Representing and comparing probabilities with kernels: Part 2

Arthur Gretton

Gatsby Computational Neuroscience Unit, University College London

MLSS Tuebingen, 2020
Comparing two samples

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

- Maximum Mean Discrepancy (MMD)...
  - ...as a difference in feature means
  - ...as an integral probability metric (not just a technicality!)

- A statistical test based on the MMD

- Next slides: training generative adversarial networks with MMD
  - Gradient regularisation and data adaptivity
Feature mean difference

- Simple example: 2 Gaussians with different means
- Answer: t-test
Feature mean difference

- Two Gaussians with same means, different variance
- Idea: look at difference in means of features of the RVs
- In Gaussian case: second order features of form $\varphi(x) = x^2$
Feature mean difference

- Two Gaussians with same means, different variance
- Idea: look at difference in **means of features** of the RVs
- In Gaussian case: second order features of form $\varphi(x) = x^2$
Feature mean difference

- Gaussian and Laplace distributions
- Same mean *and* same variance
- Difference in means using **higher order features**...RKHS
Infinitely many features using kernels

**Kernels: dot products of features**

Feature map \( \varphi(x) \in \mathcal{F} \),

\[ \varphi(x) = [\ldots \varphi_i(x) \ldots] \in l_2 \]

For positive definite \( k \),

\[ k(x, x') = \langle \varphi(x), \varphi(x') \rangle \mathcal{F} \]

Infinitely many features \( \varphi(x) \), dot product in closed form!

**Exponentiated quadratic kernel**

\[ k(x, x') = \exp \left( -\gamma \| x - x' \|^2 \right) \]

Features: Gaussian Processes for Machine learning, Rasmussen and Williams, Ch. 4.
Infinitely many features of distributions

Given $P$ a Borel probability measure on $\mathcal{X}$, define feature map of probability $P$,

$$\mu_P = \ldots \mathbb{E}_P [\varphi_i(X)] \ldots$$

For positive definite $k(x, x')$,

$$\langle \mu_P, \mu_Q \rangle_{\mathcal{F}} = \mathbb{E}_{P,Q} k(x, y)$$

for $x \sim P$ and $y \sim Q$.

Fine print: feature map $\varphi(x)$ must be Bochner integrable for all probability measures considered. Always true if kernel bounded.
Infinitely many features of distributions

Given $P$ a Borel probability measure on $X$, define feature map of probability $P$,

$$\mu_P = [\ldots E_P [\varphi_i(X)] \ldots]$$

For positive definite $k(x, x')$,

$$\langle \mu_P, \mu_Q \rangle_{\mathcal{F}} = E_{P,Q} k(x, y)$$

for $x \sim P$ and $y \sim Q$.

Fine print: feature map $\varphi(x)$ must be Bochner integrable for all probability measures considered. Always true if kernel bounded.
**The maximum mean discrepancy**

The *maximum mean discrepancy* is the distance between feature means:

\[
MMD^2(P, Q) = \| \mu_P - \mu_Q \|^2_F
\]

\[
= \langle \mu_P, \mu_P \rangle_F + \langle \mu_Q, \mu_Q \rangle_F - 2 \langle \mu_P, \mu_Q \rangle_F
\]

\[
= \mathbb{E}_P k(X, X') + \mathbb{E}_Q k(Y, Y') - 2 \mathbb{E}_{P,Q} k(X, Y)
\]

(a) + (a) - (b)
The maximum mean discrepancy is the distance between feature means:

\[ MMD^2(P, Q) = \|\mu_P - \mu_Q\|^2_F \]

\[ = \langle \mu_P, \mu_P \rangle_F + \langle \mu_Q, \mu_Q \rangle_F - 2\langle \mu_P, \mu_Q \rangle_F \]

\[ = \mathbb{E}_P k(X, X') + \mathbb{E}_Q k(Y, Y') - 2\mathbb{E}_{P,Q} k(X, Y) \]

(a) (a) (b)
The maximum mean discrepancy

The **maximum mean discrepancy** is the distance between feature means:

\[
MMD^2(P, Q) = \| \mu_P - \mu_Q \|^2_F \\
= \langle \mu_P, \mu_P \rangle_F + \langle \mu_Q, \mu_Q \rangle_F - 2 \langle \mu_P, \mu_Q \rangle_F \\
= \mathbb{E}_P k(X, X') + \mathbb{E}_Q k(Y, Y') - 2\mathbb{E}_{P,Q} k(X, Y)
\]

(a) = within distrib. similarity, (b) = cross-distrib. similarity.
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 $k(\text{fish}_i, \text{fish}_j)$
Illustration of MMD

The maximum mean discrepancy:

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

$$- \frac{2}{n^2} \sum_{i,j} k(\text{dog}_i, \text{fish}_j)$$
MMD as an integral probability metric

Are $P$ and $Q$ different?

Samples from $P$ and $Q$
MMD as an integral probability metric

Are $P$ and $Q$ different?
MMD as an integral probability metric

Integral probability metric:
Find a "well behaved function" $f(x)$ to maximize

$$E_P f(X) - E_Q f(Y)$$
MMD as an integral probability metric

Integral probability metric:
Find a "well behaved function" $f(x)$ to maximize

$$E_P f(X) - E_Q f(Y)$$
MMD as an integral probability metric

Maximum mean discrepancy: smooth function for $P$ vs $Q$

$$MMD(P, Q; F) := \sup_{\|f\|_F \leq 1} \left[ \mathbb{E}_P f(X) - \mathbb{E}_Q f(Y) \right]$$

($F$ = unit ball in RKHS $\mathcal{F}$)
MMD as an integral probability metric

Maximum mean discrepancy: smooth function for $P$ vs $Q$

$$MMD(P, Q; F) := \sup_{\|f\|_F \leq 1} \left[ \mathbb{E}_P f(X) - \mathbb{E}_Q f(Y) \right]$$

($F =$ unit ball in RKHS $\mathcal{F}$)

Functions are linear combinations of features:

$$f(x) = \langle f, \varphi(x) \rangle_{\mathcal{F}} = \sum_{\ell=1}^{\infty} f_\ell \varphi_\ell(x) = \begin{bmatrix} f_1 \\ f_2 \\ f_3 \\ \vdots \end{bmatrix}^\top$$

$$\|f\|_{\mathcal{F}}^2 := \sum_{i=1}^{\infty} f_i^2 \leq 1$$
MMD as an integral probability metric

Maximum mean discrepancy: smooth function for $P$ vs $Q$

\[
MMD(\mathcal{P}, \mathcal{Q}; F) := \sup_{\|f\|_{\mathcal{F}} \leq 1} \left[ \mathbb{E}_P f(X) - \mathbb{E}_Q f(Y) \right]
\]

($F$ = unit ball in RKHS $\mathcal{F}$)
MMD as an integral probability metric

Maximum mean discrepancy: smooth function for $P$ vs $Q$

\[
MMD(\mathcal{P}, \mathcal{Q}; F) := \sup_{\|f\|_{\mathcal{F}} \leq 1} \left[ \mathbb{E}_P f(X) - \mathbb{E}_Q f(Y) \right]
\]

($F$ = unit ball in RKHS $\mathcal{F}$)

For characteristic RKHS $\mathcal{F}$, $MMD(\mathcal{P}, \mathcal{Q}; F) = 0$ iff $\mathcal{P} = \mathcal{Q}$

Other choices for witness function class:

- Bounded continuous [Dudley, 2002]
- Bounded variation 1 (Kolmogorov metric) [Müller, 1997]
- Bounded Lipschitz (Wasserstein distances) [Dudley, 2002]
MMD as an integral probability metric

Maximum mean discrepancy: smooth function for $P$ vs $Q$

$$MMD(P, Q; F) := \sup_{\|f\|_F \leq 1} [E_Pf(X) - E_Qf(Y)]$$

($F = \text{unit ball in RKHS } \mathcal{F}$)

Expectations of functions are linear combinations of expected features

$$E_P(f(X)) = \langle f, E_P\varphi(X) \rangle_{\mathcal{F}} = \langle f, \mu_P \rangle_{\mathcal{F}}$$

(always true if kernel is bounded)
Integral prob. metric vs feature difference

The MMD:

$$MMD(P, Q; F) = \sup_{\|f\|_{\mathcal{F}} \leq 1} \left[ \mathbb{E}_P f(X) - \mathbb{E}_Q f(Y) \right]$$

![Witness f for Gauss and Laplace densities](image-url)
Integral prob. metric vs feature difference

The MMD:

\[
MMD(P, Q; F) = \sup_{f \in \mathcal{F}, \|f\|_{\mathcal{F}} \leq 1} \left[ \mathbb{E}_P f(X) - \mathbb{E}_Q f(Y) \right]
\]

\[
= \sup_{f \in \mathcal{F}, \|f\|_{\mathcal{F}} \leq 1} \langle f, \mu_P - \mu_Q \rangle_{\mathcal{F}}
\]

use

\[
\mathbb{E}_P f(X) = \langle \mu_P, f \rangle_{\mathcal{F}}
\]
The MMD:

\[ MMD(P, Q; F) = \sup_{\|f\|_{\mathcal{F}} \leq 1} \| \mathbb{E}_P f(X) - \mathbb{E}_Q f(Y) \| \]

\[ = \sup_{\|f\|_{\mathcal{F}} \leq 1} \langle f, \mu_P - \mu_Q \rangle_{\mathcal{F}} \]
The MMD:

\[ \text{MMD}(P, Q; F) = \sup_{\|f\|_{\mathcal{F}} \leq 1} \left[ E_P f(X) - E_Q f(Y) \right] \]

\[ = \sup_{\|f\|_{\mathcal{F}} \leq 1} \langle f, \mu_P - \mu_Q \rangle_{\mathcal{F}} \]
Integral prob. metric vs feature difference

The MMD:

\[
\text{MMD}(P, Q; F) = \sup_{\|f\|_{\mathcal{F}} \leq 1} \left[ \mathbb{E}_P f(X) - \mathbb{E}_Q f(Y) \right] = \sup_{\|f\|_{\mathcal{F}} \leq 1} \langle f, \mu_P - \mu_Q \rangle_{\mathcal{F}}
\]

\[
f^* = \frac{\mu_P - \mu_Q}{\|\mu_P - \mu_Q\|}
\]
Integral prob. metric vs feature difference

The MMD:

\[
MMD(P, Q; F) = \sup_{\|f\|_{\mathcal{F}} \leq 1} \left[ \mathbb{E}_P f(X) - \mathbb{E}_Q f(Y) \right] \\
= \sup_{\|f\|_{\mathcal{F}} \leq 1} \langle f, \mu_P - \mu_Q \rangle_{\mathcal{F}} \\
= \|\mu_P - \mu_Q\|
\]

Function view and feature view equivalent
Construction of MMD witness

Construction of empirical witness function (proof: next slide!)

Observe $X = \{x_1, \ldots, x_n\} \sim P$

Observe $Y = \{y_1, \ldots, y_n\} \sim Q$
Construction of MMD witness

Construction of empirical \textit{witness function} \ (proof: next slide!)
Construction of MMD witness

Construction of empirical witness function (proof: next slide!)
Construction of MMD witness

Construction of empirical witness function (proof: next slide!)
Derivation of empirical witness function

Recall the witness function expression

\[ f^* \propto \mu_P - \mu_Q \]
Derivation of empirical witness function

Recall the witness function expression

$$f^* \propto \mu_P - \mu_Q$$

The empirical feature mean for $P$

$$\hat{\mu}_P := \frac{1}{n} \sum_{i=1}^{n} \varphi(x_i)$$
Derivation of empirical witness function

Recall the witness function expression

\[ f^* \propto \mu_P - \mu_Q \]

The empirical feature mean for \( P \)

\[ \hat{\mu}_P := \frac{1}{n} \sum_{i=1}^{n} \varphi(x_i) \]

The empirical witness function at \( v \)

\[ f^*(v) = \langle f^*, \varphi(v) \rangle_{\mathcal{F}} \]
Derivation of empirical witness function

Recall the witness function expression

\[ f^* \propto \mu_P - \mu_Q \]

The empirical feature mean for \( P \)

\[ \hat{\mu}_P := \frac{1}{n} \sum_{i=1}^{n} \varphi(x_i) \]

The empirical witness function at \( v \)

\[ f^*(v) = \langle f^*, \varphi(v) \rangle_{\mathcal{F}} \]

\[ \propto \langle \hat{\mu}_P - \hat{\mu}_Q, \varphi(v) \rangle_{\mathcal{F}} \]
Derivation of empirical witness function

Recall the witness function expression

\[ f^* \propto \mu_P - \mu_Q \]

The empirical feature mean for \( P \)

\[ \hat{\mu}_P := \frac{1}{n} \sum_{i=1}^{n} \varphi(x_i) \]

The empirical witness function at \( v \)

\[ f^*(v) = \langle f^*, \varphi(v) \rangle_F \]

\[ \propto \langle \hat{\mu}_P - \hat{\mu}_Q, \varphi(v) \rangle_F \]

\[ = \frac{1}{n} \sum_{i=1}^{n} k(x_i, v) - \frac{1}{n} \sum_{i=1}^{n} k(y_i, v) \]

Don’t need explicit feature coefficients

\[ f^* := [ f_1^*, f_2^*, \ldots ] \]
Interlude: divergence measures
Divergences
Divergences

Integral prob. metrics

\[ D_H(P, Q) = \sup_{g \in \mathcal{H}} |\mathbb{E}_{X \sim P} g(X) - \mathbb{E}_{Y \sim Q} g(Y)| \]

\[ D_{\phi}(P, Q) = \int_X q(x) \phi \left( \frac{p(x)}{q(x)} \right) \, dx \]
The integral probability metrics

Integral prob. metrics

\[ D_\mathcal{H}(P, Q) = \sup_{g \in \mathcal{H}} |E_X \sim P g(X) - E_Y \sim Q g(Y)| \]

wasserstein

MMD

\[ D_\phi(P, Q) = \int_X q(x) \phi \left( \frac{p(x)}{q(x)} \right) dx \]

\[ \phi \text{-divergences} \]
The $\phi$-divergences

\[ D_H(P, Q) = \sup_{g \in \mathcal{H}} |E_X \sim P g(X) - E_Y \sim Q g(Y)| \]

$\phi$-divergences

- Hellinger
- KL
- Pearson chi$^2$

\[ D_\phi(P, Q) = \int_X q(x) \phi \left( \frac{p(x)}{q(x)} \right) dx \]
Divergences

\[ D_{\mathcal{H}}(P, Q) = \sup_{g \in \mathcal{H}} |\mathbb{E}_{X \sim P} g(X) - \mathbb{E}_{Y \sim Q} g(Y)| \]

\[ D_{\phi}(P, Q) = \int_X q(x) \phi \left( \frac{p(x)}{q(x)} \right) dx \]

Integral prob. metrics

- Wasserstein
- MMD

\( \phi \)-divergences

- Hellinger
- KL
- Pearson chi\(^2\)
Divergences

Integral prob. metrics

$D_\mathcal{H}(P, Q)
= \sup_{g \in \mathcal{H}} |E_{X \sim P}g(X) - E_{Y \sim Q}g(Y)|$

MMD

$\text{TV}$

$\phi$-divergences

Hellinger

$D_\phi(P, Q)
= \int_x q(x)\phi\left(\frac{p(x)}{q(x)}\right) dx$

KL

Pearson chi$^2$

Sriperumbudur, Fukumizu, G, Schoelkopf, Lanckriet, EJS (2012)
Two-Sample Testing with MMD
A statistical test using MMD

The empirical MMD:

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

\[ - \frac{2}{n^2} \sum_{i,j} k(x_i, y_j) \]

How does this help decide whether \( P = Q \)?
A statistical test using MMD

The empirical MMD:

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

Perspective from statistical hypothesis testing:

- **Null hypothesis** $\mathcal{H}_0$ when $P = Q$
  - should see $\hat{\text{MMD}}^2$ "close to zero".
- **Alternative hypothesis** $\mathcal{H}_1$ when $P \neq Q$
  - should see $\hat{\text{MMD}}^2$ "far from zero"
A statistical test using MMD

The empirical MMD:

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

Perspective from statistical hypothesis testing:

- **Null hypothesis** \( \mathcal{H}_0 \) when \( P = Q \)
  
  - should see \( \widehat{\text{MMD}}^2 \) “close to zero”.

- **Alternative hypothesis** \( \mathcal{H}_1 \) when \( P \neq Q \)
  
  - should see \( \widehat{\text{MMD}}^2 \) “far from zero”

Want Threshold \( c_\alpha \) for \( \widehat{\text{MMD}}^2 \) to get false positive rate \( \alpha \)
Behaviour of $\hat{MMD}^2$ when $P \neq Q$

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

- Laplace with different $y$-variance.
- $\sqrt{n} \times \hat{MMD}^2 = 1.2$
Behaviour of $\overline{\text{MMD}}^2$ when $P \neq Q$

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

- Laplace with different $y$-variance.
- $\sqrt{n} \times \overline{\text{MMD}}^2 = 1.2$

Number of MMDs: 1

\[
\begin{array}{c|c|c|c|c|c|c}
\text{Prob. of } \sqrt{n} \times \overline{\text{MMD}}^2 & 0 & 1 & 2 & 3 & 4 & 5 \\
\text{Value} & 0 & 1 & 0 & 0 & 0 & 0 \\
\end{array}
\]
Behaviour of $\widehat{\text{MMD}}^2$ when $P \neq Q$

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

- Laplace with different $y$-variance.
- $\sqrt{n} \times \widehat{\text{MMD}}^2 = 1.5$

Number of MMDs: 2
Behaviour of $\hat{MMD}^2$ when $P \neq Q$

Repeat this 150 times …

Number of MMDs: 150
Behaviour of $\overline{MMD}^2$ when $P \neq Q$

Repeat this 300 times …

Number of MMDs: 300

![Histogram](image-url)
Behaviour of $\sqrt{n}\times\text{MMD}^2$ when $P \neq Q$

Repeat this 3000 times …
Asymptotics of $\overline{\text{MMD}}^2$ when $P \neq Q$

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

$$\frac{\overline{\text{MMD}}^2 - \text{MMD}^2(P, Q)}{\sqrt{V_n(P, Q)}} \xrightarrow{D} \mathcal{N}(0, 1),$$

where variance $V_n(P, Q) = O(n^{-1})$. 

MMD density under $\mathcal{H}_1$

Two Laplace distributions with different variances
Behaviour of $\overbrace{MMD^2}$ 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 $\overbrace{\text{MMD}^2}^{\text{MMD}^2}$ when $P = Q$

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

Number of MMDs: 20
Behaviour of $\hat{MMD}^2$ when $P = Q$

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

Number of MMDs: 50

![Histogram of $n \times \hat{MMD}^2$](chart.png)
Behaviour of $\hat{\text{MMD}}^2$ when $P = Q$

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

Number of MMDs: 100

![Histogram showing the distribution of $n \times \hat{\text{MMD}}^2$ for $P = Q = \mathcal{N}(0, 1)$]
Behaviour of $\overline{\text{MMD}}^2$ when $P = Q$

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

Number of MMDs: 1000
Asymptotics of $\hat{MMD}^2$ when $P = Q$

Where $P = Q$, statistic has asymptotic distribution

$$n\hat{MMD}^2 \sim \sum_{l=1}^{\infty} \lambda_l [z_l^2 - 2]$$

where

$$\lambda_i \psi_i(x') = \int_{X} \tilde{k}(x, x') \psi_i(x) dP(x)$$

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

A summary of the asymptotics:

![Graph showing the probability of $n \times \hat{MMD}^2$ against $n \times \hat{MMD}^2$. Red line represents $P = Q$, blue line represents $P \neq Q$. The graph peaks at $n \times \hat{MMD}^2$ values close to 1.]
A statistical test

Test construction: (G., Borgwardt, Rasch, Schoelkopf, and Smola, JMLR 2012)

\[ c_\alpha = 1 - \alpha \text{ quantile when } P = Q \]
How do we get test threshold $c_\alpha$?

Original empirical MMD for dogs and fish:

$$X = \begin{bmatrix} \text{dog} & \text{dog} & \text{dog} & \ldots \end{bmatrix}$$

$$Y = \begin{bmatrix} \text{fish} & \text{fish} & \text{fish} & \ldots \end{bmatrix}$$

$$\widehat{\text{MMD}}^2 = \frac{1}{n(n-1)} \sum_{i \neq j} k(x_i, x_j) + \frac{1}{n(n-1)} \sum_{i \neq j} k(y_i, y_j) - \frac{2}{n^2} \sum_{i,j} k(x_i, y_j)$$
How do we get test threshold $c_\alpha$?

Permuted dog and fish samples (merdogs):

$$\tilde{X} = \begin{bmatrix} \text{fish} & \text{dog} & \text{fish} & \ldots \end{bmatrix}$$

$$\tilde{Y} = \begin{bmatrix} \text{dog} & \text{fish} & \text{dog} & \ldots \end{bmatrix}$$
How do we get test threshold $c_\alpha$?

Permuted dog and fish samples (merdogs):

$$\tilde{X} = \begin{bmatrix} \text{fish} & \text{dog} & \ldots \end{bmatrix}$$

$$\tilde{Y} = \begin{bmatrix} \text{dog} & \text{fish} & \ldots \end{bmatrix}$$

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

Permutation simulates $P = Q$
How to choose the best kernel: optimising the kernel parameters
The best test for the job

- A test’s power depends on $k(x, x')$, $P$, and $Q$ (and $n$)
- With characteristic kernel, MMD test has power $\rightarrow 1$ as $n \rightarrow \infty$ 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 $k(x, x')$, $P$, and $Q$ (and $n$)
- With characteristic kernel, MMD test has power $\rightarrow 1$ as $n \rightarrow \infty$ for any (fixed) problem
  - But, for many $P$ and $Q$, will have terrible power with reasonable $n$!
- You *can* choose a good kernel for a given problem
- You *can’t* get one kernel that has good finite-sample power for all problems
  - No one test can have all that power
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 \( \sigma \): for any \( P \) and \( Q \), power \( \rightarrow 1 \) as \( n \rightarrow \infty \)
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 \( \sigma \): for any \( P \) and \( Q \), power \( \to 1 \) as \( n \to \infty \)
- But choice of \( \sigma \) 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 \( \sigma \): for any \( P \) and \( Q \), power \( \rightarrow 1 \) as \( n \rightarrow \infty \)
- But choice of \( \sigma \) 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 \(\sigma\): for any \(P\) and \(Q\), power \(\to 1\) as \(n \to \infty\)

- But choice of \(\sigma\) 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 \( \sigma \): for any \( P \) and \( Q \), power \( \rightarrow 1 \) as \( n \rightarrow \infty \)

- But choice of \( \sigma \) is very important for finite \( n \)...
Choosing a kernel for the test

- Simple choice: exponentiated quadratic

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

- \textit{Characteristic:} for any \( \sigma \): for any \( P \) and \( Q \), power \( \rightarrow 1 \) as \( n \rightarrow \infty \)
- But choice of \( \sigma \) is very important for finite \( n \)...
- \( \ldots \) and some problems (e.g. images) might have no good choice for \( \sigma \)
Maximising test power same as minimizing false negatives

c_α = 1 − α quantile when P = Q
Optimizing kernel for test power

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

$$\Pr_1 \left( \frac{\hat{\text{MMD}}^2}{n} > \hat{c}_\alpha \right)$$
Optimizing kernel for test power

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

\[
\Pr_1 \left( n \overline{\operatorname{MMD}^2} > \hat{c}_\alpha \right) \\
\rightarrow \Phi \left( \frac{\operatorname{MMD}^2(P, Q)}{\sqrt{V_n(P, Q)}} - \frac{c_\alpha}{n \sqrt{V_n(P, Q)}} \right)
\]

where

- $\Phi$ is the CDF of the standard normal distribution.
- $\hat{c}_\alpha$ is an estimate of $c_\alpha$ test threshold.
The power of our test ($\Pr_1$ denotes probability under $P \neq Q$):

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

\[ \rightarrow \Phi \left( \frac{\text{MMD}^2(P, Q)}{\sqrt{V_n(P, Q)}} - \frac{c_\alpha}{n \sqrt{V_n(P, Q)}} \right) \]

For large $n$, second term negligible!
Optimizing kernel for test power

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

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

$$\rightarrow \Phi \left( \frac{\text{MMD}^2(P, Q)}{\sqrt{V_n(P, Q)}} - \frac{c_\alpha}{n \sqrt{V_n(P, Q)}} \right)$$

To maximize test power, maximize

$$\frac{\text{MMD}^2(P, Q)}{\sqrt{V_n(P, Q)}}$$
Data splitting

Choose a kernel $k$ maximizing

$$\frac{\overline{MMD}^2}{\sqrt{\hat{V}_n(P,Q)}}$$

Use chosen $k$ for MMD test
Learning a kernel helps a lot

Kernel with deep learned features:

\[ k_\theta(x, y) = [(1 - \epsilon) \kappa(\Phi_\theta(x), \Phi_\theta(y)) + \epsilon] q(x, y) \]

\( \kappa \) and \( q \) are Gaussian kernels
Learning a kernel helps a lot

Kernel with deep learned features:

\[ k_\theta(x, y) = [(1 - \epsilon)\kappa(\Phi_\theta(x), \Phi_\theta(y)) + \epsilon] \ q(x, y) \]

\( \kappa \) and \( q \) are Gaussian kernels

- CIFAR-10 vs CIFAR-10.1, null rejected 75% of time
Learning a kernel helps a lot

Kernel with deep learned features:

\[ k_\theta(x, y) = [(1 - \epsilon) \kappa(\Phi_\theta(x), \Phi_\theta(y)) + \epsilon] q(x, y) \]

\( \kappa \) and \( q \) are Gaussian kernels

- CIFAR-10 vs CIFAR-10.1, null rejected 75% of time
Questions?

- A brief introduction to RKHS

- **Maximum Mean Discrepancy (MMD)**
  - ...as a difference in feature means
  - ...as an integral probability metric (not just a technicality!)

- A statistical test based on the MMD
MMD for GAN training
Training implicit generative models

- Have: One collection of samples $X$ from unknown distribution $P$.
- Goal: generate samples $Q$ that look like $P$

Using a critic $D(P, Q)$ to train a GAN

(Binkowski, Sutherland, Arbel, G., ICLR 2018),
(Arbel, Sutherland, Binkowski, G., NeurIPS 2018)
Visual notation: GAN setting
Visual notation: GAN setting
Critic functions

\[ D_H(P, Q) = \sup_{g \in \mathcal{H}} |E_{X \sim P} g(X) - E_{Y \sim Q} g(Y)| \]

\[ D_\phi(P, Q) = \int_X q(x) \phi \left( \frac{p(x)}{q(x)} \right) dx \]
What I won’t cover: the generator

Under review as a conference paper at ICLR 2016

Figure 1: DCGAN generator used for LSUN scene modeling. A 100 dimensional uniform distribution $Z$ is projected to a small spatial extent convolutional representation with many feature maps. A series of four fractionally-strided convolutions (in some recent papers, these are wrongly called deconvolutions) then convert this high level representation into a $64 \times 64$ pixel image. Notably, no fully connected or pooling layers are used.

suggested value of 0.9 resulted in training oscillation and instability while reducing it to 0.5 helped stabilize training.

4.1 LSUN

As visual quality of samples from generative image models has improved, concerns of over-fitting and memorization of training samples have risen. To demonstrate how our model scales with more data and higher resolution generation, we train a model on the LSUN bedrooms dataset containing a little over 3 million training examples. Recent analysis has shown that there is a direct link between how fast models learn and their generalization performance (Hardt et al., 2015). We show samples from one epoch of training (Fig.2), mimicking online learning, in addition to samples after convergence (Fig.3), as an opportunity to demonstrate that our model is not producing high quality samples via simply overfitting/memorizing training examples. No data augmentation was applied to the images.

4.1.1 DEDUPLICATION

To further decrease the likelihood of the generator memorizing input examples (Fig.2) we perform a simple image de-duplication process. We fit a 3072-128-3072 de-noising dropout regularized RELU autoencoder on 32x32 downsampled center-crops of training examples. The resulting code layer activations are then binarized via thresholding the ReLU activation which has been shown to be an effective information preserving technique (Srivastava et al., 2014) and provides a convenient form of semantic-hashing, allowing for linear time de-duplication. Visual inspection of hash collisions showed high precision with an estimated false positive rate of less than 1 in 100. Additionally, the technique detected and removed approximately 275,000 near duplicates, suggesting a high recall.

4.2 FACES

We scraped images containing human faces from random web image queries of peoples names. The people names were acquired from dbpedia, with a criterion that they were born in the modern era. This dataset has 3M images from 10K people. We run an OpenCV face detector on these images, keeping the detections that are sufficiently high resolution, which gives us approximately 350,000 face boxes. We use these face boxes for training. No data augmentation was applied to the images.

Radford, Metz, Chintala, ICLR 2016
F-divergence as critic

An unhelpful critic? Jensen-Shannon,

\[ D_{JS}(P, Q) = \frac{1}{2} D_{KL}(p, \frac{p+q}{2}) + \frac{1}{2} D_{KL}(q, \frac{p+q}{2}) \]

\[ D_{JS}(P, Q) = \log 2 \]
F-divergence as critic

An unhelpful critic? Jensen-Shannon,

Goodfellow et al. (NeurIPS 2014), Arjovsky and Bottou [ICLR 2017]

\[ D_{JS}(P, Q) = \frac{1}{2} D_{KL} \left( p, \frac{p + q}{2} \right) + \frac{1}{2} D_{KL} \left( q, \frac{p + q}{2} \right) \]

\[ D_{JS}(P, Q) = \log 2 \]
F-divergence as critic

An unhelpful critic? Jensen-Shannon,
Goodfellow et al. (NeurIPS 2014), Arjovsky and Bottou [ICLR 2017]

\[ D_{JS}(P, Q) = \frac{1}{2} D_{KL}(p, \frac{p+q}{2}) + \frac{1}{2} D_{KL}(q, \frac{p+q}{2}) \]

What is done in practice?

• Add “instance noise” to the reference and generator observations
  Sonderby et al. [arXiv 2016], Arjovsky and Bottou [ICLR 2017]
• or (approx. equivalently) a data-dependent gradient penalty for the variational critic
  Roth et al [NeurIPS 2017], Nagarajan and Kolter [NeurIPS 2017], Mescheder et al. [ICML 2018]
F-divergence as critic

An unhelpful critic? Jensen-Shannon,
Goodfellow et al. (NeurIPS 2014), Arjovsky and Bottou [ICLR 2017]

\[
D_{JS}(P, Q) = \frac{1}{2} D_{KL}(p, \frac{p+q}{2}) + \frac{1}{2} D_{KL}(q, \frac{p+q}{2})
\]

What is done in practice?

- Use a variational approximation to the critic, alternate generator and critic training
  Goodfellow et al. [NeurIPS 2014], Nowozin et al. [NeurIPS 2016]
F-divergence as critic

An unhelpful critic? Jensen-Shannon, Goodfellow et al. (NeurIPS 2014), Arjovsky and Bottou [ICLR 2017]

\[
D_{JS}(P, Q) = \frac{1}{2} D_{KL}(p, \frac{p+q}{2}) + \frac{1}{2} D_{KL}(q, \frac{p+q}{2})
\]

What is done in practice?

- Use a variational approximation to the critic, alternate generator and critic training Goodfellow et al. [NeurIPS 2014], Nowozin et al. [NeurIPS 2016]
- Add “instance noise” to the reference and generator observations Sonderby et al. [arXiv 2016], Arjovsky and Bottou [ICLR 2017]
F-divergence as critic

An unhelpful critic? Jensen-Shannon,

Goodfellow et al. (NeurIPS 2014), Arjovsky and Bottou [ICLR 2017]

\[ D_{JS}(P, Q) = \frac{1}{2} D_{KL} \left( p, \frac{p+q}{2} \right) + \frac{1}{2} D_{KL} \left( q, \frac{p+q}{2} \right) \]

What is done in practice?

- Use a variational approximation to the critic, alternate generator and critic training
  Goodfellow et al. [NeurIPS 2014], Nowozin et al. [NeurIPS 2016]
- Add “instance noise” to the reference and generator observations
  Sonderby et al. [arXiv 2016], Arjovsky and Bottou [ICLR 2017]
  - ...or (approx. equivalently) a data-dependent gradient penalty for the variational critic
    Roth et al [NeurIPS 2017], Nagarajan and Kolter [NeurIPS 2017], Mescheder et al. [ICML 2018]
Wasserstein distance as critic

A helpful critic witness:

\[ W_1(P, Q) = \sup_{\|f\|_{L^1} \leq 1} E_P f(X) - E_Q f(Y). \]

\[ \|f\|_L := \sup_{x \neq y} \frac{|f(x) - f(y)|}{\|x - y\|} \]

\[ W_1 = 0.88 \]

Santambrogio, Optimal Transport for Applied Mathematicians (2015, Section 5.4)
G Peyré, M Cuturi, Computational Optimal Transport (2019)
Wasserstein distance as critic

A helpful critic witness:

\[ W_1(P, Q) = \sup_{\|f\|_{L^1} \leq 1} E_P f(X) - E_Q f(Y). \]

\[ \|f\|_L := \sup_{x \neq y} |f(x) - f(y)| / \|x - y\| \]

\[ W_1 = 0.65 \]

Santambrogio, Optimal Transport for Applied Mathematicians (2015, Section 5.4)
G Peyré, M Cuturi, Computational Optimal Transport (2019)
MMD as critic

A helpful critic witness:

\[ MMD(P, Q) = \sup_{\|f\|_{\mathcal{F}} \leq 1} E_P f(X) - E_Q f(Y). \]

MMD = 1.8
MMD as critic

A helpful critic witness:

\[ MMD(P, Q) = \sup_{\|f\|_{\mathcal{F}} \leq 1} E_P f(X) - E_Q f(Y) \]

\[ \text{MMD} = 1.1 \]
MMD as critic

An unhelpful critic witness: $MMD(P, Q)$ with a narrow kernel.

$MMD=0.64$
MMD as critic

An **unhelpful** critic witness:

\[ MMD(P, Q) \] with a narrow kernel.

\[ MMD = 0.64 \]
Gradient penalty: the regularisation viewpoint
MMD for GAN critic

Can you use **MMD as a critic** to train GANs?

**From ICML 2015:**

---

**Generative Moment Matching Networks**

---

Yujia Li\(^1\)
Kevin Swersky\(^1\)
Richard Zemel\(^{1,2}\)

\(^1\)Department of Computer Science, University of Toronto, Toronto, ON, CANADA
\(^2\)Canadian Institute for Advanced Research, Toronto, ON, CANADA

**From UAI 2015:**

---

**Training generative neural networks via Maximum Mean Discrepancy optimization**

---

Gintare Karolina Dziugaite
University of Cambridge

Daniel M. Roy
University of Toronto

Zoubin Ghahramani
University of Cambridge
MMD for GAN critic

Can you use **MMD as a critic** to train GANs?

Need better image features.
CNN features for IPM witness functions

- Add convolutional features!
- The critic (teacher) also needs to be trained.

\[ \mathcal{K}(x, y) = h_\psi^\top(x) h_\psi(y) \]
where \( h_\psi(x) \) is a CNN map:

- **Wasserstein GAN** Arjovsky et al. [ICML 2017]
- **WGAN-GP** Gulrajani et al. [NeurIPS 2017]

\[ \mathcal{K}(x, y) = k(h_\psi(x), h_\psi(y)) \]
where \( h_\psi(x) \) is a CNN map, \( k \) is e.g. an exponentiated quadratic kernel

- **MMD** Li et al., [NeurIPS 2017]
- **Cramer** Bellemare et al. [2017]
- **Coulomb** Unterthiner et al., [ICLR 2018]
- **Demystifying MMD GANs** Binkowski, Sutherland, Arbel, G., [ICLR 2018]
CNN features for IPM witness functions

- Add convolutional features!
- The critic (teacher) also needs to be trained.

\[ K(x, y) = h_\psi^\top(x) h_\psi(y) \]

where \( h_\psi(x) \) is a CNN map:

- **Wasserstein GAN** Arjovsky et al. [ICML 2017]
- **WGAN-GP** Gulrajani et al. [NeurIPS 2017]

\[ K(x, y) = k(h_\psi(x), h_\psi(y)) \]

where \( h_\psi(x) \) is a CNN map, \( k \) is e.g. an exponentiated quadratic kernel

- **MMD** Li et al., [NeurIPS 2017]
- **Cramer** Bellemare et al. [2017]
- **Coulomb** Unterthiner et al., [ICLR 2018]
- **Demystifying MMD GANs** Binkowski, Sutherland, Arbel, G., [ICLR 2018]
Witness function, kernels on deep features

Reminder: witness function, $k(x, y)$ is exponentiated quadratic
Witness function, kernels on deep features

Reminder: witness function,

\[ k(h_\psi(x), h_\psi(y)) \] with nonlinear \( h_\psi \) and exp. quadratic \( k \)
Challenges for learned critic features

Learned critic features:

MMD with kernel $k(h_{\psi}(x), h_{\psi}(y))$ must give useful gradient to generator.
Challenges for learned critic features

Learned critic features:
MMD with kernel $k(h_\psi(x), h_\psi(y))$ must give useful gradient to generator.

Relation with test power?
If the MMD with kernel $k(h_\psi(x), h_\psi(y))$ gives a powerful test, will it be a good critic?
Challenges for learned critic features

Learned critic features:

MMD with kernel $k(h_\psi(x), h_\psi(y))$ must give useful gradient to generator.

Relation with test power?

If the MMD with kernel $k(h_\psi(x), h_\psi(y))$ gives a powerful test, will it be a good critic?
A simple 2-D example

Samples from target $P$ and model $Q$
A simple 2-D example

Witness gradient, MMD with exp. quad. kernel $k(x, y)$
A simple 2-D example

What the kernels $k(x, y)$ look like

MMD Gaussian
On gradient regularizers for MMD GANs

Michael Arbel  
Gatsby Computational Neuroscience Unit  
University College London  
michael.n.arbel@gmail.com

Mikołaj Bińkowski  
Department of Mathematics  
Imperial College London  
mikbinkowski@gmail.com

Dougal J. Sutherland  
Gatsby Computational Neuroscience Unit  
University College London  
dougal@gmail.com

Arthur Gretton  
Gatsby Computational Neuroscience Unit  
University College London  
arthur.gretton@gmail.com
A data-adaptive gradient penalty: NeurIPS 2018

- **New gradient regulariser** Arbel, Sutherland, Binkowski, G. [NeurIPS 2018]
- **Also related to Sobolev GAN** Mroueh et al. [ICLR 2018]

Maximise scaled MMD over critic features:

$$\text{SMMD}(P, \lambda) = \sigma_{P,\lambda} \text{ MMD}$$

where

$$\sigma_{P,\lambda}^2 = \lambda + \int k(h_{\psi}(x), h_{\psi}(x)) dP(x) + \sum_{i=1}^{d} \int \partial_i \partial_{i+d} k(h_{\psi}(x), h_{\psi}(x)) dP(x)$$
A data-adaptive gradient penalty: NeurIPS 2018

- **New gradient regulariser** Arbel, Sutherland, Binkowski, G. [NeurIPS 2018]
- Also related to **Sobolev GAN** Mroueh et al. [ICLR 2018]

Maximise scaled MMD over critic features:

$$\text{SMMD}(\mathcal{P}, \lambda) = \sigma_{\mathcal{P}, \lambda} \text{ MMD}$$

where

$$\sigma_{\mathcal{P}, \lambda}^2 = \lambda + \int k(h_\psi(x), h_\psi(x))d\mathcal{P}(x) + \sum_{i=1}^{d} \int \partial_i \partial_{i+d} k(h_\psi(x), h_\psi(x)) d\mathcal{P}(x)$$

Idea: rather than regularise the **critic** or **witness function**, regularise features directly
Simple 2-D example revisited

Samples from target $P$ and model $Q$
Use kernels $k(h_{\psi}(x), h_{\psi}(y))$ with features

$$h_{\psi}(x) = L_3\left(\begin{bmatrix} x \\ L_2(L_1(x)) \end{bmatrix}\right)$$

where $L_1, L_2, L_3$ are fully connected with quadratic nonlinearity.
Simple 2-D example revisited

Witness gradient, maximize $SMMD(P, \lambda)$
to learn $h_\psi(x)$ for $k(h_\psi(x), h_\psi(y))$

vector field movie, use Acrobat Reader to play
Simple 2-D example revisited

What the kernels $k(h_{\psi}(x), h_{\psi}(y))$ look like

isolines movie, use Acrobat Reader to play
Our empirical observations

Data-adaptive critic loss:

- Witness function class for $SMMD(P, \lambda)$ depends on $P$.
  - Without data-dependent regularisation, maximising MMD over features $h_\psi$ of kernel $k(h_\psi(x), h_\psi(y))$ can be unhelpful.
  - WGAN-GP is a pretty good data-dependent regularisation strategy.

- Similar regularisation strategies apply to variational form in f-GANs.

Roth et al [NeurIPS 2017, eq. 19 and 20]
Our empirical observations

Data-adaptive critic loss:

- Witness function class for $SMMD(P, \lambda)$ depends on $P$.
  - Without data-dependent regularisation, maximising MMD over features $h_\psi$ of kernel $k(h_\psi(x), h_\psi(y))$ can be unhelpful.
  - WGAN-GP is a pretty good data-dependent regularisation strategy.

- Similar regularisation strategies apply to variational form in f-GANs
  Roth et al [NeurIPS 2017, eq. 19 and 20]

Alternate critic and generator training:

- Weaker critics can give better signals to poor (early stage) generators.
- Incomplete training of the critic is also a regularisation strategy.
Don’t just use gradient regularizers!

Spectral norm regularizer (effectively smooths critic class; ICLR 2018):

**Spectral Normalization**

for Generative Adversarial Networks

Takeru Miyato¹, Toshiki Kataoka¹, Masanori Koyama², Yuichi Yoshida³

¹Preferred Networks, Inc. ²Ritsumeikan University ³National Institute of Informatics

Entropic regularizer (avoid mode collapse):


Statistics > Machine Learning

[Submitted on 9 Oct 2019]

Prescribed Generative Adversarial Networks

Adji B. Dieng, Francisco J. R. Ruiz, David M. Blei, Michalis K. Titsias
Evaluation and experiments
Benchmarks for comparison (all from ICLR 2018)

SPECTRAL NORMALIZATION FOR GENERATIVE ADVERSARIAL NETWORKS

Takeru Miyato¹, Toshiki Kataoka¹, Masanori Koyama², Yuichi Yoshida³
{miyato, kataoka}@preferred.jp
koyama.masanori1@gmail.com
yoshida@iizuka.ac.jp
²Ritsumeikan University ³National Institute of Informatics

SOBOLEV GAN

Youssef Mroueh¹, Chun-Liang Li²¹, Tom Sercu¹¹, Anant Raj⁹¹, & Yu Cheng¹
⟩ IBM Research AI
i Carnegie Mellon University
į Max Planck Institute for Intelligent Systems
# denotes Equal Contribution
{mroueh,chengyu}@us.ibm.com, chunlial@cs.cmu.edu,
tom.sercul@ibm.com,anant.raj@tuebingen.mpg.de

DEMYSTIFYING MMD GANs

Mikołaj Bifkowksi
Department of Mathematics
Imperial College London
mikbinkowski@gmail.com

BOUNDARY-SEEKING GENERATIVE ADVERSARIAL NETWORKS

R Devon Hjelm³
MILA, University of Montréal, IVADO
eronneus@gmail.com

Dougal J. Sutherland, Michael Arbel & Arthur Gretton
Gatsby Computational Neuroscience Unit
University College London
dougal@cs.gatsby.ucl.ac.uk, michael.n.arbel,arthur.gretton@gmail.com

Athul Paul Jacob³
MILA, MSR, University of Waterloo
apjacob@edu.uwaterloo.ca

Our ICLR 2018 paper

We combine with scaled MMD

67/76
Results: unconditional imagenet $64 \times 64$

KID scores:

- **BGAN:**
  47

- **SN-GAN:**
  44

- **SMMD GAN:**
  35

ILSVRC2012 (ImageNet) dataset, 1,281,167 images, resized to $64 \times 64$. 1000 classes.
Results: unconditional imagenet 64×64

KID scores:

- **BGAN:**
  47

- **SN-GAN:**
  44

- **SMMD GAN:**
  35

ILSVRC2012 (ImageNet) dataset, 1 281 167 images, resized to 64 × 64. 1000 classes.
Results: unconditional imagenet 64×64

KID scores:

- **BGAN:**
  47
- **SN-GAN:**
  44
- **SMMD GAN:**
  35

ILSVRC2012 (ImageNet) dataset, 1 281 167 images, resized to 64 × 64. 1000 classes.
Summary

- GAN critics rely on two sources of regularisation
  - Regularisation by incomplete training
  - Data-dependent gradient regulariser
- Some advantages of hybrid kernel/neural features:
  - MMD loss still a valid critic when features not optimal (unlike WGAN-GP)
  - Kernel features do some of the “work”, so simpler \( h_\psi \) features possible.

“Demystifying MMD GANs,” including KID score, ICLR 2018: https://github.com/mbinkowski/MMD-GAN
Gradient regularised MMD, NeurIPS 2018: https://github.com/MichaelArbel/Scaled-MMD-GAN
Post-credit scene: Generalised Energy-Based Models


Statistics > Machine Learning

[Submitted on 10 Mar 2020 (v1), last revised 24 Jun 2020 (this version, v3)]

Generalized Energy Based Models

Michael Arbel, Liang Zhou, Arthur Gretton

https://github.com/MichaelArbel/GeneralizedEBM
Critic features from DCGAN: an $f$-filter critic has $f$, $2f$, $4f$ and $8f$ convolutional filters in layers 1-4. LSUN $64 \times 64$. 

\[ k(h_\psi(x), h_\psi(y)), f = 64, \text{ KID}=3 \]

\[ h_\psi^\top(x)h_\psi(y), f = 64, \text{ KID}=4 \]
Linear vs nonlinear kernels

- **Critic features from DCGAN**: an $f$-filter critic has $f$, $2f$, $4f$, and $8f$ convolutional filters in layers 1-4. LSUN $64 \times 64$.

\[
k(h_\psi(x), h_\psi(y)), \quad f = 16, \quad \text{KID}=9
\]

\[
h_\psi^\top(x)h_\psi(y), \quad f = 16, \quad \text{KID}=37
\]
Evaluation of GANs

The inception score?  Salimans et al. [NeurIPS 2016]

Based on the classification output $p(y|x)$ of the inception model Szegedy et al. [ICLR 2014],

$$E_X \exp KL(P(y|X) || P(y)).$$

High when:
- predictive label distribution $P(y|x)$ has low entropy (good quality images)
- label entropy $P(y)$ is high (good variety).
Evaluation of GANs

The inception score? Salimans et al. [NeurIPS 2016]

Based on the classification output $p(y|x)$ of the inception model Szegedy et al. [ICLR 2014],

$$E_X \exp KL(P(y|X)\|P(y)).$$

High when:

- predictive label distribution $P(y|x)$ has low entropy (good quality images)
- label entropy $P(y)$ is high (good variety).

Problem: relies on a trained classifier! Can’t be used on new categories (celeb, bedroom...).
Evaluation of GANs

The Frechet inception distance? Heusel et al. [NeurIPS 2017]

Fits **Gaussians** to features in the inception architecture (pool3 layer):

\[
FID(P, Q) = \|\mu_P - \mu_Q\|^2 + \text{tr}(\Sigma_P) + \text{tr}(\Sigma_Q) - 2\text{tr} \left( (\Sigma_P \Sigma_Q)^{1/2} \right)
\]

where \(\mu_P\) and \(\Sigma_P\) are the feature mean and covariance of \(P\).
Evaluation of GANs

The Frechet inception distance? Heusel et al. [NeurIPS 2017]

Fits Gaussians to features in the inception architecture (pool3 layer):

\[ FID(P, Q) = ||\mu_P - \mu_Q||^2 + \text{tr}(\Sigma_P) + \text{tr}(\Sigma_Q) - 2\text{tr}\left( (\Sigma_P \Sigma_Q)^{\frac{1}{2}} \right) \]

where \( \mu_P \) and \( \Sigma_P \) are the feature mean and covariance of \( P \)

Problem: bias. For finite samples can consistently give incorrect answer.

- Bias demo,
  CIFAR-10 train vs test
Evaluation of GANs

The FID can give the wrong answer in theory.

Assume $m$ samples from $P$ and $n \to \infty$ samples from $Q$.

Given two alternatives:

$$P_1 \sim \mathcal{N}(0, (1 - m^{-1})^2) \quad P_2 \sim \mathcal{N}(0, 1) \quad Q \sim \mathcal{N}(0, 1).$$

Clearly,

$$FID(P_1, Q) = \frac{1}{m^2} > FID(P_2, Q) = 0.$$

Given $m$ samples from $P_1$ and $P_2$, 

$$FID(\hat{P}_1, Q) < FID(\hat{P}_2, Q).$$
Evaluation of GANs

The FID can give the wrong answer in theory.
Assume $m$ samples from $P$ and $n \to \infty$ samples from $Q$.
Given two alternatives:

\[
P_1 \sim \mathcal{N}(0, (1 - m^{-1})^2) \quad P_2 \sim \mathcal{N}(0, 1) \quad Q \sim \mathcal{N}(0, 1).
\]

Clearly,

\[
FID(P_1, Q) = \frac{1}{m^2} > FID(P_2, Q) = 0
\]

Given $m$ samples from $P_1$ and $P_2$,

\[
FID(\hat{P}_1, Q) < FID(\hat{P}_2, Q).
\]
Evaluation of GANs

The FID can give the wrong answer in theory.
Assume $m$ samples from $P$ and $n \rightarrow \infty$ samples from $Q$.
Given two alternatives:

$$P_1 \sim \mathcal{N}(0, (1 - m^{-1})^2) \quad P_2 \sim \mathcal{N}(0, 1) \quad Q \sim \mathcal{N}(0, 1).$$

Clearly,

$$FID(P_1, Q) = \frac{1}{m^2} > FID(P_2, Q) = 0$$

Given $m$ samples from $P_1$ and $P_2$,

$$FID(\hat{P}_1, Q) < FID(\hat{P}_2, Q).$$
Evaluation of GANs

The FID can give the wrong answer in theory.
Assume $m$ samples from $P$ and $n \to \infty$ samples from $Q$.
Given two alternatives:

$$P_1 \sim \mathcal{N}(0, (1 - m^{-1})^2) \quad P_2 \sim \mathcal{N}(0, 1) \quad Q \sim \mathcal{N}(0, 1).$$

Clearly,

$$FID(P_1, Q) = \frac{1}{m^2} > FID(P_2, Q) = 0$$

Given $m$ samples from $P_1$ and $P_2$,

$$FID(\hat{P}_1, Q) < FID(\hat{P}_2, Q).$$
Evaluation of GANs

The FID can give the wrong answer in practice.

Let $d = 2048$, and define

$$P_1 = \text{relu}(\mathcal{N}(0, I_d)) \quad P_2 = \text{relu}(\mathcal{N}(1, .8\Sigma+.2I_d)) \quad Q = \text{relu}(\mathcal{N}(1, I_d))$$

where $\Sigma = \frac{4}{d} CC^T$, with $C$ a $d \times d$ matrix with iid standard normal entries.

For a random draw of $C$:

$$FID(P_1, Q) \approx 1123.0 > 1114.8 \approx FID(P_2, Q)$$

With $m = 50000$ samples,

$$FID(\hat{P}_1, Q) \approx 1133.7 < 1136.2 \approx FID(\hat{P}_2, Q)$$

At $m = 100000$ samples, the ordering of the estimates is correct.

This behavior is similar for other random draws of $C$. 
**Evaluation of GANs**

The FID can give the **wrong answer in practice.**

Let $d = 2048$, and define

$$P_1 = \text{relu}(\mathcal{N}(0, I_d)) \quad P_2 = \text{relu}(\mathcal{N}(1, .8\Sigma+.2I_d)) \quad Q = \text{relu}(\mathcal{N}(1, I_d))$$

where $\Sigma = \frac{4}{d} C C^T$, with $C$ a $d \times d$ matrix with iid standard normal entries.

For a random draw of $C$:

$$FID(P_1, Q) \approx 1123.0 > 1114.8 \approx FID(P_2, Q)$$

With $m = 50000$ samples,

$$FID(\hat{P}_1, Q) \approx 1133.7 < 1136.2 \approx FID(\hat{P}_2, Q)$$

At $m = 100000$ samples, the ordering of the estimates is correct. This behavior is similar for other random draws of $C$. 
Evaluation of GANs

The FID can give the wrong answer in practice.
Let \( d = 2048 \), and define

\[
P_1 = \text{relu}(\mathcal{N}(0, I_d)) \quad P_2 = \text{relu}(\mathcal{N}(1, .8\Sigma + .2I_d)) \quad Q = \text{relu}(\mathcal{N}(1, I_d))
\]

where \( \Sigma = \frac{4}{d} CC^T \), with \( C \) a \( d \times d \) matrix with iid standard normal entries.

For a random draw of \( C \):

\[
FID(P_1, Q) \approx 1123.0 > 1114.8 \approx FID(P_2, Q)
\]

With \( m = 50000 \) samples,

\[
FID(\hat{P}_1, Q) \approx 1133.7 < 1136.2 \approx FID(\hat{P}_2, Q)
\]

At \( m = 100000 \) samples, the ordering of the estimates is correct.
This behavior is similar for other random draws of \( C \).
Evaluation of GANs

The FID can give the wrong answer in practice.

Let $d = 2048$, and define

\[ P_1 = \text{relu}(\mathcal{N}(0, I_d)) \quad P_2 = \text{relu}(\mathcal{N}(1, .8\Sigma + .2I_d)) \quad Q = \text{relu}(\mathcal{N}(1, I_d)) \]

where $\Sigma = \frac{4}{d} CC^T$, with $C$ a $d \times d$ matrix with iid standard normal entries.

For a random draw of $C$:

\[ FID(P_1, Q) \approx 1123.0 > 1114.8 \approx FID(P_2, Q) \]

With $m = 50000$ samples,

\[ FID(\hat{P}_1, Q) \approx 1133.7 < 1136.2 \approx FID(\hat{P}_2, Q) \]

At $m = 100000$ samples, the ordering of the estimates is correct.

This behavior is similar for other random draws of $C$. 
The kernel inception distance (KID)

The Kernel inception distance Binkowski, Sutherland, Arbel, G. [ICLR 2018]

Measures similarity of the samples’ representations in the inception architecture (pool3 layer)

MMD with kernel

\[ k(x, y) = \left( \frac{1}{d} x^\top y + 1 \right)^3. \]

- Checks match for feature means, variances, skewness
- Unbiased: eg CIFAR-10 train/test
The kernel inception distance (KID)

The Kernel inception distance Binkowski, Sutherland, Arbel, G. [ICLR 2018]
Measures similarity of the samples’ representations in the inception architecture (pool3 layer)

\[ k(x, y) = \left( \frac{1}{d} x^\top y + 1 \right)^3. \]

- Checks match for feature means, variances, skewness
- Unbiased: eg CIFAR-10 train/test

...“but isn’t KID is computationally costly?”
The kernel inception distance (KID)

The Kernel inception distance Binkowski, Sutherland, Arbel, G. [ICLR 2018]

Measures similarity of the samples’ representations in the inception architecture (pool3 layer)

MMD with kernel

\[ k(x, y) = \left( \frac{1}{d} x^\top y + 1 \right)^3. \]

- Checks match for feature means, variances, skewness
- Unbiased: eg CIFAR-10 train/test

...“but isn’t KID is computationally costly?”

“Block” KID implementation is cheaper than FID: see paper (or use Tensorflow implementation)!
The kernel inception distance (KID)

The Kernel inception distance Binkowski, Sutherland, Arbel, G. [ICLR 2018]

Measures similarity of the samples’ representations in the inception architecture (pool3 layer)

MMD with kernel

$$k(x, y) = \left( \frac{1}{d} x^\top y + 1 \right)^3.$$  

- Checks match for feature means, variances, skewness
- **Unbiased**: eg CIFAR-10 train/test

Also used for automatic learning rate adjustment: if $KID(\hat{P}_{t+1}, Q)$ not significantly better than $KID(\hat{P}_t, Q)$ then reduce learning rate.

[Bounliphone et al. ICLR 2016]

Related: “An empirical study on evaluation metrics of generative adversarial networks”, Xu et al. [arxiv, June 2018]