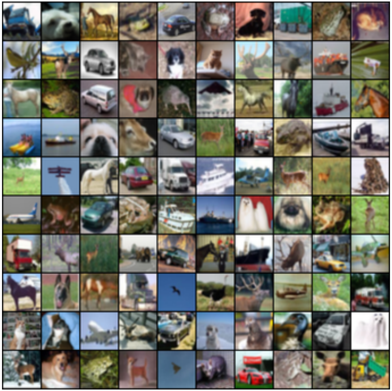
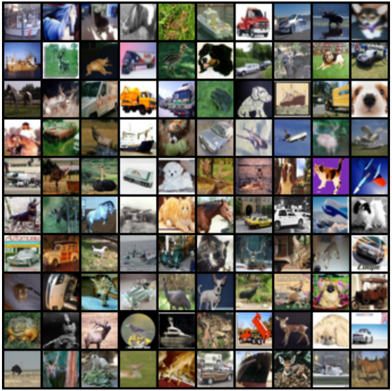
Two-Sample Testing

The problem



CIFAR-10 test set (Krizhevsky 2009)

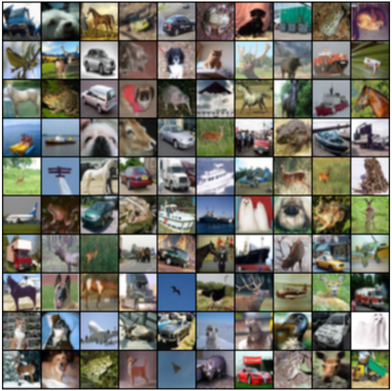
$X \sim P$



CIFAR-10.1 (Recht+ ICML 2019)

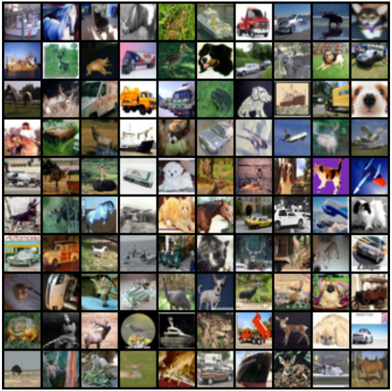
$Y \sim Q$

The problem



CIFAR-10 test set (Krizhevsky 2009)

$X \sim P$

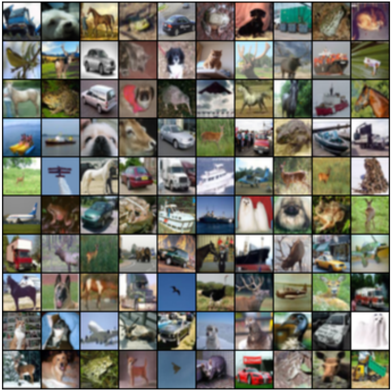


CIFAR-10.1 (Recht+ ICML 2019)

$Y \sim Q$

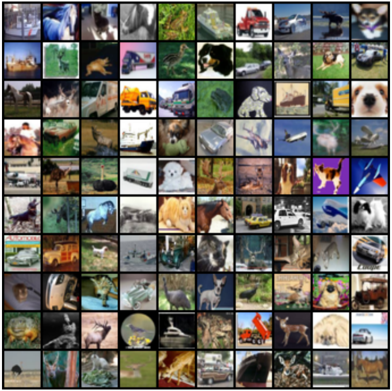
- Are the distributions P and Q the same?

The problem



CIFAR-10 test set (Krizhevsky 2009)

$X \sim P$



CIFAR-10.1 (Recht+ ICML 2019)

$Y \sim Q$

- Are the distributions P and Q the same?
- Remember that $MMD(P, Q) = 0$ iff $P = Q$

Estimating the MMD

$$MMD(P, Q)^2 = \|\mu_P - \mu_Q\|^2$$

Estimating the MMD

$$
\begin{aligned}
MMD(P, Q)^2 &= \|\mu_P - \mu_Q\|^2 \\
&= \langle \mu_P, \mu_P \rangle - 2\langle \mu_P, \mu_Q \rangle + \langle \mu_Q, \mu_Q \rangle
\end{aligned}
$$

Estimating the MMD

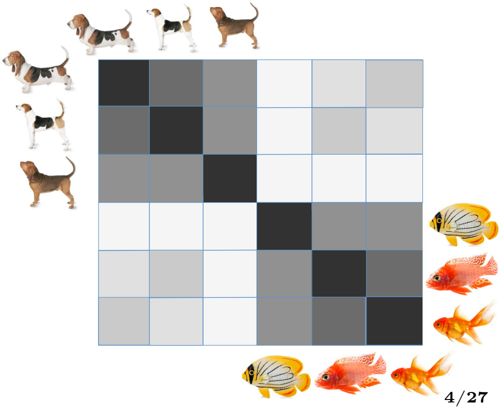
$$\begin{aligned} MMD(P, Q)^2 &= \|\mu_P - \mu_Q\|^2 \\ &= \langle \mu_P, \mu_P \rangle - 2\langle \mu_P, \mu_Q \rangle + \langle \mu_Q, \mu_Q \rangle \\ &= \mathbf{E}\left[\langle \varphi(X), \varphi(X') \rangle - 2\langle \varphi(X), \varphi(Y) \rangle + \langle \varphi(Y), \varphi(Y') \rangle \right] \end{aligned}$$

Estimating the MMD

$$
\begin{aligned}
MMD(P, Q)^2 &= \|\mu_P - \mu_Q\|^2 \\
&= \langle \mu_P, \mu_P \rangle - 2\langle \mu_P, \mu_Q \rangle + \langle \mu_Q, \mu_Q \rangle \\
&= \mathbf{E}\left[\langle \varphi(X), \varphi(X') \rangle - 2\langle \varphi(X), \varphi(Y) \rangle + \langle \varphi(Y), \varphi(Y') \rangle\right] \\
&= \mathbf{E}\left[k(X, X') - 2k(X, Y) + k(Y, Y')\right]
\end{aligned}
$$

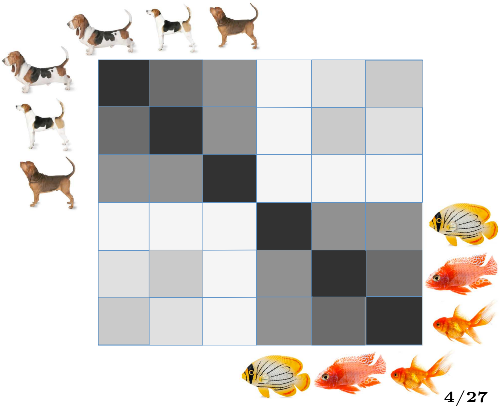
Estimating the MMD

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



Estimating the MMD

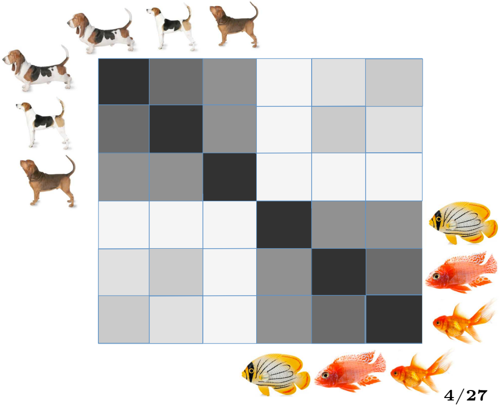
- Dogs $(= P)$ and fish $(= Q)$ example
- Each entry is one of $k(\text{dog}_i, \text{dog}_j)$, $k(\text{dog}_i, \text{fish}_j)$, or $k(\text{fish}_i, \text{fish}_j)$
- $MMD(P, Q)^2 = \mathbf{E}\left[k(\text{dog}_i, \text{dog}_j) + k(\text{fish}_i, \text{fish}_j) - 2k(\text{dog}_i, \text{fish}_j)\right]$



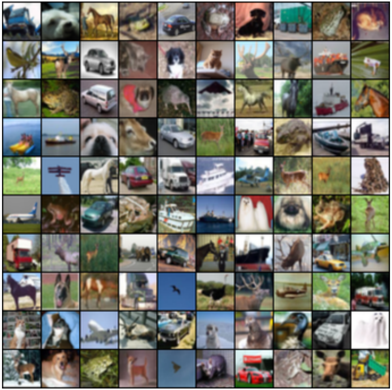
Estimating the MMD

- Dogs $(= P)$ and fish $(= Q)$ example
- Each entry is one of $k(\text{dog}_i, \text{dog}_j)$, $k(\text{dog}_i, \text{fish}_j)$, or $k(\text{fish}_i, \text{fish}_j)$
- $MMD(P, Q)^2 = \mathbf{E}\left[k(\text{dog}_i, \text{dog}_j) + k(\text{fish}_i, \text{fish}_j) - 2k(\text{dog}_i, \text{fish}_j)\right]$

$$
\begin{aligned}
\widehat{MMD}^2 = \\
&\frac{1}{n(n-1)} \sum_{i \neq j} k(\text{dog}_i, \text{dog}_j) \\
&+ \frac{1}{n(n-1)} \sum_{i \neq j} k(\text{fish}_i, \text{fish}_j) \\
&- \frac{2}{n^2} \sum_{i,j} k(\text{dog}_i, \text{fish}_j)
\end{aligned}
$$

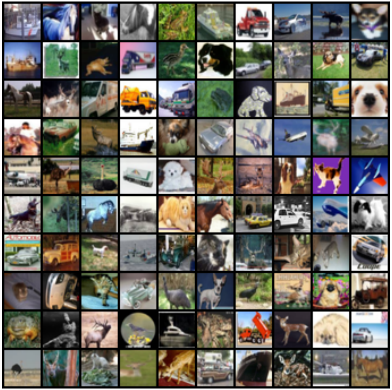


Using a divergence estimator



CIFAR-10 test set (Krizhevsky 2009)

$X \sim P$

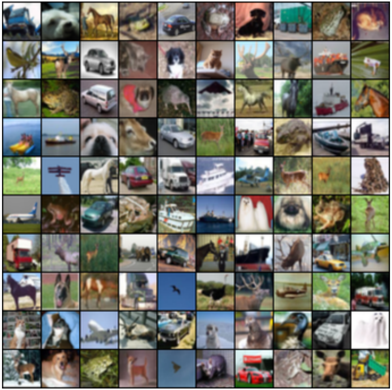


CIFAR-10.1 (Recht+ ICML 2019)

$Y \sim Q$

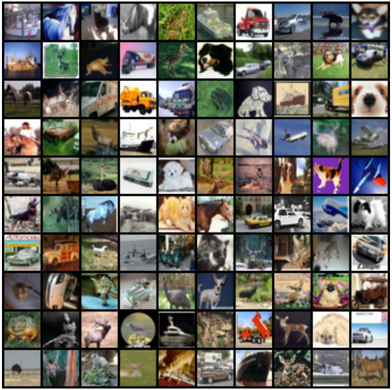
- Say we get $\widehat{MMD}^2 = 0.09116$

Using a divergence estimator



CIFAR-10 test set (Krizhevsky 2009)

$X \sim P$

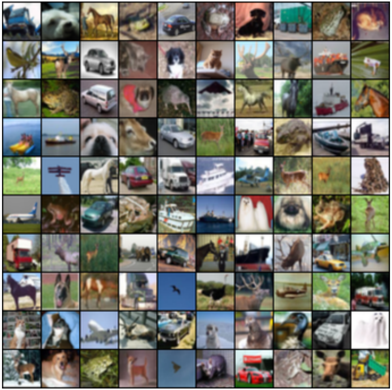


CIFAR-10.1 (Recht+ ICML 2019)

$Y \sim Q$

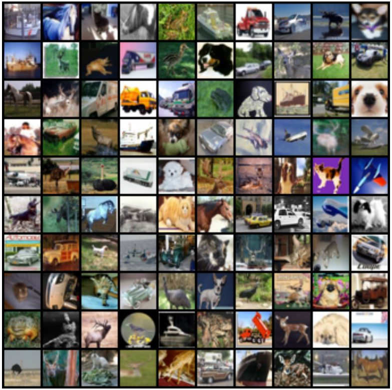
- Say we get $\widehat{MMD}^2 = 0.09116$

Using a divergence estimator



CIFAR-10 test set (Krizhevsky 2009)

$X \sim P$

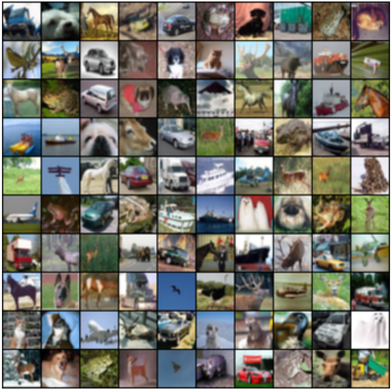


CIFAR-10.1 (Recht+ ICML 2019)

$Y \sim Q$

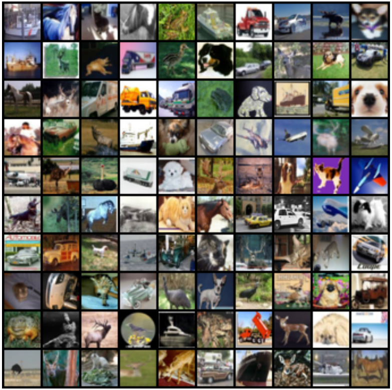
- Say we get $\widehat{MMD}^2 = 0.09116$
- ...great. Is the true MMD zero? Equivalently: is $P = Q$?

Using a divergence estimator



CIFAR-10 test set (Krizhevsky 2009)

$X \sim P$



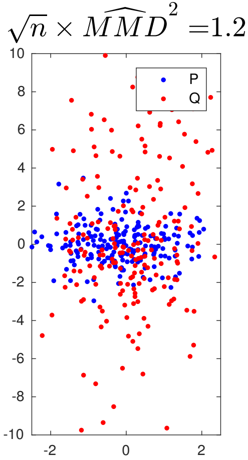
CIFAR-10.1 (Recht+ ICML 2019)

$Y \sim Q$

- Say we get $\widehat{MMD}^2 = 0.09116$
- ...great. Is the true MMD zero? Equivalently: is $P = Q$?
- We need to know "how random" $\widehat{MMD}^2$ is...

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

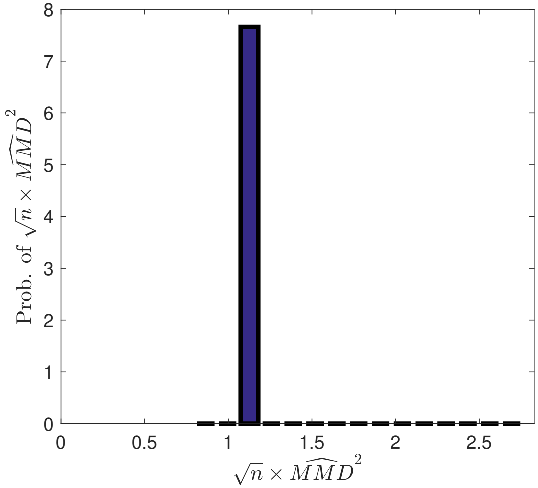
- P, Q Laplace with different variances in y
- Draw $n = 200$ i.i.d samples from P and Q



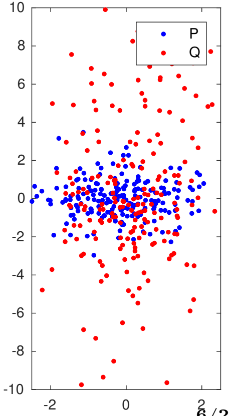
Behavior of $\widehat{MMD}^2$ when $P \neq Q$

- P, Q Laplace with different variances in y
- Draw $n = 200$ i.i.d samples from P and Q

Number of MMDs: 1

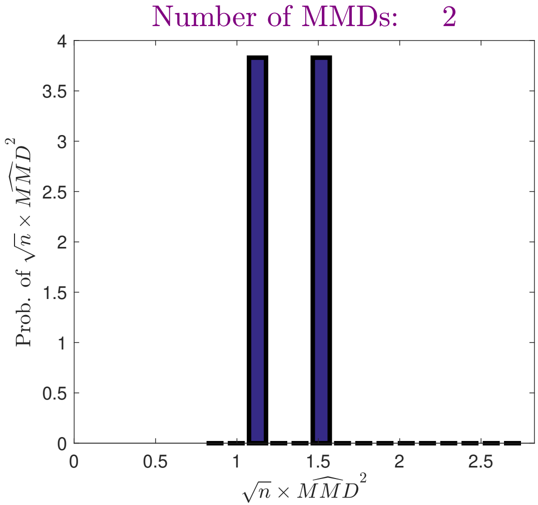


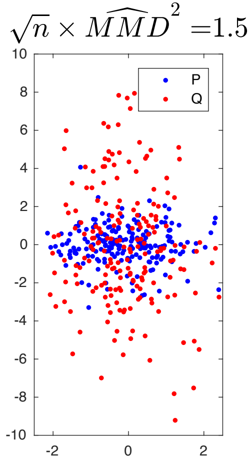
$\sqrt{n} \times \widehat{MMD}^2 = 1.2$



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

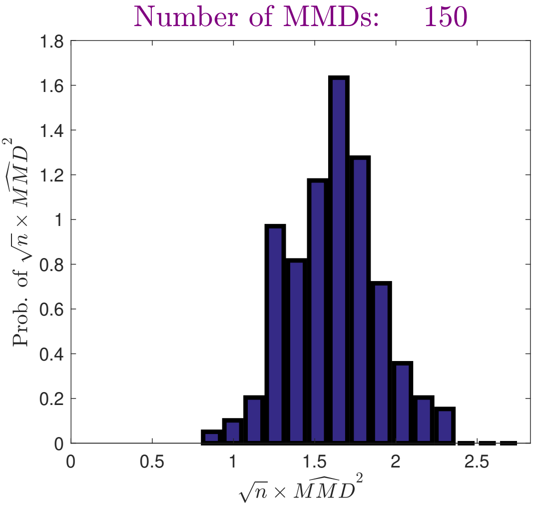
- P, Q Laplace with different variances in y
- Draw $n = 200$ new i.i.d samples from P and Q





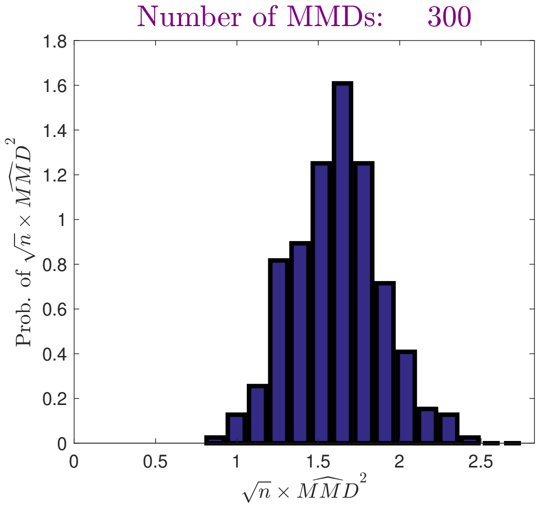
Behavior of $\widehat{MMD}^2$ when $P \neq Q$

- P, Q Laplace with different variances in y
- Draw $n = 200$ i.i.d samples from P and Q, 150 times



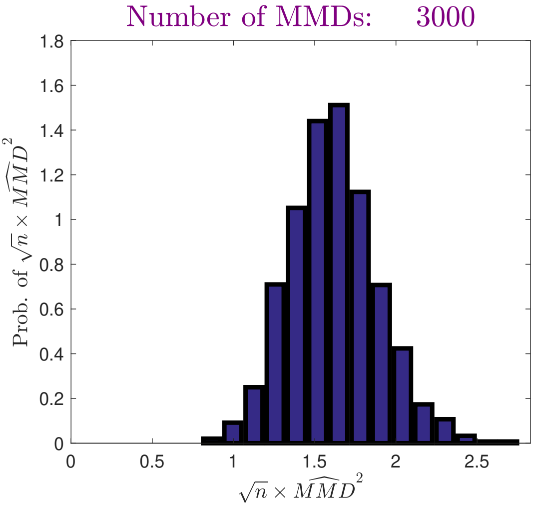
Behavior of $\widehat{MMD}^2$ when $P \neq Q$

- P, Q Laplace with different variances in y
- Draw $n = 200$ i.i.d samples from P and Q, 300 times



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

- P, Q Laplace with different variances in y
- Draw $n = 200$ i.i.d samples from P and Q, 3000 times

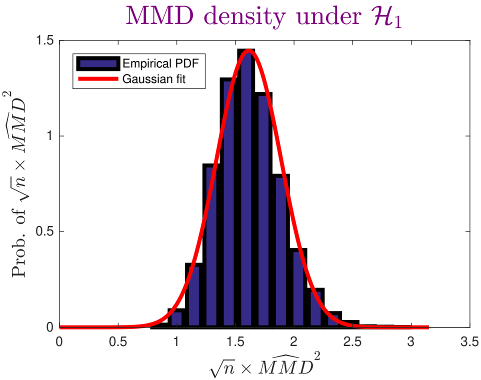


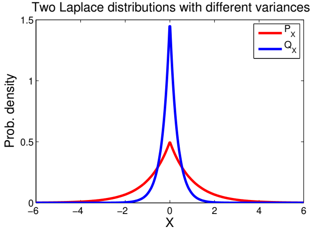
Asymptotics of $\widehat{MMD}^2$ when $P \neq Q$

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

$$\sqrt{n}\,\frac{\widehat{\mathrm{MMD}}^2 - \mathrm{MMD}(P, Q)}{\sigma_{H_1}} \xrightarrow{D} \mathcal{N}(0, 1),$$

where $\sigma_{H_1}^2/n$ is asymptotic variance (depends on P, Q, k).



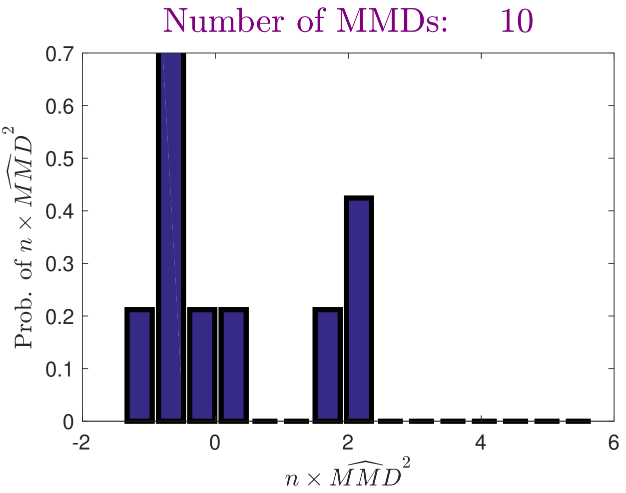


Behavior of $\widehat{MMD}^2$ when $P = Q$

What about when P and Q are the same?

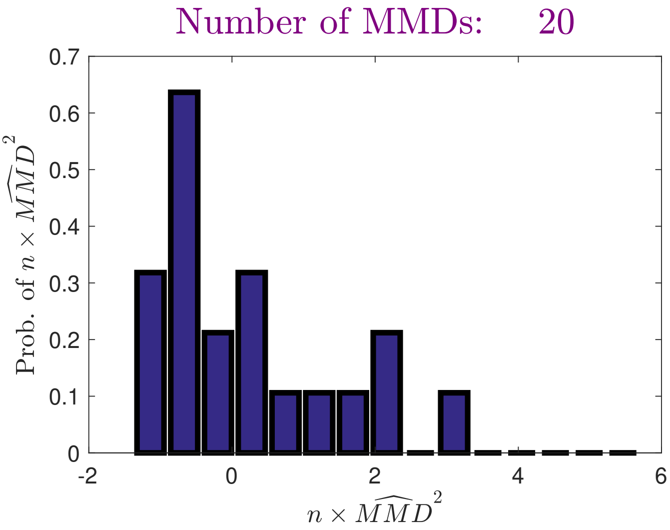
Behavior of $\widehat{MMD}^2$ when $P = Q$

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



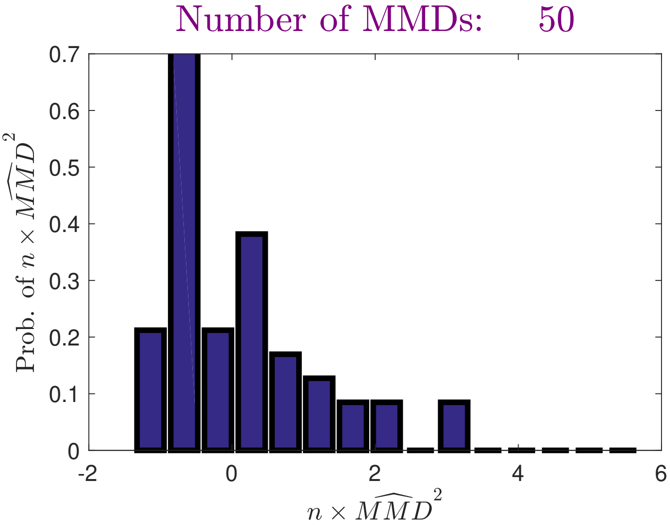
Behavior of $\widehat{MMD}^2$ when $P = Q$

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



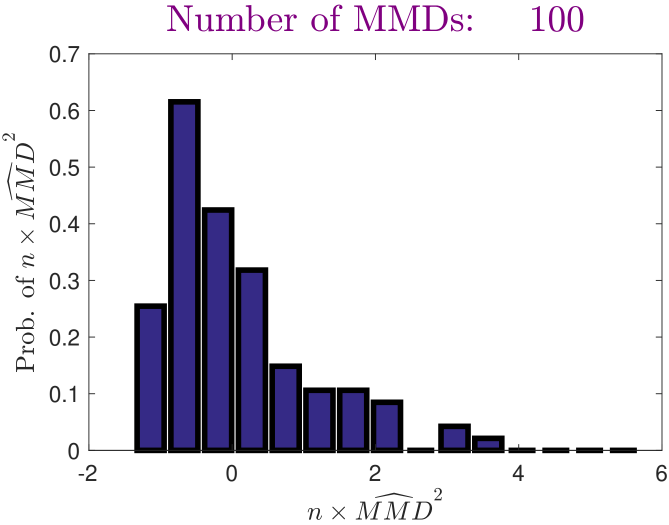
Behavior of $\widehat{MMD}^2$ when $P = Q$

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



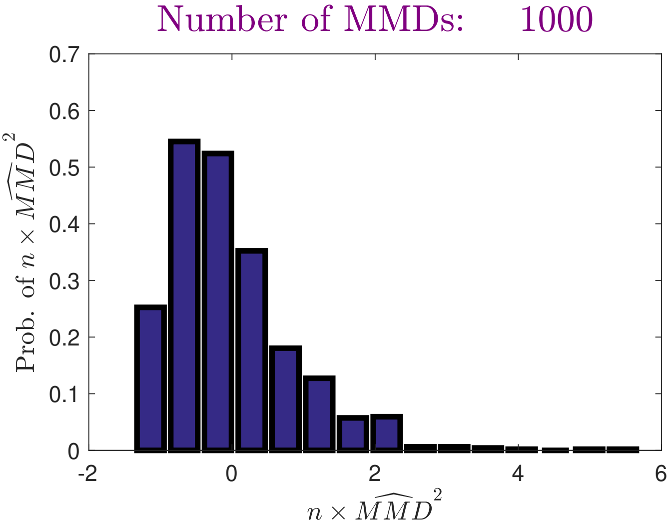
Behavior of $\widehat{MMD}^2$ when $P = Q$

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



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

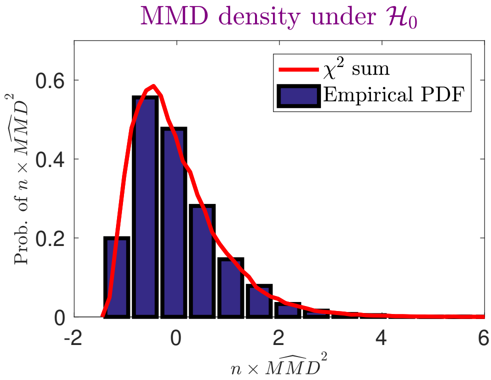
- Case of $P = Q = \mathcal{N}(0, 1)$



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

Where $P = Q$, statistic has asymptotic distribution

$$n\widehat{\text{MMD}}^2 \sim \sum_{l=1}^{\infty} \lambda_l \left[z_l^2 - 2 \right]$$



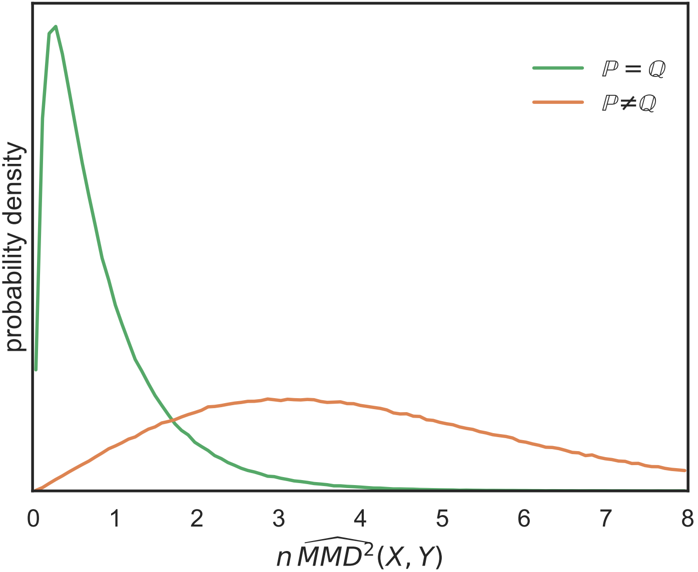
where

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

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

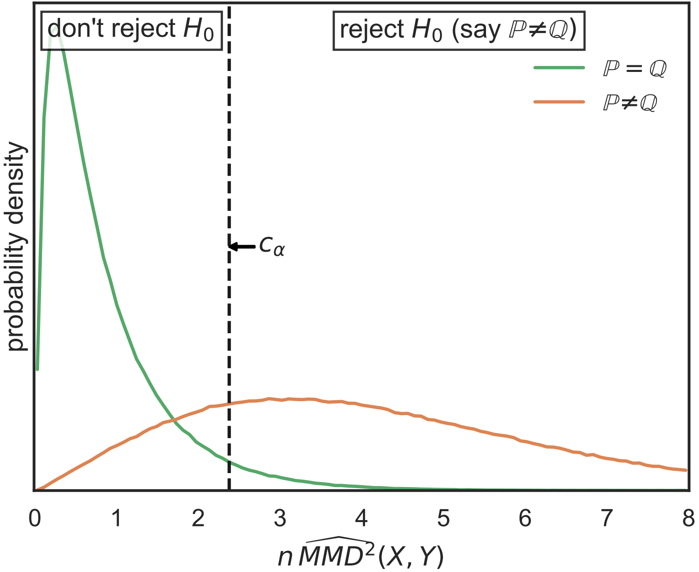
Statistical testing

A summary of the asymptotics:



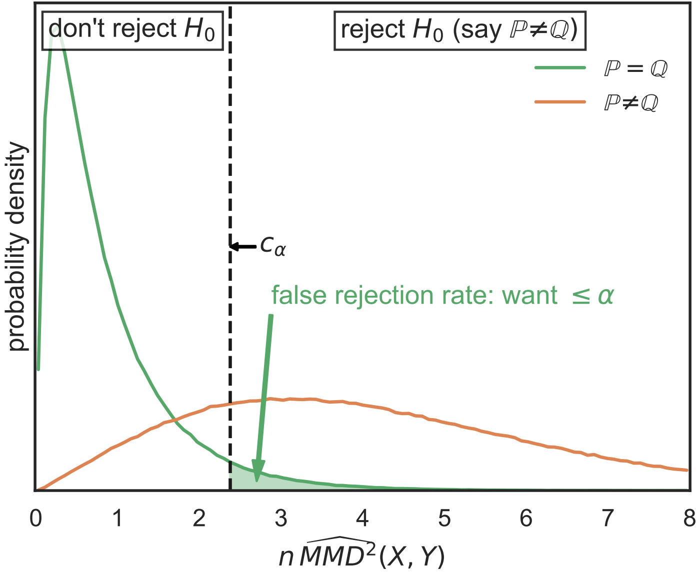
Statistical testing

Test construction: (Gretton+, JMLR 2012)



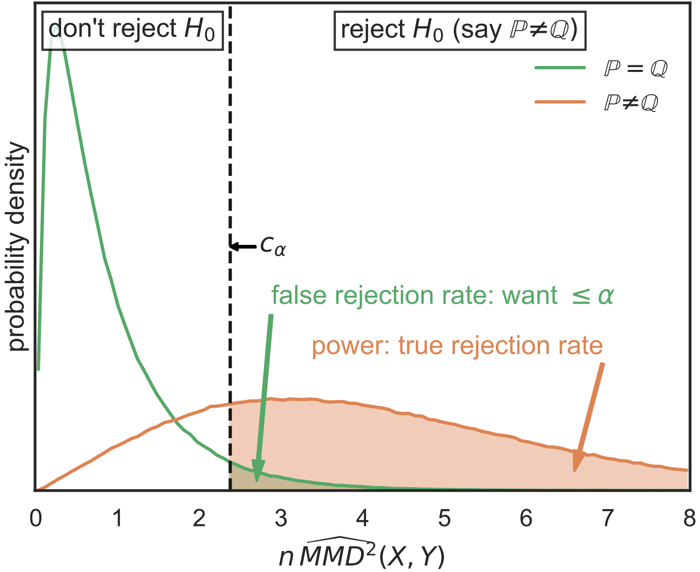
Statistical testing

Test construction: (Gretton+, JMLR 2012)



Statistical testing

Test construction: (Gretton+, JMLR 2012)



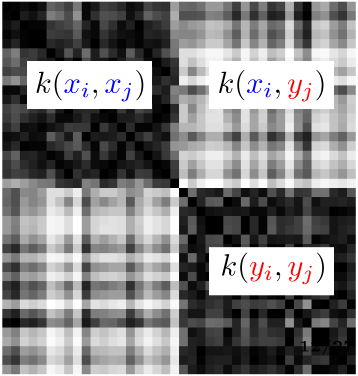
How do we get the test threshold c_α?

Original empirical MMD for dogs and fish:

$X = [$  $\dots]$

$Y = [$  $\dots]$

$$\widehat{MMD}^2 = \frac{1}{n(n-1)} \sum_{i \neq j} k(x_i, x_j) + \frac{1}{n(n-1)} \sum_{i \neq j} k(y_i, y_j) - \frac{2}{n^2} \sum_{i,j} k(x_i, y_j)$$

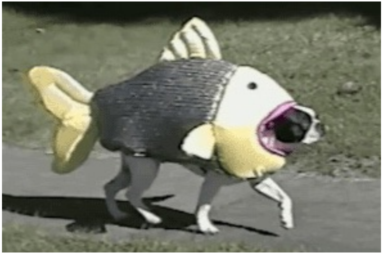


How do we get the test threshold c_α?

Permuted dog and fish samples (**merdogs**):

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

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



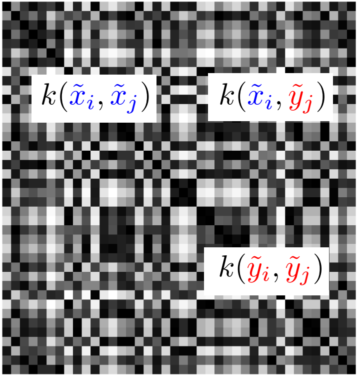
How do we get the test threshold c_α?

Permuted dog and fish samples (**merdogs**):

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

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

$$\widehat{MMD}^2 = \frac{1}{n(n-1)} \sum_{i \neq j} k(\tilde{\mathbf{x}}_i, \tilde{\mathbf{x}}_j) + \frac{1}{n(n-1)} \sum_{i \neq j} k(\tilde{\mathbf{y}}_i, \tilde{\mathbf{y}}_j) - \frac{2}{n^2} \sum_{i,j} k(\tilde{\mathbf{x}}_i, \tilde{\mathbf{y}}_j)$$



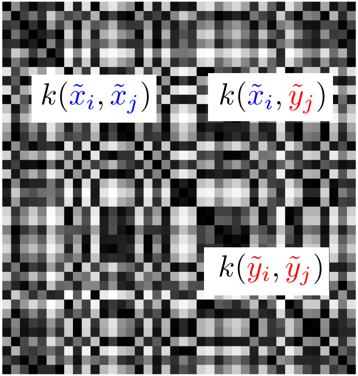
How do we get the test threshold c_α?

Permuted dog and fish samples (**merdogs**):

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

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

$$\widehat{MMD}^2 = \frac{1}{n(n-1)} \sum_{i \neq j} k(\tilde{x}_i, \tilde{x}_j) + \frac{1}{n(n-1)} \sum_{i \neq j} k(\tilde{y}_i, \tilde{y}_j) - \frac{2}{n^2} \sum_{i,j} k(\tilde{x}_i, \tilde{y}_j)$$



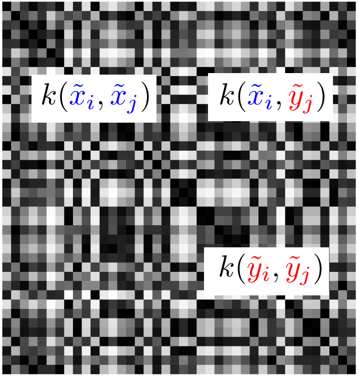
How do we get the test threshold c_α?

Permuted dog and fish samples (**merdogs**):

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

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

$$\widehat{MMD}^2 = \frac{1}{n(n-1)} \sum_{i \neq j} k(\tilde{x}_i, \tilde{x}_j) + \frac{1}{n(n-1)} \sum_{i \neq j} k(\tilde{y}_i, \tilde{y}_j) - \frac{2}{n^2} \sum_{i,j} k(\tilde{x}_i, \tilde{y}_j)$$



- This simulates $P = Q$

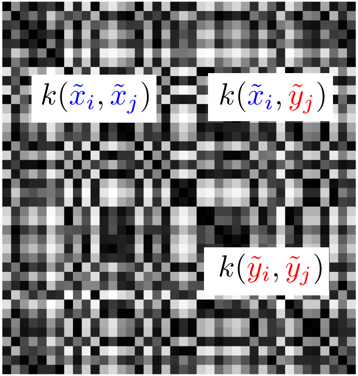
How do we get the test threshold c_α?

Permuted dog and fish samples (**merdogs**):

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

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

$$\widehat{MMD}^2 = \frac{1}{n(n-1)} \sum_{i \neq j} k(\tilde{x}_i, \tilde{x}_j) + \frac{1}{n(n-1)} \sum_{i \neq j} k(\tilde{y}_i, \tilde{y}_j) - \frac{2}{n^2} \sum_{i,j} k(\tilde{x}_i, \tilde{y}_j)$$



- This simulates $P = Q$
- Repeat, set c_α to quantile

Choosing a kernel for the test

- Simple choice: exponentiated quadratic

$$k(x, y) = \exp\left(-\frac{1}{2\sigma^2}\|x - y\|^2\right)$$

- *Characteristic* for any σ: for any P and Q, power $\to 1$ as $n \to \infty$

Choosing a kernel for the test

- Simple choice: exponentiated quadratic

$$k(x, y) = \exp\left(-\frac{1}{2\sigma^2}\|x - y\|^2\right)$$

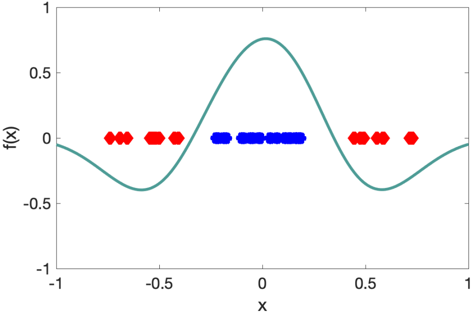
- *Characteristic* for any σ: for any P and Q, power $\to 1$ as $n \to \infty$
- But choice of σ is very important for finite n...

Choosing a kernel for the test

- Simple choice: exponentiated quadratic

$$k(x, y) = \exp\left(-\frac{1}{2\sigma^2}\|x - y\|^2\right)$$

- *Characteristic* for any σ: for any P and Q, power $\to 1$ as $n \to \infty$
- But choice of σ is very important for finite n...

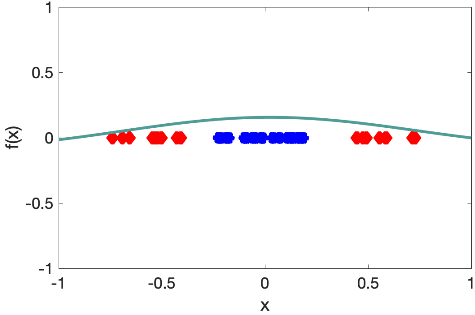


Choosing a kernel for the test

- Simple choice: exponentiated quadratic

$$k(x, y) = \exp\left(-\frac{1}{2\sigma^2}\|x - y\|^2\right)$$

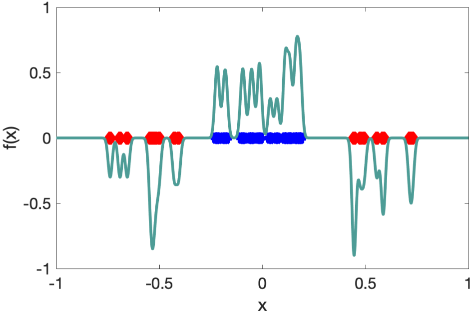
- *Characteristic* for any σ: for any P and Q, power $\to 1$ as $n \to \infty$
- But choice of σ is very important for finite n...



Choosing a kernel for the test

- Simple choice: exponentiated quadratic
$$k(x, y) = \exp\left(-\frac{1}{2\sigma^2}\|x - y\|^2\right)$$

- *Characteristic* for any σ: for any P and Q, power $\to 1$ as $n \to \infty$
- But choice of σ is very important for finite n...



Choosing a kernel for the test

- Simple choice: exponentiated quadratic

$$k(x, y) = \exp\left(-\frac{1}{2\sigma^2}\|x - y\|^2\right)$$

- *Characteristic* for any σ: for any P and Q, power $\to 1$ as $n \to \infty$
- But choice of σ is very important for finite n...
- ...and some problems (e.g. images) might have no good choice for σ

Choosing a kernel for the test

- Often helpful to use a relevant representation $\Phi : \mathcal{X} \to \mathbb{R}^d$, eg:

$$k(x, y) = k_{\text{top}}(\Phi(x), \Phi(y))$$

.

Choosing a kernel for the test

- Often helpful to use a relevant representation $\Phi : \mathcal{X} \to \mathbb{R}^d$, eg:

$$k(x, y) = k_{\text{top}}(\Phi(x), \Phi(y))$$

  - Take Φ as predictions of a pretrained classifier on a related domain

.

Choosing a kernel for the test

- Often helpful to use a relevant representation $\Phi : \mathcal{X} \to \mathbb{R}^d$, eg:

$$k(x, y) = k_{\text{top}}(\Phi(x), \Phi(y))$$

  - Take Φ as predictions of a pretrained classifier on a related domain
    - Related to Adversarial Accuracy (Yang+ ICLR 2017) and Inception Score (Salimans+ NeurIPS 2016).

Choosing a kernel for the test

- Often helpful to use a relevant representation $\Phi : \mathcal{X} \to \mathbb{R}^d$, eg:

$$k(x, y) = k_{\text{top}}(\Phi(x), \Phi(y))$$

  - Take Φ as predictions of a pretrained classifier on a related domain
    - Related to Adversarial Accuracy (Yang+ ICLR 2017) and Inception Score (Salimans+ NeurIPS 2016). We'll come back to this!

Choosing a kernel for the test

- Often helpful to use a relevant representation $\Phi : \mathcal{X} \to \mathbb{R}^d$, eg:

$$k(x, y) = k_{\text{top}}(\Phi(x), \Phi(y))$$

  - Take Φ as predictions of a pretrained classifier on a related domain
    - Related to Adversarial Accuracy (Yang+ ICLR 2017) and Inception Score (Salimans+ NeurIPS 2016). We'll come back to this!
  - Take Φ as late hidden layer from pretrained related classifier
    - KID (Bińkowski, Sutherland+ ICLR 2018), Xu+ (arXiv:1806.07755)

Choosing a kernel for the test

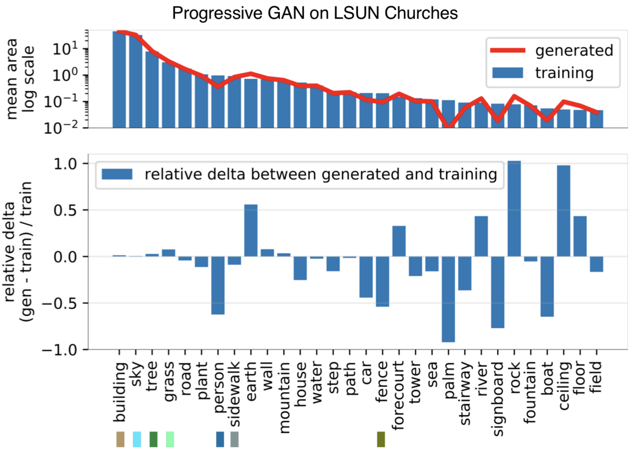
- Often helpful to use a relevant representation $\Phi : \mathcal{X} \to \mathbb{R}^d$, eg:

$$k(x, y) = k_{\mathrm{top}}(\Phi(x), \Phi(y))$$

- Take Φ as predictions of a pretrained classifier on a related domain
  - Related to Adversarial Accuracy (Yang+ ICLR 2017) and Inception Score (Salimans+ NeurIPS 2016). We'll come back to this!
- Take Φ as late hidden layer from pretrained related classifier
  - KID (Bińkowski, Sutherland+ ICLR 2018), Xu+ (arXiv:1806.07755)
  - Closely related to FID (Heusel+ NeurIPS 2017) but
    much nicer statistical properties,
    more correlated with human judgement (Zhou, Gordon+ NeurIPS 2019)

Choosing a kernel for the test

- Bau et al. (ICCV 2019) compare counts of pixel categories



(a) generated vs training object segmentation statistics

What about tests for other distances?

- Sometimes, nice closed forms for threshold (like a t test)
- Asymptotic behavior of KALE, Wasserstein, … mostly unknown
- But permutation tests usually work!

Choosing the best test

The best test for the job

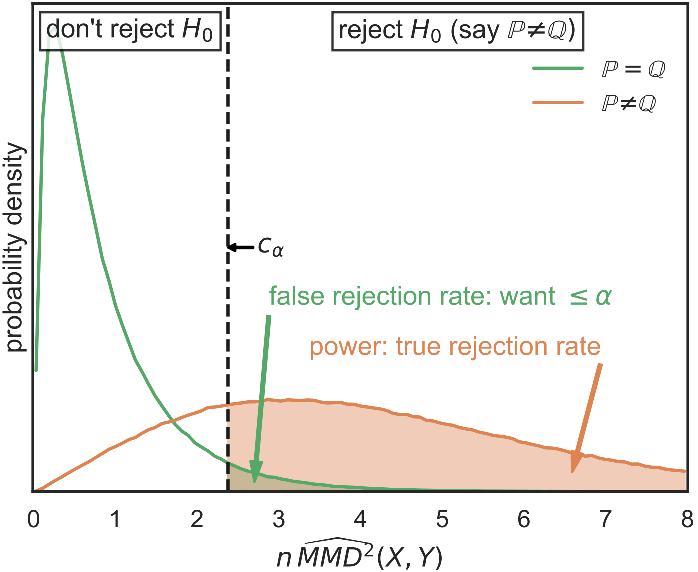
- A test's power depends on P and Q (and n)
- Many MMDs have power $\to 1$ as $n \to \infty$ for any (fixed) problem
  - But, for many P and Q, will have terrible power with reasonable n!

The best test for the job

- A test's power depends on P and Q (and n)
- Many MMDs have power $\to 1$ as $n \to \infty$ for any (fixed) problem
  - But, for many P and Q, will have terrible power with reasonable n!
- Can maybe pick a good kernel manually for a given problem
- Can't get one that has good finite-sample power for all problems
  - No one test can have all that power

Choosing test power

- Best test (of level α) is the one with highest test power



Optimizing MMD for test power

The power of our test ($\Pr_1$ denotes probability under $P \neq Q$):

$$\Pr_1\left(n\widehat{MMD}^2 > \hat{c}_\alpha\right)$$

- $\hat{c}_\alpha$ is an estimate of the test threshold c_α

Optimizing MMD for test power

The power of our test ($\Pr_1$ denotes probability under $P \neq Q$):

$$\Pr_1\left(n\widehat{MMD}^2 > \hat{c}_\alpha\right)$$
$$= \Pr_1\left(\sqrt{n}\frac{\widehat{MMD}^2 - MMD^2}{\sigma_{H_1}} > \frac{\hat{c}_\alpha}{\sqrt{n}\sigma_{H_1}} - \frac{\sqrt{n}MMD^2}{\sigma_{H_1}}\right)$$

- $\hat{c}_\alpha$ is an estimate of the test threshold c_α

Optimizing MMD for test power

The power of our test ($\Pr_1$ denotes probability under $P \neq Q$):

$$
\begin{aligned}
&\Pr_1\left(n\widehat{MMD}^2 > \hat{c}_\alpha\right)\\
&\quad = \Pr_1\left(\sqrt{n}\frac{\widehat{MMD}^2 - MMD^2}{\sigma_{H_1}} > \frac{\hat{c}_\alpha}{\sqrt{n}\sigma_{H_1}} - \frac{\sqrt{n}MMD^2}{\sigma_{H_1}}\right)\\
&\quad \to \Phi\left(\sqrt{n}\frac{MMD^2}{\sigma_{H_1}} - \frac{c_\alpha}{\sqrt{n}\sigma_{H_1}}\right)
\end{aligned}
$$

- $\hat{c}_\alpha$ is an estimate of the test threshold c_α
- Φ is the CDF of the standard normal distribution

Optimizing MMD for test power

The power of our test ($\Pr_1$ denotes probability under $P \neq Q$):

$$\begin{aligned}
&\Pr_1\left(n\widehat{MMD}^2 > \hat{c}_\alpha\right) \\
&\quad = \Pr_1\left(\sqrt{n}\frac{\widehat{MMD}^2 - MMD^2}{\sigma_{H_1}} > \frac{\hat{c}_\alpha}{\sqrt{n}\sigma_{H_1}} - \frac{\sqrt{n}MMD^2}{\sigma_{H_1}}\right) \\
&\quad \to \Phi\left(\sqrt{n}\frac{MMD^2}{\sigma_{H_1}} - \frac{c_\alpha}{\sqrt{n}\sigma_{H_1}}\right)
\end{aligned}$$

- For large n, second term is negligible!

Optimizing MMD for test power

The power of our test ($\Pr_1$ denotes probability under $P \neq Q$):

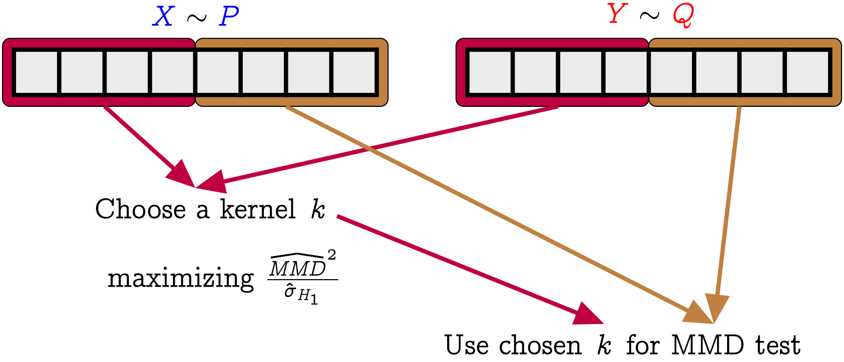
$$\Pr_1\left(n\widehat{MMD}^2 > \hat{c}_\alpha\right)$$
$$= \Pr_1\left(\sqrt{n}\frac{\widehat{MMD}^2 - MMD^2}{\sigma_{H_1}} > \frac{\hat{c}_\alpha}{\sqrt{n}\sigma_{H_1}} - \frac{\sqrt{n}MMD^2}{\sigma_{H_1}}\right)$$
$$\to \Phi\left(\sqrt{n}\frac{MMD^2}{\sigma_{H_1}} - \frac{c_\alpha}{\sqrt{n}\sigma_{H_1}}\right)$$

- To maximize test power, choose k to maximize (Sutherland+ ICLR 2017)

$$\frac{MMD^2(P, Q)}{\sigma_{H_1}(P, Q)}$$

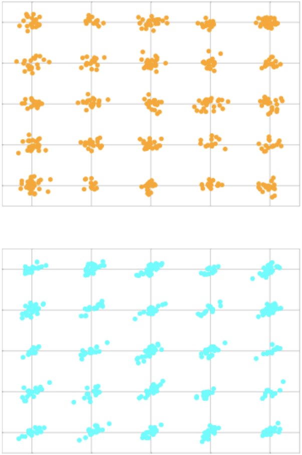
  - Estimator is differentiable in kernel parameters!

Data splitting

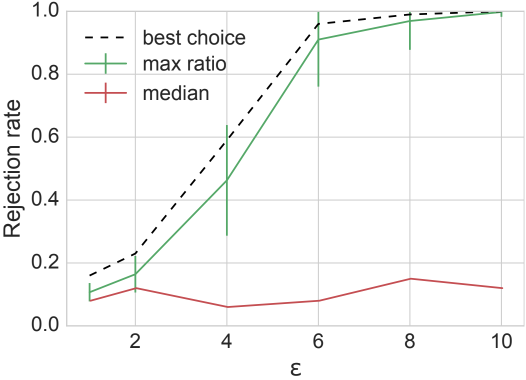


Learning a kernel helps a lot

- Even just learning a bandwidth. . . (Sutherland+ ICLR 2017)

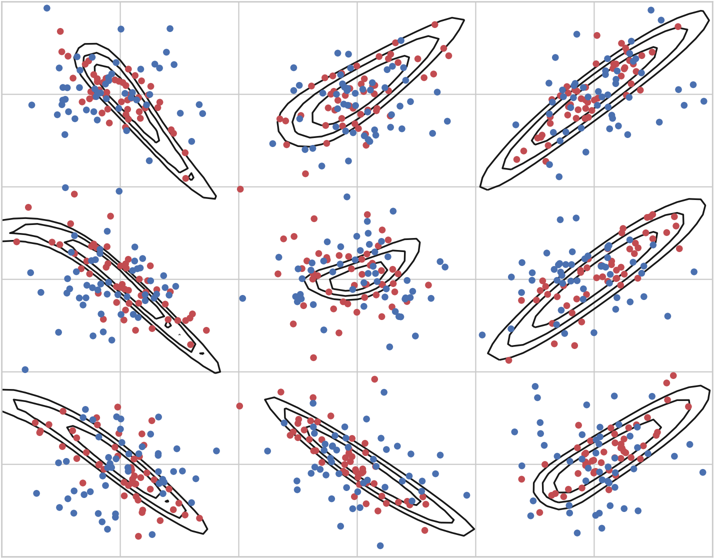


$\varepsilon = 6$



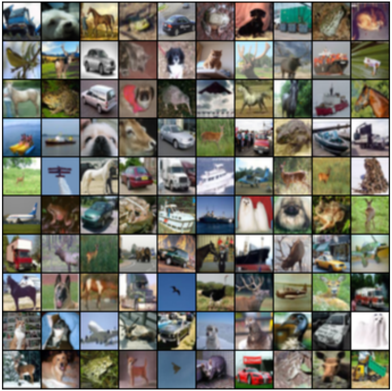
Learning a kernel helps a lot

- Even just learning a bandwidth. . . (Sutherland+ ICLR 2017)
- . . . but you can learn a lot more: $k_\theta(x, y) = k_{\text{top}}(\Phi_\theta(x), \Phi_\theta(y))$



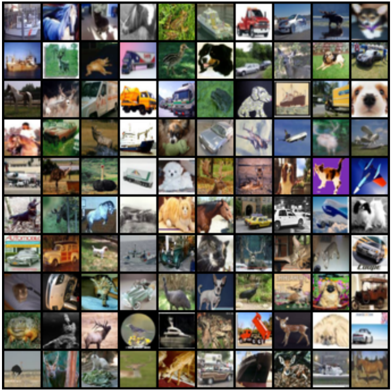
Learning a kernel helps a lot

- Even just learning a bandwidth... (Sutherland+ ICLR 2017)
- ...but you can learn a lot more: $k_\theta(x, y) = k_{\text{top}}(\Phi_\theta(x), \Phi_\theta(y))$
  - Learning a deep kernel for CIFAR-10 vs CIFAR-10.1 rejects the null



CIFAR-10 test set (Krizhevsky 2009)

$X \sim P$

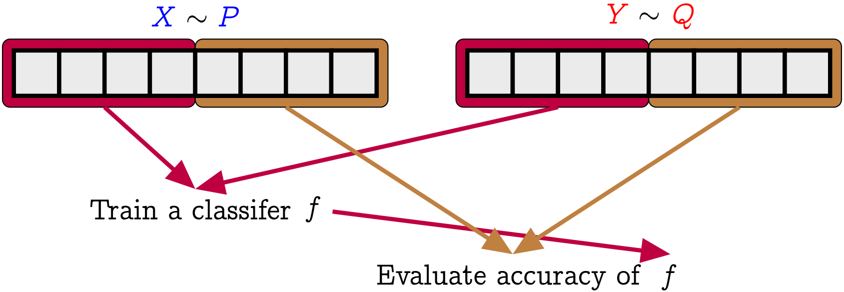


CIFAR-10.1 (Recht+ ICML 2019)

$Y \sim Q$

Alternative approach: Classifier two-sample tests

- Train a classifier $f : \mathcal{X} \to \{1, -1\}$ on P from Q
- Test statistic: accuracy on test set (Lopez-Paz and Oquab, ICLR 2017)



Alternative approach: Classifier two-sample tests

- Train a classifier $f : \mathcal{X} \to \{1, -1\}$ on P from Q
- Test statistic: accuracy on test set (Lopez-Paz and Oquab, ICLR 2017)
- Almost exactly equivalent:

$$k_f(x, y) = \frac{1}{4}\, \mathbb{1}(f(x) > 0)\; \mathbb{1}(f(y) > 0)$$

gives

$$MMD(P, Q) = \left| \text{accuracy} - \frac{1}{2} \right|$$

Alternative approach: Classifier two-sample tests

- Train a classifier $f : \mathcal{X} \to \{1, -1\}$ on P from Q
- Test statistic: accuracy on test set (Lopez-Paz and Oquab, ICLR 2017)
- Almost exactly equivalent:

$$k_f(x, y) = \frac{1}{4}\, \mathbb{1}(f(x) > 0)\, \mathbb{1}(f(y) > 0)$$

gives

$$MMD(P, Q) = \left| \text{accuracy} - \frac{1}{2} \right|$$

  - σ_{H_1} decreases with acc: maximizing $\frac{MMD^2}{\sigma_{H_1}}$ exactly maximizes power

Alternative approach: Classifier two-sample tests

- Train a classifier $f : \mathcal{X} \to \{1, -1\}$ on P from Q
- Test statistic: accuracy on test set (Lopez-Paz and Oquab, ICLR 2017)
- Almost exactly equivalent:

$$k_f(x, y) = \frac{1}{4}\,\mathbb{1}(f(x) > 0)\,\mathbb{1}(f(y) > 0)$$

gives

$$MMD(P, Q) = \left|\text{accuracy} - \frac{1}{2}\right|$$

- 0-1 kernel inflates variance, decreases test power

Alternative approach: Classifier two-sample tests

- Train a classifier $f : \mathcal{X} \to \{1, -1\}$ on P from Q
- Test statistic: accuracy on test set (Lopez-Paz and Oquab, ICLR 2017)
- Almost exactly equivalent:

$$k_f(x, y) = \frac{1}{4}\, \mathbb{1}(f(x) > 0)\, \mathbb{1}(f(y) > 0)$$

gives

$$MMD(P, Q) = \left| \text{accuracy} - \frac{1}{2} \right|$$

- 0-1 kernel inflates variance, decreases test power
  - Intermediate option: $k(x, y) = f(x)\, f(y)$

Alternative approach: Classifier two-sample tests

- Train a classifier $f : \mathcal{X} \to \{1, -1\}$ on P from Q
- Test statistic: accuracy on test set (Lopez-Paz and Oquab, ICLR 2017)
- Almost exactly equivalent:

$$k_f(x, y) = \frac{1}{4}\, \mathbb{1}(f(x) > 0)\, \mathbb{1}(f(y) > 0)$$

gives

$$MMD(P, Q) = \left| \text{accuracy} - \frac{1}{2} \right|$$

- 0-1 kernel inflates variance, decreases test power
  - Intermediate option: $k(x, y) = f(x)\, f(y)$
- Empricially: deep kernel > linear > 0-1

Alternative approach: Classifier two-sample tests

- Train a classifier $f : \mathcal{X} \to \{1, -1\}$ on P from Q
- Test statistic: accuracy on test set (Lopez-Paz and Oquab, ICLR 2017)
- Almost exactly equivalent:

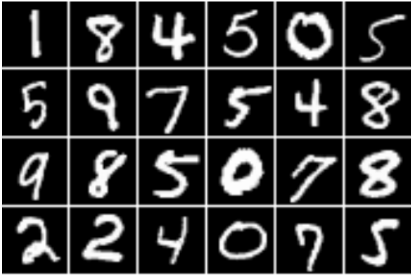
$$k_f(x, y) = \frac{1}{4}\, \mathbb{1}(f(x) > 0)\, \mathbb{1}(f(y) > 0)$$

gives

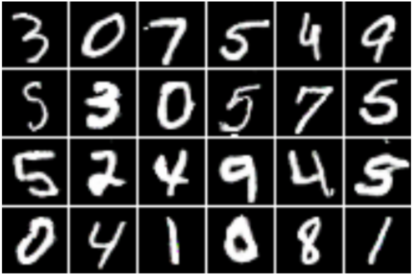
$$MMD(P, Q) = \left| \text{accuracy} - \frac{1}{2} \right|$$

- 0-1 kernel inflates variance, decreases test power
  - Intermediate option: $k(x, y) = f(x)\, f(y)$
- Also trains for cross-entropy, instead of power directly(ish)
- Empricially: deep kernel $>$ linear $>$ 0-1, $\frac{\widehat{MMD}^2}{\hat{\sigma}_{H_1}}$ $>$ cross-entropy

Interpreting the learned kernel

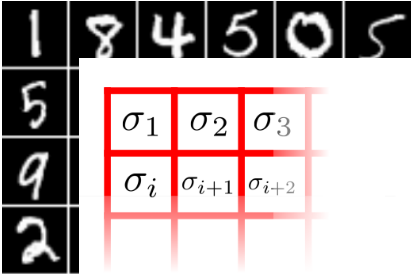


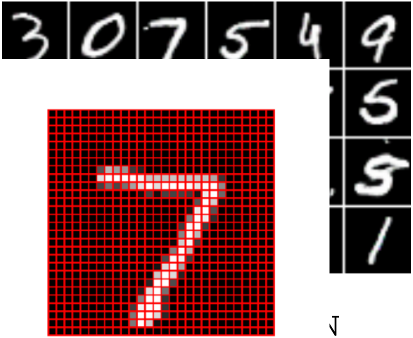
MNIST samples



Samples from a GAN

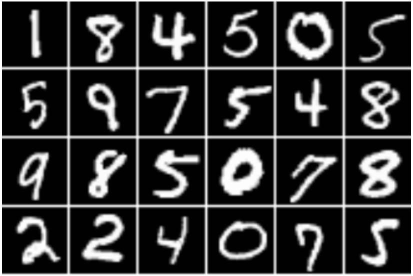
Interpreting the learned kernel



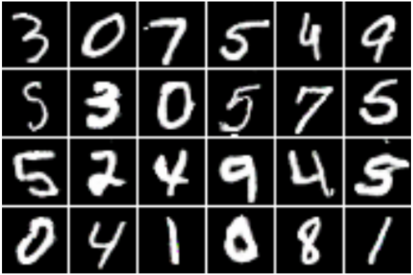


$$k(\boxed{4}, \boxed{2}) = \prod_{i=1}^{D} \exp\left(\frac{-(\boxed{4}[i] - \boxed{2}[i])^2}{\sigma_i^2}\right)$$

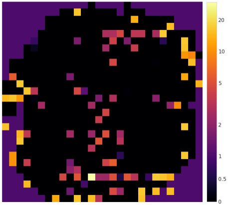
Interpreting the learned kernel



MNIST samples



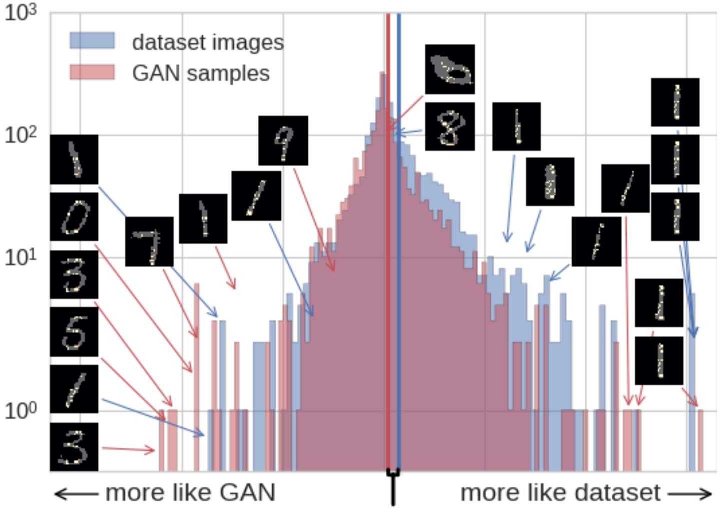
Samples from a GAN



ARD map

- Power for **optimized ARD kernel**: 1.00 at $\alpha = 0.01$
- Power for optimized RBF kernel: 0.57 at $\alpha = 0.01$

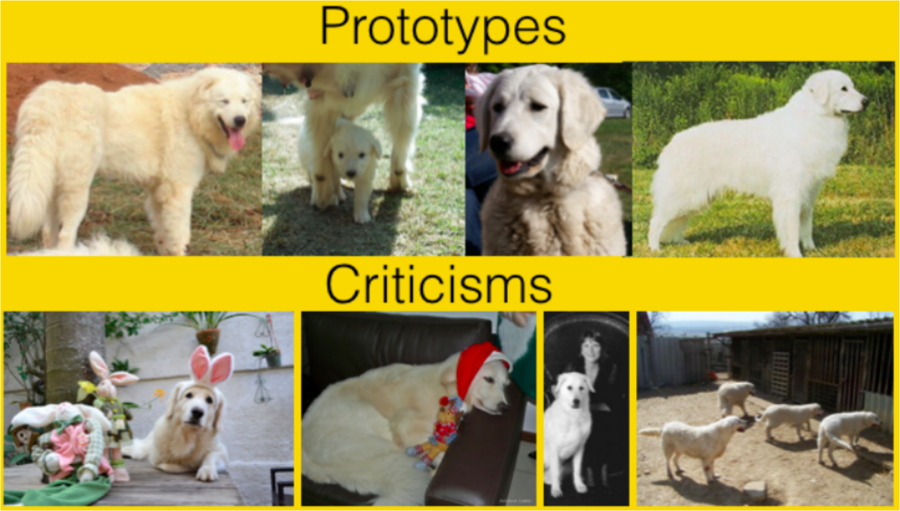
Interpreting points with largest witness function values



$\mathrm{MMD}^2 = 0.0001$

(Sutherland+ ICLR 2017)

Interpreting points with largest witness function values



(Kim+ NeurIPS 2016)

Main references and further reading

- MMD asymptotics and test construction:
  - Gretton, Borgwardt, Rasch, Schölkopf, Smola. A kernel two-sample test (2012)
- Kernels for tests on images:
  - Bińkowski, Sutherland, Arbel, Gretton. Demystifying MMD GANs (2018)
  - Bau, Zhu, Wulff, Peebles, Strobelt, Zhou, Torralba. Seeing What a GAN Cannot Generate (2019)
- Another approach: random 1d projection is almost surely consistent
  - Heller, Heller. Multivariate tests of association based on univariate tests (2016)
- Optimizing test kernels / classifiers:
  - Sutherland, Tung, Strathmann, De, Ramdas, Smola, Gretton. Generative Models and Model Criticism via Optimized Maximum Mean Discrepancy (2017)
    - Also our not-quite-on-arXiv-yet followup...
(with Feng Liu, Wenkai Xu, Jie Lu, Guangquang Zhang)
  - Lopez-Paz, Oquab. Revisiting Classifier Two-Sample Tests (2017)
- Interpreting via witness functions:
  - Lloyd, Ghahramani. Statistical Model Criticism using Kernel Two Sample Tests (2015)
  - Kim, Khanna, Koyejo. Examples are not Enough, Learn to Criticize! Criticism for Interpretability (2016)