Learning features to compare distributions

Arthur Gretton

Gatsby Computational Neuroscience Unit, University College London

NIPS 2016 Workshop on Adversarial Learning, Barcelona Spain
Goal of this talk

- **Have:** Two collections of samples $X, Y$ from unknown distributions $P$ and $Q$.
- **Goal:** Learn distinguishing features that indicate how $P$ and $Q$ differ.
Goal of this talk

- **Have:** Two collections of samples $X$, $Y$ from unknown distributions $P$ and $Q$.
- **Goal:** Learn distinguishing features that indicate how $P$ and $Q$ differ.
Divergences
Divergences

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

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

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

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

Integral prob. metrics

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

MMD

F-divergences

Hellinger

KL

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

Pearson chi^2
Divergences

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

\[ D_f(P, Q) = \int_{\mathcal{X}} q(x) f \left( \frac{p(x)}{q(x)} \right) \, dx \]

Sriperumbudur, Fukumizu, G, Schoelkopf, Lanckriet (2012)
Overview

The Maximum mean discrepancy:
- How to compute and interpret the MMD
- How to train the MMD
- Application to troubleshooting GANs

The ME test statistic:
- Informative, linear time features for comparing distributions
- How to learn these features

TL;DR: Variance matters.
The maximum mean discrepancy

Are \( P \) and \( Q \) different?

\[
\begin{align*}
P(x) & \\
Q(y) & 
\end{align*}
\]
Maximum mean discrepancy (on sample)
Observe $X = \{x_1, \ldots, x_n\} \sim P$

Observe $Y = \{y_1, \ldots, y_n\} \sim Q$
Maximum mean discrepancy (on sample)

Gaussian kernel on $x_i$

Gaussian kernel on $y_i$
Maximum mean discrepancy (on sample)

\[ \hat{\mu}_P(v) := \frac{1}{m} \sum_{i=1}^{m} k(x_i, v) \]

\[ \hat{\mu}_Q(v) : \text{mean embedding of } Q \]

\[ \hat{\mu}_P(v) : \text{mean embedding of } P \]
Maximum mean discrepancy (on sample)

\[ \hat{\mu}_P(v) : \text{mean embedding of } P \]

\[ \hat{\mu}_Q(v) : \text{mean embedding of } Q \]

\[ \text{witness}(v) = \hat{\mu}_P(v) - \hat{\mu}_Q(v) \]
Maximum mean discrepancy (on sample)

\[ \overline{\text{MMD}}^2 = \| \text{witness}(v) \|_F^2 \]

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

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

- 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 $k(\text{fish}_i, \text{fish}_j)$
Overview

The maximum mean discrepancy:

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

$$- \frac{2}{n^2} \sum_{i,j} k(\text{dog}_i, \text{fish}_j)$$
Asymptotics of MMD

- The MMD:

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

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

but how to choose the kernel?
Asymptotics of MMD

- The MMD:

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

but how to choose the kernel?

- Perspective from statistical hypothesis testing:
  - When \( P = Q \) then \( \hat{\text{MMD}}^2 \) “close to zero”.
  - When \( P \neq Q \) then \( \hat{\text{MMD}}^2 \) “far from zero”

- Threshold \( c_\alpha \) for \( \hat{\text{MMD}}^2 \) gives false positive rate \( \alpha \)
A statistical test

- \( n \times \hat{\text{MMD}}^2 \)

- \( P=Q \) vs. \( P \neq Q \)

- False negatives: \( c_\alpha = 1-\alpha \) quantile when \( P=Q \)

- Best kernel gives lowest false negative rate (=highest power)

- Can you train for this?
A statistical test

Best kernel gives lowest false negative rate (=highest power)
Best kernel gives lowest false negative rate (highest power)

.... but can you train for this?
Asymptotics of MMD

When \( P \neq Q \), statistic is asymptotically normal,

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

where \( \text{MMD}(P, Q) \) is population MMD, and \( V_n(P, Q) = O(n^{-1}) \).

![MMD distribution and Gaussian fit under H1](image-url)
Asymptotics of MMD

Where $P = Q$, statistic has asymptotic distribution

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

where

$$\lambda_l \psi_l(x') = \int \chi $$

and

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

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

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

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

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

$$\rightarrow 1 - \Phi \left( \frac{c_\alpha}{n \sqrt{V_n(P, Q)}} - \frac{\text{MMD}^2(P, Q)}{\sqrt{V_n(P, Q)}} \right)$$

where

- $\Phi$ is the CDF of the standard normal distribution.
- $\hat{c}_\alpha$ is an estimate of $c_\alpha$ test threshold.
Optimizing test power

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

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

$$\rightarrow 1 - \Phi \left( \frac{c_\alpha}{n \sqrt{V_n(P, Q)}} - \frac{\text{MMD}^2(P, Q)}{\sqrt{V_n(P, Q)}} \right)$$

First term asymptotically negligible!
Optimizing test power

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

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

$$\rightarrow 1 - \Phi \left( \frac{c_\alpha}{n \sqrt{V_n(P, Q)}} - \frac{\text{MMD}^2(P, Q)}{\sqrt{V_n(P, Q)}} \right)$$

To maximize test power, maximize

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

(Sutherland, Tung, Strathmann, De, Ramdas, Smola, G., in review for ICLR 2017)

Code: github.com/dougalsutherland/opt-mmd
Troubleshooting for generative adversarial networks

MNIST samples

Samples from a GAN
Troubleshooting for generative adversarial networks

MNIST samples

Samples from a GAN

- Power for optimized ARD kernel: 1.00 at $\alpha = 0.01$
- Power for optimized RBF kernel: 0.57 at $\alpha = 0.01$
Benchmarking generative adversarial networks

\[ \text{MMD}^2 = 0.0001 \]
The ME statistic and test
Distinguishing Feature(s)

\[ \hat{\mu}_P(v) : \text{mean embedding of } P \]

\[ \hat{\mu}_Q(v) : \text{mean embedding of } Q \]

\[ \text{witness}(v) = \hat{\mu}_P(v) - \hat{\mu}_Q(v) \]
Distinguishing Feature(s)

Take square of witness (only worry about amplitude)
Distinguishing Feature(s)

- New test statistic: \( \text{witness}^2 \) at a single \( v^* \);
- Linear time in number \( n \) of samples
- ....but how to choose best feature \( v^* \)?
Distinguishing Feature(s)

Best feature = \( v^* \) that maximizes \( \text{witness}^2(v) \)??
Distinguishing Feature(s)

Sample size \( n = 3 \)
Distinguishing Feature(s)

Sample size $n = 50$
Distinguishing Feature(s)

Sample size $n = 500$
Distinguishing Feature(s)

\[ P(x) \]
\[ Q(y) \]
\[ \text{witness}^2(v) \]

Population witness\(^2\) function
Distinguishing Feature(s)

\[ P(x) \]
\[ Q(y) \]
\[ \text{witness}^2(v) \]
Variance of witness function

- Variance at $v = \text{variance of } X \text{ at } v + \text{variance of } Y \text{ at } v$.
- ME Statistic: $\hat{\lambda}_n(v) := n \frac{\text{witness}^2(v)}{\text{variance of } v}$.

Best location is $v$ that maximizes $\hat{\lambda}_n(v)$. Improve performance using multiple locations $f_j v \in \mathcal{J} j = 1$. 

25/28
Variance of witness function

- Variance at $v = \text{variance of } X \text{ at } v + \text{variance of } Y \text{ at } v$.
- ME Statistic: $\hat{\lambda}_n(v) := n \frac{\text{witness}^2(v)}{\text{variance of } v}$. 

Best location is $v$ that maximizes $\hat{\lambda}_n(v)$. 

Improve performance using multiple locations $f_{\mathbf{v}}_{j\in J}$.
Variance of witness function

- Variance at $v = \text{variance of } X \text{ at } v + \text{variance of } Y \text{ at } v$.
- ME Statistic: $\hat{\lambda}_n(v) := n \frac{\text{witness}^2(v)}{\text{variance of } v}$.

Best location is $v$ that maximizes $\hat{\lambda}_n(v)$.

Improve performance using multiple locations $f_{v_j} = g_{v_j}$.
Variance of witness function

- Variance at $v = \text{variance of } X \text{ at } v + \text{variance of } Y \text{ at } v$.
- ME Statistic: $\hat{\lambda}_n(v) := n \frac{\text{witness}^2(v)}{\text{variance of } v}$.

Best location is $v$ that maximizes $\hat{\lambda}_n(v)$.

Improve performance using multiple locations $f(v) \forall v \in J = 1, 2, 5, 25/28$.
Variance of witness function

- Variance at $v = \text{variance of } X \text{ at } v + \text{variance of } Y \text{ at } v$.
- ME Statistic: $\hat{\lambda}_n(v) := n \frac{\text{witness}^2(v)}{\text{variance of } v}$. 

Best location is $v$ that maximizes $\hat{\lambda}_n(v)$.

Improve performance using multiple locations $f_{v_j}$ for $j = 1, 2, \ldots, 25/28$. 

![Graph showing the witness function and variance function with the blue box highlighting the best location](image-url)
Variance of witness function

- Variance at \( v \) = variance of \( X \) at \( v \) + variance of \( Y \) at \( v \).
- ME Statistic: \( \hat{\lambda}_n(v) := n \frac{\text{witness}^2(v)}{\text{variance of } v} \).
Variance of witness function

- Variance at $v = \text{variance of } X \text{ at } v + \text{variance of } Y \text{ at } v$.
- ME Statistic: $\hat{\lambda}_n(v) := n \frac{\text{witness}^2(v)}{\text{variance of } v}$.

Best location is $v^*$ that maximizes $\hat{\lambda}_n$.

Improve performance using multiple locations $\{v^*_j\}_{j=1}^J$. 
Distinguishing Positive/Negative Emotions

+:
- happy neutral surprised
- afraid angry disgusted

- 35 females and 35 males
  (Lundqvist et al., 1998).
- $48 \times 34 = 1632$ dimensions.
  Pixel features.
- Sample size: 402.

- The proposed test achieves maximum test power in time $O(n)$.
- Informative features: differences at the nose, and smile lines.
Distinguishing Positive/Negative Emotions

+ : happy neutral surprised

− : afraid angry disgusted

The proposed test achieves maximum test power in time $O(n)$.

Informative features: differences at the nose, and smile lines.
Distinguishing Positive/Negative Emotions

+ : happy neutral surprised

- : afraid angry disgusted

The proposed test achieves maximum test power in time $O(n)$.

Informative features: differences at the nose, and smile lines.
Distinguishing Positive/Negative Emotions

+ : happy neutral surprised

− : afraid angry disgusted

- The proposed test achieves **maximum test power** in time $O(n)$.
- Informative features: differences at the nose, and smile lines.
Distinguishing Positive/Negative Emotions

+ : happy neutral surprised

- : afraid angry disgusted

Learned feature

- The proposed test achieves **maximum test power** in time $O(n)$.
- **Informative features**: differences at the nose, and smile lines.
Distinguishing Positive/Negative Emotions

+ : happy neutral surprised
- : afraid angry disgusted

Learned feature

- The proposed test achieves **maximum test power** in time $O(n)$.
- **Informative features**: differences at the nose, and smile lines.

Code: https://github.com/wittawatj/interpretable-test
Final thoughts

Witness function approaches:

- **Diversity of samples:**
  - MMD test uses pairwise similarities between all samples
  - ME test uses similarities to $J$ reference features

- **Disjoint support of generator/data distributions**
  - Witness function is smooth

Other discriminator heuristics:

- **Diversity of samples** by minibatch heuristic (add as feature distances to neighbour samples) Salimans et al. (2016)

- **Disjoint support** treated by adding noise to “blur” images Arjovsky and Bottou (2016), Sønderby et al (2016)
Co-authors

Students and postdocs:
- Kacper Chwialkowski (at Voleon)
- Wittawat Jitkrittum
- Heiko Strathmann
- Dougal Sutherland

Collaborators
- Kenji Fukumizu
- Krikamol Muandet
- Bernhard Schoelkopf
- Bharath Sriperumbudur
- Zoltan Szabo

Questions?
Testing against a probabilistic model
Statistical model criticism

\[ MMD(P, Q) = \|f^*\|^2 = \sup_{\|f\|_{\mathcal{F}} \leq 1}[E_Q f - E_P f] \]

\( f^*(x) \) is the witness function

Can we compute MMD with samples from \( Q \) and a model \( P \)?

**Problem:** usually can’t compute \( E_P f \) in closed form.
To get rid of $E_p f$ in

$$\sup_{\|f\|_{\mathcal{F}} \leq 1} \left[ E_q f - E_p f \right]$$

we define the **Stein operator**

$$T_p f = \partial_x f + f \left( \partial_x \log p \right)$$

Then

$$E_p T_p f = 0$$

subject to appropriate boundary conditions. (Oates, Girolami, Chopin, 2016)
Maximum Stein Discrepancy

Stein operator

\[ T_p f = \partial_x f + f \partial_x (\log p) \]

Maximum Stein Discrepancy (MSD)

\[ MSD(p, q, \mathcal{F}) = \sup_{\|g\|_{\mathcal{F}} \leq 1} E_q T_p g - E_p T_p g \]
Maximum Stein Discrepancy

Stein operator

\[ T_p f = \partial_x f + f \partial_x (\log p) \]

Maximum Stein Discrepancy (MSD)

\[ MSD(p, q, \mathcal{F}) = \sup \| g \|_{\mathcal{F}} \leq 1 \quad E_q T_p g - E_p T_p g \]
Maximum Stein Discrepancy

Stein operator

\[ T_p f = \partial_x f + f \partial_x (\log p) \]

Maximum Stein Discrepancy (MSD)

\[ MSD(p, q, \mathcal{F}) = \sup_{\|g\|_{\mathcal{F}} \leq 1} E_q T_p g - E_p T_p g = \sup_{\|g\|_{\mathcal{F}} \leq 1} E_q T_p g \]
Maximum Stein Discrepancy

Stein operator

\[ T_p f = \partial_x f + f \partial_x (\log p) \]

Maximum Stein Discrepancy (MSD)

\[
MSD(p, q, \mathcal{F}) = \sup_{\|g\|_{\mathcal{F}} \leq 1} E_q T_p g - E_p T_p g = \sup_{\|g\|_{\mathcal{F}} \leq 1} E_q T_p g
\]
Maximum Stein Discrepancy

Stein operator

\[ T_p f = \partial_x f + f \partial_x (\log p) \]

Maximum Stein Discrepancy (MSD)

\[ MSD(p, q, F) = \sup_{\|g\|_F \leq 1} E_q T_p g - E_p T_p g = \sup_{\|g\|_F \leq 1} E_q T_p g \]
Maximum stein discrepancy

Closed-form expression for MSD: given $Z, Z' \sim q$, then (Chwialkowski, Strathmann, G., 2016) (Liu, Lee, Jordan 2016)

$$MSD(p, q, \mathcal{F}) = E_q h_p(Z, Z')$$

where

$$h_p(x, y) := \partial_x \log p(x) \partial_x \log p(y) k(x, y)$$
$$+ \partial_y \log p(y) \partial_x k(x, y)$$
$$+ \partial_x \log p(x) \partial_y k(x, y)$$
$$+ \partial_x \partial_y k(x, y)$$

and $k$ is RKHS kernel for $\mathcal{F}$

Only depends on kernel and $\partial_x \log p(x)$. Do not need to normalize $p$, or sample from it.
Statistical model criticism

Test the hypothesis that a Gaussian process model, learned from data *, is a good fit for the test data (example from Lloyd and Ghahramani, 2015)

Code: https://github.com/karlnapf/kernel_goodness_of_fit
Test the hypothesis that a Gaussian process model, learned from data *, is a good fit for the test data.