the identity mapping: $\Phi(x) = x$.

Polynomial Kernel

Each $m \times n$ grayscale image is converted to a vector of length $p = mn$. Given $X \in \mathbb{R}^{n \times p}$, the linear kernel is given by

$$K(x, x') = \langle x, x' \rangle = \langle \Phi(x), \Phi(x') \rangle.$$

The kernel matrix is given simply by $XX^T \succeq 0$. This corresponds to the identity mapping: $\Phi(x) = x$.

The homogeneous polynomial kernel,

$$K(x, x') = \langle \Phi(x), \Phi(x') \rangle = \langle x, x' \rangle^d,$$

corresponds to the mapping

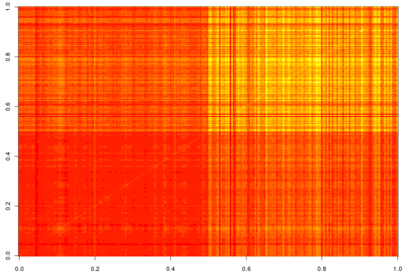
$\Phi(x) = [x_1^d, \ldots, x_p^d, x_1^{d-1} x_2, \ldots, x_p^{d-1} x_{p-1}]^T \in \mathbb{R}^{d'}$, where $d' = \binom{d+N-1}{d}$.

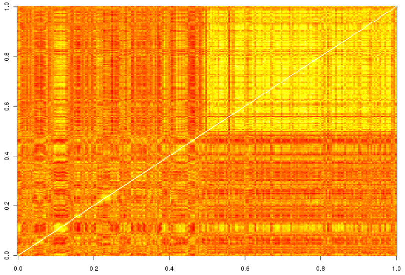
Standardization

In order to mitigate the effects of global differences in illumination, each vector is scaled so that it has mean zero and unit norm.

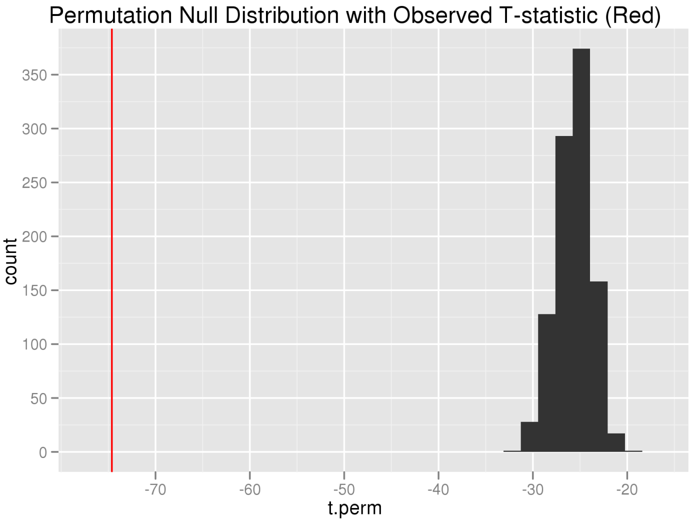
In order to mitigate the effects of global differences in illumination, each vector is scaled so that it has mean zero and unit norm.

Unscaled linear kernel matrix, left; scaled, right

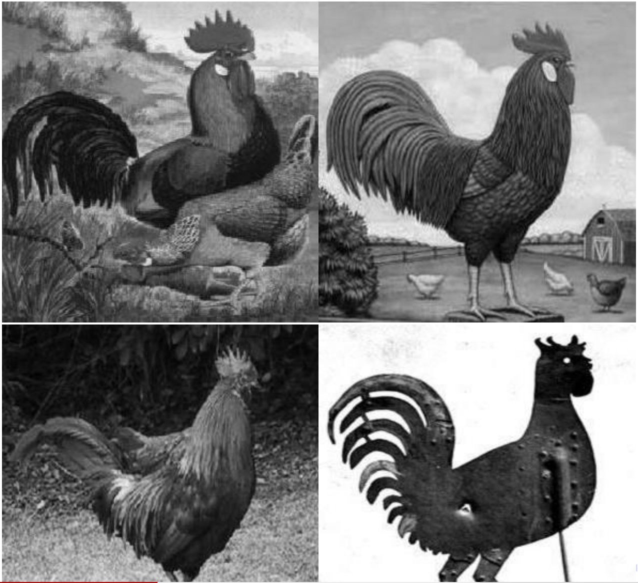




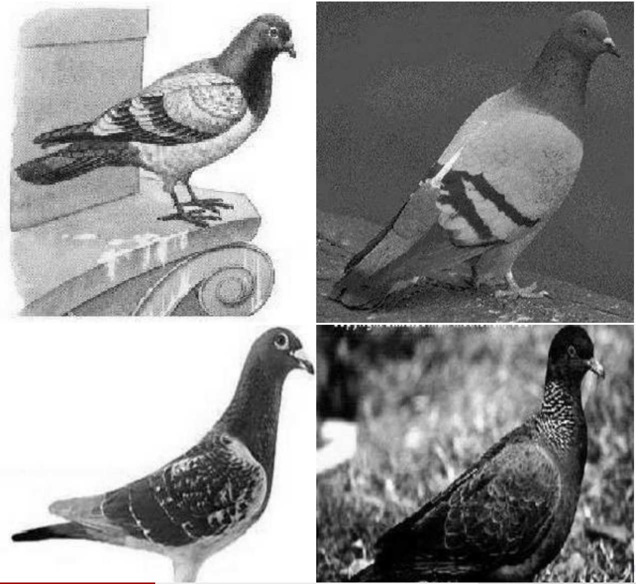
Car/Airplane Example (Linear Kernel)



Roosters

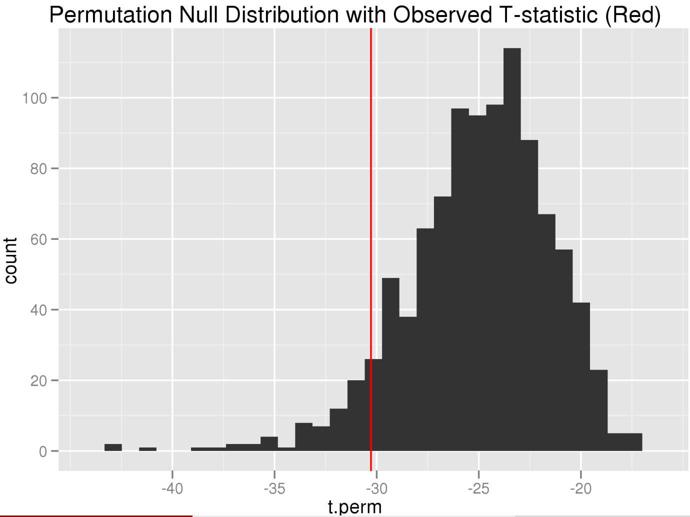


Pigeons



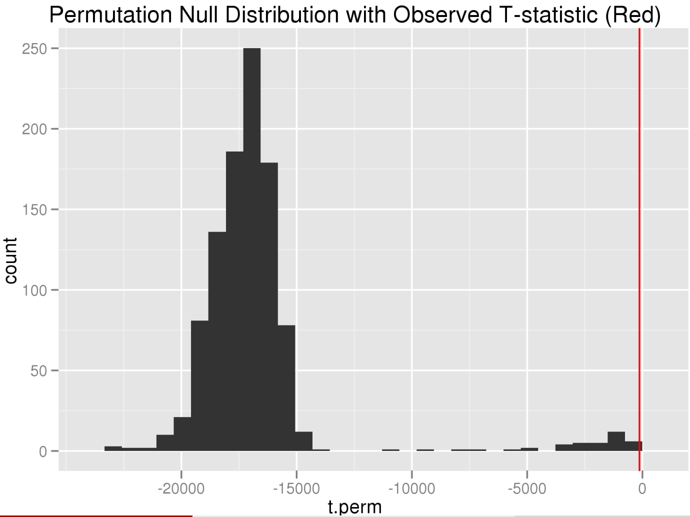
Rooster/Pigeon Example (Linear Kernel)

$p = .138$



Rooster/Pigeon Example (Inhomogeneous Degree 4)

$p < .001$



Future Work

- String Kernels:

Future Work

- String Kernels: k-spectrum, decay factors
- Side Information:

Future Work

- String Kernels: k-spectrum, decay factors
- Side Information: phylogenetic tree, Twitter post times
- Heterogeneous Data (Wikipedia pages):

Future Work

- String Kernels: k-spectrum, decay factors
- Side Information: phylogenetic tree, Twitter post times
- Heterogeneous Data (Wikipedia pages): optimal combinations of kernels via SDPs, KL divergence

References I

J. Friedman, "On Multivariate Goodness–of–Fit and Two–Sample Testing," *Proceedings of Phystat2003, http://www.slac.stanford.edu/econf/C*, vol. 30908, 2003.

L. Fei-Fei, R. Fergus, and P. Perona, "Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories," *Computer Vision and Image Understanding*, vol. 106, no. 1, pp. 59–70, 2007.