Optimized MMD for Detecting Distribution Shift

Arthur Gretton

Gatsby Computational Neuroscience Unit,
Google DeepMind

Monitoring ML Models Under Drift, ICLR2026

Comparing two samples

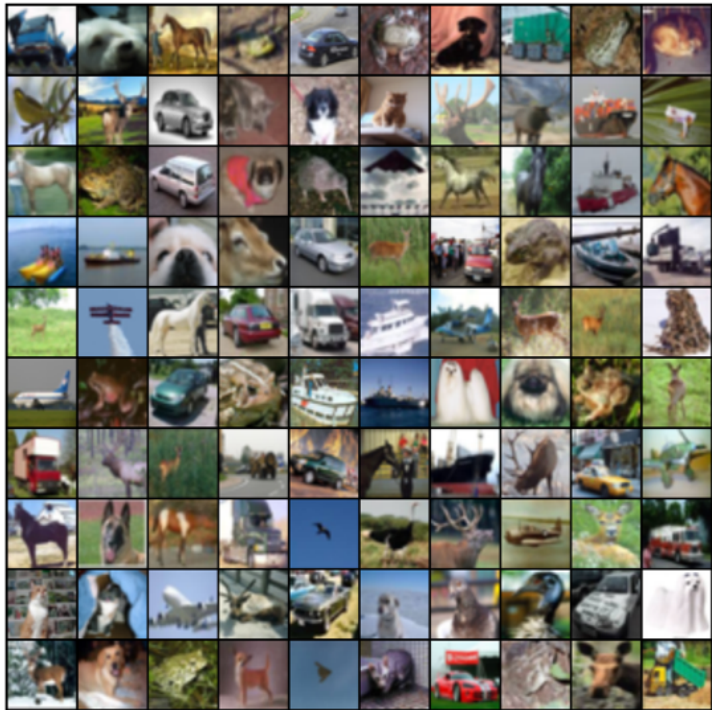
- Given: Samples from unknown distributions P and Q.
- Goal: do P and Q differ?

$\sim P$

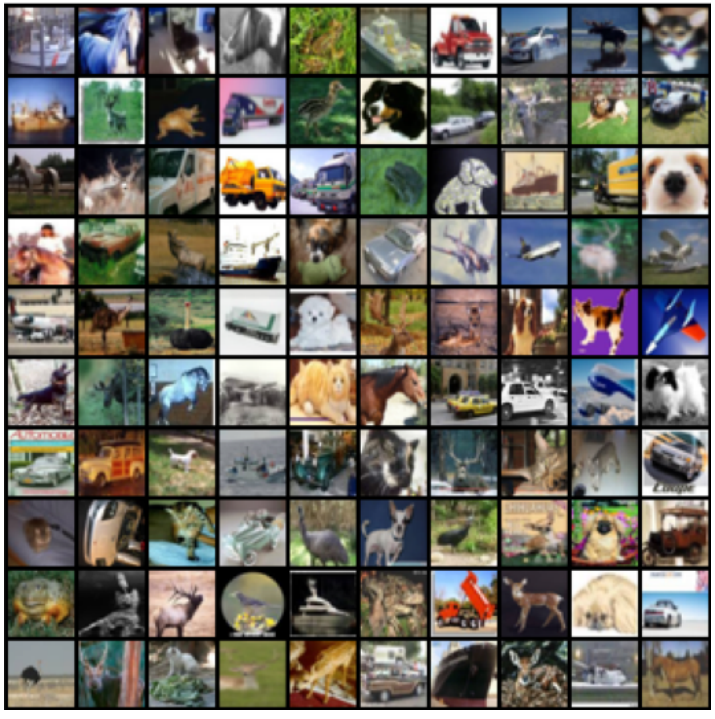
$\sim Q$

MMD for detecting drift

- Goal: do P and Q differ?



CIFAR 10 samples



Cifar 10.1 samples

Significant difference?

Krizhevsky, Hinton, Learning multiple layers of features from tiny images. (2009)
Recht, Roelofs, Schmidt, Shankar, Do CIFAR-10 Classifiers Generalize to CIFAR-10? (2018).

Outline

Two sample testing

- Test statistic: Maximum Mean Discrepancy (MMD)...
- Statistical testing with the MMD
- "How to choose the best kernel"
  - using fusing
  - using aggregation
- No sample splitting!

Outline

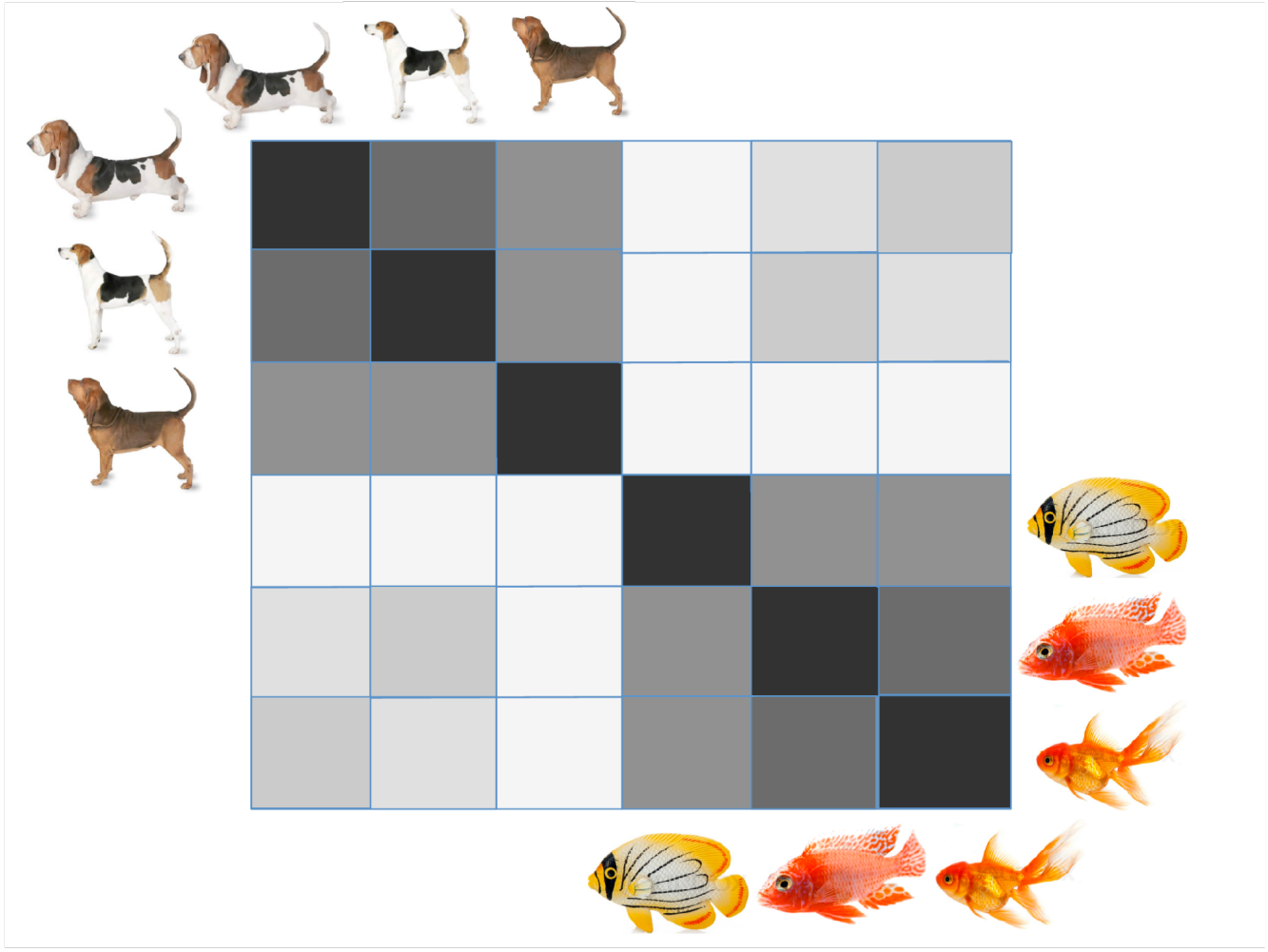
Two sample testing

- Test statistic: Maximum Mean Discrepancy (MMD)...
- Statistical testing with the MMD
- "How to choose the best kernel"
  - using fusing
  - using aggregation
- No sample splitting!

The last MMD papers you'll ever need!

Illustration of MMD

- Dogs ($= P$) and fish ($= Q$) example revisited
- Each entry is one of $k(\text{dog}_i, \text{dog}_j)$, $k(\text{dog}_i, \text{fish}_j)$, or $k(\text{fish}_i, \text{fish}_j)$



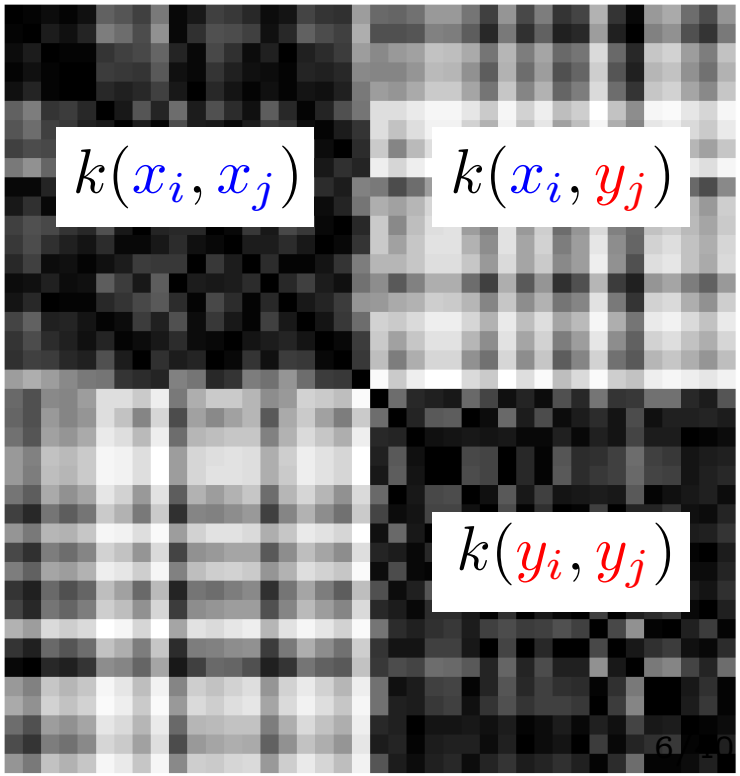
The maximum mean discrepancy

Original empirical MMD for dogs and fish:

$X = [$  $\dots]$

$Y = [$  $\dots]$

$$\begin{aligned}
&\mathrm{MMD}^2_\lambda(p, q) \\
&= \frac{1}{m(m-1)} \sum_{i \neq j} k_\lambda(x_i, x_j) \\
&+ \frac{1}{n(n-1)} \sum_{i \neq j} k_\lambda(y_i, y_j) \\
&- \frac{2}{mn} \sum_{i,j} k_\lambda(x_i, y_j)
\end{aligned}$$



Population maximum mean discrepancy

The population maximum mean discrepancy

$$\mathrm{MMD}^2_\lambda(p, q) = \underbrace{\mathrm{E}_p k_\lambda(X, X')}_{\text{(a)}} + \underbrace{\mathrm{E}_q k_\lambda(Y, Y')}_{\text{(a)}} - 2\underbrace{\mathrm{E}_{p,q} k_\lambda(X, Y)}_{\text{(b)}}$$

(a)= within distrib. similarity, (b)= cross-distrib. similarity.

Kernel: $k_\lambda(x, y) := \prod_{i=1}^{d} \frac{1}{\lambda} K_i\left(\frac{x_i - y_i}{\lambda}\right)$ Bandwidth: $\lambda \in (0, \infty)$

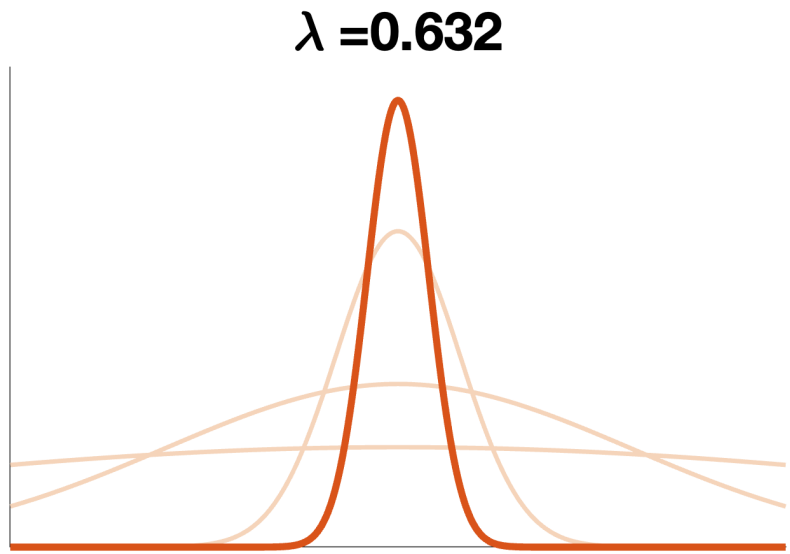
Set of bandwidths: $\Lambda := \{\lambda_i\}_{i=1}^{|\Lambda|}$

Kernels and bandwidths

Kernel: $k_\lambda(x, y) := \prod_{i=1}^{d} \frac{1}{\lambda} K_i\left(\frac{x_i - y_i}{\lambda}\right)$ Bandwidth: $\lambda \in (0, \infty)$

Characteristic kernels: $\mathrm{MMD}^2_\lambda(p, q) = 0$ iff $p = q$.

Gaussian ($K_i(u) \propto e^{-u^2}$), Laplace ($K_i(u) \propto e^{-|u|}$), Matérn, Energy,...

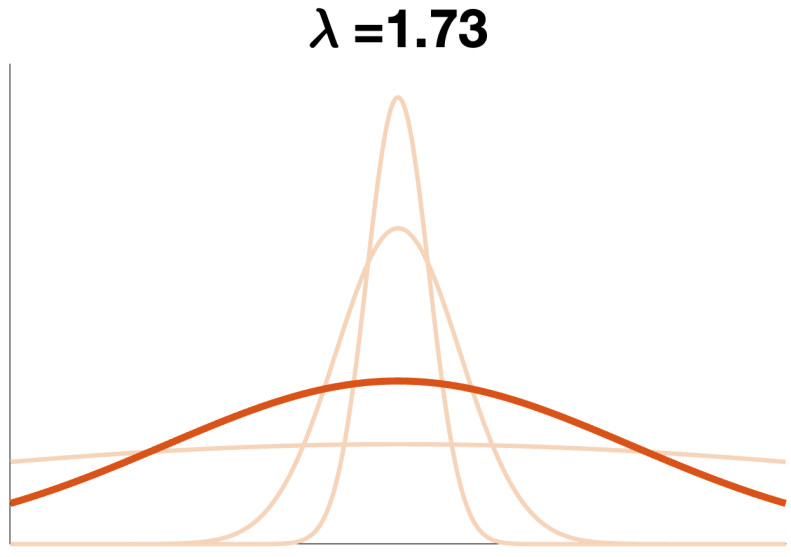


Kernels and bandwidths

Kernel: $k_\lambda(x, y) := \prod_{i=1}^{d} \frac{1}{\lambda} K_i\left(\frac{x_i - y_i}{\lambda}\right)$ Bandwidth: $\lambda \in (0, \infty)$

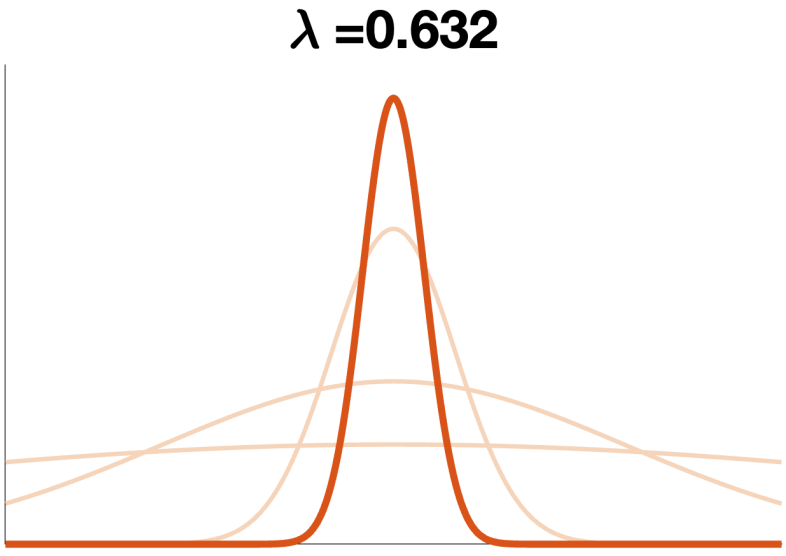
Characteristic kernels: $\mathrm{MMD}^2_\lambda(p, q) = 0$ iff $p = q$.

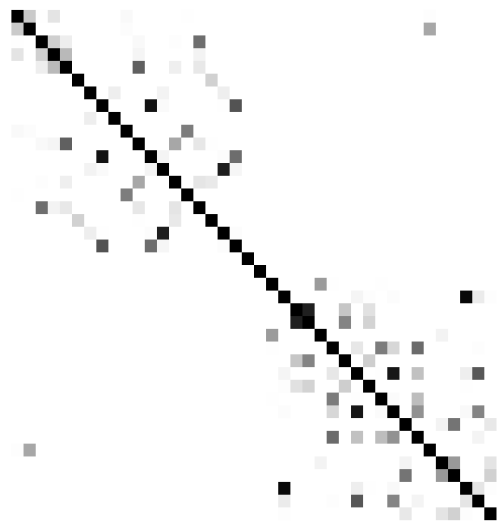
Gaussian ($K_i(u) \propto e^{-u^2}$), Laplace ($K_i(u) \propto e^{-|u|}$), Matérn, Energy,...



What effect do different kernels have?

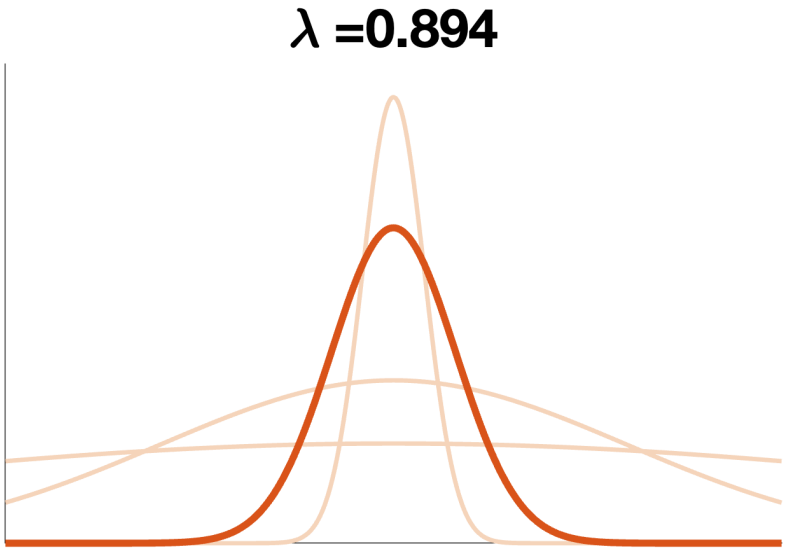
$$\widehat{\mathrm{MMD}}^2_\lambda(\mathbb{X}_m, \mathbb{Y}_n) = \frac{1}{m(m-1)} \sum_{i \neq j} k_\lambda(x_i, x_j) + \frac{1}{n(n-1)} \sum_{i \neq j} k_\lambda(\mathrm{y}_i, \mathrm{y}_j) - \frac{2}{mn} \sum_{i,j} k_\lambda(x_i, \mathrm{y}_j)$$

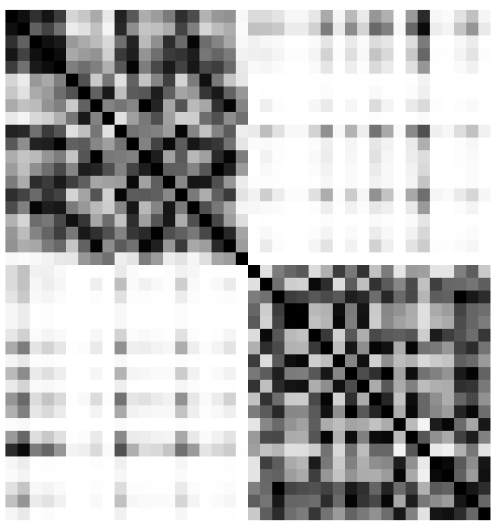




What effect do different kernels have?

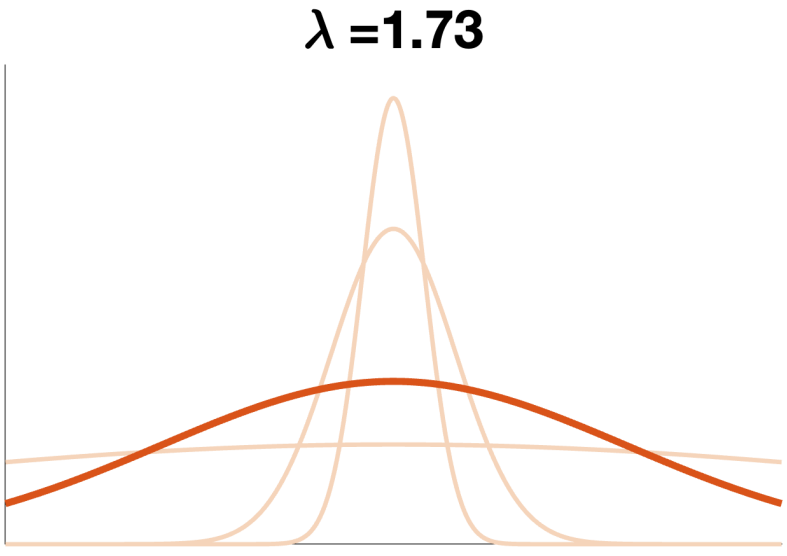
$$\widehat{\mathrm{MMD}}^2_\lambda(\mathbb{X}_m, \mathbb{Y}_n) = \frac{1}{m(m-1)} \sum_{i \neq j} k_\lambda(x_i, x_j) + \frac{1}{n(n-1)} \sum_{i \neq j} k_\lambda(\mathrm{y}_i, \mathrm{y}_j) - \frac{2}{mn} \sum_{i,j} k_\lambda(x_i, \mathrm{y}_j)$$

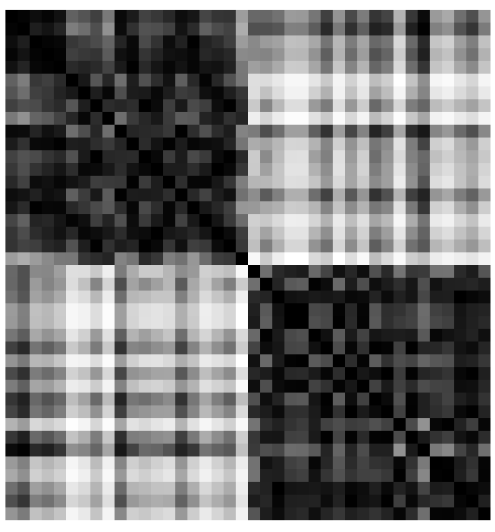




What effect do different kernels have?

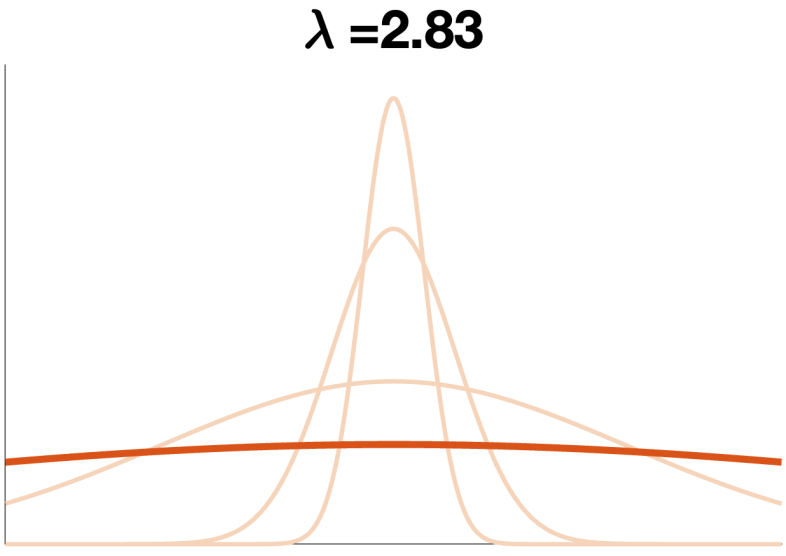
$$\widehat{\mathrm{MMD}}^2_\lambda(\mathbb{X}_m, \mathbb{Y}_n) = \frac{1}{m(m-1)} \sum_{i \neq j} k_\lambda(x_i, x_j) + \frac{1}{n(n-1)} \sum_{i \neq j} k_\lambda(\mathrm{y}_i, \mathrm{y}_j) - \frac{2}{mn} \sum_{i,j} k_\lambda(x_i, \mathrm{y}_j)$$

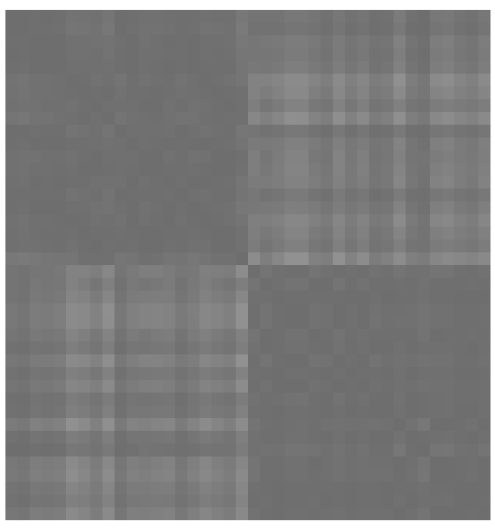




What effect do different kernels have?

$$\widehat{\mathrm{MMD}}^2_\lambda(\mathbb{X}_m, \mathbb{Y}_n) = \frac{1}{m(m-1)} \sum_{i \neq j} k_\lambda(x_i, x_j) + \frac{1}{n(n-1)} \sum_{i \neq j} k_\lambda(\mathrm{y}_i, \mathrm{y}_j) - \frac{2}{mn} \sum_{i,j} k_\lambda(x_i, \mathrm{y}_j)$$





Two-Sample Testing with MMD

MMD statistical test

The empirical MMD:

$$\widehat{\mathrm{MMD}}^2_\lambda(\mathbb{X}_m, \mathbb{Y}_n) = \frac{1}{m(m-1)} \sum_{i \neq j} k_\lambda(x_i, x_j) + \frac{1}{n(n-1)} \sum_{i \neq j} k_\lambda(y_i, y_j) - \frac{2}{mn} \sum_{i,j} k_\lambda(x_i, y_j)$$

Two-sample MMD test:

$$\Delta^\lambda_\alpha(\mathbb{X}_m, \mathbb{Y}_n) := \mathbb{1}\left(\widehat{\mathrm{MMD}}^2_\lambda(\mathbb{X}_m, \mathbb{Y}_n) > q^\lambda_{1-\alpha}\right)$$

MMD statistical test

The empirical MMD:

$$\widehat{\mathrm{MMD}}^2_\lambda(\mathbb{X}_m, \mathbb{Y}_n) = \frac{1}{m(m-1)} \sum_{i \neq j} k_\lambda(x_i, x_j) + \frac{1}{n(n-1)} \sum_{i \neq j} k_\lambda(y_i, y_j) - \frac{2}{mn} \sum_{i,j} k_\lambda(x_i, y_j)$$

Two-sample MMD test:

$$\Delta^\lambda_\alpha(\mathbb{X}_m, \mathbb{Y}_n) := \mathbb{1}\left(\widehat{\mathrm{MMD}}^2_\lambda(\mathbb{X}_m, \mathbb{Y}_n) > q^\lambda_{1-\alpha}\right)$$

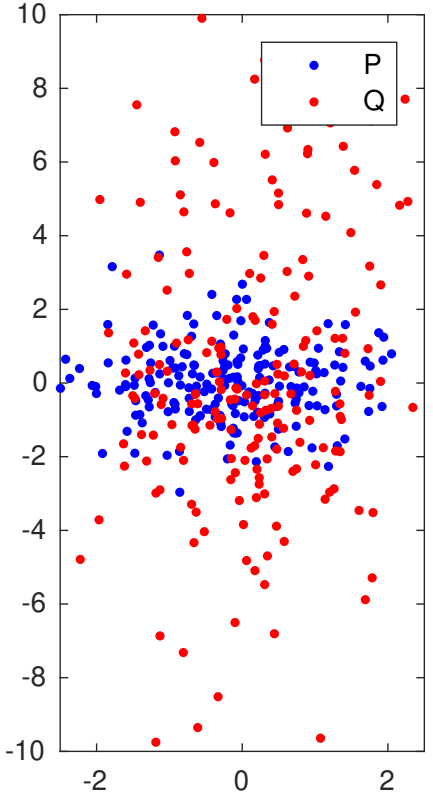
Want threshold $q^\lambda_{1-\alpha}$ for test $\widehat{\mathrm{MMD}}^2_\lambda(\mathbb{X}_m, \mathbb{Y}_n)$ to get false positive rate α

Behaviour of $\widehat{MMD}^2$ when $P \neq Q$

Draw $n = 200$ i.i.d samples from P and Q

- Laplace with different y-variance.
- $\sqrt{n} \times \widehat{\mathrm{MMD}}^2_\lambda(\mathbb{X}_m, \mathbb{Y}_n) = 1.2$

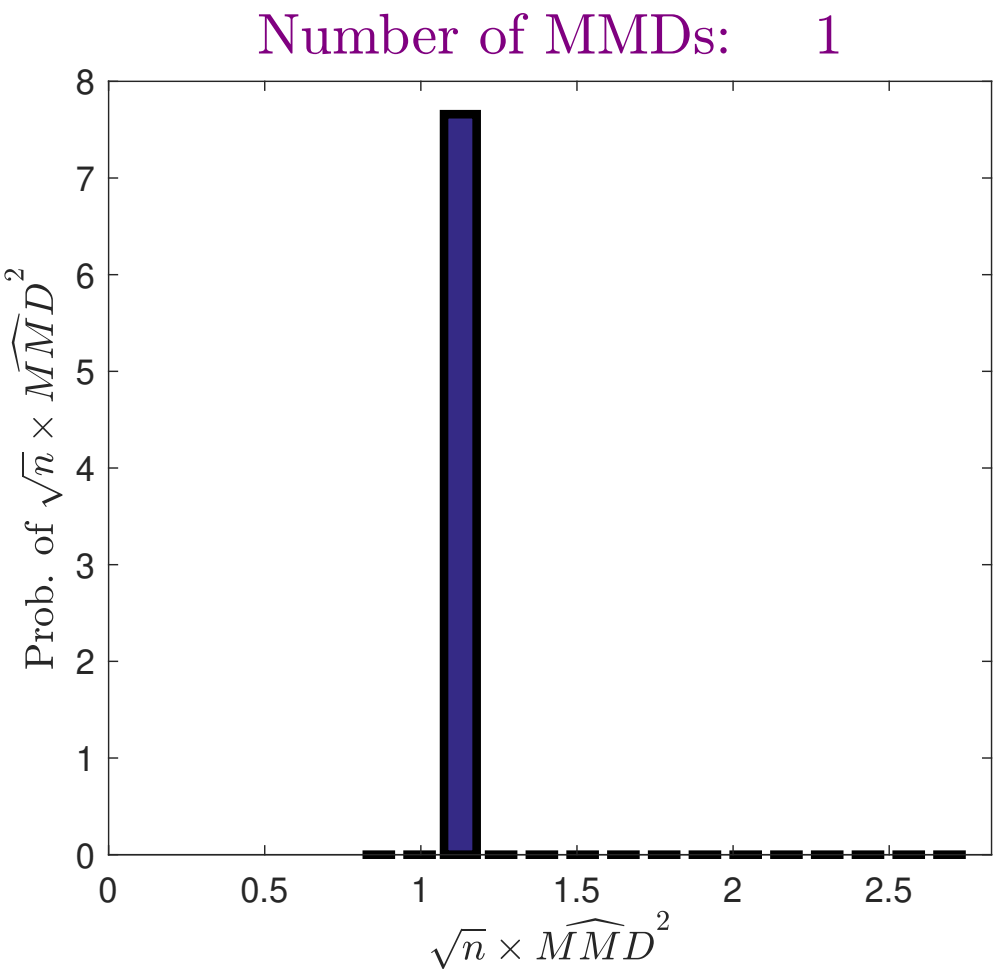
$\sqrt{n} \times \widehat{MMD}^2 = 1.2$



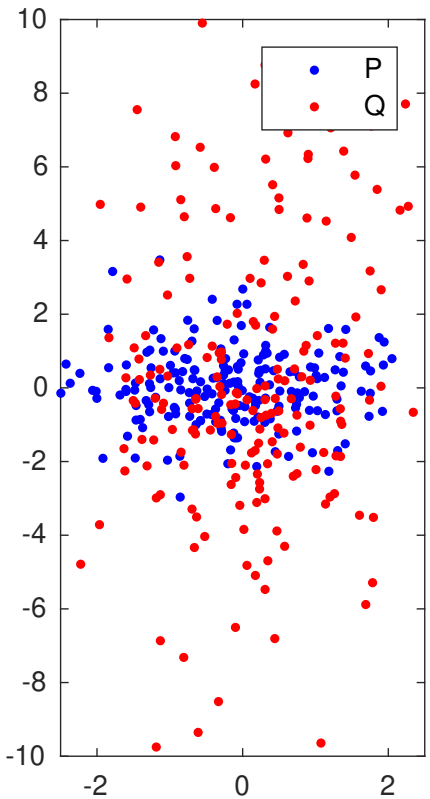
Behaviour of $\widehat{MMD}^2$ when $P \neq Q$

Draw $n = 200$ i.i.d samples from P and Q

- Laplace with different y-variance.
- $\sqrt{n} \times \widehat{\mathrm{MMD}}^2_\lambda(\mathbb{X}_m, \mathbb{Y}_n) = 1.2$



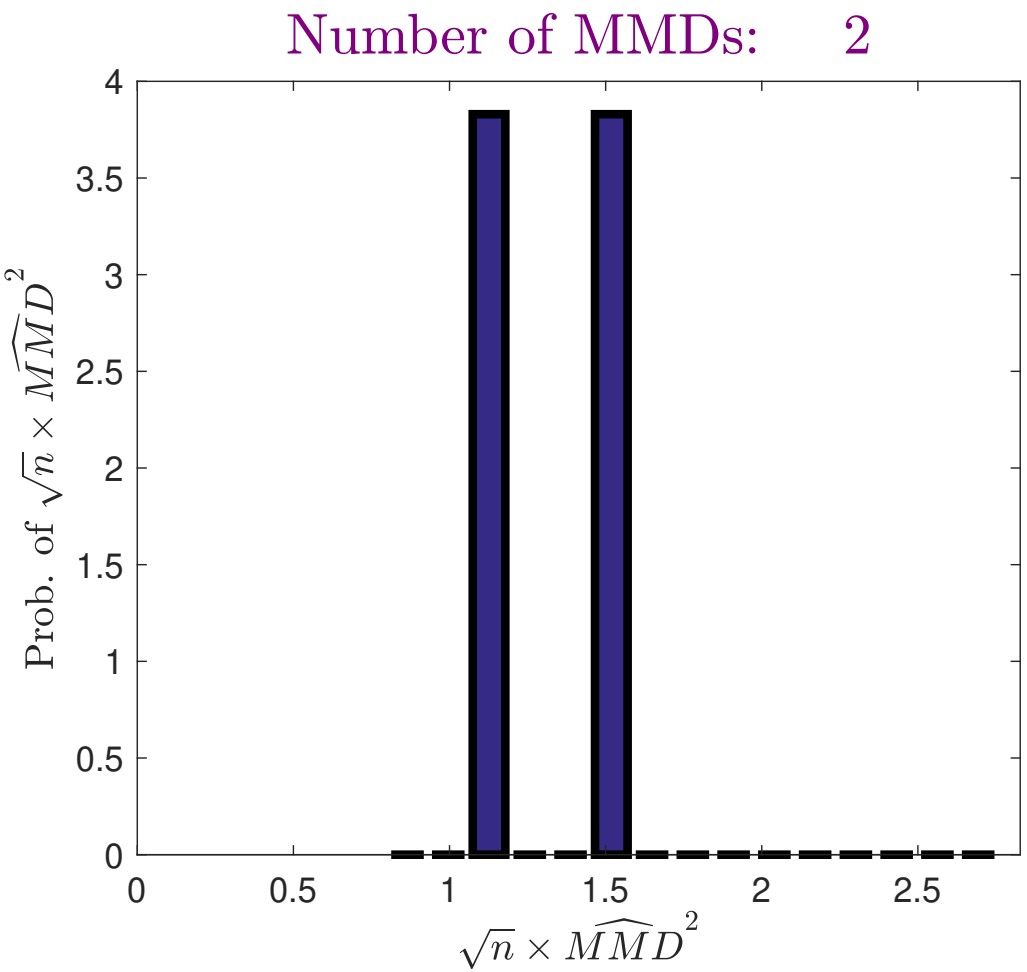
$\sqrt{n} \times \widehat{MMD}^2 = 1.2$



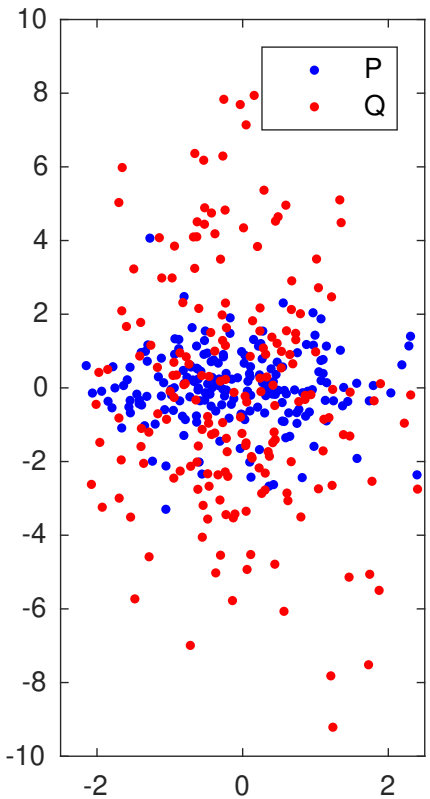
Behaviour of $\widehat{MMD}^2$ when $P \neq Q$

Draw $n = 200$ new samples from P and Q

- Laplace with different y-variance.
- $\sqrt{n} \times \widehat{\mathrm{MMD}}^2_\lambda(\mathbb{X}_m, \mathbb{Y}_n) = 1.5$

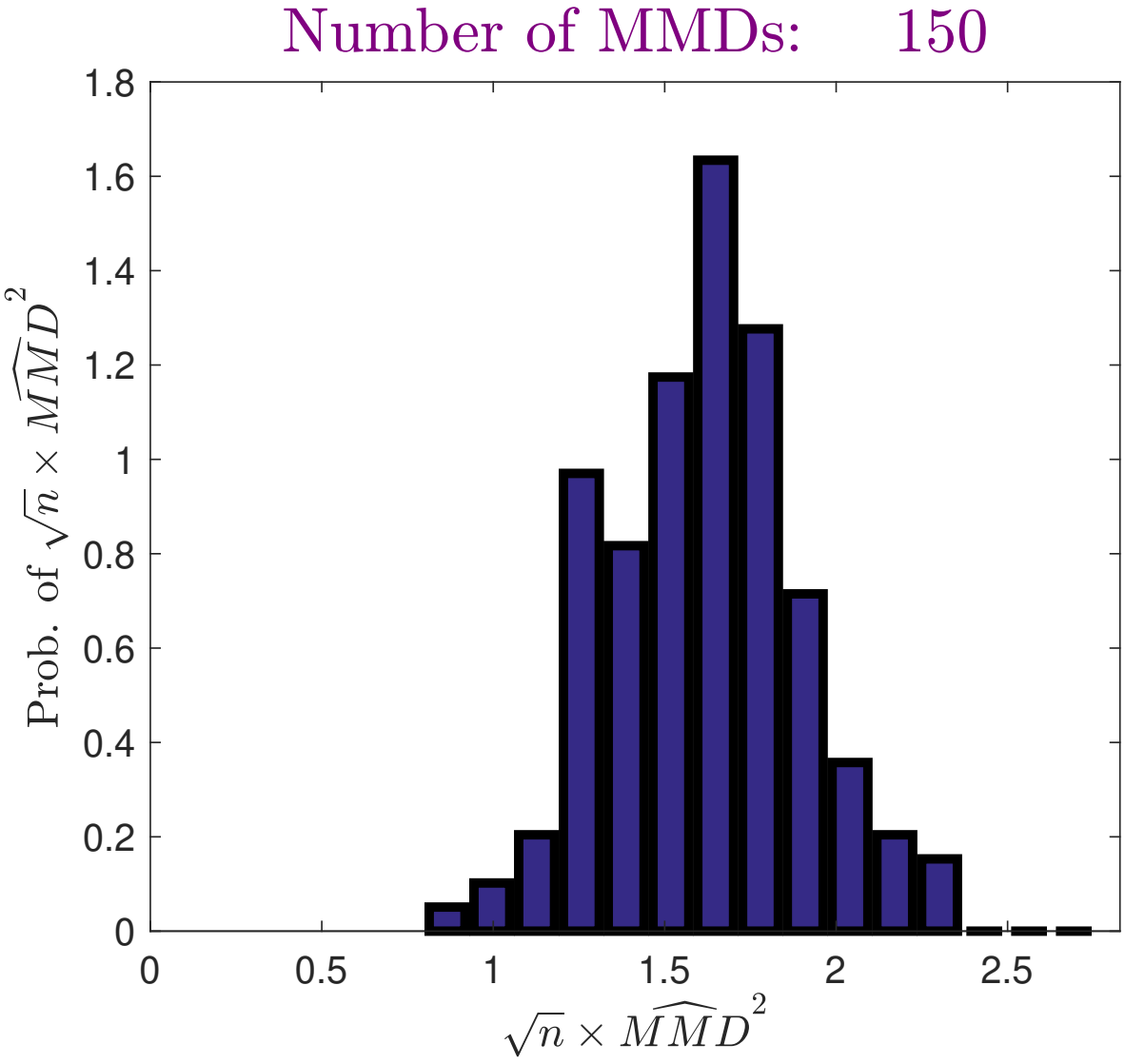


$\sqrt{n} \times \widehat{MMD}^2 = 1.5$



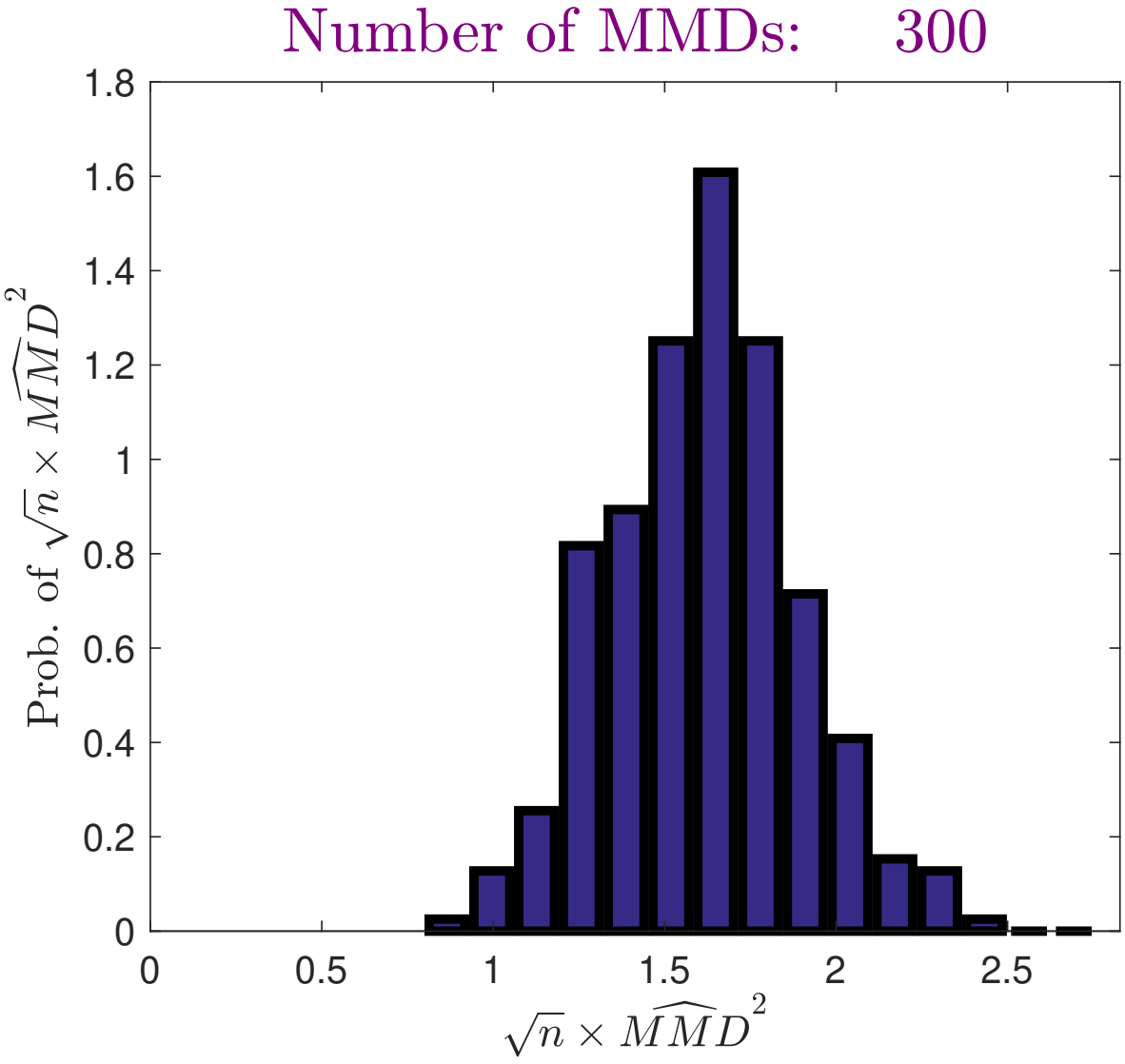
Behaviour of $\widehat{MMD}^2$ when $P \neq Q$

Repeat this 150 times ...



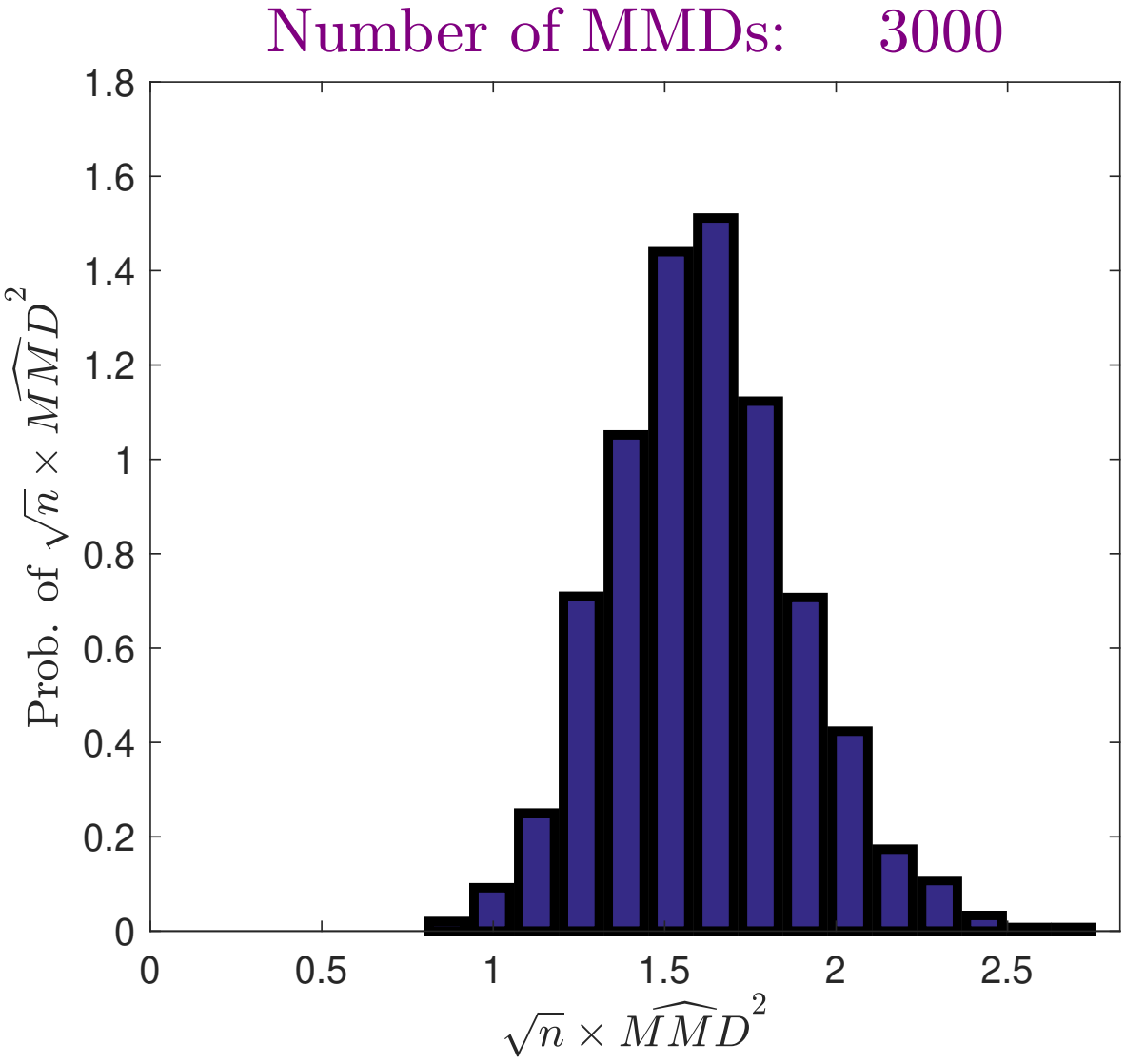
Behaviour of $\widehat{MMD}^2$ when $P \neq Q$

Repeat this 300 times ...



Behaviour of $\widehat{MMD}^2$ when $P \neq Q$

Repeat this 3000 times ...

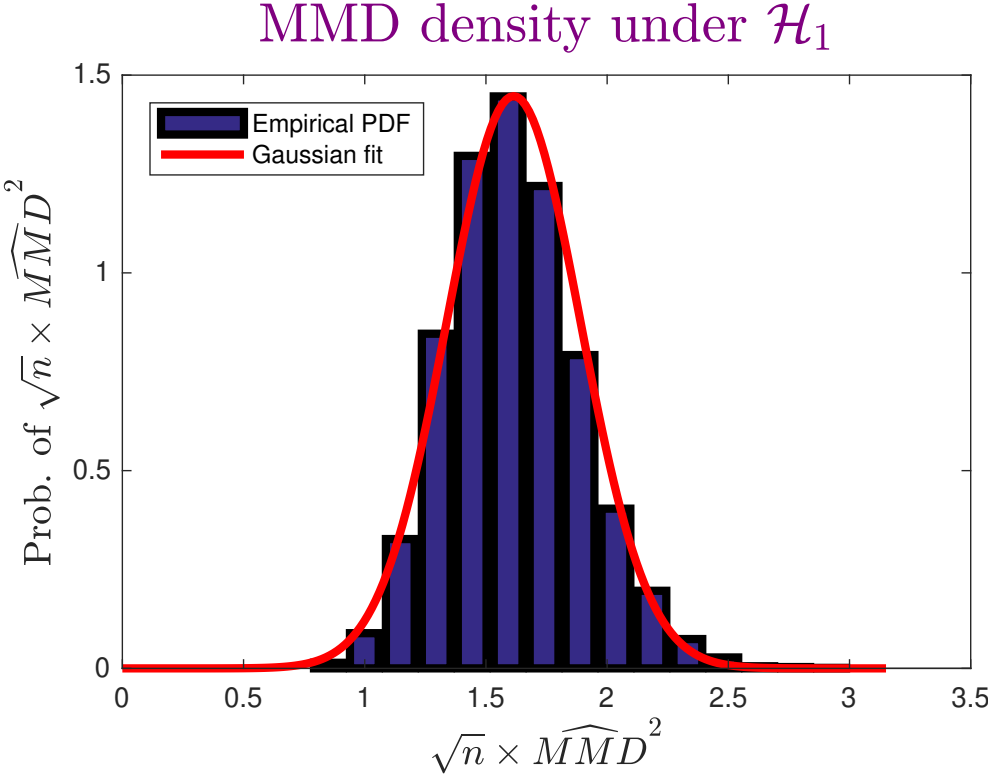


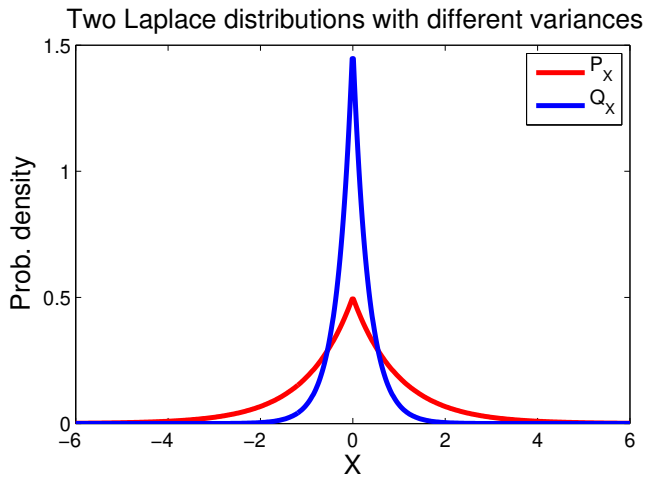
Asymptotics of $\widehat{MMD}^2$ when $P \neq Q$

When $P \neq Q$, statistic is asymptotically normal,

$$\sqrt{n}\left(\widehat{\mathrm{MMD}}^2_\lambda(\mathbb{X}_m, \mathbb{Y}_n) - \mathrm{MMD}^2_\lambda(p, q)\right) \xrightarrow{D} \mathcal{N}(0, \mathrm{V}_\lambda(p, q)),$$

with variance $\mathrm{V}_\lambda(p, q)$.



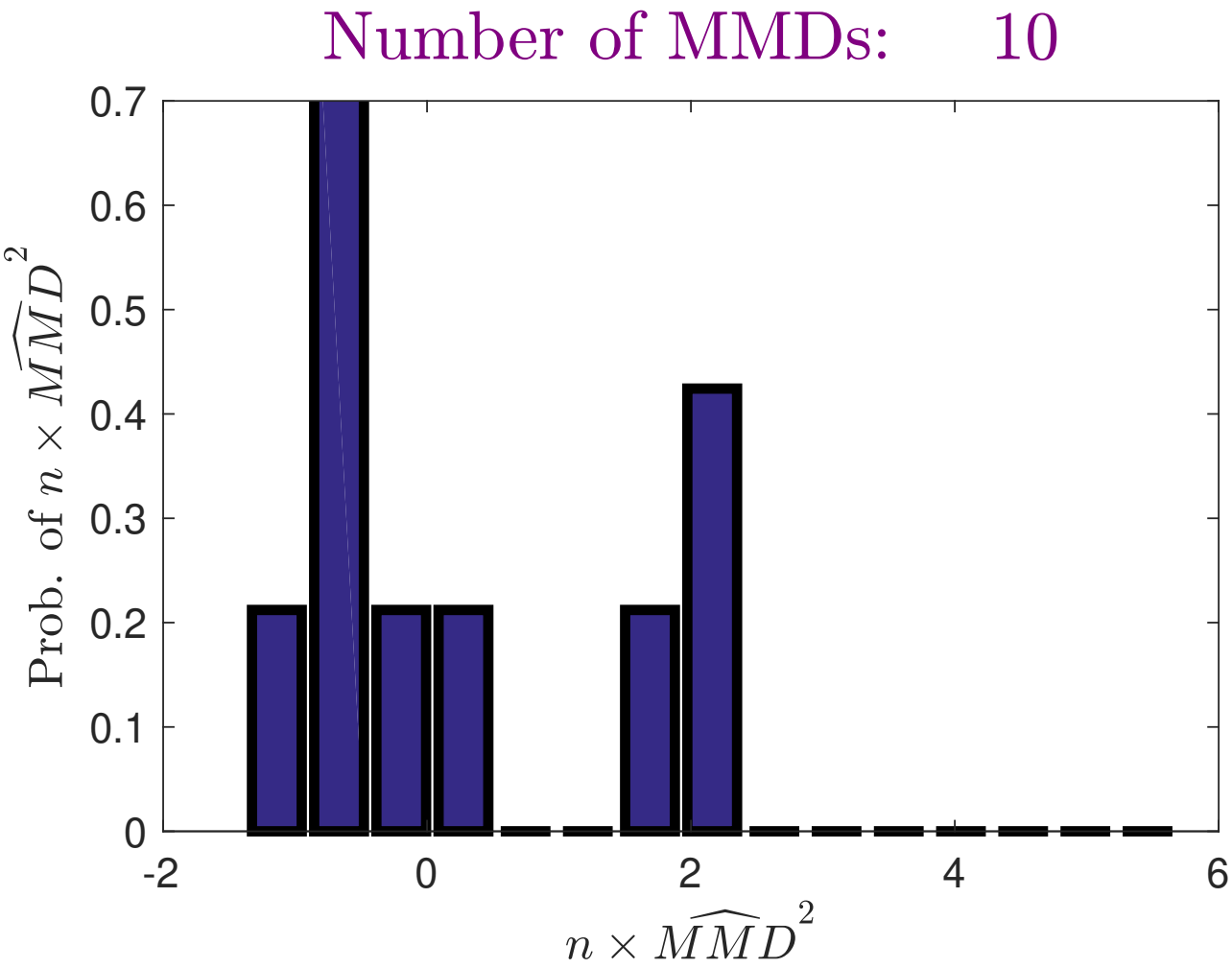


Behaviour of $\widehat{MMD}^2$ when $P = Q$

What happens when P and Q are the same?

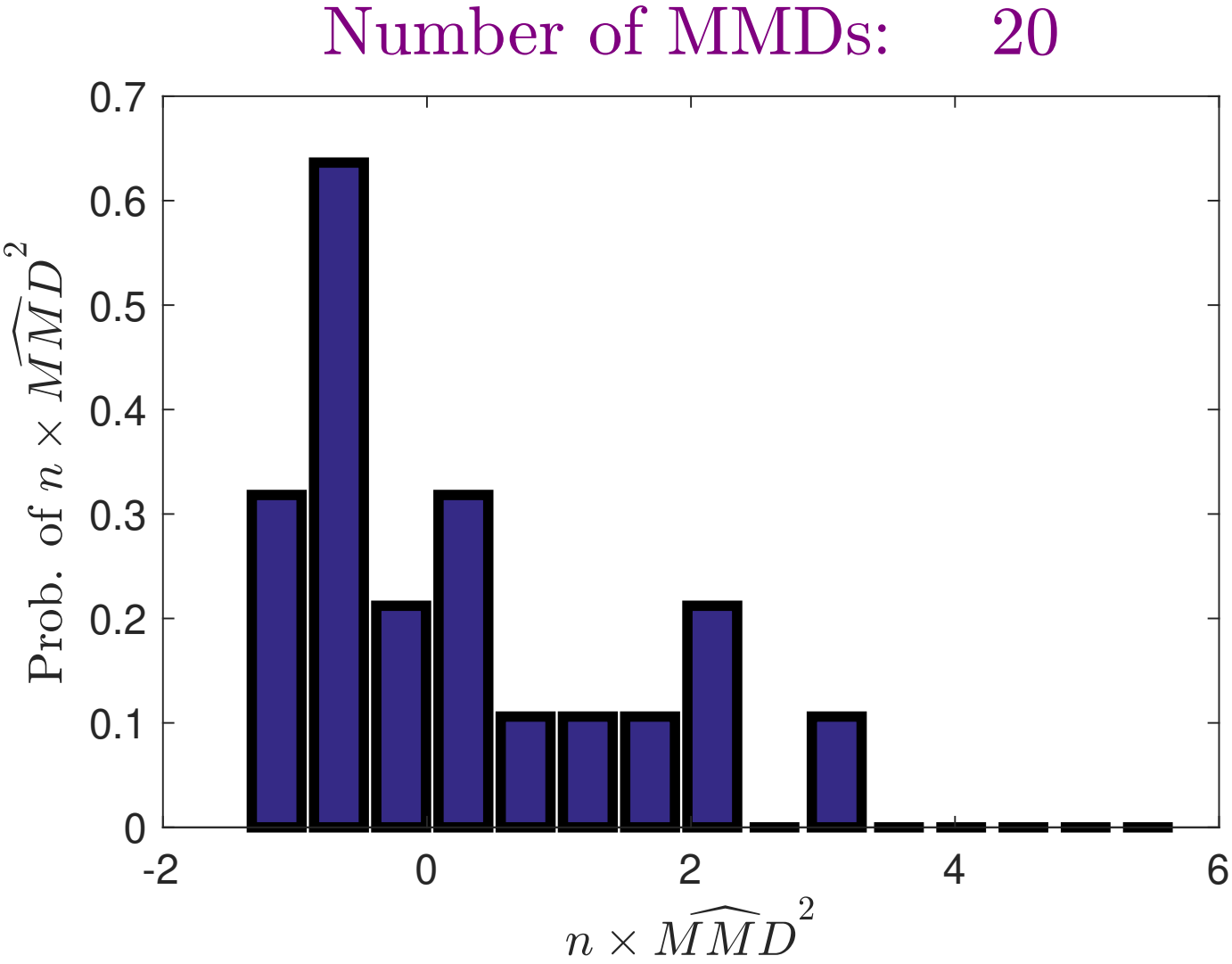
Behaviour of $\widehat{MMD}^2$ when $P = Q$

- Case of $P = Q = \mathcal{N}(0, 1)$



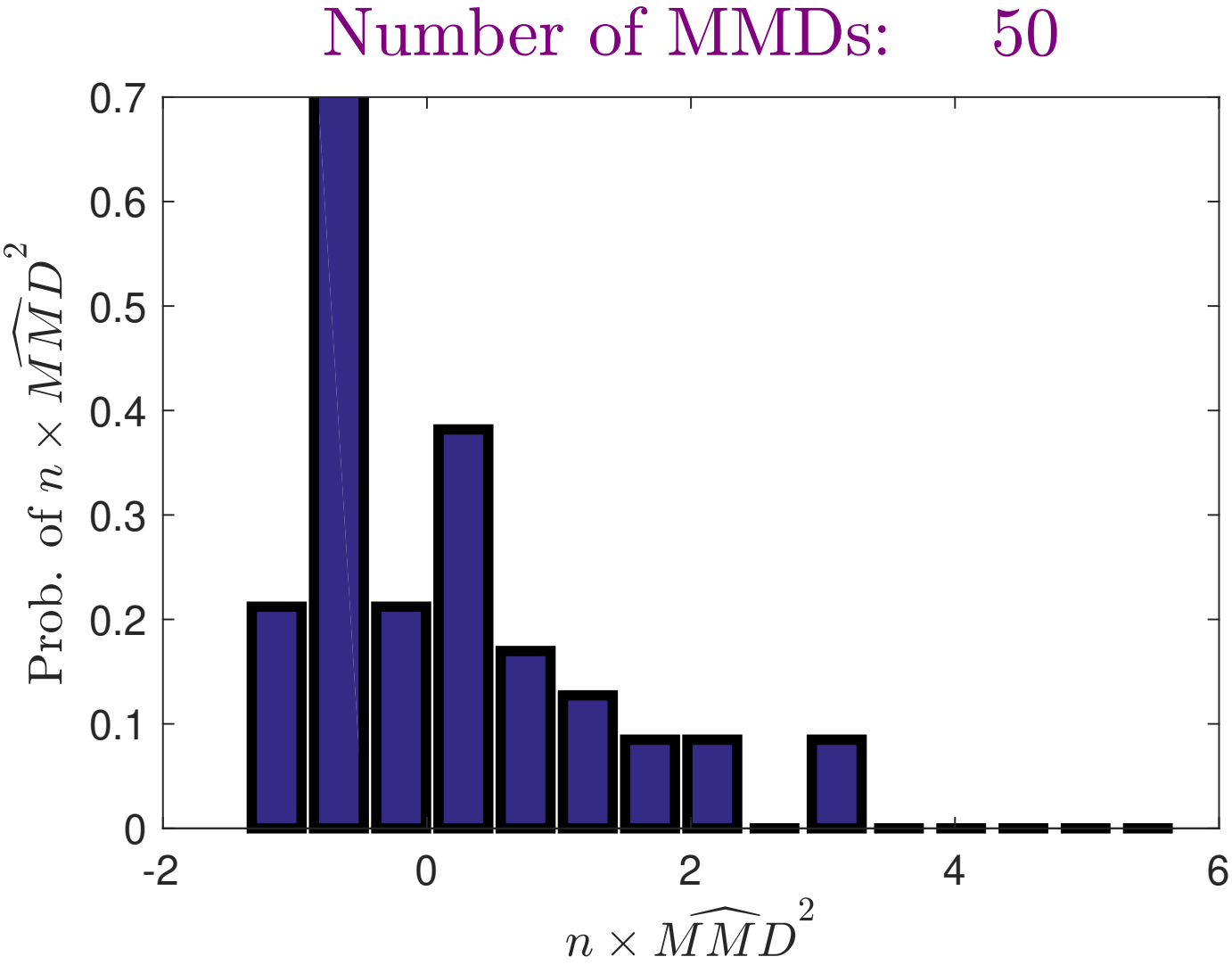
Behaviour of $\widehat{MMD}^2$ when $P = Q$

- Case of $P = Q = \mathcal{N}(0, 1)$



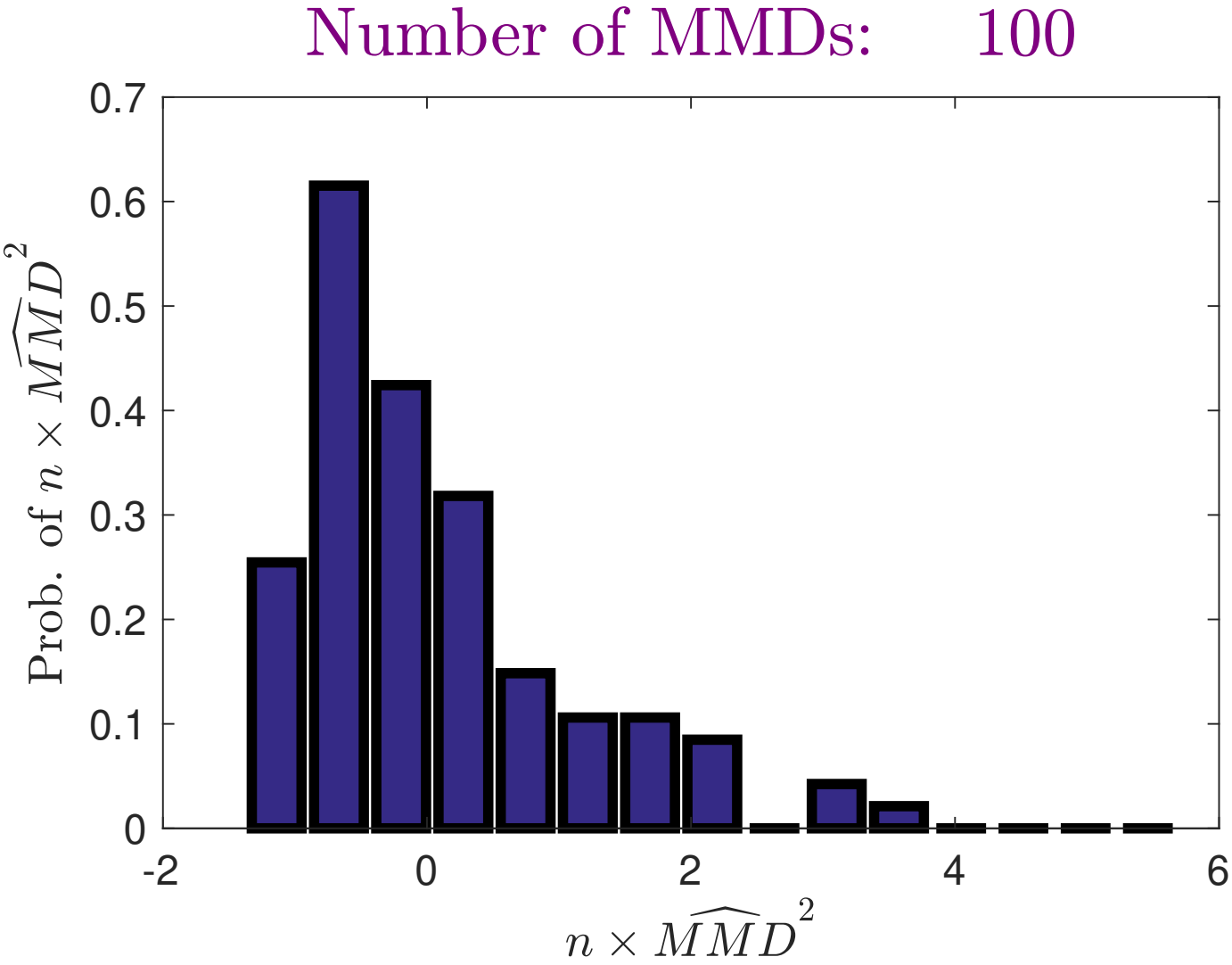
Behaviour of $\widehat{MMD}^2$ when $P = Q$

- Case of $P = Q = \mathcal{N}(0, 1)$



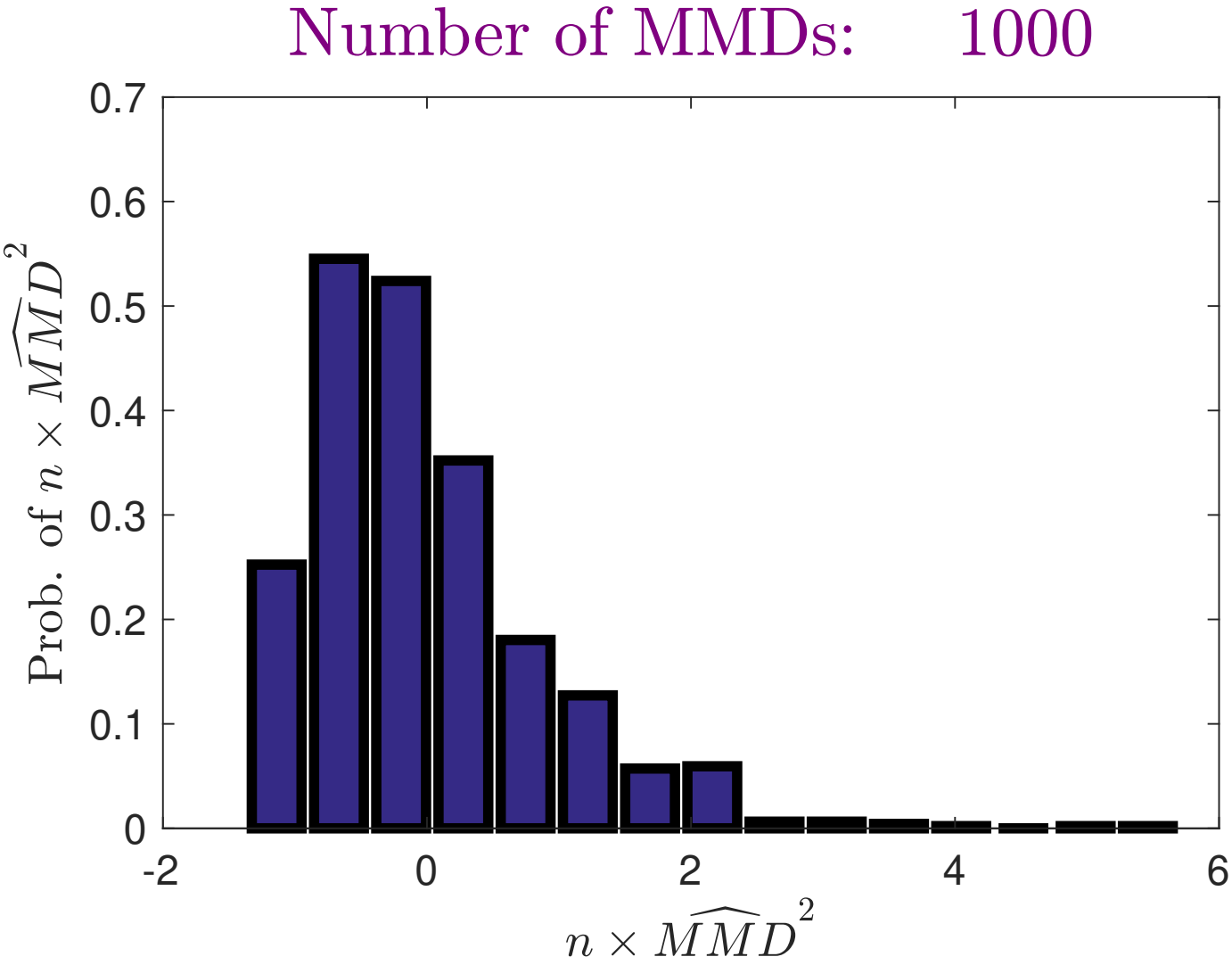
Behaviour of $\widehat{MMD}^2$ when $P = Q$

- Case of $P = Q = \mathcal{N}(0, 1)$



Behaviour of $\widehat{MMD}^2$ when $P = Q$

- Case of $P = Q = \mathcal{N}(0, 1)$

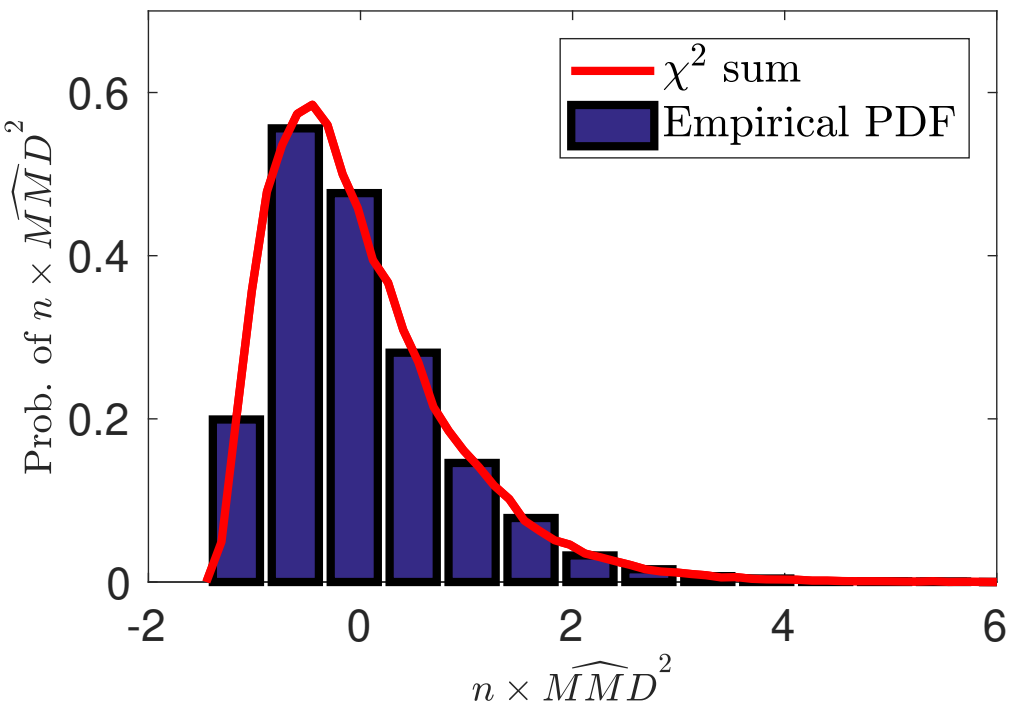


Asymptotics of $\widehat{MMD}^2$ when $P = Q$

$P = Q$, statistic has asymptotic distribution

$$(m+n)\widehat{\mathrm{MMD}}^2_\lambda(\mathbb{X}_m, \mathbb{Y}_n) \sim \sum_{l=1}^{\infty} \lambda_l \left[z_l^2 - 2\right]$$

MMD density under $\mathcal{H}_0$



where

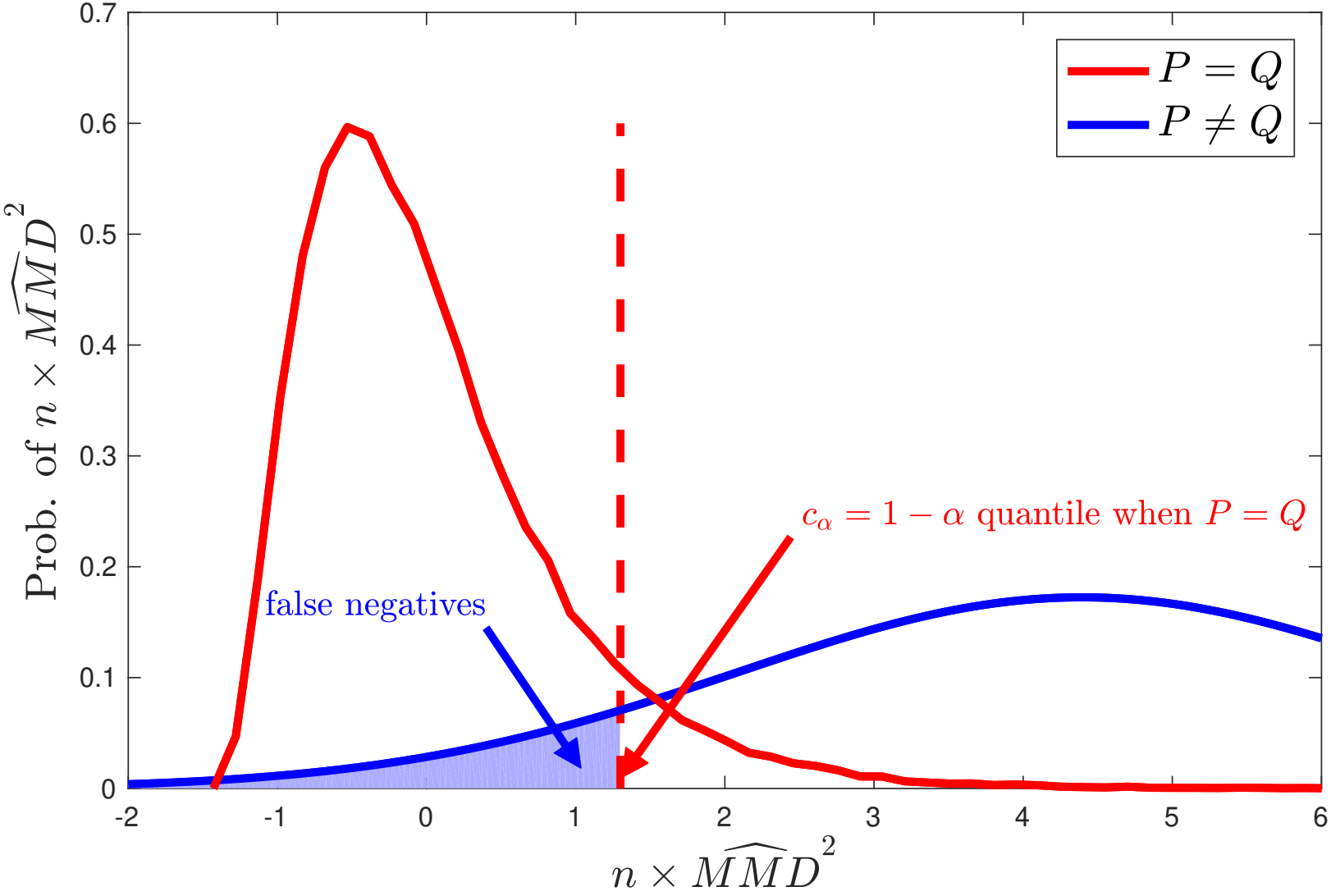
$$\lambda_i \psi_i(x') = \int_{\mathcal{X}} \underbrace{\tilde{k}_\lambda(x, x')}_{\text{centred}} \psi_i(x) dP(x)$$

$$z_l \sim \mathcal{N}(0, 2) \quad \text{i.i.d.}$$

Example plot: $P = \mathcal{N}(0, 1)$

A statistical test

Test construction:



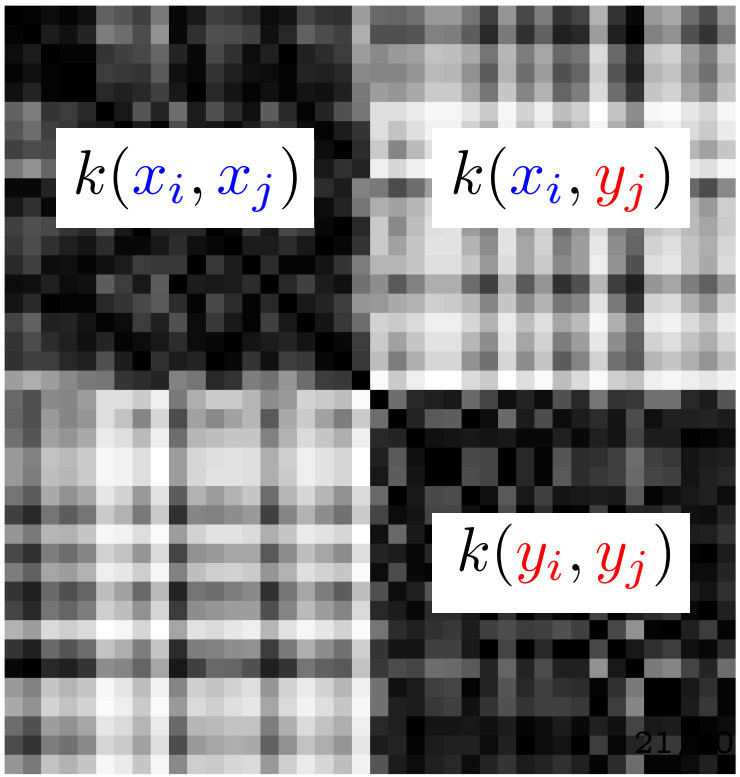
Test threshold $q_{1-\alpha}^{\lambda}$

Original empirical MMD for dogs and fish:

$X = [$  $\dots]$

$Y = [$  $\dots]$

$$\widehat{\mathrm{MMD}}^2_{\lambda}(\mathbb{X}_m, \mathbb{Y}_n)$$
$$= \frac{1}{n(n-1)} \sum_{i \neq j} k(x_i, x_j)$$
$$+ \frac{1}{n(n-1)} \sum_{i \neq j} k(y_i, y_j)$$
$$- \frac{2}{n^2} \sum_{i,j} k(x_i, y_j)$$

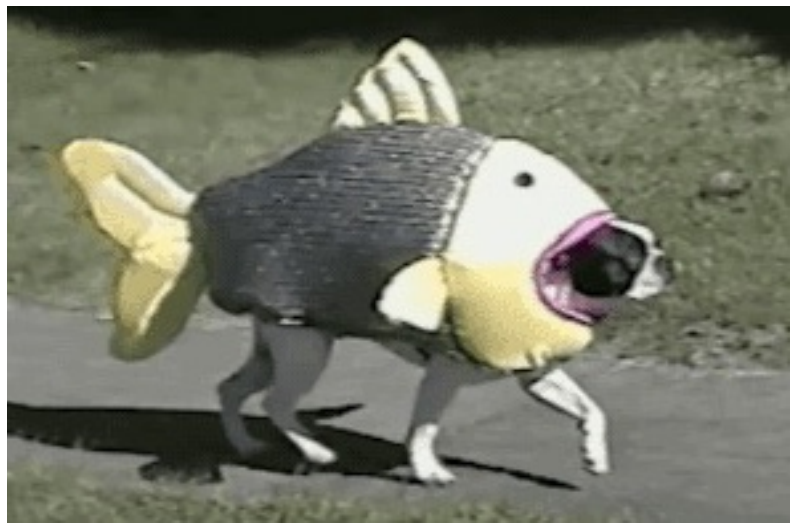


Test threshold $q^{\lambda}_{1-\alpha}$

Permuted dog and fish samples (merdogs):

$\widetilde{X} = [$    $\dots\ ]$

$\widetilde{Y} = [$    $\dots\ ]$



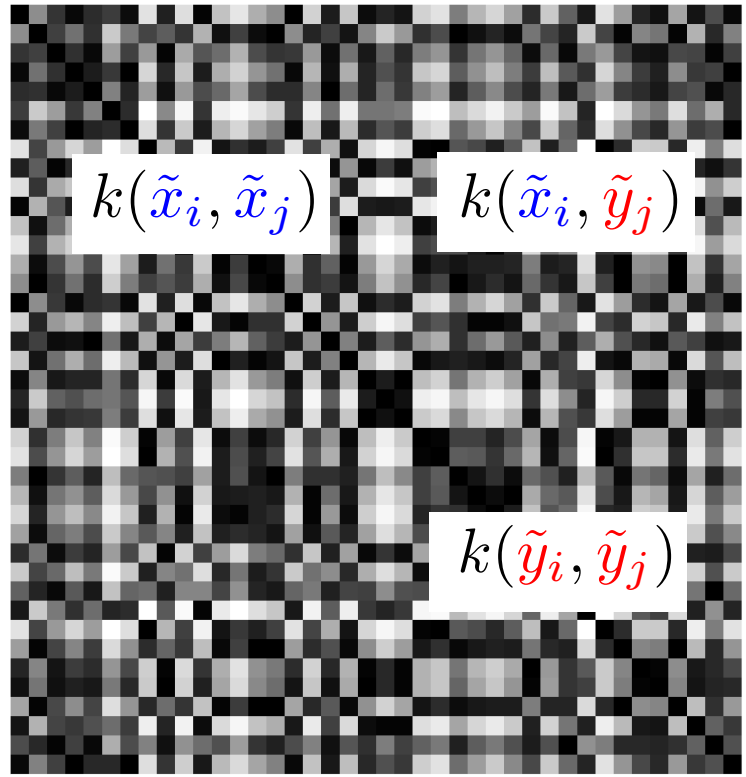
Test threshold $q^{\lambda}_{1-\alpha}$

Permuted dog and fish samples (merdogs):

$\widetilde{X} = [$  $\dots\ ]$

$\widetilde{Y} = [$  $\dots\ ]$

$$\widehat{\mathrm{MMD}}^2_{\lambda}(\tilde{\mathbb{X}}_m, \tilde{\mathbb{Y}}_n)$$
$$= \frac{1}{n(n-1)} \sum_{i \neq j} k(\tilde{x}_i, \tilde{x}_j)$$
$$+ \frac{1}{n(n-1)} \sum_{i \neq j} k(\tilde{y}_i, \tilde{y}_j)$$
$$- \frac{2}{n^2} \sum_{i,j} k(\tilde{x}_i, \tilde{y}_j)$$



Permutation simulates $P = Q$

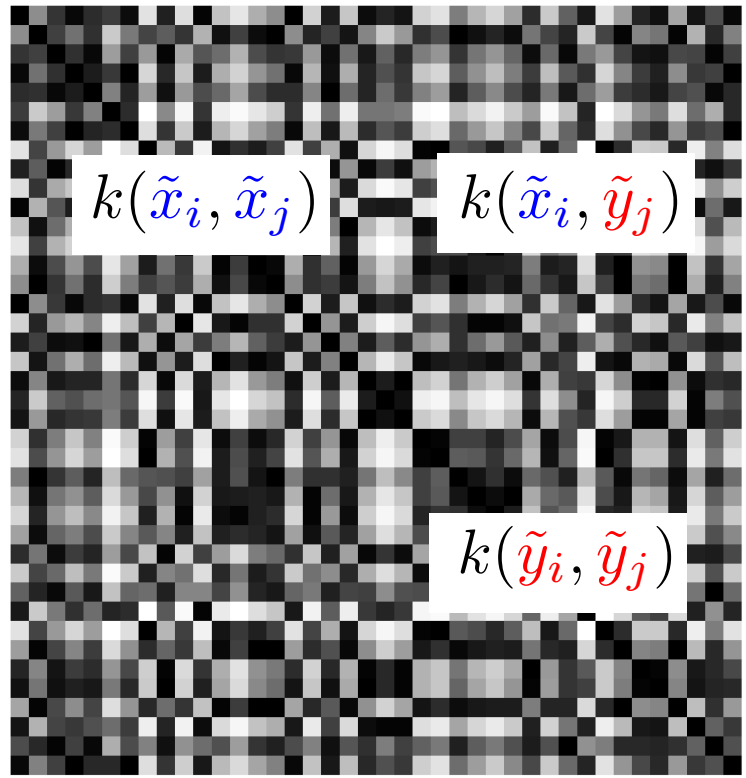
Test threshold $q^{\lambda}_{1-\alpha}$

Permuted dog and fish samples (merdogs):

$\widetilde{X} = [$  $\dots\ ]$

$\widetilde{Y} = [$  $\dots\ ]$

Using threshold $q^{\lambda}_{1-\alpha}$ gives exact level α (upper bound on false positive rate) at finite n and number of permutations (when unpermuted statistic included in pool)



How to choose your kernel?

MMD Fuse

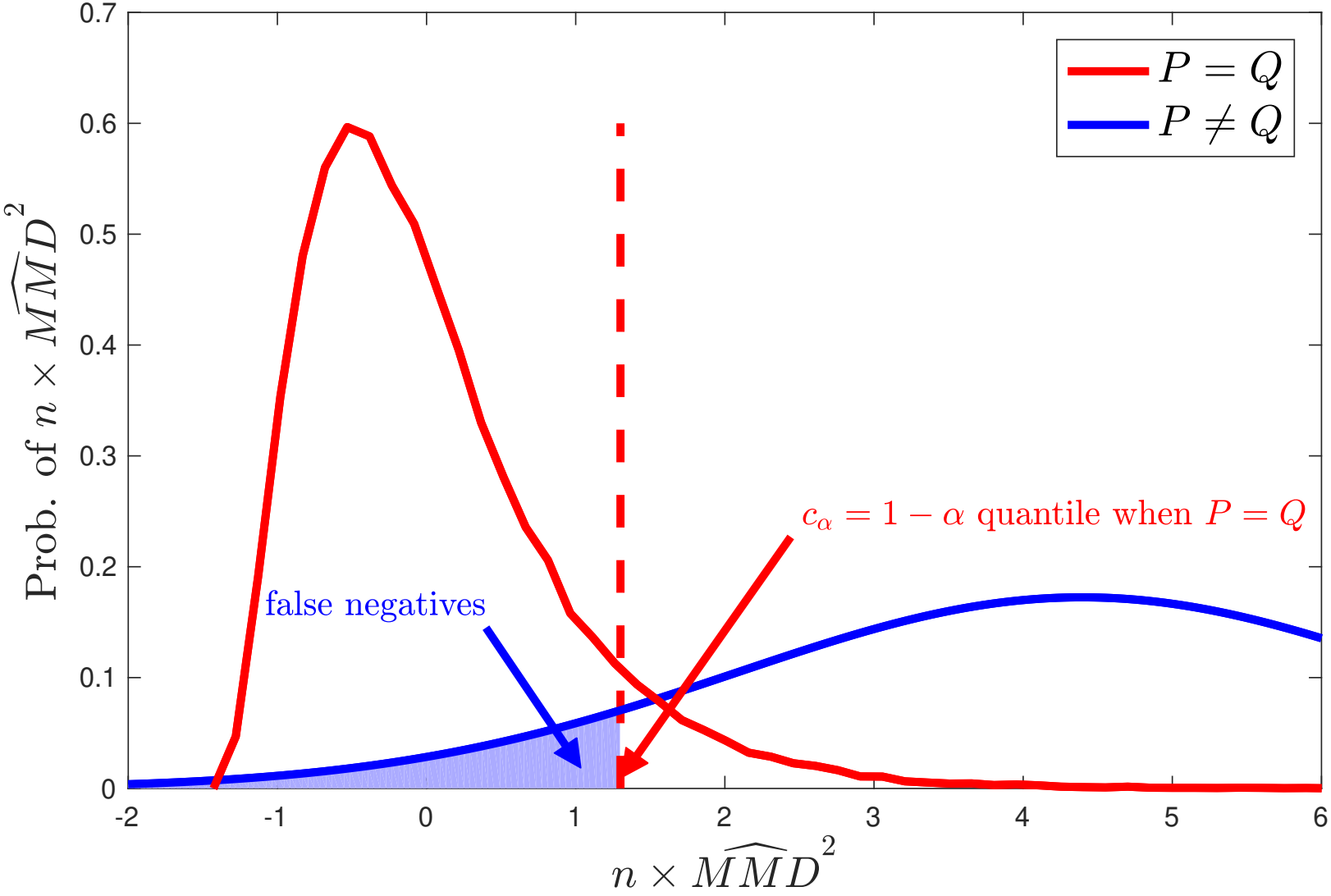
MMD Fuse (NeurIPS 2023):

Code:

https://github.com/antoninschrab/mmdfuse

A statistical test

Test construction:



Test power - intuition

The power of our test (assume $m = n$):

$$\Pr_{P \neq Q}\left(n\widehat{\mathrm{MMD}}^2_{\lambda}(\mathbb{X}_m, \mathbb{Y}_n) > q^{\lambda}_{1-\alpha} \right)$$

Test power - intuition

The power of our test (assume $m = n$):

$$\Pr_{P \neq Q}\left(n\widehat{\mathrm{MMD}}^2_\lambda(\mathbb{X}_m, \mathbb{Y}_n) > \hat{q}^\lambda_{1-\alpha}\right)$$
$$\to \Phi\left(\frac{\mathrm{MMD}^2_\lambda(p, q)}{\sqrt{n^{-1}\mathrm{V}_\lambda(p, q)}} - \frac{q^\lambda_{1-\alpha}}{n\sqrt{n^{-1}\mathrm{V}_\lambda(p, q)}}\right)$$

where

- Φ is the CDF of the standard normal distribution.
- $\mathrm{V}_\lambda(p, q)$ is asymptotic variance of $\sqrt{n}\left(\widehat{\mathrm{MMD}}^2_\lambda(\mathbb{X}_m, \mathbb{Y}_n) - \mathrm{MMD}^2_\lambda(p, q)\right)$
- $\hat{q}^\lambda_{1-\alpha}$ is an estimate of $q^\lambda_{1-\alpha}$ test threshold.

Test power - intuition

The power of our test (assume $m = n$):

$$\Pr_{P \neq Q}\left(n\widehat{\mathrm{MMD}}^2_\lambda(\mathbb{X}_m, \mathbb{Y}_n) > \hat{q}^\lambda_{1-\alpha} \right)$$

$$\to \Phi\left(\underbrace{\frac{\mathrm{MMD}^2_\lambda(p,q)}{\sqrt{n^{-1}\mathrm{V}_\lambda(p,q)}}}_{O(n^{1/2})} - \underbrace{\frac{q^\lambda_{1-\alpha}}{n\sqrt{n^{-1}\mathrm{V}_\lambda(p,q)}}}_{O(n^{-1/2})} \right)$$

For large n, second term negligible!

Test power - intuition

The power of our test (assume $m = n$):

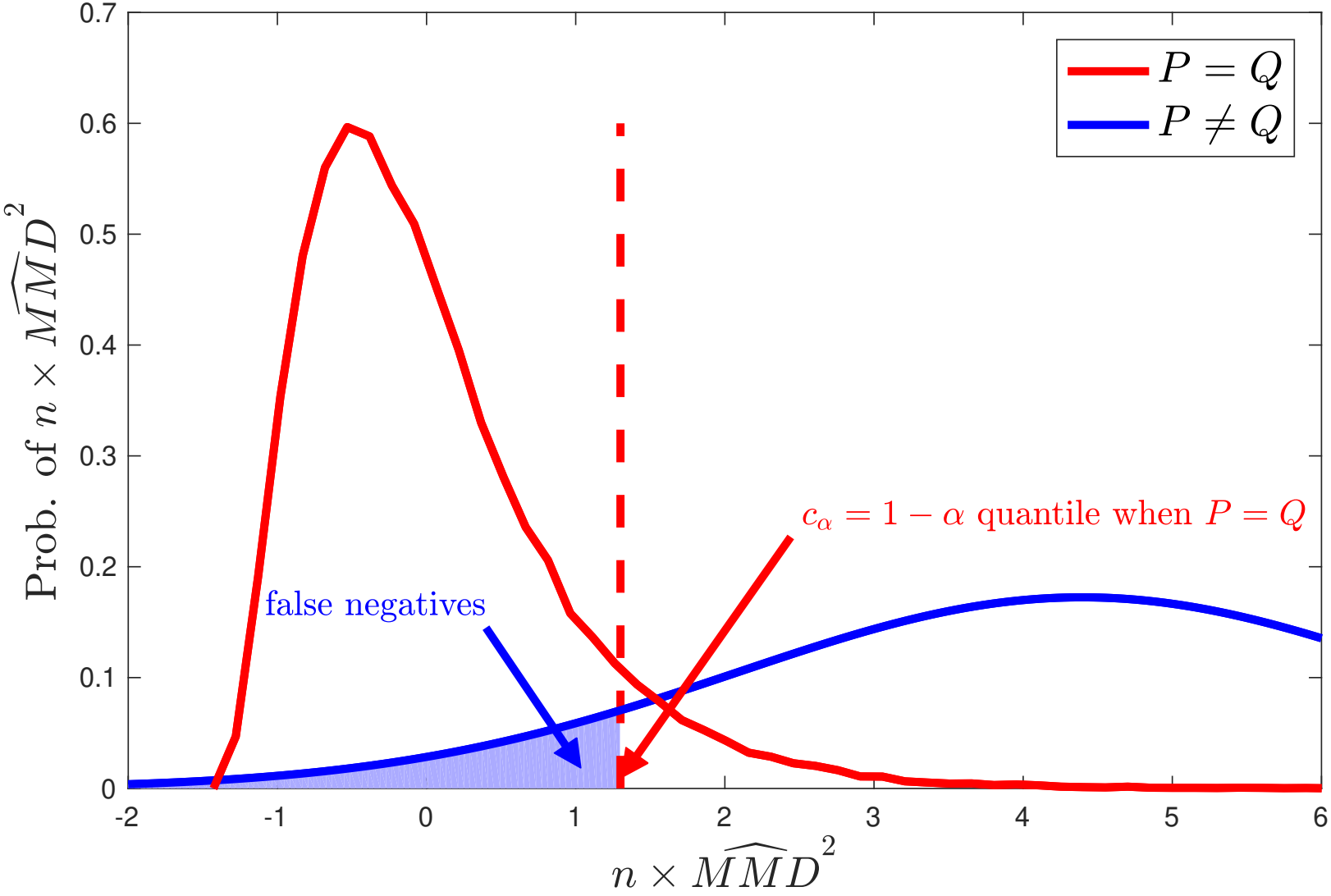
$$\Pr_{P \neq Q}\left(n\widehat{\mathrm{MMD}}^2_\lambda(\mathbb{X}_m, \mathbb{Y}_n) > \hat{q}^\lambda_{1-\alpha}\right)$$
$$\to \Phi\left(\frac{\mathrm{MMD}^2_\lambda(p,q)}{\sqrt{n^{-1}\mathrm{V}_\lambda(p,q)}} - \frac{q^\lambda_{1-\alpha}}{n\sqrt{n^{-1}\mathrm{V}_\lambda(p,q)}}\right)$$

To maximize test power,

$$\lambda^\star := \operatorname*{argmax}_{\lambda\in\Lambda} \frac{\mathrm{MMD}^2_\lambda(p,q)}{\sqrt{\mathrm{V}_\lambda(p,q)}}$$

Test power - intuition

Test construction:



[Schrab, Biggs, G. 2023]

You choose? You fuse!

Use a softmax over the normalized MMDs

$$\frac{1}{\eta}\log\left(\frac{1}{|\Lambda|}\sum_{\lambda\in\Lambda}\exp\left(\eta\ \widehat{\mathrm{MMD}}^2_\lambda(\mathbb{X}_m,\mathbb{Y}_n)\ /\ \sqrt{\widehat{\mathrm{N}}_\lambda(\mathbb{X}_m,\mathbb{Y}_n)}\right)\right)$$

You choose? You fuse!

Use a softmax over the normalized MMDs

$$\frac{1}{\eta}\log\left(\frac{1}{|\Lambda|}\sum_{\lambda\in\Lambda}\exp\left(\eta\ \widehat{\mathrm{MMD}}^2_\lambda(\mathbb{X}_m,\mathbb{Y}_n)\ /\ \sqrt{\widehat{\mathrm{N}}_\lambda(\mathbb{X}_m,\mathbb{Y}_n)}\right)\right)$$

Variance $\mathrm{V}_\lambda(p,q)$ replaced with permutation-invariant proxy

$$\widehat{\mathrm{N}}_\lambda(\mathbb{X}_m,\mathbb{Y}_n) := \frac{1}{(m+n)(m+n-1)}\sum_{1\le i\neq j\le m+n} k_\lambda(Z_i,Z_j)^2$$

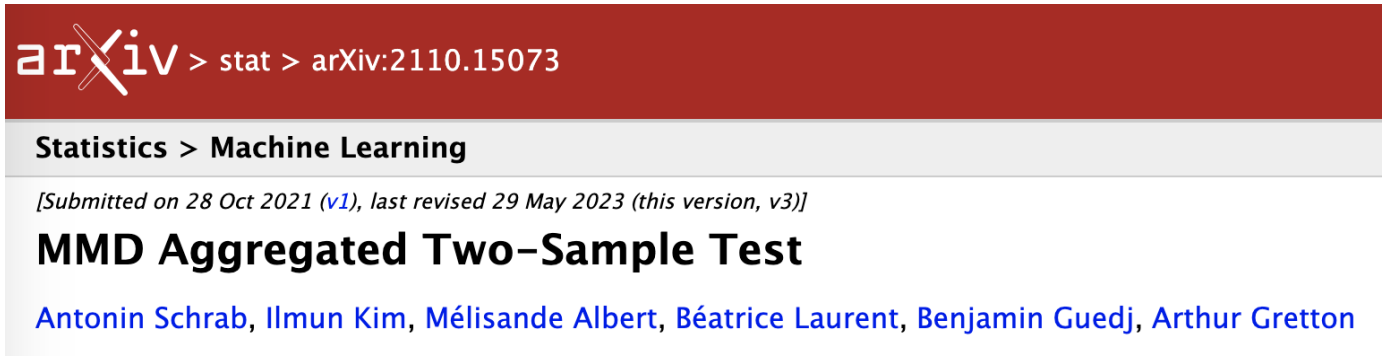
where

$$\mathbb{Z}_{m+n} = (\mathbb{X}_m,\mathbb{Y}_n)$$

and $\eta\sim(m+n)$.

MMD Aggregation

MMD Aggregated Two-Sample Test (JMLR 2023):

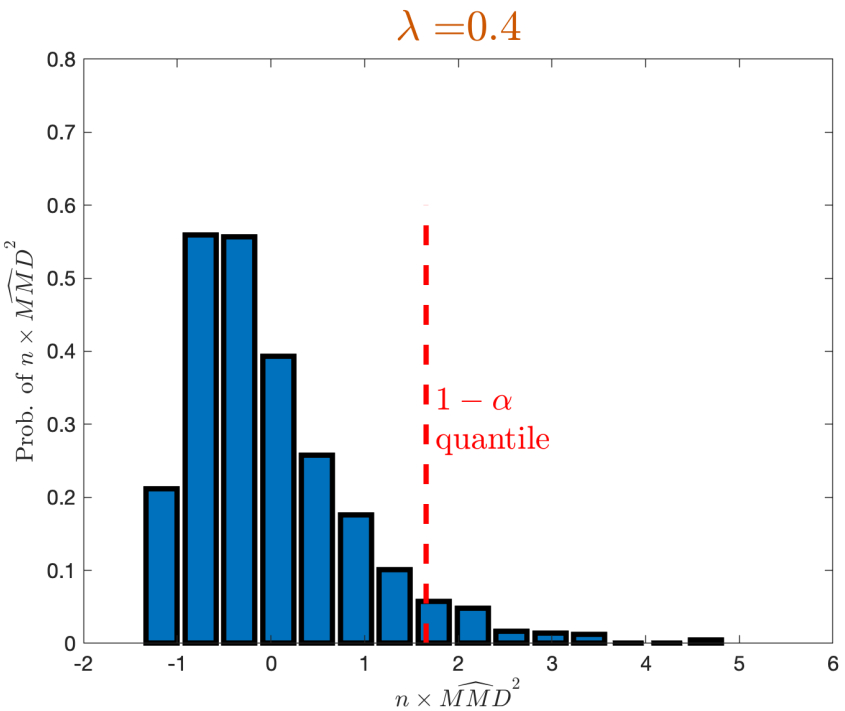


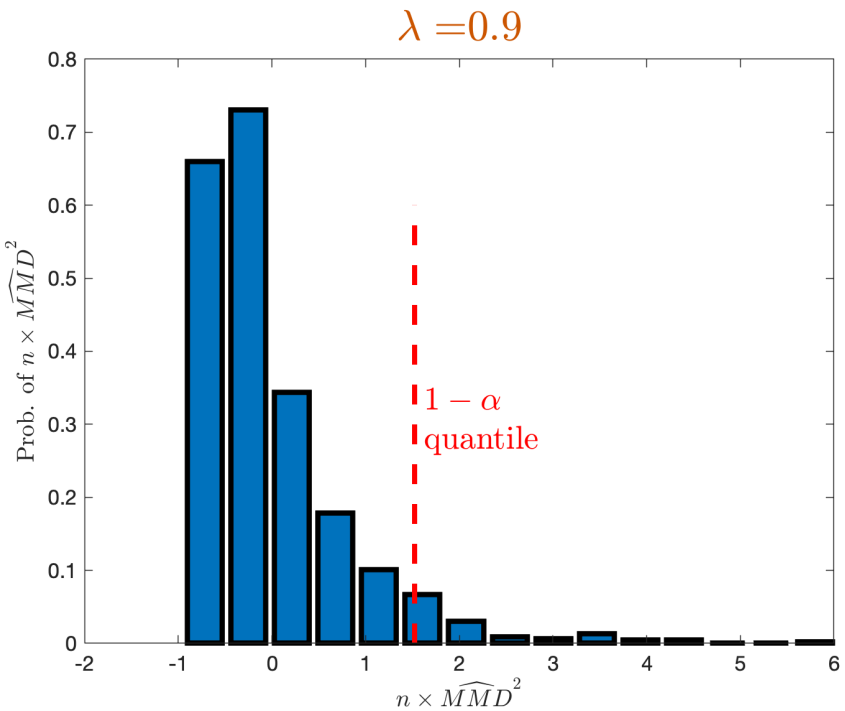
Code:

https://github.com/antoninschrab/mmdagg-paper

The test quantile

Individual tests for each λ have false positive rate α....

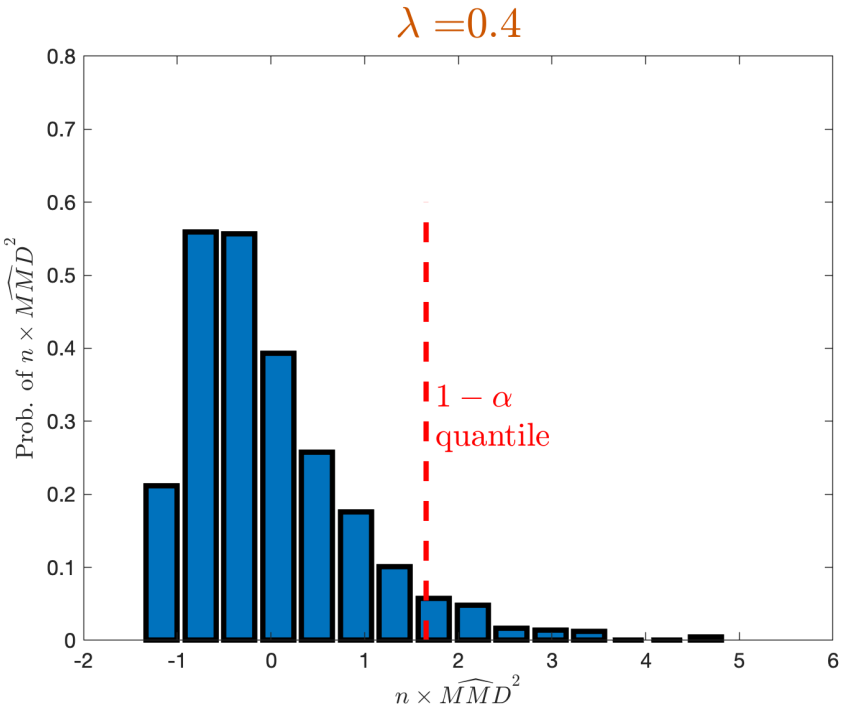


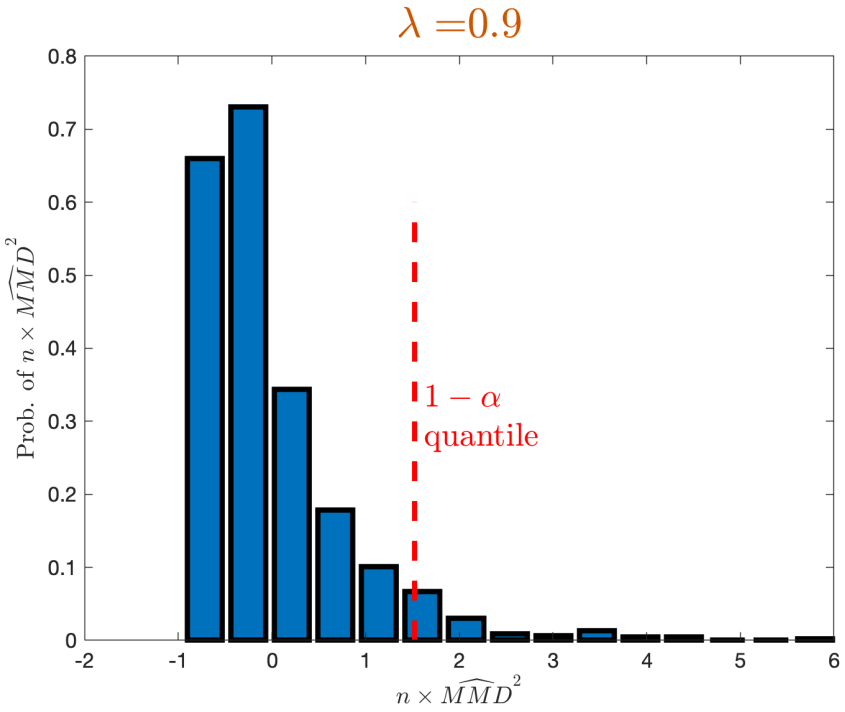


The test quantile

Individual tests for each λ have false positive rate α....

... but we want overall rejection rate for any kernel $\lambda \in \Lambda$ to be $\leq \alpha$



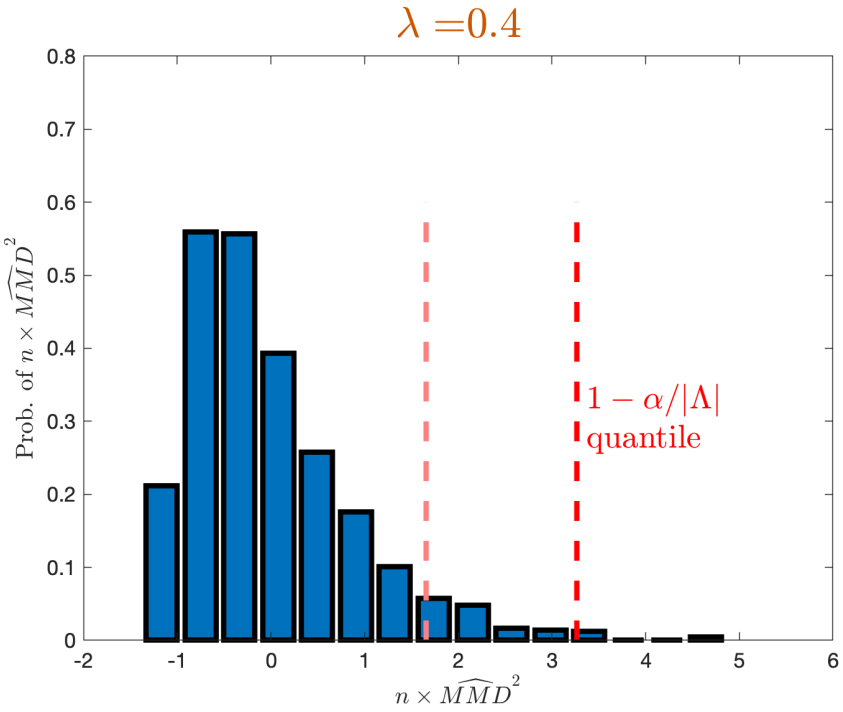


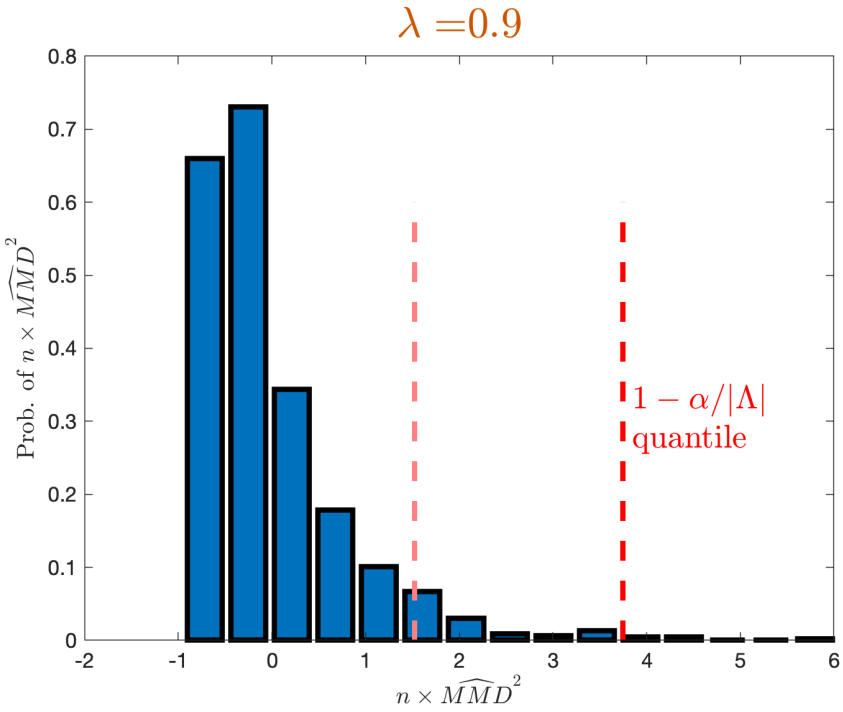
Bonferroni correction

Overall rate of rejection for any kernel $\lambda \in \Lambda$ is $\leq \alpha$:

Use quantile $\alpha/|\Lambda|$

This is correct (Bonferroni)...



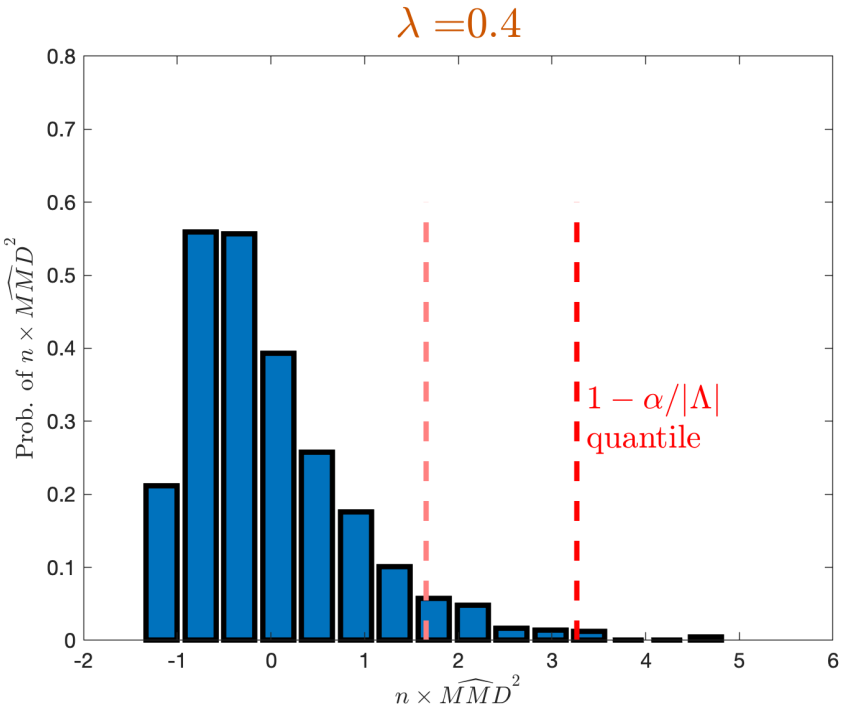


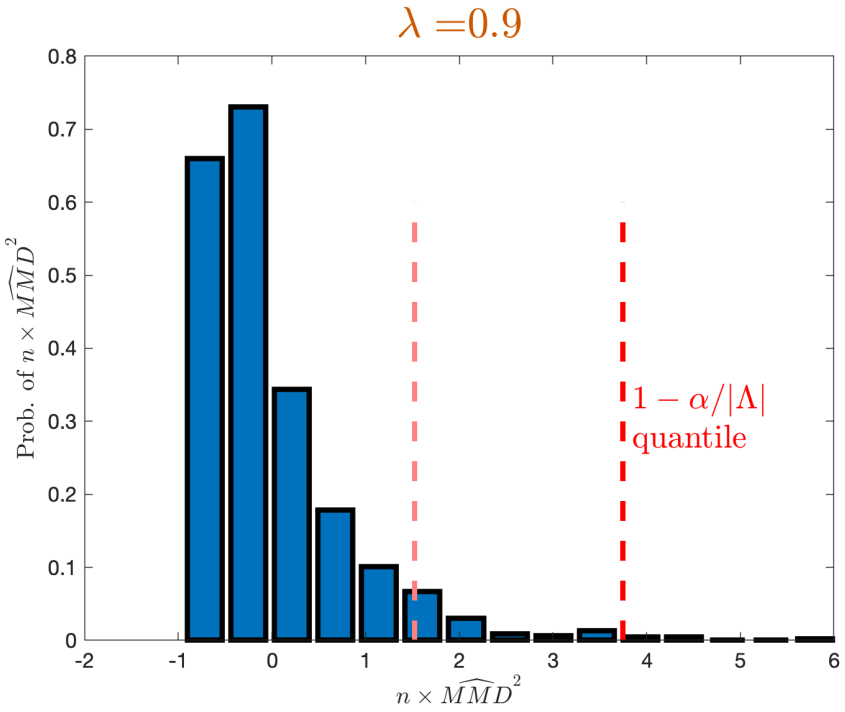
Bonferroni correction

Overall rate of rejection for any kernel $\lambda \in \Lambda$ is $\leq \alpha$:

Use quantile $\alpha/|\Lambda|$

This is correct (Bonferroni)..... but very conservative





[Schrab, Kim, Albert, Laurent, Guedj, G. 2023]

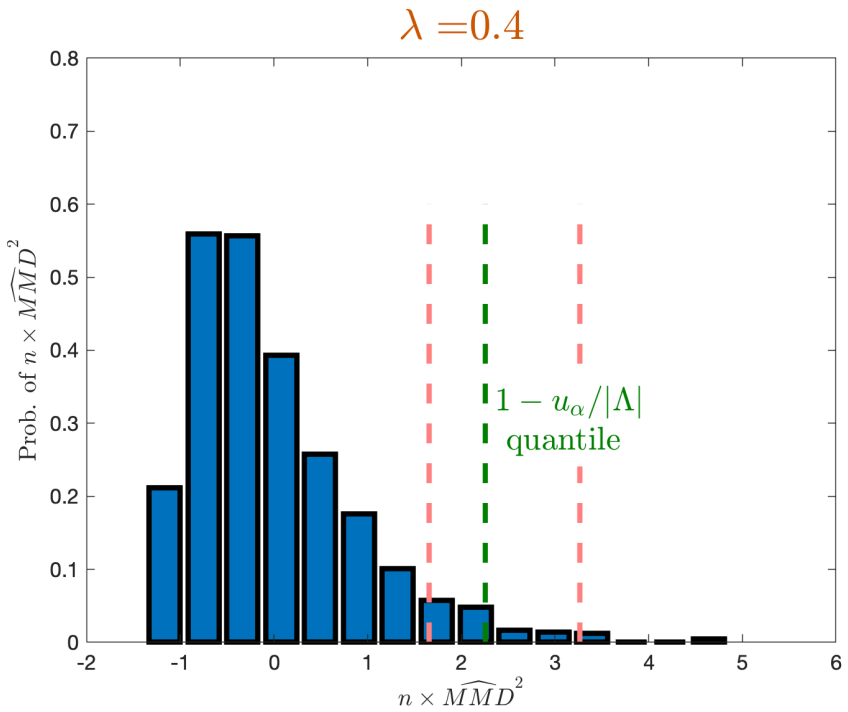
Aggregation

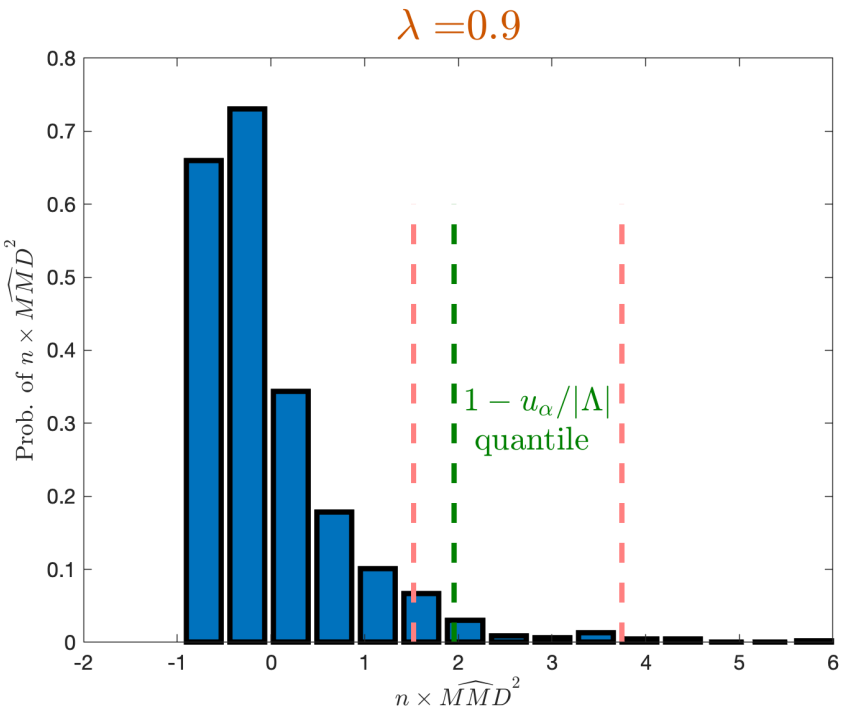
Overall rate of rejection for any kernel $\lambda \in \Lambda$ is $\leq \alpha$:

Use quantile $u_\alpha / |\Delta|$

with u_α chosen to give overall rejection rate α

... but how?





Aggregation

Overall rate of rejection for any kernel $\lambda \in \Lambda$ is $\leq \alpha$:

$$\text{Use quantile } u_\alpha / |\Delta|$$

with u_α chosen to give overall rejection rate α

... but how?

Do another permutation

$$\left(\tilde{\mathbb{X}}_m, \tilde{\mathbb{Y}}_n\right) \sim p \times p$$

Using $\left(\tilde{\mathbb{X}}_m, \tilde{\mathbb{Y}}_n\right)$, search over u_α such that prob. of any $\lambda \in \Lambda$ rejecting is $\leq \alpha$.

Aggregation

Overall rate of rejection for any kernel $\lambda \in \Lambda$ is $\leq \alpha$:

$$\text{Use quantile } u_\alpha / |\Delta|$$

with u_α chosen to give overall rejection rate α

... but how?

Do another permutation

$$\left(\tilde{\mathbb{X}}_m, \tilde{\mathbb{Y}}_n\right) \sim p \times p$$

Using $\left(\tilde{\mathbb{X}}_m, \tilde{\mathbb{Y}}_n\right)$, search over u_α such that prob. of any $\lambda \in \Lambda$ rejecting is $\leq \alpha$.

Time complexity

$$\mathcal{O}\left(|\Lambda| \left(B_1 + B_2\right) (m+n)^2\right)$$

where B_1 permutations to get quantiles, B_2 permutations to get u_α.

Aggregation

Overall rate of rejection for any kernel $\lambda \in \Lambda$ is $\leq \alpha$:

$$\text{Use quantile } u_\alpha / |\Delta|$$

with u_α chosen to give overall rejection rate α

... but how?

Do another permutation

$$\left(\tilde{\mathbb{X}}_m, \tilde{\mathbb{Y}}_n\right) \sim p \times p$$

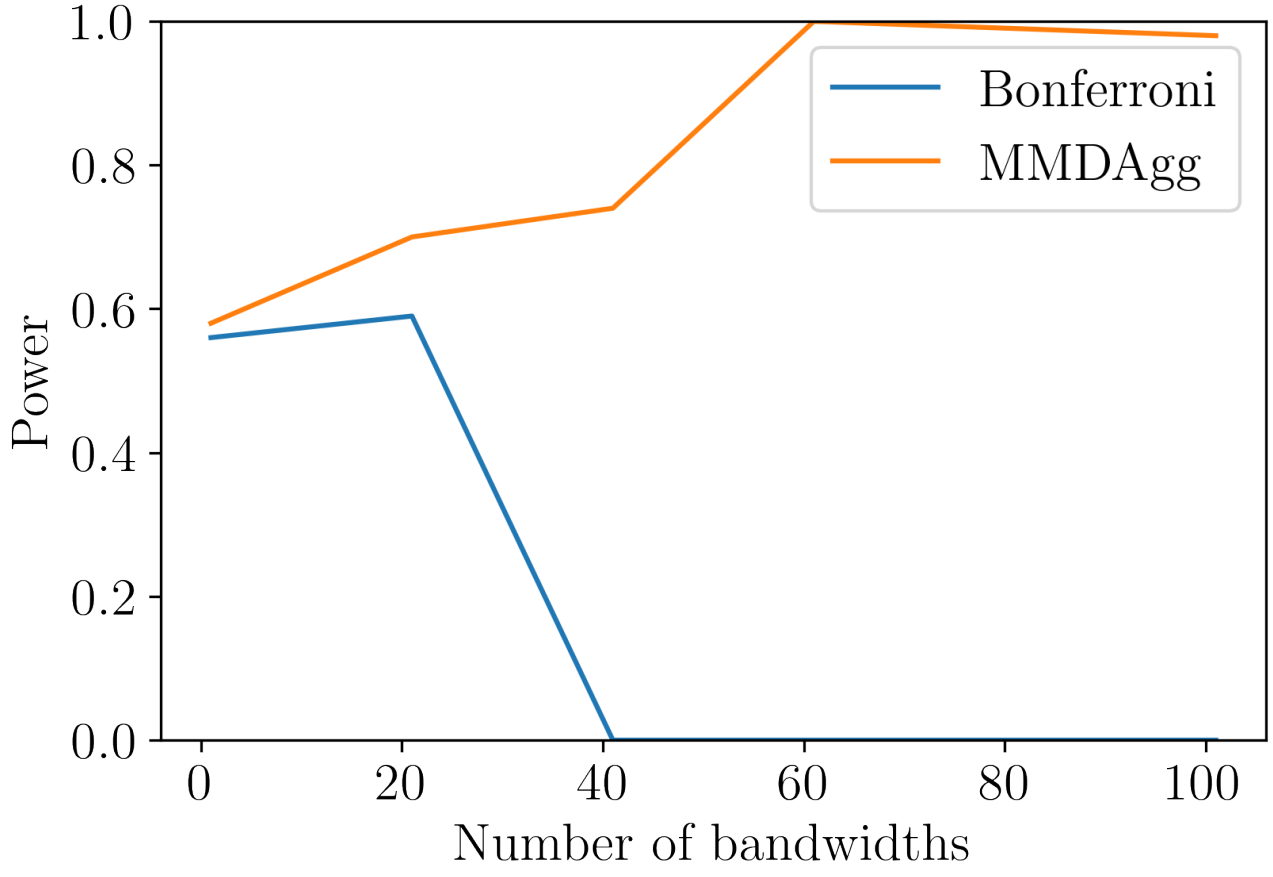
Using $\left(\tilde{\mathbb{X}}_m, \tilde{\mathbb{Y}}_n\right)$, search over u_α such that prob. of any $\lambda \in \Lambda$ rejecting is $\leq \alpha$.

Test is minimax optimal if $p - q$ smooth (in a Sobolev ball), and subject to reasonable[1] conditions on kernels.

1. Kernel is a product of one-dimensional translation invariant characteristic kernels which are absolutely and square integrable.

Multiple testing correction comparison

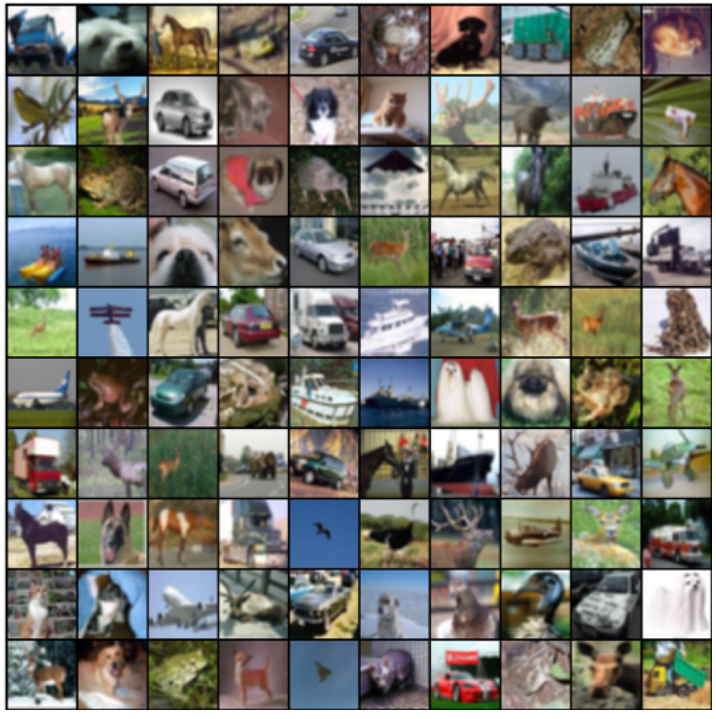
Simple example: 3-d Gaussians with different means



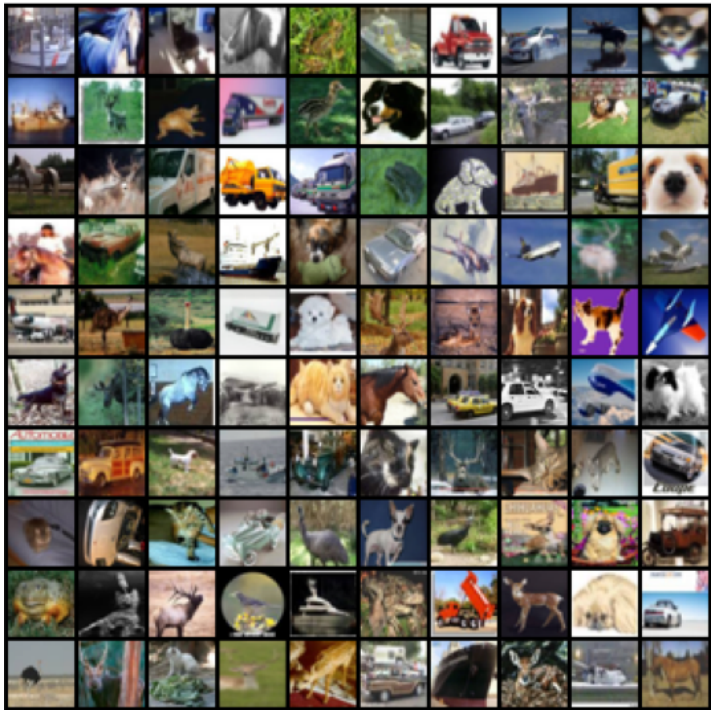
$$\Lambda(i) := \left\{ 2^{\ell} \lambda_{\text{med}} : \ell \in \{-i, \dots, i\} \right\} \text{ for } i \in \{0, 10, 20, 30, 40, 50\}$$

MMD for detecting drift

- Goal: do P and Q differ?



CIFAR 10 samples



Cifar 10.1 samples

Significant difference?

Krizhevsky, Hinton, Learning multiple layers of features from tiny images. (2009)
Recht, Roelofs, Schmidt, Shankar, Do CIFAR-10 Classifiers Generalize to CIFAR-10? (2018).

CIFAR-10 and CIFAR-10.1

Test level $\alpha = 0.05$. Average rejections over 1000 repetitions.

Tests	Power
MMD-FUSE	0.937
MMDAgg	0.883
MMD-D	0.744
CTT	0.711
MMD-Median	0.678
ACTT	0.652
ME	0.588
AutoML	0.544
C2ST-L	0.529
C2ST-S	0.452
MMD-O	0.316
MMDAggInc	0.281
SCF	0.171

Outline

Two sample testing

- Test statistic: Maximum Mean Discrepancy (MMD)
- Statistical testing with the MMD
- "How to choose the best kernel"
  - ...with fusing: https://github.com/antoninschrab/mmdfuse
  - ...with aggregation: https://github.com/antoninschrab/mmdagg-paper
- No sample splitting!

Research support

Work supported by:

The Gatsby Charitable Foundation

Google Deepmind

Google DeepMind

Questions?


THE KERNEL
BREWERY
LONDON
BALTIC PORTER
7.3% ABV
THE KERNEL
EXPORT STOUT

[Schrab et al., 2023, Corollary 10]

MMDAgg test power guarantee

Theorem (MMDAgg minimax adaptivity)

$$\Lambda^\star := \left\{ 2^{-\ell} \mathbb{1}_d : \ell \in \left\{ 1, \dots, \left\lceil \frac{2}{d} \log_2 \left(\frac{m+n}{\ln(\ln(m+n))} \right) \right\rceil \right\} \right\}, \quad w_\lambda := \frac{6}{\pi^2 \ell^2}$$

Assuming $p - q \in \mathcal{S}_d^s(R)$, *the condition*

$$\|p - q\|_2 \geq \frac{C}{\sqrt{\beta}} \left(\frac{m+n}{\ln(\ln(m+n))} \right)^{-2s/(4s+d)}$$

guarantees control of the type II error of MMDAgg

$$\mathbb{P}_{p \times q} \left(\Delta_\alpha^{\Lambda^\star}(\mathbb{X}_m, \mathbb{Y}_n) = 0 \right) \leq \beta.$$

Minimax rate over Sobolev balls: $(m+n)^{-2s/(4s+d)}$

Minimax adaptive over $\{\mathcal{S}_d^s(R) : s > 0, R > 0\}$

Unlike the MMD test $\Delta_\alpha^{\lambda^\star}$, MMDAgg $\Delta_\alpha^{\Lambda^\star}$ is independent of s