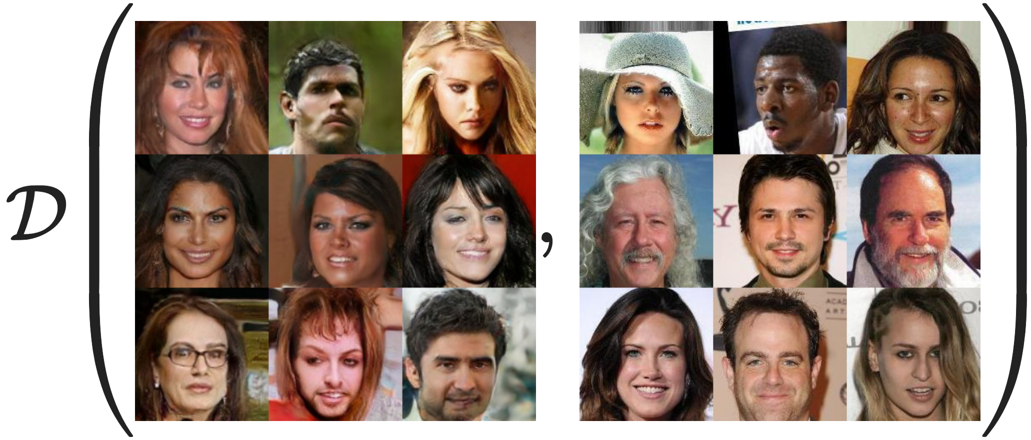
Deep kernel-based distances between distributions

Danica J. Sutherland

Based on work with:

Michael Arbel, Mikołaj Bińkowski, Soumyajit De,
Arthur Gretton, Feng Liu, Jie Lu,
Aaditya Ramdas, Alex Smola, Heiko Strathmann,
Hsiao-Yu (Fish) Tung, Wenkai Xu, Guangquan Zhang



PIHOT kick-off, 30 Jan 2021

What's a kernel again?

- Linear classifiers: $\hat{y}(x) = \mathrm{sign}(f(x))$, $f(x) = w^\mathsf{T}\,(x, 1)$

What's a kernel again?

- Linear classifiers: $\hat{y}(x) = \mathrm{sign}(f(x))$, $f(x) = w^{\mathsf{T}}(x, 1)$

What's a kernel again?

- Linear classifiers: $\hat{y}(x) = \operatorname{sign}(f(x))$, $f(x) = w^\mathsf{T}(x, 1)$

What's a kernel again?

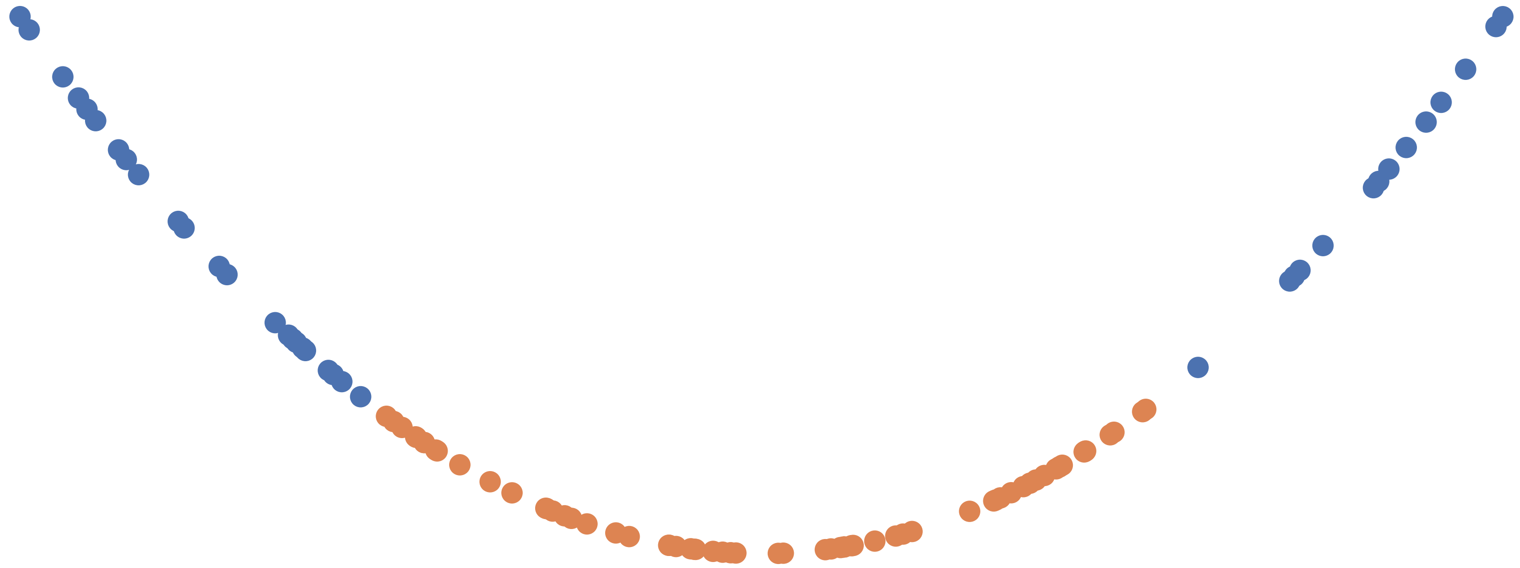
- Linear classifiers: $\hat{y}(x) = \operatorname{sign}(f(x))$, $f(x) = w^\mathsf{T}(x, 1)$
- Use a "richer" x:

$$f(x) = w^\mathsf{T}\left(x, x^2, 1\right) = w^\mathsf{T}\phi(x)$$

What's a kernel again?

- Linear classifiers: $\hat{y}(x) = \operatorname{sign}(f(x))$, $f(x) = w^\mathsf{T} (x, 1)$
- Use a "richer" x:

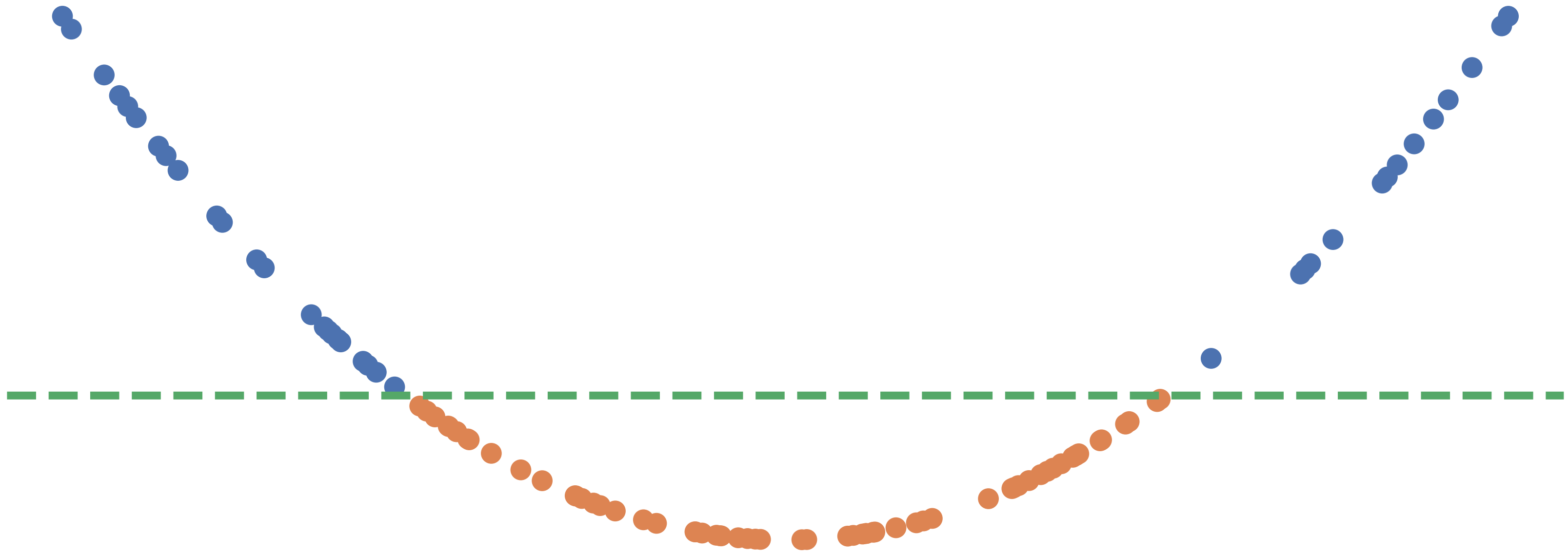
$$f(x) = w^\mathsf{T} \left(x, x^2, 1\right) = w^\mathsf{T} \phi(x)$$



What's a kernel again?

- Linear classifiers: $\hat{y}(x) = \operatorname{sign}(f(x))$, $f(x) = w^\mathsf{T}(x, 1)$
- Use a "richer" x:

$$f(x) = w^\mathsf{T}\left(x, x^2, 1\right) = w^\mathsf{T}\phi(x)$$



What's a kernel again?

- Linear classifiers: $\hat{y}(x) = \operatorname{sign}(f(x))$, $f(x) = w^\mathsf{T} (x, 1)$
- Use a “richer” x:

$$f(x) = w^\mathsf{T} \left(x, x^2, 1\right) = w^\mathsf{T} \phi(x)$$

- Can avoid explicit $\phi(x)$; instead $k(x, y) = \langle \phi(x), \phi(y) \rangle_{\mathcal{H}}$

What's a kernel again?

- Linear classifiers: $\hat{y}(x) = \operatorname{sign}(f(x))$, $f(x) = w^\mathsf{T} \left(x, 1\right)$
- Use a “richer” x:

$$f(x) = w^\mathsf{T} \left(x, x^2, 1\right) = w^\mathsf{T} \phi(x)$$

- Can avoid explicit $\phi(x)$; instead $k(x, y) = \langle \phi(x), \phi(y) \rangle_{\mathcal{H}}$
- “Kernelized” algorithms access data only through $k(x, y)$

$$f(x) = \langle w, \phi(x) \rangle_{\mathcal{H}} = \sum_{i=1}^{n} \alpha_i k(X_i, x)$$

Reproducing Kernel Hilbert Space (RKHS)

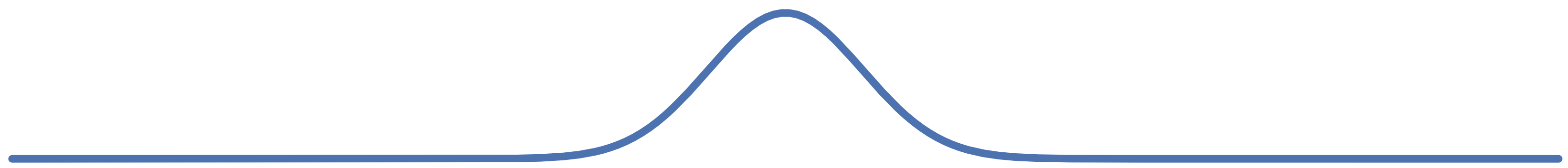
- Ex: Gaussian RBF

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

Reproducing Kernel Hilbert Space (RKHS)

- Ex: Gaussian RBF

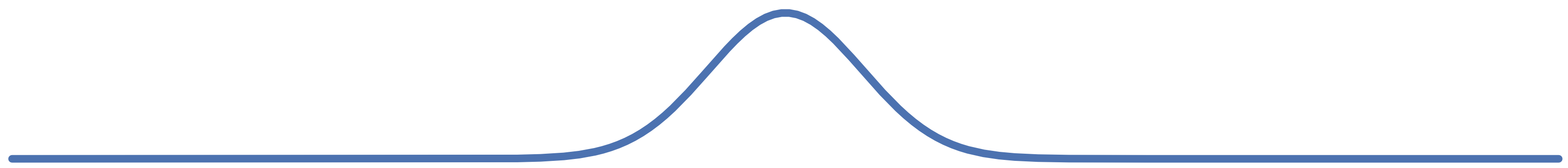
$$k(x, y) = \exp\left(-\frac{\|x - y\|^2}{2\sigma^2}\right)$$



Reproducing Kernel Hilbert Space (RKHS)

- Ex: Gaussian RBF / exponentiated quadratic / squared exponential / ...

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

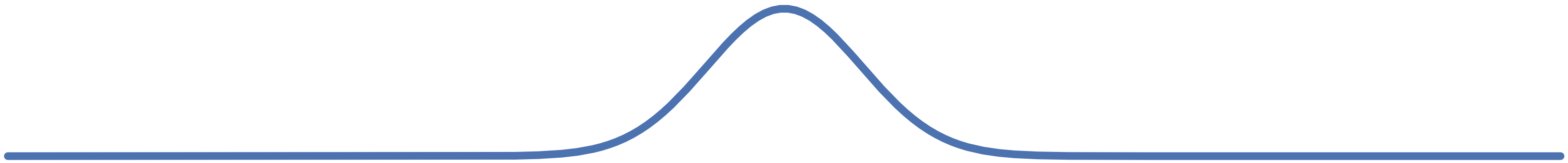


Reproducing Kernel Hilbert Space (RKHS)

- Ex: Gaussian RBF / exponentiated quadratic / squared exponential / …

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

- Reproducing property: $\langle f, \phi(x) \rangle_{\mathcal{H}} = f(x)$ for $f \in \mathcal{H}$

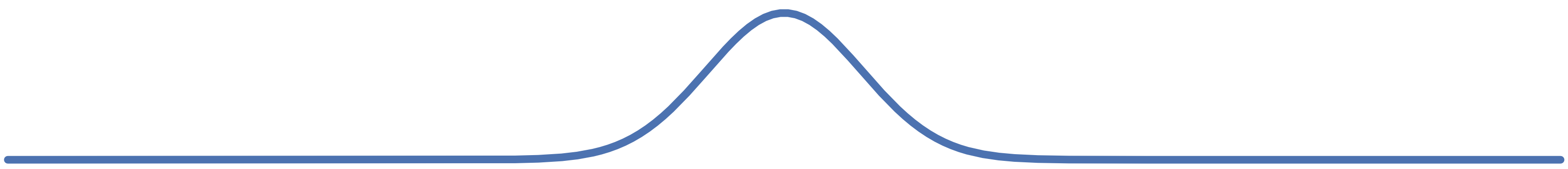


Reproducing Kernel Hilbert Space (RKHS)

- Ex: Gaussian RBF / exponentiated quadratic / squared exponential / …

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

- Reproducing property: $\langle f, \phi(x) \rangle_{\mathcal{H}} = f(x)$ for $f \in \mathcal{H}$
- $\mathcal{H} = \mathrm{cl}(\{\sum_{i=1}^{n} \alpha_i \phi(X_i) \mid n \geq 0, \alpha \in \mathbb{R}^n, X_i \in \mathcal{X}\})$



Reproducing Kernel Hilbert Space (RKHS)

- Ex: Gaussian RBF / exponentiated quadratic / squared exponential / …

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

- Reproducing property: $\langle f, \phi(x) \rangle_{\mathcal{H}} = f(x)$ for $f \in \mathcal{H}$
- $\mathcal{H} = \mathrm{cl}(\{\sum_{i=1}^n \alpha_i \phi(X_i) \mid n \geq 0, \alpha \in \mathbb{R}^n, X_i \in \mathcal{X}\})$

Reproducing Kernel Hilbert Space (RKHS)

- Ex: Gaussian RBF / exponentiated quadratic / squared exponential / ...

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

- Reproducing property: $\langle f, \phi(x) \rangle_{\mathcal{H}} = f(x)$ for $f \in \mathcal{H}$
- $\mathcal{H} = \operatorname{cl}(\{\sum_{i=1}^{n} \alpha_i \phi(X_i) \mid n \geq 0, \alpha \in \mathbb{R}^n, X_i \in \mathcal{X}\})$

Reproducing Kernel Hilbert Space (RKHS)

- Ex: Gaussian RBF / exponentiated quadratic / squared exponential / …

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

- Reproducing property: $\langle f, \phi(x) \rangle_{\mathcal{H}} = f(x)$ for $f \in \mathcal{H}$
- $\mathcal{H} = \operatorname{cl}(\{\sum_{i=1}^{n} \alpha_i \phi(X_i) \mid n \geq 0, \alpha \in \mathbb{R}^n, X_i \in \mathcal{X}\})$

Reproducing Kernel Hilbert Space (RKHS)

- Ex: Gaussian RBF / exponentiated quadratic / squared exponential / …

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

- Reproducing property: $\langle f, \phi(x) \rangle_{\mathcal{H}} = f(x)$ for $f \in \mathcal{H}$
- $\mathcal{H} = \mathrm{cl}(\{\sum_{i=1}^n \alpha_i \phi(X_i) \mid n \geq 0, \alpha \in \mathbb{R}^n, X_i \in \mathcal{X}\})$
- $\|\sum_i \alpha_i \phi(X_i)\|_{\mathcal{H}}^2 = \alpha^\mathsf{T} K \alpha$, where $K_{ij} = k(X_i, X_j)$

Reproducing Kernel Hilbert Space (RKHS)

- Ex: Gaussian RBF / exponentiated quadratic / squared exponential / …

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

- Reproducing property: $\langle f, \phi(x) \rangle_{\mathcal{H}} = f(x)$ for $f \in \mathcal{H}$
- $\mathcal{H} = \operatorname{cl}(\{\sum_{i=1}^{n} \alpha_i \phi(X_i) \mid n \geq 0, \alpha \in \mathbb{R}^n, X_i \in \mathcal{X}\})$
- $\|\sum_i \alpha_i \phi(X_i)\|_{\mathcal{H}}^2 = \alpha^\mathsf{T} K \alpha$, where $K_{ij} = k(X_i, X_j)$
- $\operatorname{\mathbf{argmin}}_{f \in \mathcal{H}} L(f(X_1), \ldots, f(X_n)) + \lambda \|f\|_{\mathcal{H}}^2$ is in $\{\sum_{i=1}^{n} \alpha_i \phi(X_i) \mid \alpha \in \mathbb{R}^n\}$ – the representer theorem

Maximum Mean Discrepancy (MMD)

$$\mathrm{MMD}_k(\mathbb{P}, \mathbb{Q}) = \sup_{\|f\|_{\mathcal{H}} \leq 1} \mathop{\mathbb{E}}_{X \sim \mathbb{P}}[f(X)] - \mathop{\mathbb{E}}_{Y \sim \mathbb{Q}}[f(Y)]$$

Maximum Mean Discrepancy (MMD)

$$\begin{aligned}
\mathrm{MMD}_k(\mathbb{P}, \mathbb{Q}) &= \sup_{\|f\|_{\mathcal{H}} \leq 1} \mathbb{E}_{X \sim \mathbb{P}}[f(X)] - \mathbb{E}_{Y \sim \mathbb{Q}}[f(Y)] \\
&= \sup_{\|f\|_{\mathcal{H}} \leq 1} \mathbb{E}_{X \sim \mathbb{P}}[\langle f, \varphi(X) \rangle_{\mathcal{H}}] - \mathbb{E}_{Y \sim \mathbb{Q}}[\langle f, \varphi(Y) \rangle_{\mathcal{H}}]
\end{aligned}$$

Maximum Mean Discrepancy (MMD)

$$\begin{aligned}
\mathrm{MMD}_k(\mathbb{P},\mathbb{Q}) &= \sup_{\|f\|_{\mathcal{H}}\le 1} \mathop{\mathbb{E}}_{X\sim\mathbb{P}}[f(X)] - \mathop{\mathbb{E}}_{Y\sim\mathbb{Q}}[f(Y)] \\
&= \sup_{\|f\|_{\mathcal{H}}\le 1} \mathop{\mathbb{E}}_{X\sim\mathbb{P}}[\langle f,\varphi(X)\rangle_{\mathcal{H}}] - \mathop{\mathbb{E}}_{Y\sim\mathbb{Q}}[\langle f,\varphi(Y)\rangle_{\mathcal{H}}] \\
&= \sup_{\|f\|_{\mathcal{H}}\le 1} \left\langle f, \mathop{\mathbb{E}}_{X\sim\mathbb{P}}[\varphi(X)] - \mathop{\mathbb{E}}_{Y\sim\mathbb{Q}}[\varphi(Y)] \right\rangle_{\mathcal{H}}
\end{aligned}$$

Maximum Mean Discrepancy (MMD)

$$
\begin{aligned}
\mathrm{MMD}_k(\mathbb{P}, \mathbb{Q}) &= \sup_{\|f\|_{\mathcal{H}} \le 1} \mathop{\mathbb{E}}_{X \sim \mathbb{P}}[f(X)] - \mathop{\mathbb{E}}_{Y \sim \mathbb{Q}}[f(Y)] \\
&= \sup_{\|f\|_{\mathcal{H}} \le 1} \mathop{\mathbb{E}}_{X \sim \mathbb{P}}[\langle f, \varphi(X) \rangle_{\mathcal{H}}] - \mathop{\mathbb{E}}_{Y \sim \mathbb{Q}}[\langle f, \varphi(Y) \rangle_{\mathcal{H}}] \\
&= \sup_{\|f\|_{\mathcal{H}} \le 1} \left\langle f, \mathop{\mathbb{E}}_{X \sim \mathbb{P}}[\varphi(X)] - \mathop{\mathbb{E}}_{Y \sim \mathbb{Q}}[\varphi(Y)] \right\rangle_{\mathcal{H}} \\
&= \sup_{\|f\|_{\mathcal{H}} \le 1} \left\langle f, \mu_{\mathbb{P}}^k - \mu_{\mathbb{Q}}^k \right\rangle_{\mathcal{H}}
\end{aligned}
$$

Maximum Mean Discrepancy (MMD)

$$
\begin{aligned}
\mathrm{MMD}_k(\mathbb{P}, \mathbb{Q}) &= \sup_{\|f\|_{\mathcal{H}} \leq 1} \mathop{\mathbb{E}}_{X \sim \mathbb{P}}[f(X)] - \mathop{\mathbb{E}}_{Y \sim \mathbb{Q}}[f(Y)] \\
&= \sup_{\|f\|_{\mathcal{H}} \leq 1} \mathop{\mathbb{E}}_{X \sim \mathbb{P}}[\langle f, \varphi(X) \rangle_{\mathcal{H}}] - \mathop{\mathbb{E}}_{Y \sim \mathbb{Q}}[\langle f, \varphi(Y) \rangle_{\mathcal{H}}] \\
&= \sup_{\|f\|_{\mathcal{H}} \leq 1} \left\langle f, \mathop{\mathbb{E}}_{X \sim \mathbb{P}}[\varphi(X)] - \mathop{\mathbb{E}}_{Y \sim \mathbb{Q}}[\varphi(Y)] \right\rangle_{\mathcal{H}} \\
&= \sup_{\|f\|_{\mathcal{H}} \leq 1} \left\langle f, \mu_{\mathbb{P}}^k - \mu_{\mathbb{Q}}^k \right\rangle_{\mathcal{H}} = \left\| \mu_{\mathbb{P}}^k - \mu_{\mathbb{Q}}^k \right\|_{\mathcal{H}}
\end{aligned}
$$

MMD as feature matching

$$\mathrm{MMD}_k(\mathbb{P}, \mathbb{Q}) = \left\| \mathop{\mathbb{E}}_{X \sim \mathbb{P}}[\varphi(X)] - \mathop{\mathbb{E}}_{Y \sim \mathbb{Q}}[\varphi(Y)] \right\|_{\mathcal{H}}$$

- $\varphi : \mathcal{X} \to \mathcal{H}$ is the *feature map* for $k(x, y) = \langle \varphi(x), \varphi(y) \rangle$

MMD as feature matching

$$\mathrm{MMD}_k(\mathbb{P}, \mathbb{Q}) = \left\| \mathop{\mathbb{E}}_{X \sim \mathbb{P}}[\varphi(X)] - \mathop{\mathbb{E}}_{Y \sim \mathbb{Q}}[\varphi(Y)] \right\|_{\mathcal{H}}$$

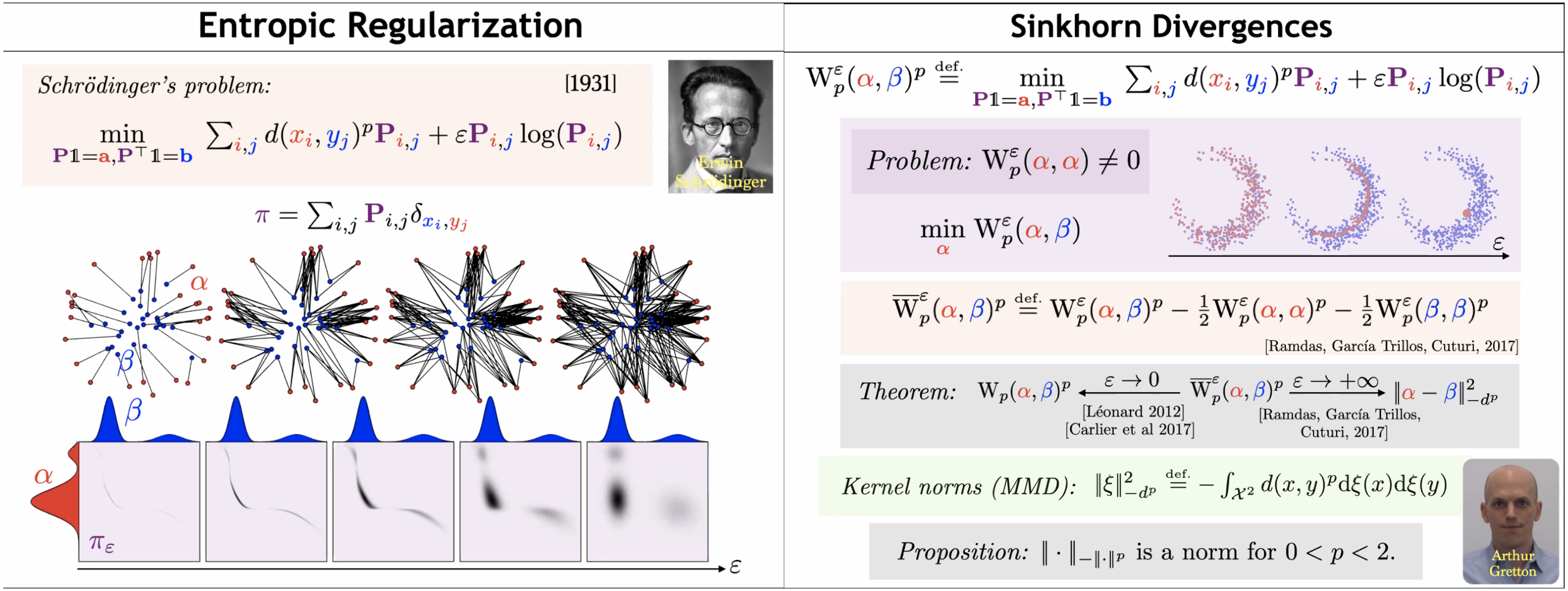
- $\varphi : X \to \mathcal{H}$ is the *feature map* for $k(x, y) = \langle \varphi(x), \varphi(y) \rangle$
- If $k(x, y) = x^\mathsf{T} y$, $\varphi(x) = x$; MMD is distance between means

MMD as feature matching

$$\mathrm{MMD}_k(\mathbb{P}, \mathbb{Q}) = \left\| \mathop{\mathbb{E}}_{X \sim \mathbb{P}} [\varphi(X)] - \mathop{\mathbb{E}}_{Y \sim \mathbb{Q}} [\varphi(Y)] \right\|_{\mathcal{H}}$$

- $\varphi : \mathcal{X} \to \mathcal{H}$ is the *feature map* for $k(x, y) = \langle \varphi(x), \varphi(y) \rangle$
- If $k(x, y) = x^\mathsf{T} y$, $\varphi(x) = x$; MMD is distance between means
- Many kernels: **infinite-dimensional** $\mathcal{H}$

MMD and OT


Entropic Regularization
Schrödinger's problem: [1931]
$\min_{\mathbf{P}\mathbb{1}=\mathbf{a},\mathbf{P}^\top\mathbb{1}=\mathbf{b}} \sum_{i,j} d(x_i, y_j)^p \mathbf{P}_{i,j} + \varepsilon \mathbf{P}_{i,j} \log(\mathbf{P}_{i,j})$
Erwin Schrödinger
$\pi = \sum_{i,j} \mathbf{P}_{i,j} \delta_{x_i, y_j}$
α
β
π_ε
ε
Sinkhorn Divergences
$\mathrm{W}_p^\varepsilon(\alpha,\beta)^p \stackrel{\text{def.}}{=} \min_{\mathbf{P}\mathbb{1}=\mathbf{a},\mathbf{P}^\top\mathbb{1}=\mathbf{b}} \sum_{i,j} d(x_i, y_j)^p \mathbf{P}_{i,j} + \varepsilon \mathbf{P}_{i,j} \log(\mathbf{P}_{i,j})$
Problem: $\mathrm{W}_p^\varepsilon(\alpha,\alpha) \neq 0$
$\min_{\alpha} \mathrm{W}_p^\varepsilon(\alpha,\beta)$
ε
$\overline{\mathrm{W}}_p^\varepsilon(\alpha,\beta)^p \stackrel{\text{def.}}{=} \mathrm{W}_p^\varepsilon(\alpha,\beta)^p - \frac{1}{2}\mathrm{W}_p^\varepsilon(\alpha,\alpha)^p - \frac{1}{2}\mathrm{W}_p^\varepsilon(\beta,\beta)^p$
[Ramdas, García Trillos, Cuturi, 2017]
Theorem: $\mathrm{W}_p(\alpha,\beta)^p \xleftarrow{\varepsilon \to 0} \overline{\mathrm{W}}_p^\varepsilon(\alpha,\beta)^p \xrightarrow{\varepsilon \to +\infty} \|\alpha-\beta\|_{-d^p}^2$
[Léonard 2012]
[Carlier et al 2017]
[Ramdas, García Trillos, Cuturi, 2017]
Kernel norms (MMD): $\|\xi\|_{-d^p}^2 \stackrel{\text{def.}}{=} -\int_{\mathcal{X}^2} d(x,y)^p \mathrm{d}\xi(x)\mathrm{d}\xi(y)$
Proposition: $\|\cdot\|_{-\|\cdot\|^p}$ is a norm for $0<p<2$.
Arthur Gretton

Estimating MMD

$$\mathrm{MMD}_k^2(\mathbb{P}, \mathbb{Q}) = \mathop{\mathbb{E}}_{X, X' \sim \mathbb{P}}[k(X, X')] + \mathop{\mathbb{E}}_{Y, Y' \sim \mathbb{Q}}[k(Y, Y')] - 2 \mathop{\mathbb{E}}_{\substack{X \sim \mathbb{P} \\ Y \sim \mathbb{Q}}}[k(X, Y)]$$

Estimating MMD

$$\mathrm{MMD}_k^2(\mathbb{P}, \mathbb{Q}) = \mathop{\mathbb{E}}_{X, X' \sim \mathbb{P}}[k(X, X')] + \mathop{\mathbb{E}}_{Y, Y' \sim \mathbb{Q}}[k(Y, Y')] - 2 \mathop{\mathbb{E}}_{\substack{X \sim \mathbb{P} \\ Y \sim \mathbb{Q}}}[k(X, Y)]$$

$$\widehat{\mathrm{MMD}}_k^2(X, Y) = \mathrm{mean}(K_{XX}) + \mathrm{mean}(K_{YY}) - 2\,\mathrm{mean}(K_{XY})$$

Estimating MMD

$$\mathrm{MMD}_k^2(\mathbb{P}, \mathbb{Q}) = \mathop{\mathbb{E}}_{X, X' \sim \mathbb{P}}[k(X, X')] + \mathop{\mathbb{E}}_{Y, Y' \sim \mathbb{Q}}[k(Y, Y')] - 2 \mathop{\mathbb{E}}_{\substack{X \sim \mathbb{P} \\ Y \sim \mathbb{Q}}}[k(X, Y)]$$

$$\widehat{\mathrm{MMD}}_k^2(X, Y) = \mathrm{mean}(K_{XX}) + \mathrm{mean}(K_{YY}) - 2\,\mathrm{mean}(K_{XY})$$

K_{XX}

	1.0	0.2	0.6
	0.2	1.0	0.5
	0.6	0.5	1.0

Estimating MMD

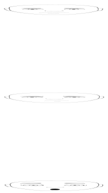
$$\mathrm{MMD}_k^2(\mathbb{P}, \mathbb{Q}) = \mathop{\mathbb{E}}_{X,X' \sim \mathbb{P}}[k(X, X')] + \mathop{\mathbb{E}}_{Y,Y' \sim \mathbb{Q}}[k(Y, Y')] - 2 \mathop{\mathbb{E}}_{\substack{X \sim \mathbb{P} \\ Y \sim \mathbb{Q}}}[k(X, Y)]$$

$$\widehat{\mathrm{MMD}}_k^2(X, Y) = \mathrm{mean}(K_{XX}) + \mathrm{mean}(K_{YY}) - 2\,\mathrm{mean}(K_{XY})$$

K_{XX}

1.0	0.2	0.6
0.2	1.0	0.5
0.6	0.5	1.0

K_{YY}



1.0	0.8	0.7
0.8	1.0	0.6
0.7	0.6	1.0

Estimating MMD

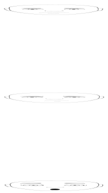
$$\mathrm{MMD}_k^2(\mathbb{P}, \mathbb{Q}) = \mathop{\mathbb{E}}_{X, X' \sim \mathbb{P}}[k(X, X')] + \mathop{\mathbb{E}}_{Y, Y' \sim \mathbb{Q}}[k(Y, Y')] - 2 \mathop{\mathbb{E}}_{\substack{X \sim \mathbb{P} \\ Y \sim \mathbb{Q}}}[k(X, Y)]$$

$$\widehat{\mathrm{MMD}}_k^2(X, Y) = \mathrm{mean}(K_{XX}) + \mathrm{mean}(K_{YY}) - 2\,\mathrm{mean}(K_{XY})$$

K_{XX}

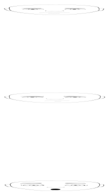
1.0	0.2	0.6
0.2	1.0	0.5
0.6	0.5	1.0

K_{YY}



1.0	0.8	0.7
0.8	1.0	0.6
0.7	0.6	1.0

K_{XY}



0.3	0.1	0.2
0.2	0.3	0.3
0.2	0.1	0.4

I: Two-sample testing

- Given samples from two unknown distributions

$$X \sim \mathbb{P} \qquad Y \sim \mathbb{Q}$$

- Question: is $\mathbb{P} = \mathbb{Q}$?

I: Two-sample testing

- Given samples from two unknown distributions

$$X \sim \mathbb{P} \qquad Y \sim \mathbb{Q}$$

- Do smokers/non-smokers get different cancers?

I: Two-sample testing

- Given samples from two unknown distributions

$$X \sim \mathbb{P} \qquad Y \sim \mathbb{Q}$$

- Do smokers/non-smokers get different cancers?
- Do Brits have the same friend network types as Americans?

I: Two-sample testing

- Given samples from two unknown distributions

$$X \sim \mathbb{P} \qquad Y \sim \mathbb{Q}$$

- Do smokers/non-smokers get different cancers?
- Do Brits have the same friend network types as Americans?
- When does my laser agree with the one on Mars?

I: Two-sample testing

- Given samples from two unknown distributions

$$X \sim \mathbb{P} \qquad Y \sim \mathbb{Q}$$

- Do smokers/non-smokers get different cancers?
- Do Brits have the same friend network types as Americans?
- When does my laser agree with the one on Mars?
- Are storms in the 2000s different from storms in the 1800s?

I: Two-sample testing

- Given samples from two unknown distributions

$$X \sim \mathbb{P} \qquad Y \sim \mathbb{Q}$$

- Do smokers/non-smokers get different cancers?
- Do Brits have the same friend network types as Americans?
- When does my laser agree with the one on Mars?
- Are storms in the 2000s different from storms in the 1800s?
- Does presence of this protein affect DNA binding? [MMDiff2]

I: Two-sample testing

- Given samples from two unknown distributions

$$X \sim \mathbb{P} \qquad Y \sim \mathbb{Q}$$

- Do smokers/non-smokers get different cancers?
- Do Brits have the same friend network types as Americans?
- When does my laser agree with the one on Mars?
- Are storms in the 2000s different from storms in the 1800s?
- Does presence of this protein affect DNA binding? [`MMDiff2`]
- Do these `dob` and `birthday` columns mean the same thing?

I: Two-sample testing

- Given samples from two unknown distributions

$$X \sim \mathbb{P} \qquad Y \sim \mathbb{Q}$$

- Do smokers/non-smokers get different cancers?
- Do Brits have the same friend network types as Americans?
- When does my laser agree with the one on Mars?
- Are storms in the 2000s different from storms in the 1800s?
- Does presence of this protein affect DNA binding? [`MMDiff2`]
- Do these `dob` and `birthday` columns mean the same thing?
- Does my generative model $\mathbb{Q}_\theta$ match $\mathbb{P}_{\text{data}}$?

I: Two-sample testing

- Given samples from two unknown distributions

$$X \sim \mathbb{P} \qquad Y \sim \mathbb{Q}$$

- Do smokers/non-smokers get different cancers?
- Do Brits have the same friend network types as Americans?
- When does my laser agree with the one on Mars?
- Are storms in the 2000s different from storms in the 1800s?
- Does presence of this protein affect DNA binding? [`MMDiff2`]
- Do these `dob` and `birthday` columns mean the same thing?
- Does my generative model $\mathbb{Q}_\theta$ match $\mathbb{P}_{\text{data}}$?
- Independence testing: is $P(X, Y) = P(X)P(Y)$?

I: Two-sample testing

- Given samples from two unknown distributions

$$X \sim \mathbb{P} \qquad Y \sim \mathbb{Q}$$

- Question: is $\mathbb{P} = \mathbb{Q}$?

I: Two-sample testing

- Given samples from two unknown distributions

$$X \sim \mathbb{P} \qquad Y \sim \mathbb{Q}$$

- Question: is $\mathbb{P} = \mathbb{Q}$?
- Hypothesis testing approach:

$$H_0 : \mathbb{P} = \mathbb{Q} \qquad H_1 : \mathbb{P} \neq \mathbb{Q}$$

I: Two-sample testing

- Given samples from two unknown distributions

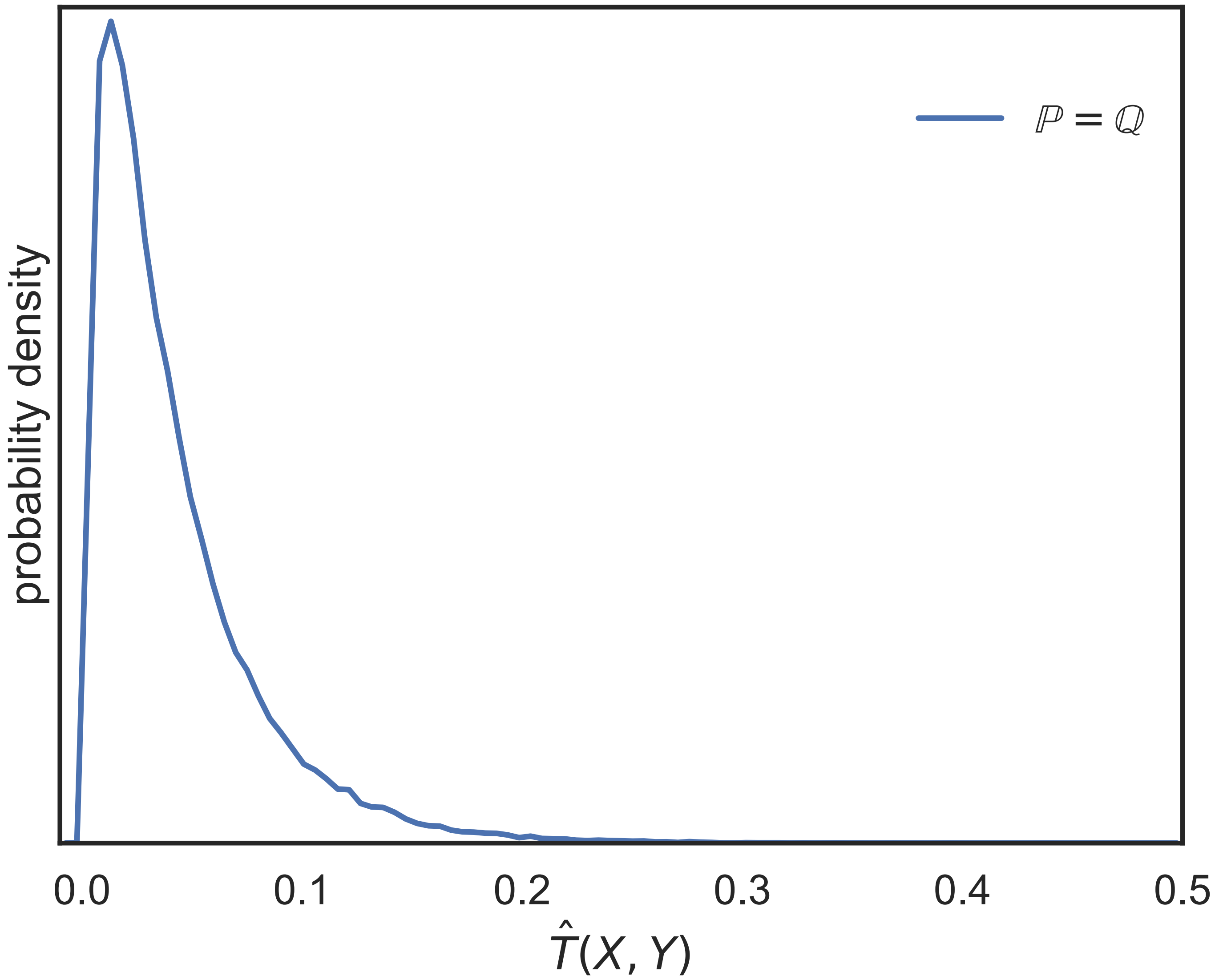
$$X \sim \mathbb{P} \qquad Y \sim \mathbb{Q}$$

- Question: is $\mathbb{P} = \mathbb{Q}$?
- Hypothesis testing approach:

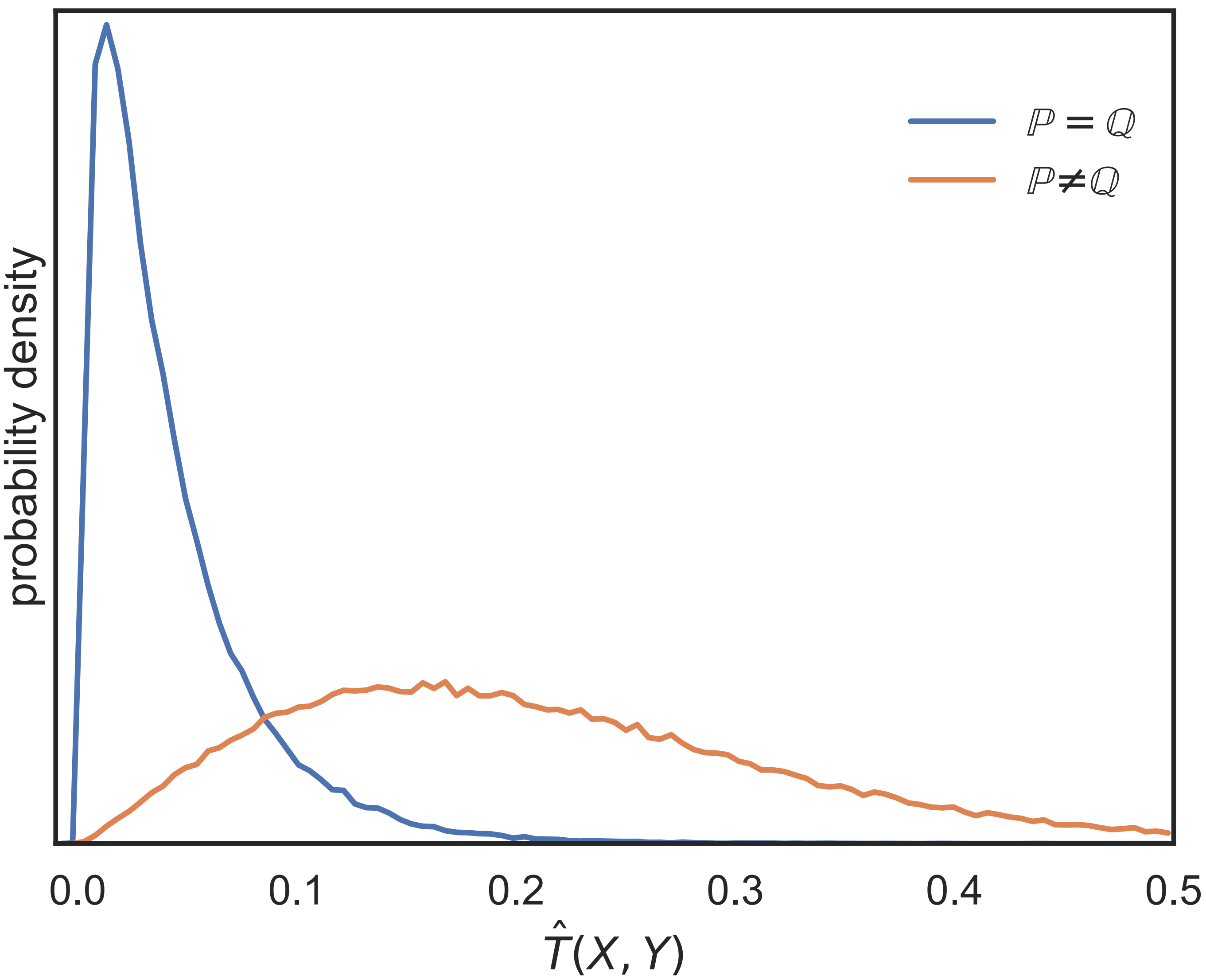
$$H_0 : \mathbb{P} = \mathbb{Q} \qquad H_1 : \mathbb{P} \neq \mathbb{Q}$$

- Reject H_0 if test statistic $\hat{T}(X, Y) > c_\alpha$

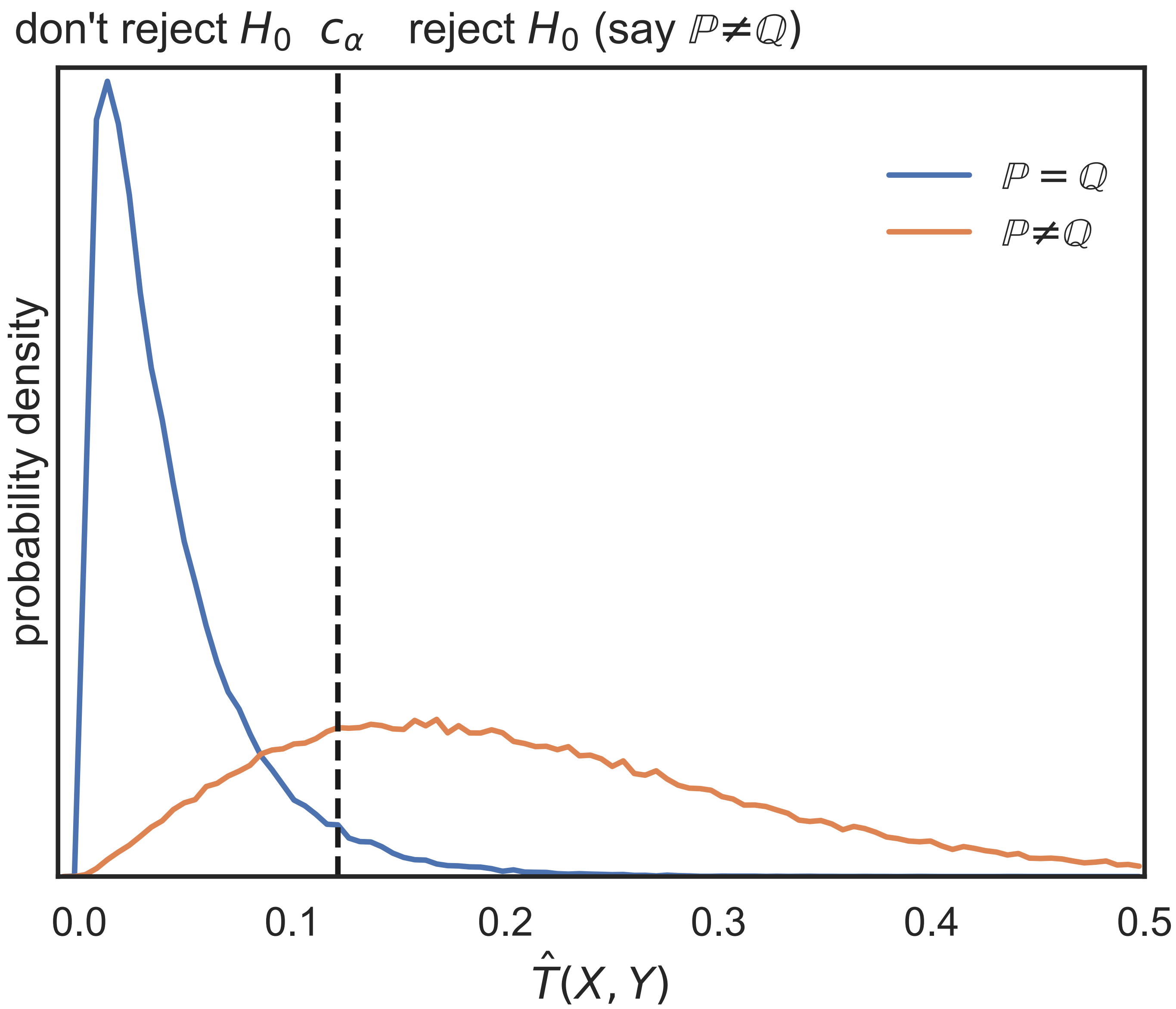
What's a hypothesis test again?



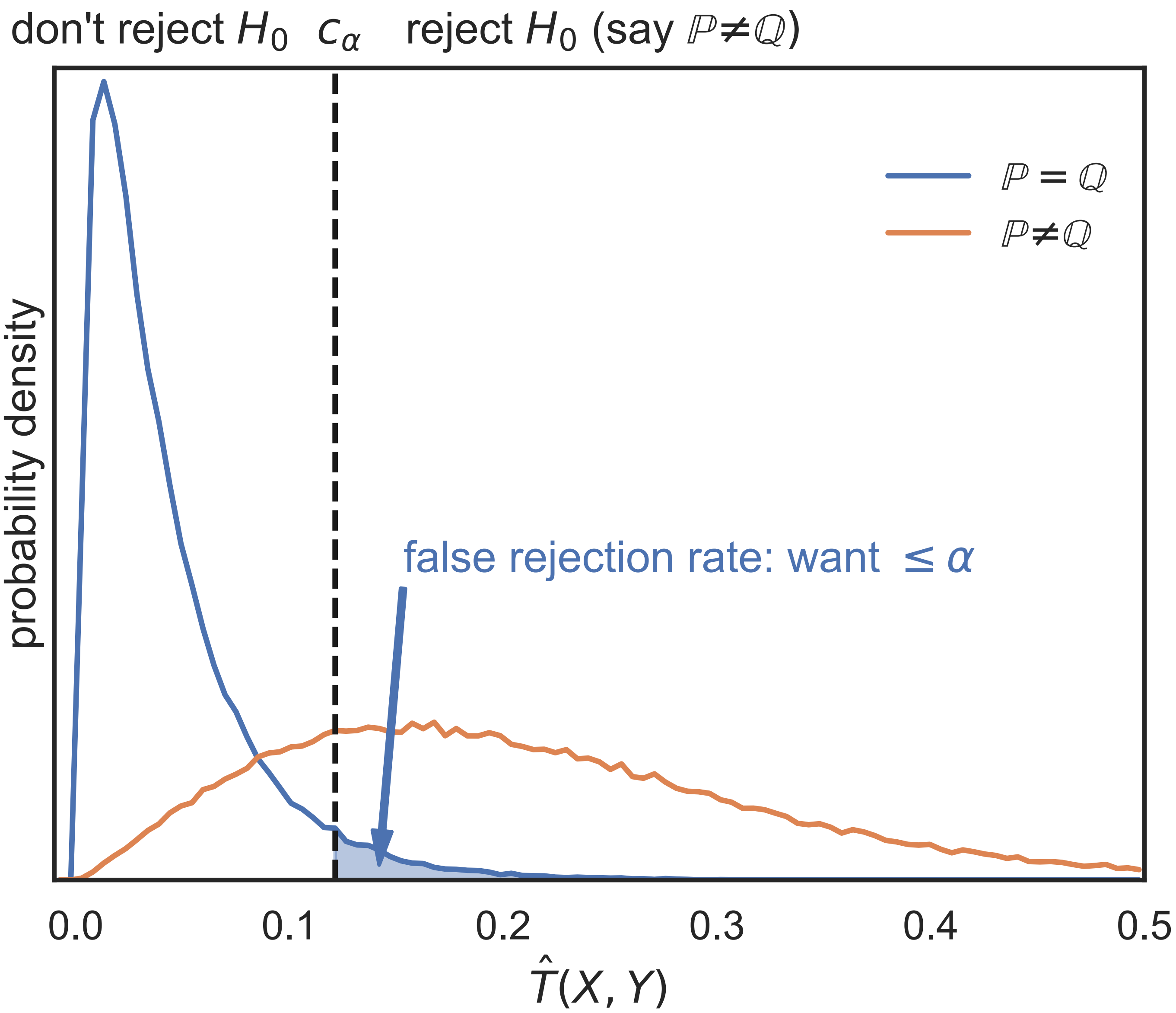
What's a hypothesis test again?



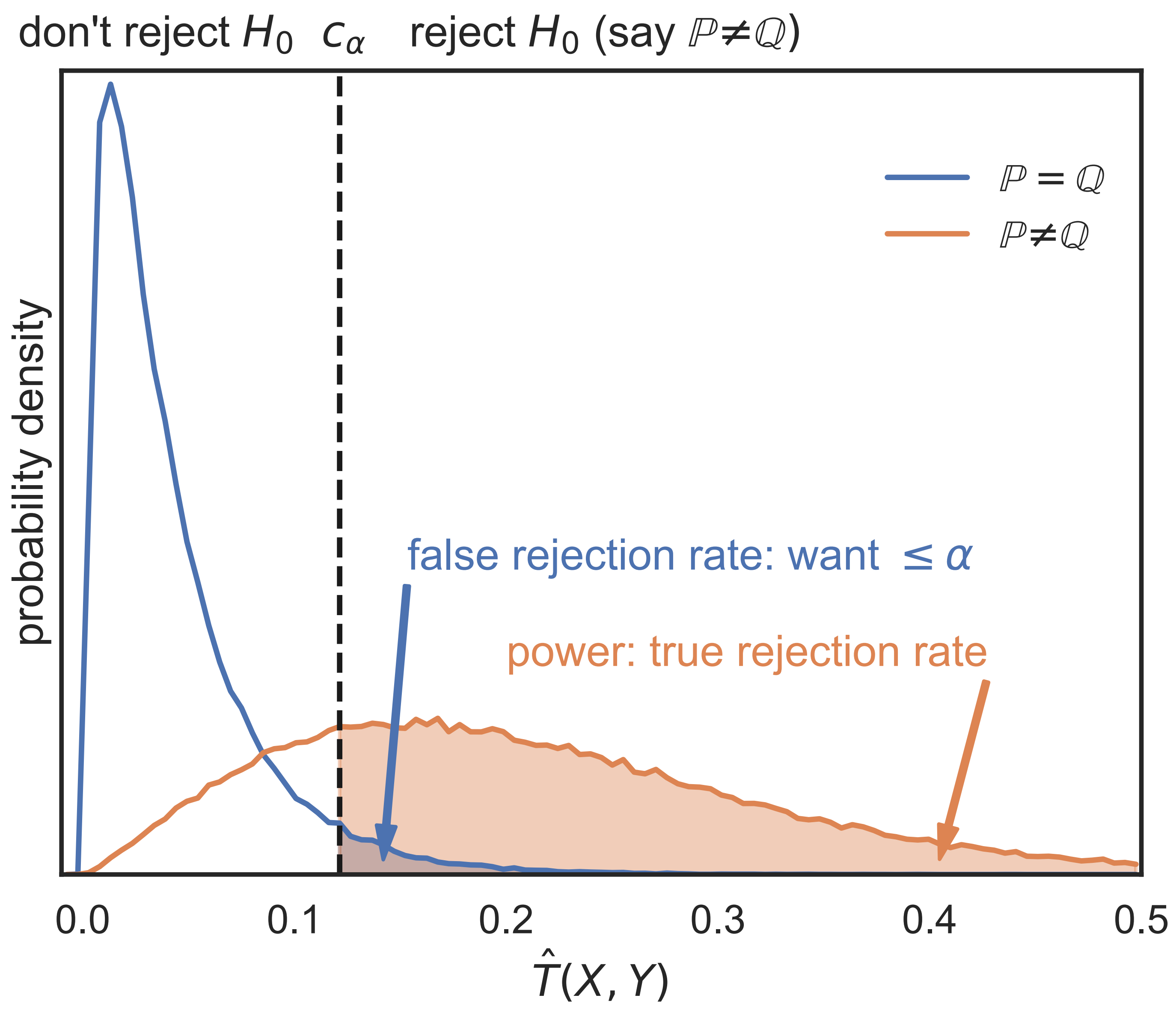
What's a hypothesis test again?



What's a hypothesis test again?



What's a hypothesis test again?



Permutation testing to find c_α

Need $\Pr_{H_0}\left(T(X, Y) > c_\alpha\right) \leq \alpha$

$$X_1 \quad X_2 \quad X_3 \quad X_4 \quad X_5 \qquad Y_1 \quad Y_2 \quad Y_3 \quad Y_4 \quad Y_5$$

$c_\alpha : 1 - \alpha$th quantile of $\left\{ \qquad \right\}$

Permutation testing to find c_α

Need $\Pr_{H_0}\left(T(X, Y) > c_\alpha\right) \leq \alpha$

X_1 X_2 X_3 X_4 X_5 Y_1 Y_2 Y_3 Y_4 Y_5

$c_\alpha : 1 - \alpha$th quantile of $\left\{ \qquad \right\}$

Permutation testing to find c_α

Need $\Pr_{H_0}\left(T(X, Y) > c_\alpha\right) \leq \alpha$

$c_\alpha : 1 - \alpha$th quantile of $\left\{ \hat{T}(\tilde{X}_1, \tilde{Y}_1), \qquad \right\}$

Permutation testing to find c_α

Need $\Pr_{H_0}\left(T(X, Y) > c_\alpha\right) \leq \alpha$

X_1 X_2 X_3 X_4 X_5 Y_1 Y_2 Y_3 Y_4 Y_5

c_α: $1-\alpha$th quantile of $\left\{\hat{T}(\tilde{X}_1, \tilde{Y}_1),\ \hat{T}(\tilde{X}_2, \tilde{Y}_2), \qquad \right\}$

Permutation testing to find c_α

Need $\Pr_{H_0}\left(T(X, Y) > c_\alpha\right) \leq \alpha$

$X_1 \quad X_2 \quad X_3 \quad X_4 \quad X_5 \qquad Y_1 \quad Y_2 \quad Y_3 \quad Y_4 \quad Y_5$

c_α: $1-\alpha$th quantile of $\left\{\hat{T}(\tilde{X}_1, \tilde{Y}_1),\ \hat{T}(\tilde{X}_2, \tilde{Y}_2),\ \cdots\right\}$

MMD-based tests

- If k is *characteristic*, $\mathrm{MMD}(\mathbb{P}, \mathbb{Q}) = 0$ iff $\mathbb{P} = \mathbb{Q}$
- Efficient permutation testing for $\widehat{\mathrm{MMD}}(X, Y)$

MMD-based tests

- If k is *characteristic*, $\mathrm{MMD}(\mathbb{P}, \mathbb{Q}) = 0$ iff $\mathbb{P} = \mathbb{Q}$
- Efficient permutation testing for $\widehat{\mathrm{MMD}}(X, Y)$
  - H_0: $n\widehat{\mathrm{MMD}}^2$ converges in distribution
  - H_1: $\sqrt{n}(\widehat{\mathrm{MMD}}^2 - \mathrm{MMD}^2)$ asymptotically normal

MMD-based tests

- If k is *characteristic*, $\mathrm{MMD}(\mathbb{P}, \mathbb{Q}) = 0$ iff $\mathbb{P} = \mathbb{Q}$
- Efficient permutation testing for $\widehat{\mathrm{MMD}}(X, Y)$
  - H_0: $n\widehat{\mathrm{MMD}}^2$ converges in distribution
  - H_1: $\sqrt{n}(\widehat{\mathrm{MMD}}^2 - \mathrm{MMD}^2)$ asymptotically normal
- Any characteristic kernel gives consistent test

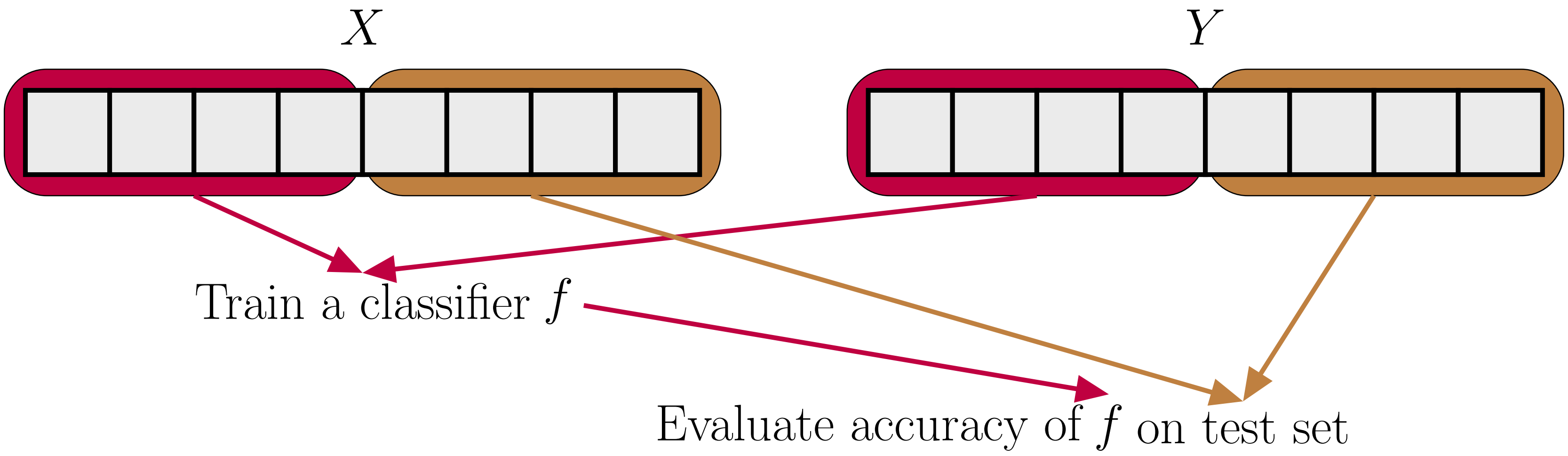
MMD-based tests

- If k is *characteristic*, $\mathrm{MMD}(\mathbb{P}, \mathbb{Q}) = 0$ iff $\mathbb{P} = \mathbb{Q}$
- Efficient permutation testing for $\widehat{\mathrm{MMD}}(X, Y)$
  - H_0: $n\widehat{\mathrm{MMD}}^2$ converges in distribution
  - H_1: $\sqrt{n}(\widehat{\mathrm{MMD}}^2 - \mathrm{MMD}^2)$ asymptotically normal
- Any characteristic kernel gives consistent test…eventually

MMD-based tests

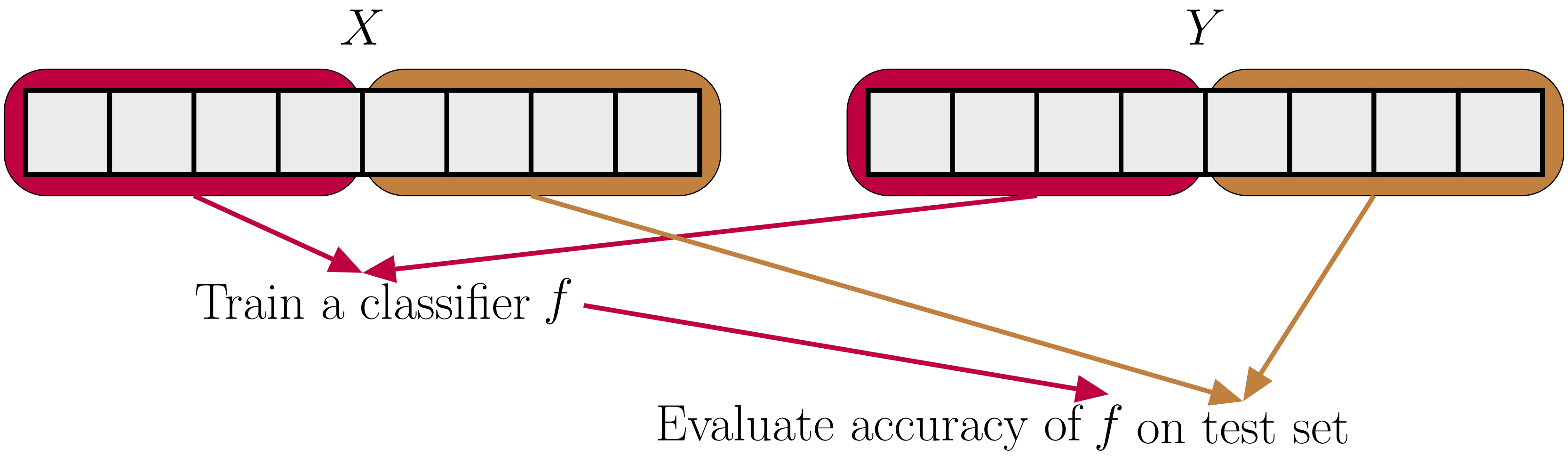
- If k is *characteristic*, $\mathrm{MMD}(\mathbb{P}, \mathbb{Q}) = 0$ iff $\mathbb{P} = \mathbb{Q}$
- Efficient permutation testing for $\widehat{\mathrm{MMD}}(X, Y)$
  - H_0: $n\widehat{\mathrm{MMD}}^2$ converges in distribution
  - H_1: $\sqrt{n}(\widehat{\mathrm{MMD}}^2 - \mathrm{MMD}^2)$ asymptotically normal
- Any characteristic kernel gives consistent test…eventually
- Need enormous n if kernel is bad for problem

Classifier two-sample tests



- $\hat{T}(X, Y)$ is the accuracy of f on the test set
- Under H_0, classification impossible: $\hat{T} \sim \mathbf{Binomial}(n, \frac{1}{2})$

Classifier two-sample tests



- $\hat{T}(X, Y)$ is the accuracy of f on the test set
- Under H_0, classification impossible: $\hat{T} \sim \mathrm{Binomial}(n, \frac{1}{2})$
- With $k(x, y) = \frac{1}{4} f(x) f(y)$ where $f(x) \in \{-1, 1\}$, get $\widehat{\mathrm{MMD}}(X, Y) = \left|\hat{T}(X, Y) - \frac{1}{2}\right|$

Optimizing test power

- Asymptotics of $\widehat{\mathrm{MMD}}^2$ give us immediately that

$$\Pr_{H_1}\left(n\widehat{\mathrm{MMD}}^2 > c_\alpha\right) \approx \Phi\left(\frac{\sqrt{n}\,\mathrm{MMD}^2}{\sigma_{H_1}} - \frac{c_\alpha}{\sqrt{n}\sigma_{H_1}}\right)$$

MMD, σ_{H_1}, c_α are constants: first term dominates

Optimizing test power

- Asymptotics of $\widehat{\mathrm{MMD}}^2$ give us immediately that

$$\Pr_{H_1}\left(n\widehat{\mathrm{MMD}}^2 > c_\alpha\right) \approx \Phi\left(\frac{\sqrt{n}\,\mathrm{MMD}^2}{\sigma_{H_1}} - \frac{c_\alpha}{\sqrt{n}\sigma_{H_1}}\right)$$

  MMD, σ_{H_1}, c_α are constants: first term dominates

- Pick k to maximize an estimate of $\mathrm{MMD}^2\,/\sigma_{H_1}$

Optimizing test power

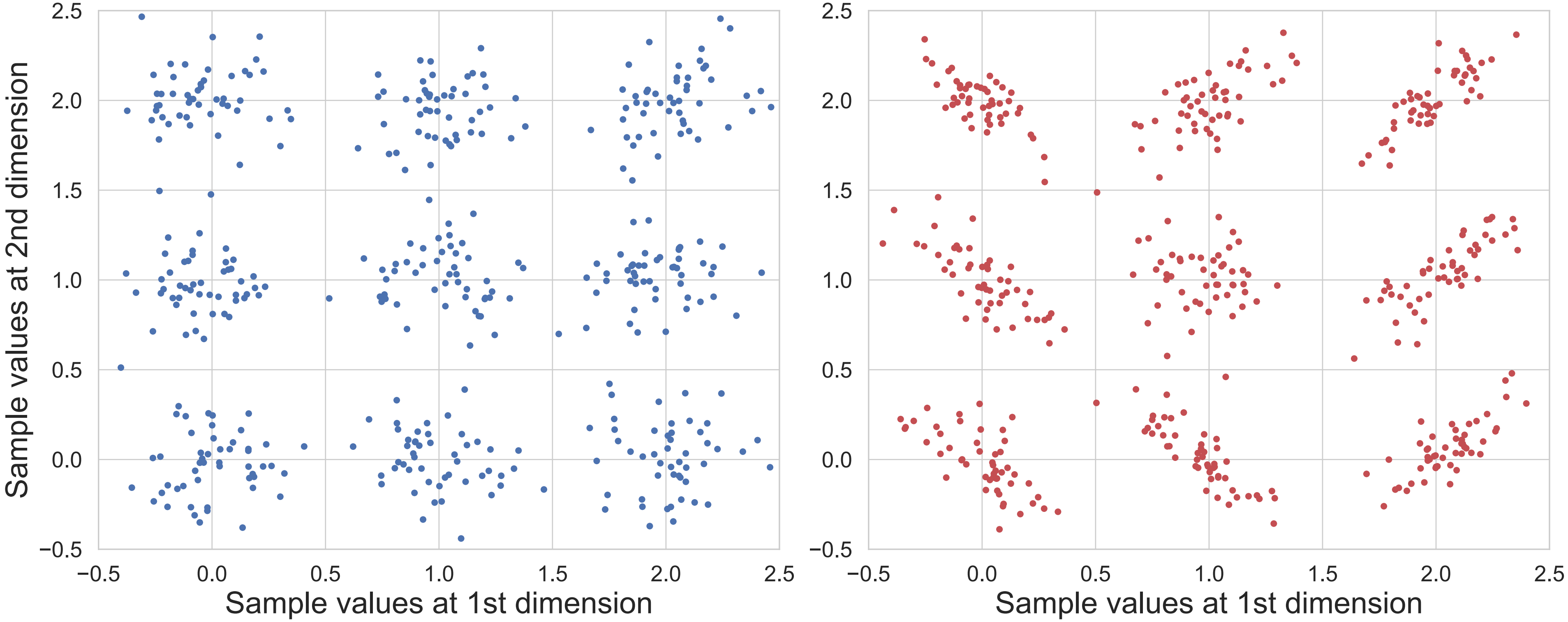
- Asymptotics of $\widehat{\mathrm{MMD}}^2$ give us immediately that

$$\Pr_{H_1}\left(n\widehat{\mathrm{MMD}}^2 > c_\alpha\right) \approx \Phi\left(\frac{\sqrt{n}\,\mathrm{MMD}^2}{\sigma_{H_1}} - \frac{c_\alpha}{\sqrt{n}\sigma_{H_1}}\right)$$

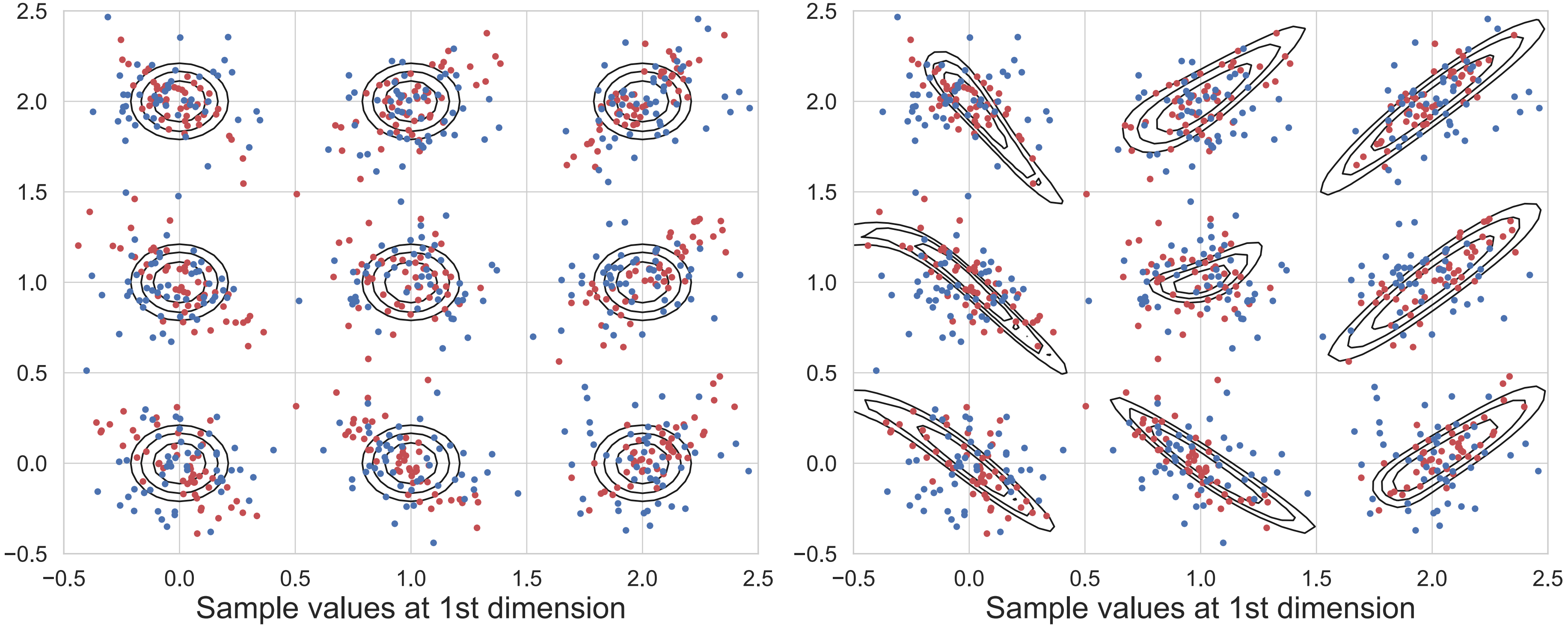
  MMD, σ_{H_1}, c_α are constants: first term dominates

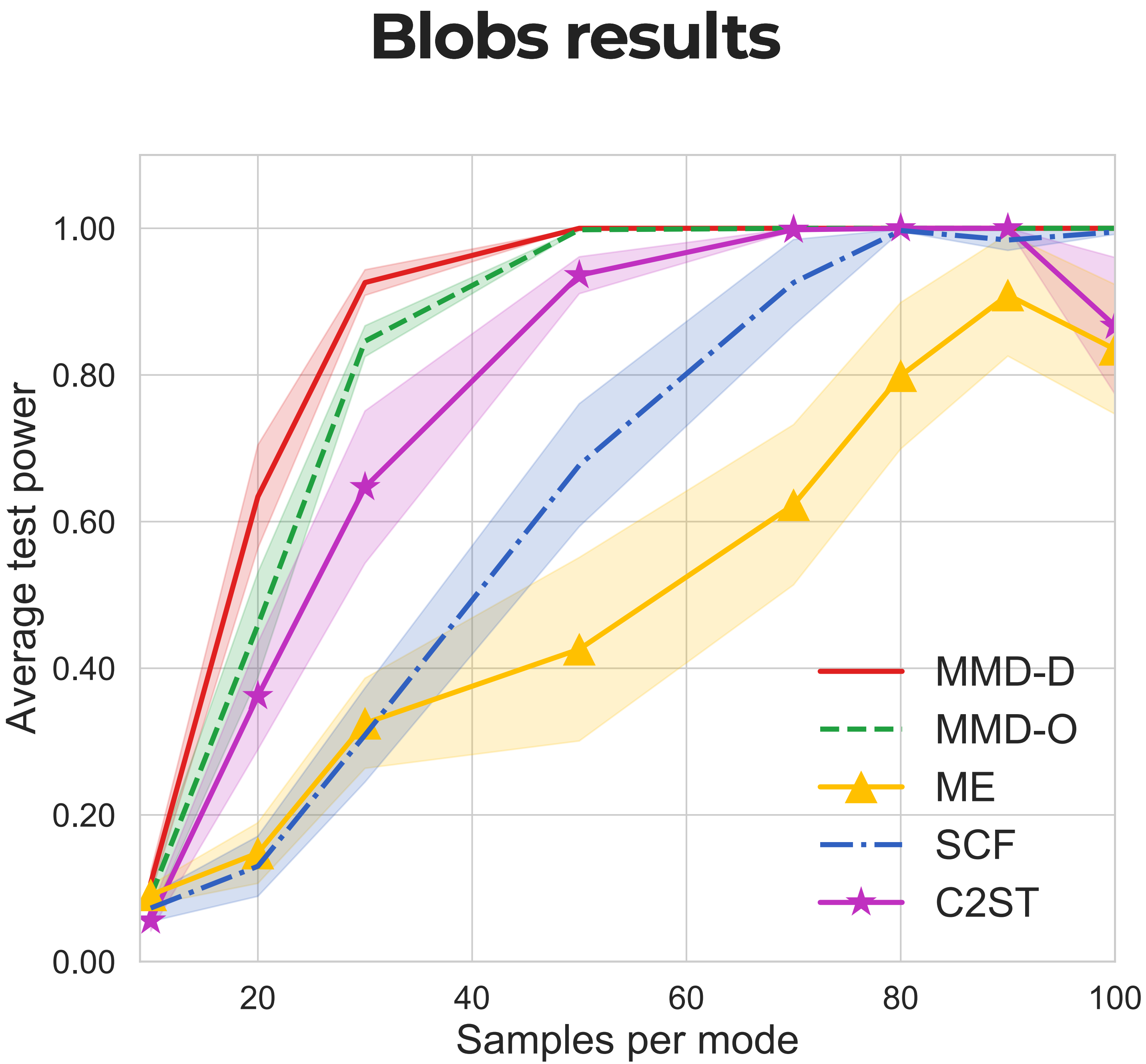
- Pick k to maximize an estimate of $\mathrm{MMD}^2 / \sigma_{H_1}$
- Can show uniform $\mathcal{O}_P(n^{-\frac{1}{3}})$ convergence of estimator

Blobs dataset

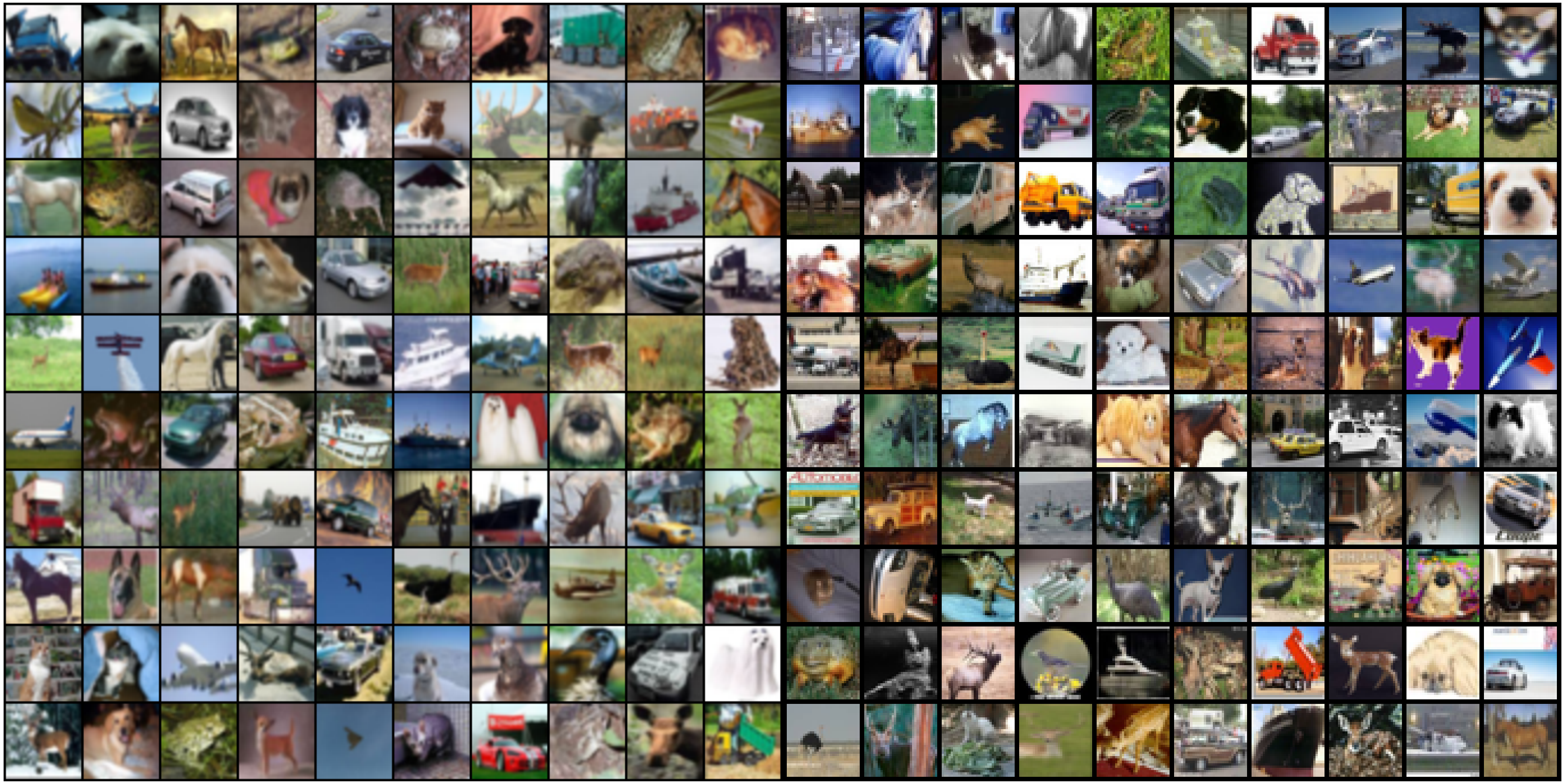


Blobs kernels




Blobs results
Average test power
1.00
0.80
0.60
0.40
0.20
0.00
20
40
60
80
100
Samples per mode
MMD-D
MMD-O
ME
SCF
C2ST

CIFAR-10 vs CIFAR-10.1



Train on 1 000, test on 1 031, repeat 10 times. Rejection rates:

ME	SCF	C2ST	MMD-O	MMD-D
0.588	0.171	0.452	0.316	**0.744**

Ablation vs classifier-based tests

	Cross-entropy			Max power		
Dataset	**Sign**	**Lin**	**Ours**	**Sign**	**Lin**	**Ours**
Blob	0.84	0.94	0.90	–	0.95	0.99
High-d Gauss. mix.	0.47	0.59	0.29	–	0.64	0.66
Higgs	0.26	0.40	0.35	–	0.30	0.40
MNIST vs GAN	0.65	0.71	0.80	–	0.94	1.00

II: Training implicit generative models

Given samples from a distribution $\mathbb{P}$ over $\mathcal{X}$,
we want a model that can produce new samples from $\mathbb{Q}_\theta \approx \mathbb{P}$

$X \sim \mathbb{P}$

$Y \sim \mathbb{Q}_\theta$

II: Training implicit generative models

we want [...] $\mathbb{Q}_\theta \approx \mathbb{P}$

II: Training implicit generative models

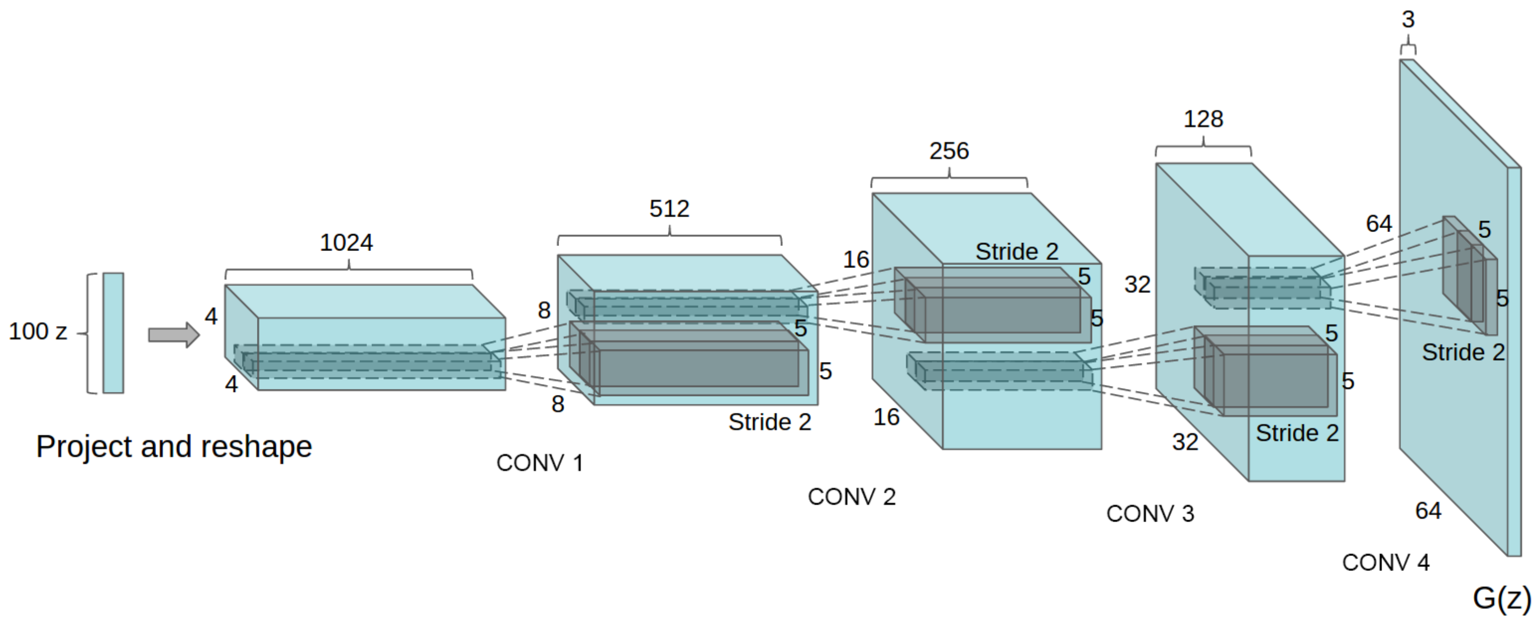
Given samples from a distribution $\mathbb{P}$ over $\mathcal{X}$,
we want a model that can produce new samples from $\mathbb{Q}_\theta \approx \mathbb{P}$

"Everybody Dance Now" [Chan et al. ICCV-19]

Generator networks

Fixed distribution of latents: $Z \sim \mathrm{Uniform}\left([-1,1]^{100}\right)$

Maps through a network: $G_\theta(Z) \sim \mathbb{Q}_\theta$

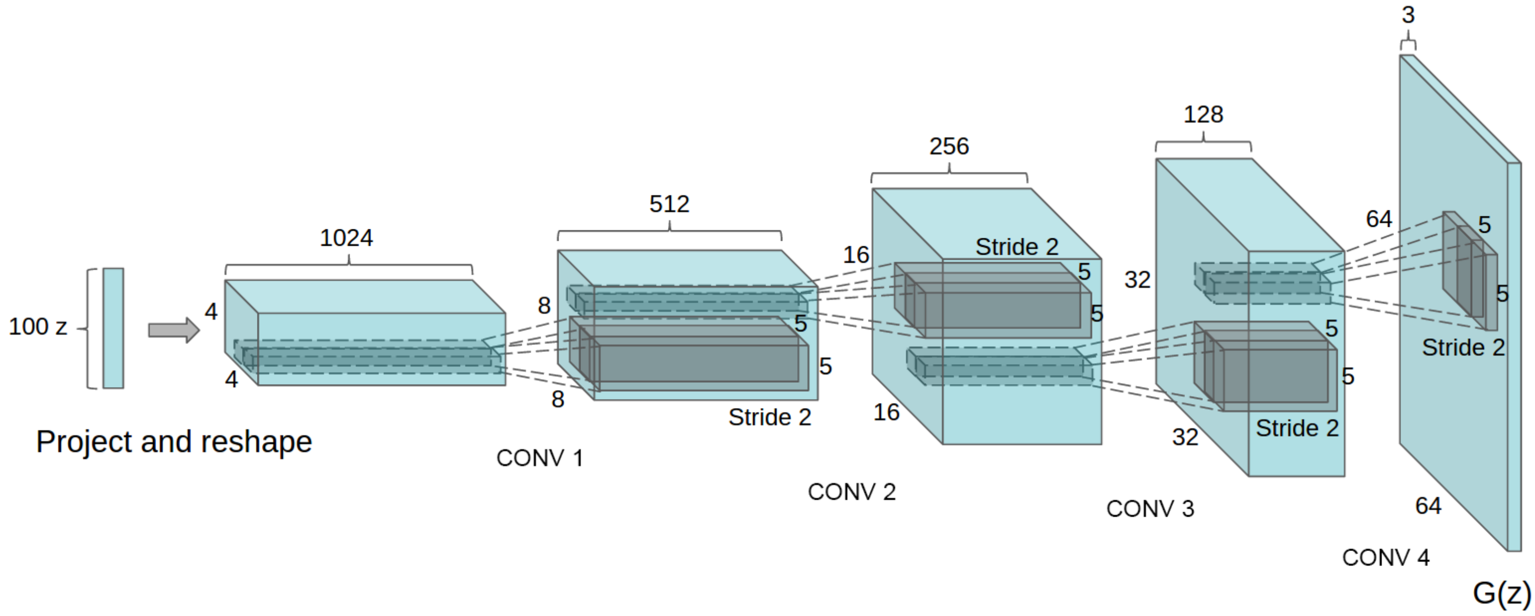


DCGAN generator [Radford+ ICLR-16]

Generator networks

Fixed distribution of latents: $Z \sim \mathrm{Uniform}\left([-1,1]^{100}\right)$

Maps through a network: $G_\theta(Z) \sim \mathbb{Q}_\theta$



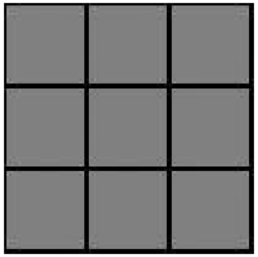
DCGAN generator [Radford+ ICLR-16]

How to choose θ?

GANs and their flaws

- GANs [Goodfellow+ NeurIPS-14] minimize discriminator accuracy (like classifier test) between $\mathbb{P}$ and $\mathbb{Q}_\theta$
- Problem: if there's a perfect classifier, discontinuous loss, no gradient to improve it [Arjovsky/Bottou ICLR-17]
- Disjoint at init:

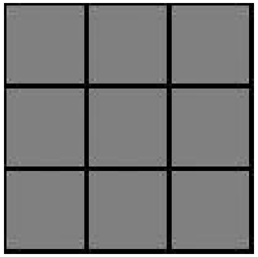
$\mathbb{P}$: 

$\mathbb{Q}_{\theta_0}$: 

GANs and their flaws

- GANs [Goodfellow+ NeurIPS-14] minimize discriminator accuracy (like classifier test) between $\mathbb{P}$ and $\mathbb{Q}_\theta$
- Problem: if there's a perfect classifier, discontinuous loss, no gradient to improve it [Arjovsky/Bottou ICLR-17]
- Disjoint at init:

$\mathbb{P}$: 

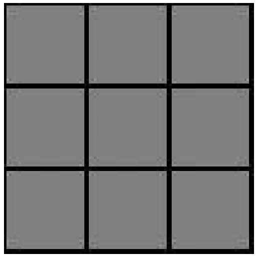
$\mathbb{Q}_{\theta_0}$: 

- For usual $G_\theta : \mathbb{R}^{100} \to \mathbb{R}^{64\times 64\times 3}$, $\mathbb{Q}_\theta$ is supported on a countable union of manifolds with dim ≤ 100

GANs and their flaws

- GANs [Goodfellow+ NeurIPS-14] minimize discriminator accuracy (like classifier test) between $\mathbb{P}$ and $\mathbb{Q}_\theta$
- Problem: if there's a perfect classifier, discontinuous loss, no gradient to improve it [Arjovsky/Bottou ICLR-17]
- Disjoint at init:

$\mathbb{P}$: 

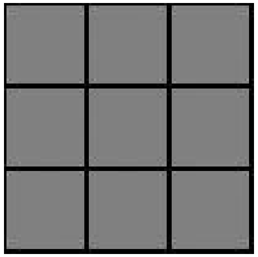
$\mathbb{Q}_{\theta_0}$: 

- For usual $G_\theta : \mathbb{R}^{100} \to \mathbb{R}^{64\times 64\times 3}$, $\mathbb{Q}_\theta$ is supported on a countable union of manifolds with dim ≤ 100
- "Natural image manifold" usually considered low-dim

GANs and their flaws

- GANs [Goodfellow+ NeurIPS-14] minimize discriminator accuracy (like classifier test) between $\mathbb{P}$ and $\mathbb{Q}_\theta$
- Problem: if there's a perfect classifier, discontinuous loss, no gradient to improve it [Arjovsky/Bottou ICLR-17]
- Disjoint at init:

$\mathbb{P}$: 

$\mathbb{Q}_{\theta_0}$: 

- For usual $G_\theta : \mathbb{R}^{100} \to \mathbb{R}^{64\times64\times3}$, $\mathbb{Q}_\theta$ is supported on a countable union of manifolds with dim ≤ 100
- “Natural image manifold” usually considered low-dim
- Won't align at init, so won't ever align

WGANs and MMD GANs

- Integral probability metrics with “smooth” $\mathcal{F}$ are continuous
- WGAN: $\mathcal{F}$ a set of neural networks satisfying $\|f\|_L \leq 1$
- WGAN-GP: instead penalize $\mathbb{E}\|\nabla_x f(x)\|$ near the data

WGANs and MMD GANs

- Integral probability metrics with “smooth” $\mathcal{F}$ are continuous
- WGAN: $\mathcal{F}$ a set of neural networks satisfying $\|f\|_L \leq 1$
- WGAN-GP: instead penalize $\mathbb{E}\|\nabla_x f(x)\|$ near the data
- Both losses are MMD with $k_\psi(x, y) = \phi_\psi(x)\phi_\psi(y)$

WGANs and MMD GANs

- Integral probability metrics with "smooth" $\mathcal{F}$ are continuous
- WGAN: $\mathcal{F}$ a set of neural networks satisfying $\|f\|_L \leq 1$
- WGAN-GP: instead penalize $\mathbb{E}\|\nabla_x f(x)\|$ near the data
- Both losses are MMD with $k_\psi(x, y) = \phi_\psi(x)\phi_\psi(y)$
  - $$\min_\theta \left[\mathcal{D}^{\Psi}_{\mathrm{MMD}}(\mathbb{P}, \mathbb{Q}_\theta) = \sup_{\psi \in \Psi} \mathrm{MMD}_\psi(\mathbb{P}, \mathbb{Q}_\theta) \right]$$

WGANs and MMD GANs

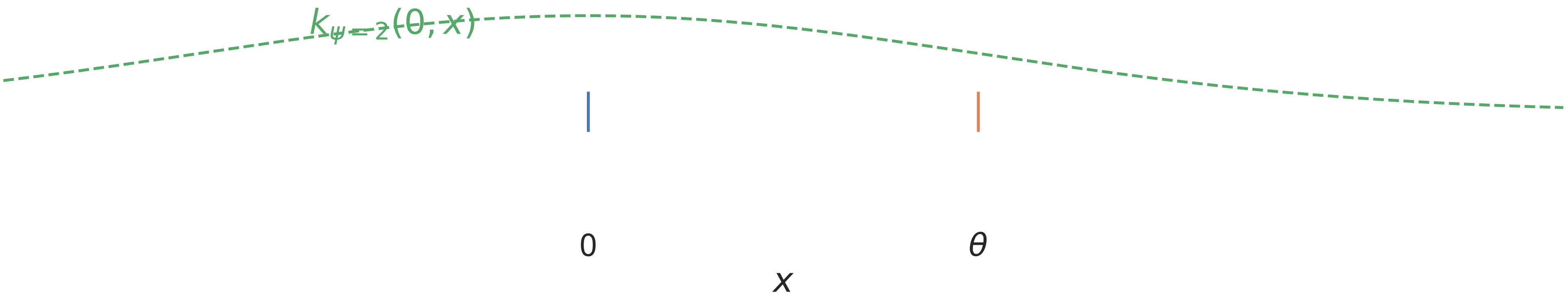
- Integral probability metrics with “smooth” $\mathcal{F}$ are continuous
- WGAN: $\mathcal{F}$ a set of neural networks satisfying $\|f\|_L \leq 1$
- WGAN-GP: instead penalize $\mathbb{E}\|\nabla_x f(x)\|$ near the data
- Both losses are MMD with $k_\psi(x, y) = \phi_\psi(x)\phi_\psi(y)$
  - $$\min_\theta \left[\mathcal{D}_{\mathrm{MMD}}^{\Psi}(\mathbb{P}, \mathbb{Q}_\theta) = \sup_{\psi \in \Psi} \mathrm{MMD}_\psi(\mathbb{P}, \mathbb{Q}_\theta) \right]$$
- Some kind of constraint on ϕ_ψ is important!

Non-smoothness of plain MMD GANs

Illustrative problem in $\mathbb{R}$, DiracGAN [Mescheder+ ICML-18]:

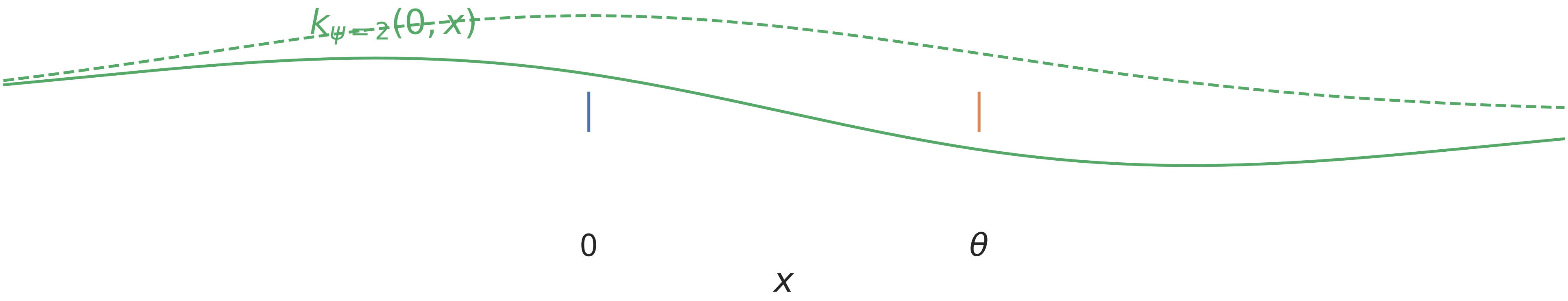
Non-smoothness of plain MMD GANs

Illustrative problem in $\mathbb{R}$, DiracGAN [Mescheder+ ICML-18]:



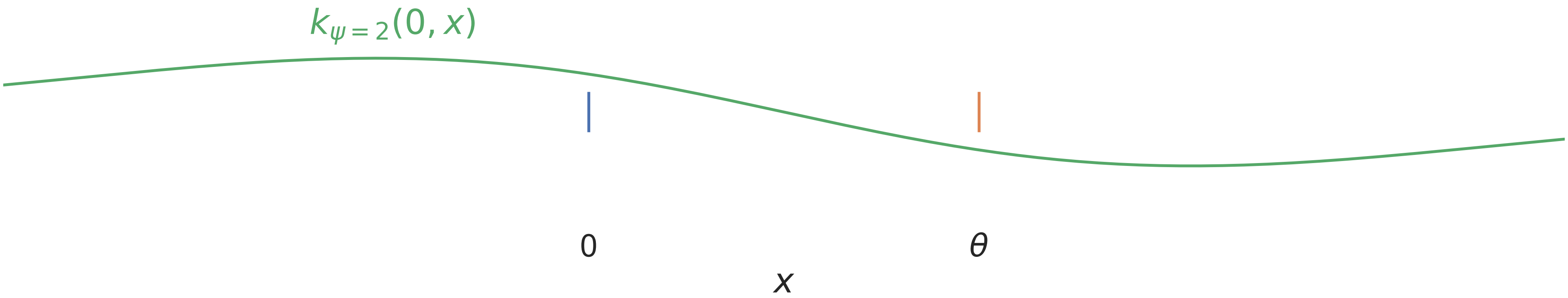
Non-smoothness of plain MMD GANs

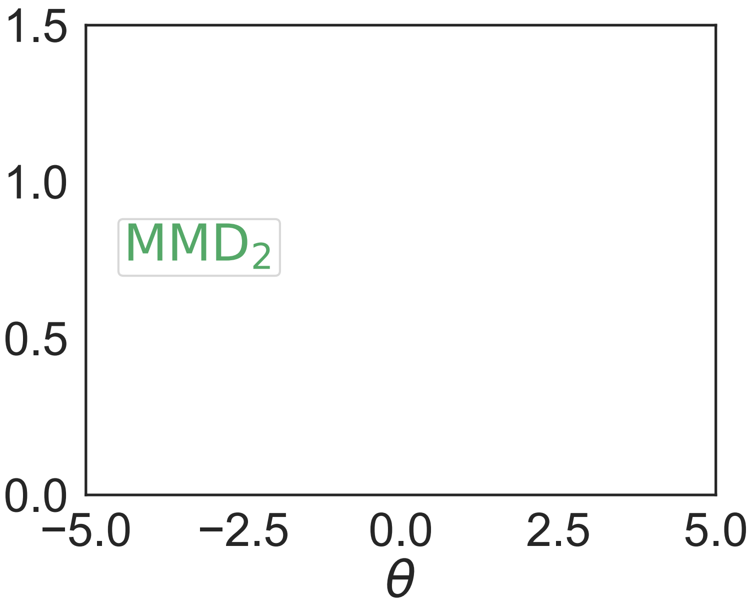
Illustrative problem in $\mathbb{R}$, DiracGAN [Mescheder+ ICML-18]:



Non-smoothness of plain MMD GANs

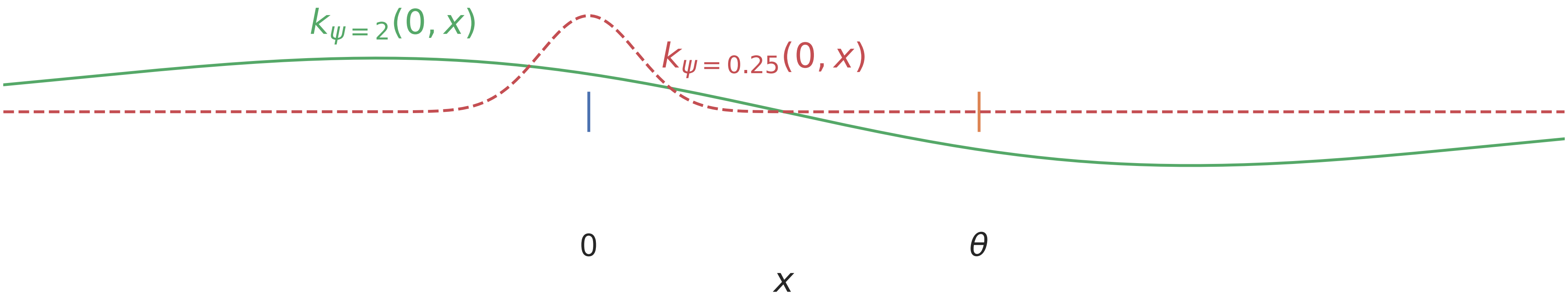
Illustrative problem in $\mathbb{R}$, DiracGAN [Mescheder+ ICML-18]:

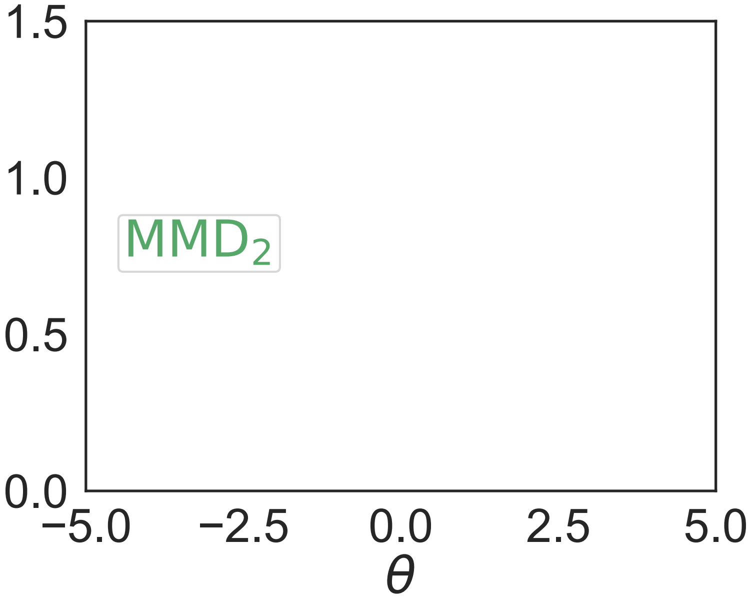




Non-smoothness of plain MMD GANs

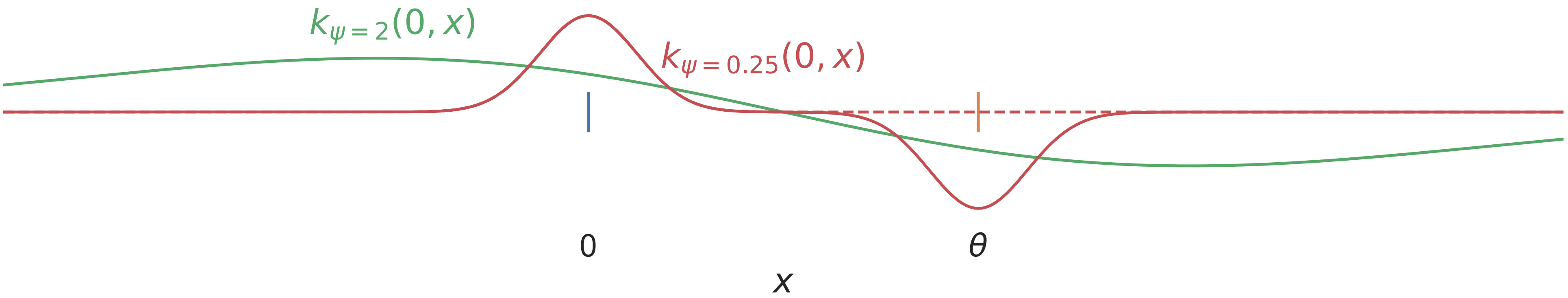
Illustrative problem in $\mathbb{R}$, DiracGAN [Mescheder+ ICML-18]:

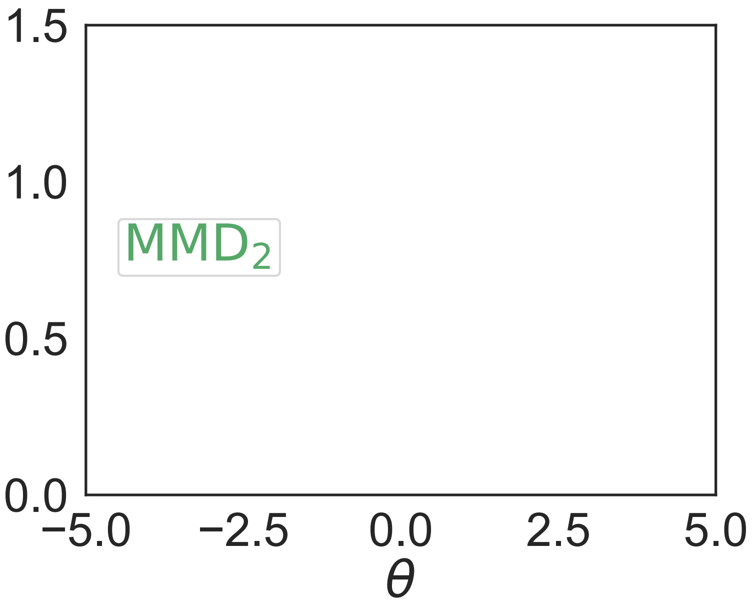




Non-smoothness of plain MMD GANs

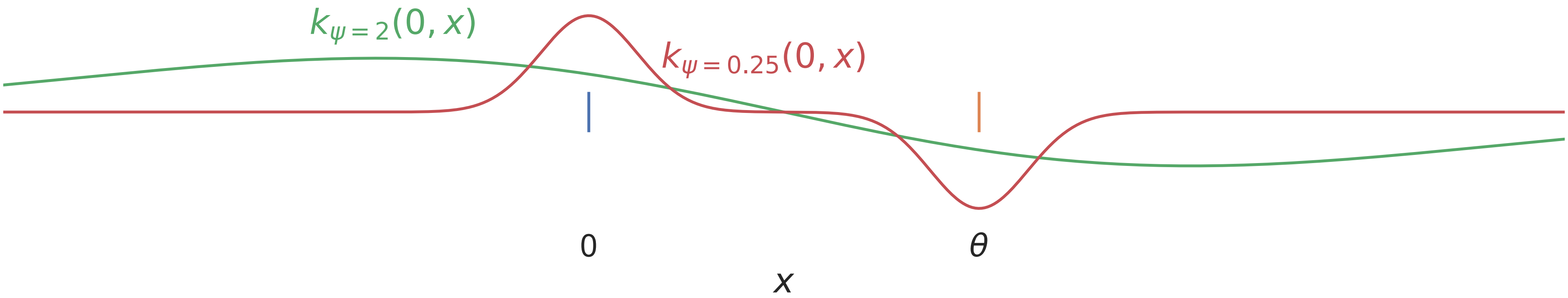
Illustrative problem in $\mathbb{R}$, DiracGAN [Mescheder+ ICML-18]:

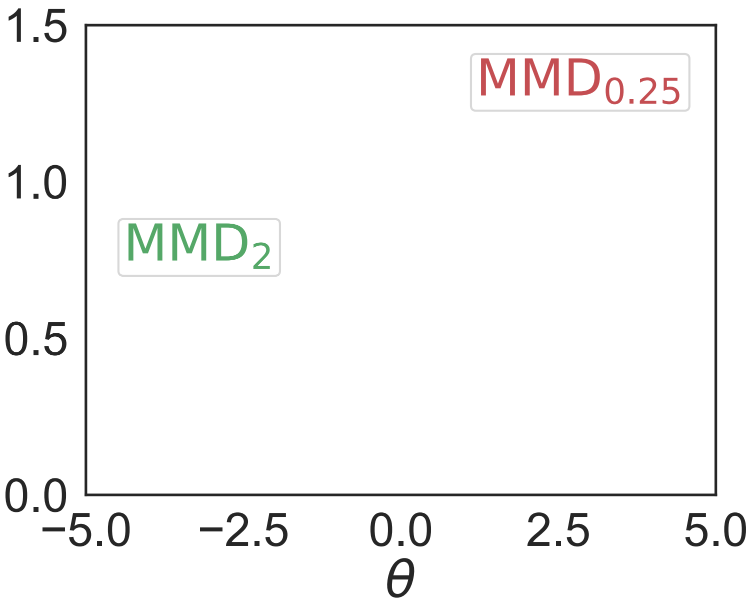




Non-smoothness of plain MMD GANs

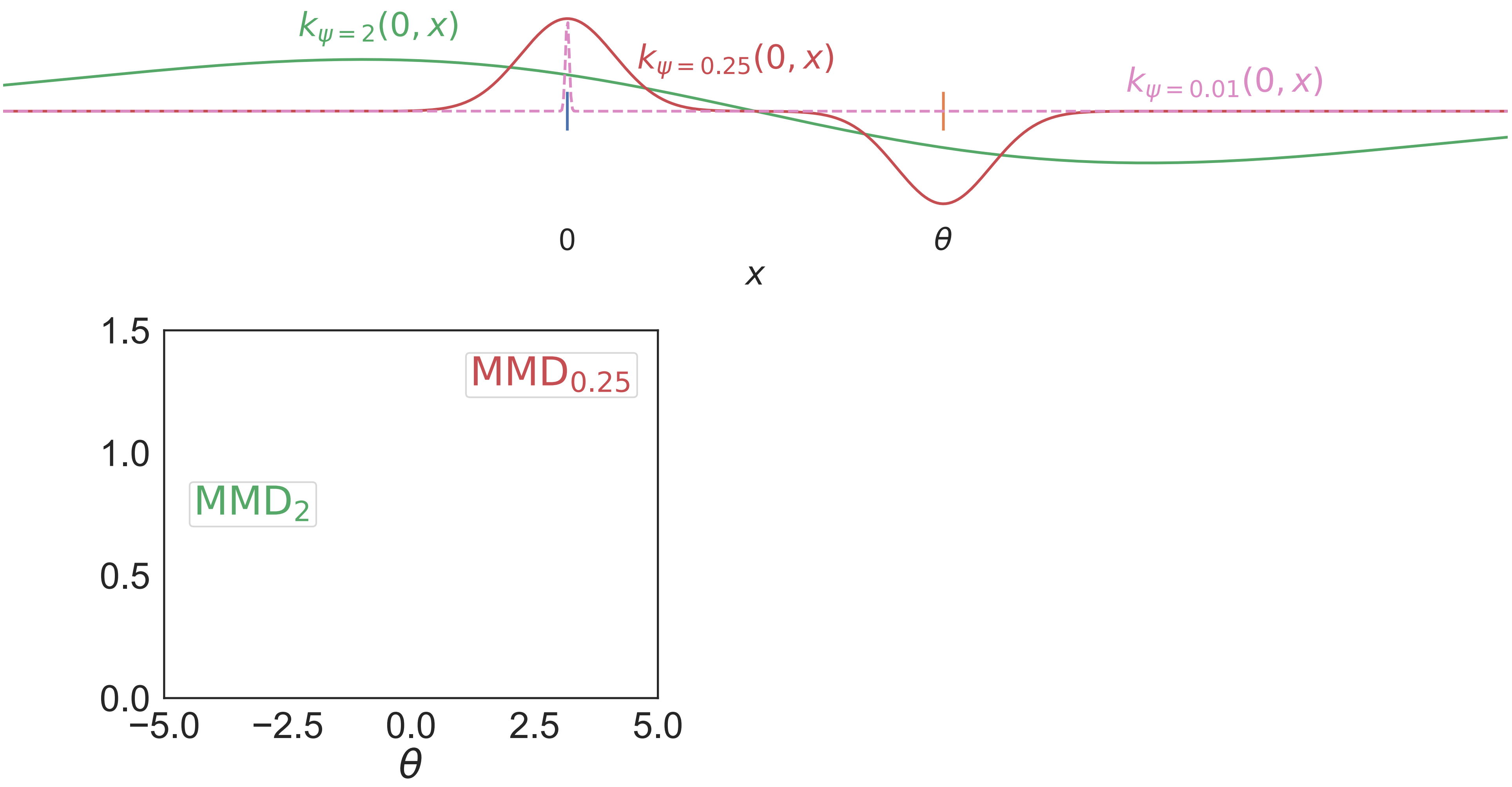
Illustrative problem in $\mathbb{R}$, DiracGAN [Mescheder+ ICML-18]:





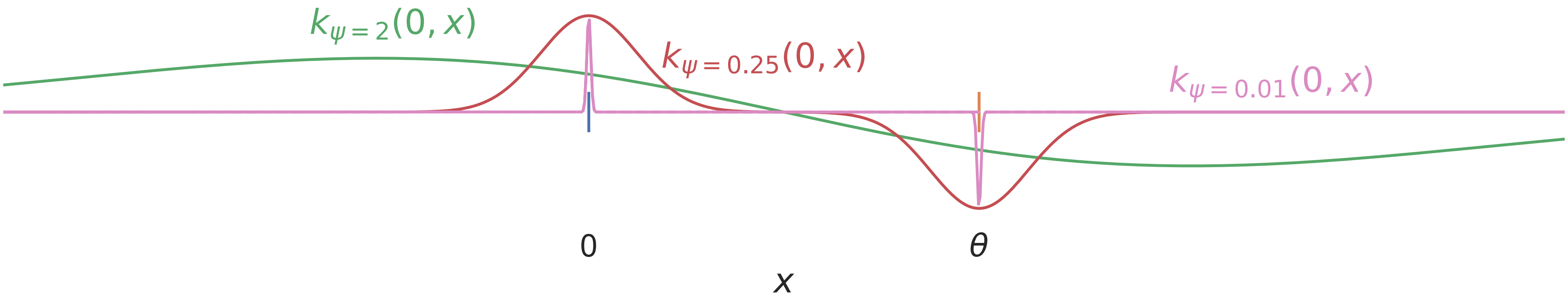
Non-smoothness of plain MMD GANs

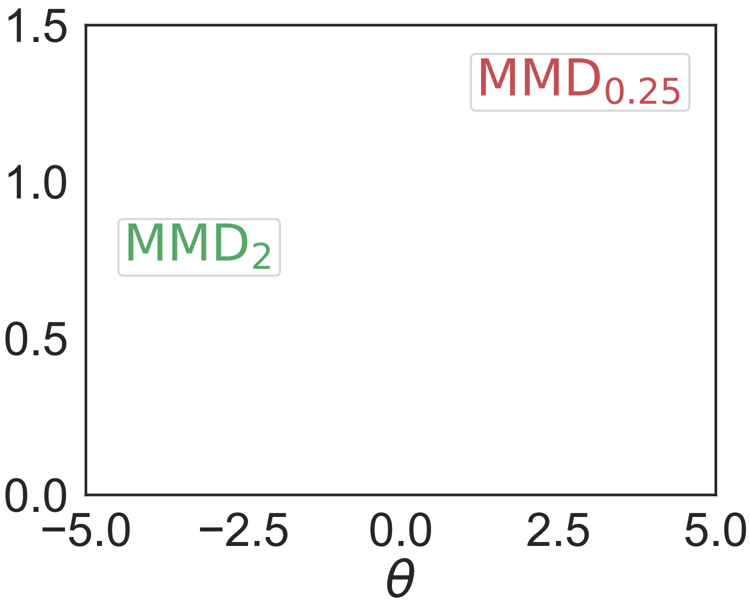
Illustrative problem in $\mathbb{R}$, DiracGAN [Mescheder+ ICML-18]:



Non-smoothness of plain MMD GANs

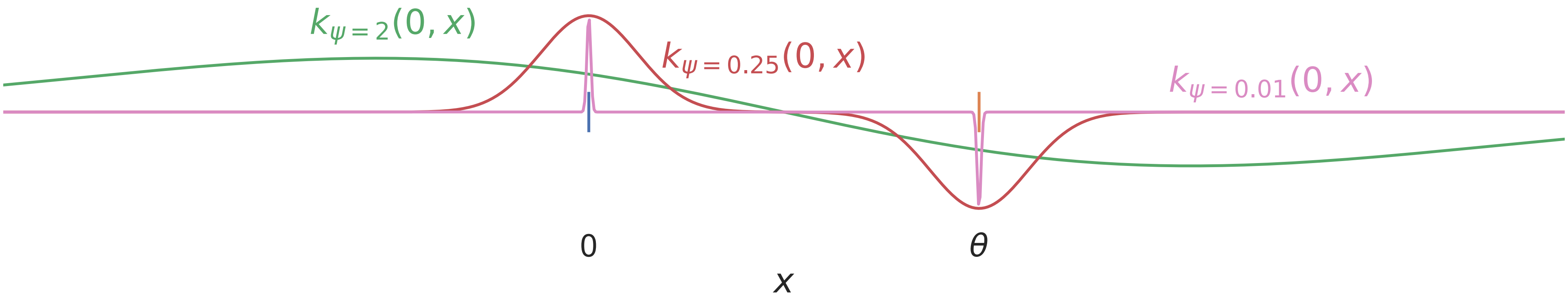
Illustrative problem in $\mathbb{R}$, DiracGAN [Mescheder+ ICML-18]:

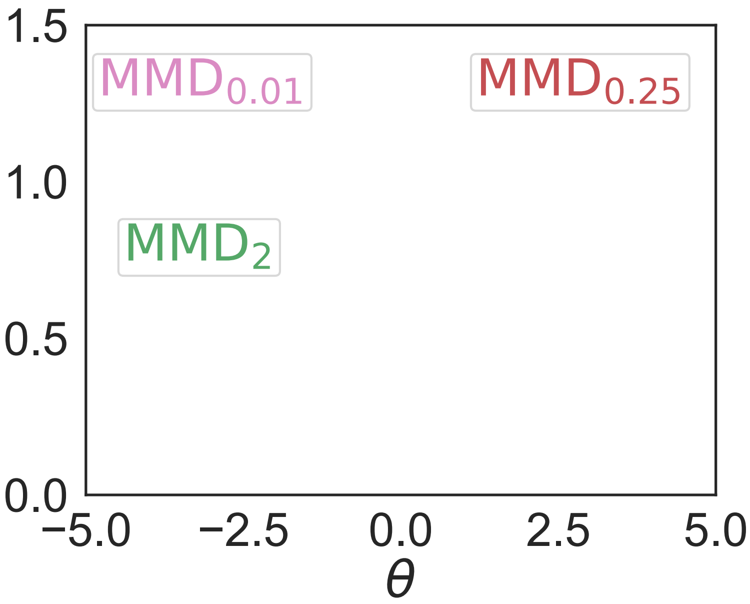




Non-smoothness of plain MMD GANs

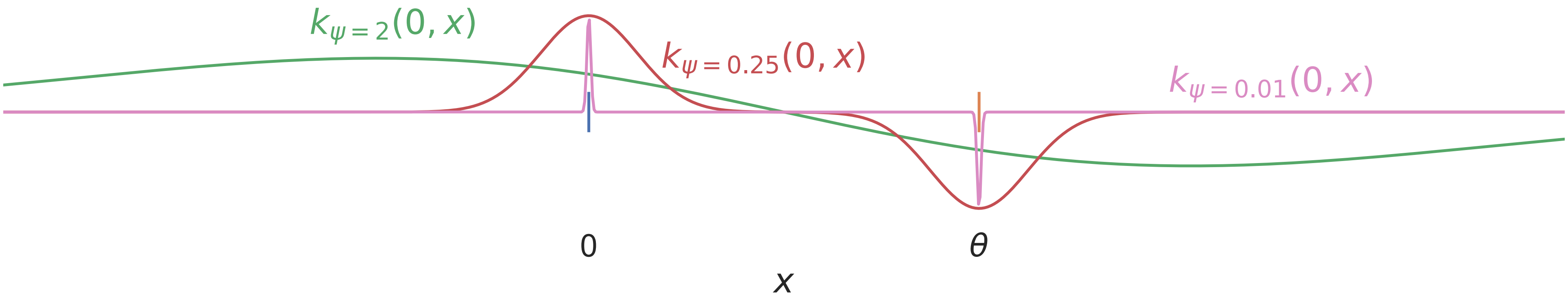
Illustrative problem in $\mathbb{R}$, DiracGAN [Mescheder+ ICML-18]:

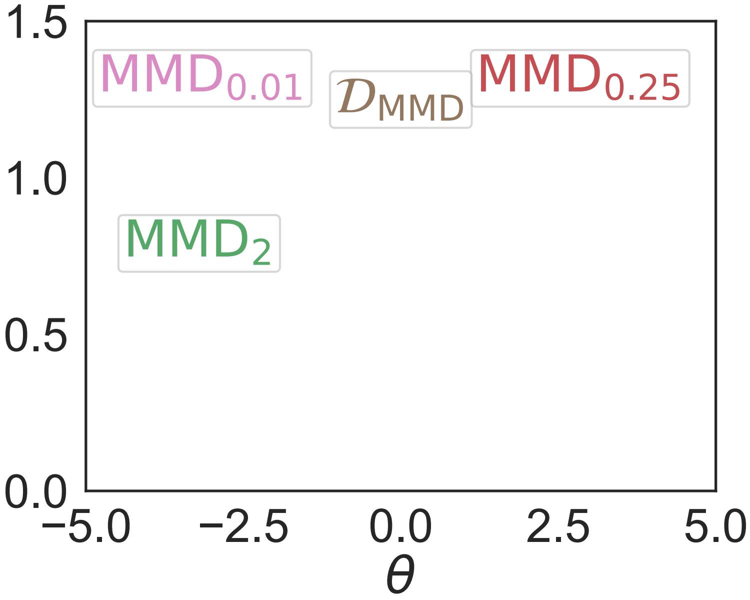




Non-smoothness of plain MMD GANs

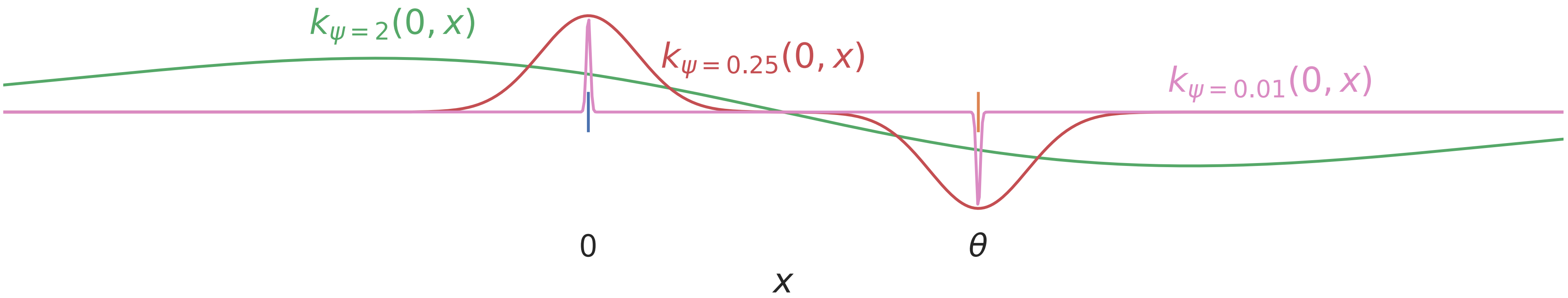
Illustrative problem in $\mathbb{R}$, DiracGAN [Mescheder+ ICML-18]:

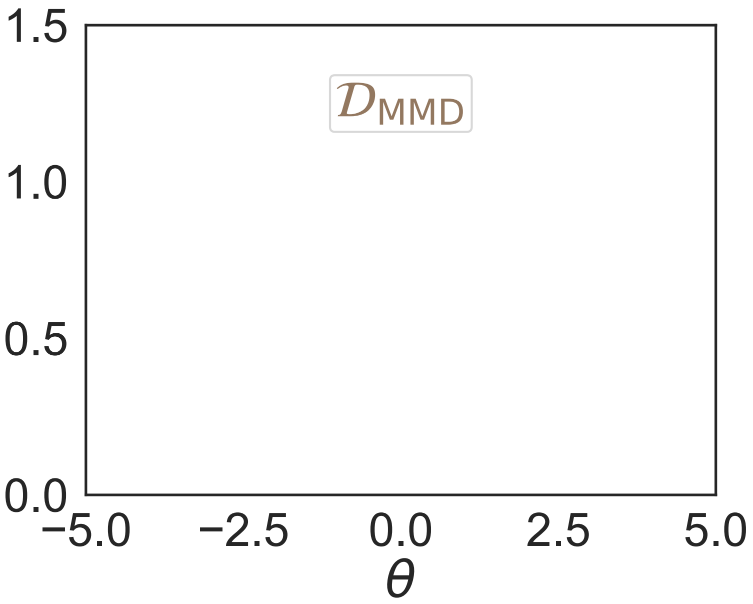




Non-smoothness of plain MMD GANs

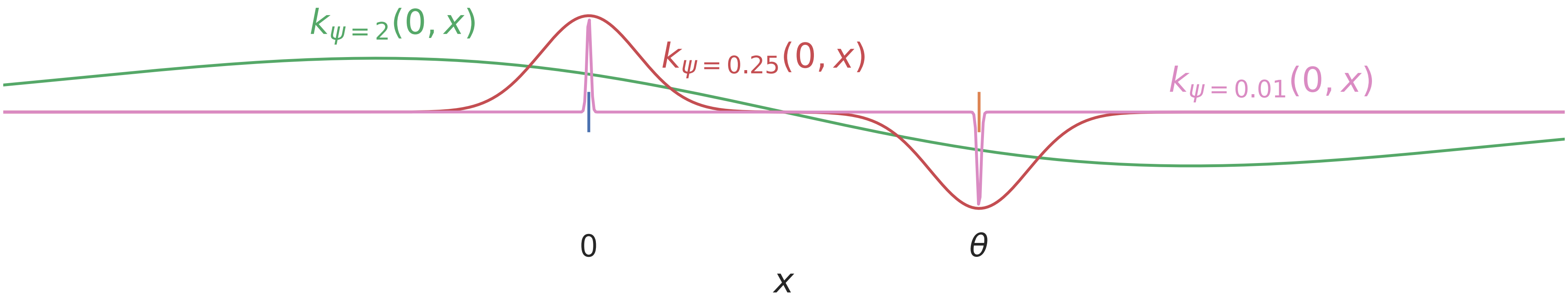
Illustrative problem in $\mathbb{R}$, DiracGAN [Mescheder+ ICML-18]:

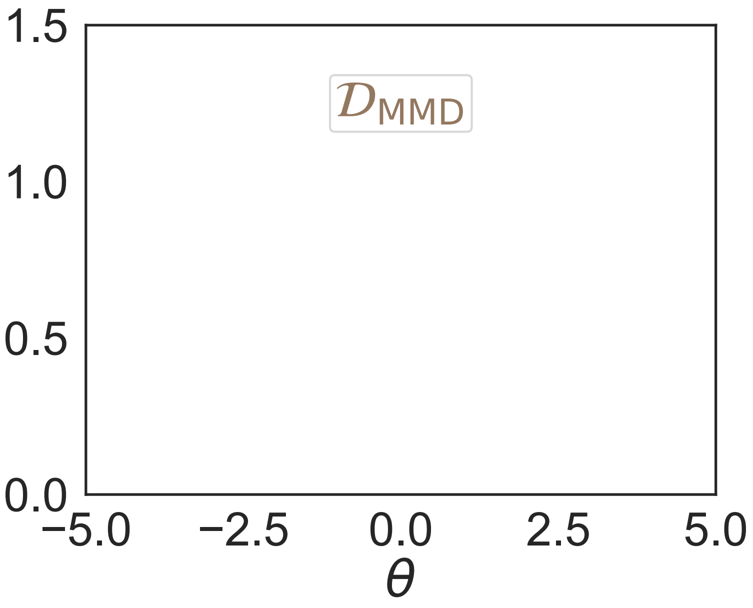


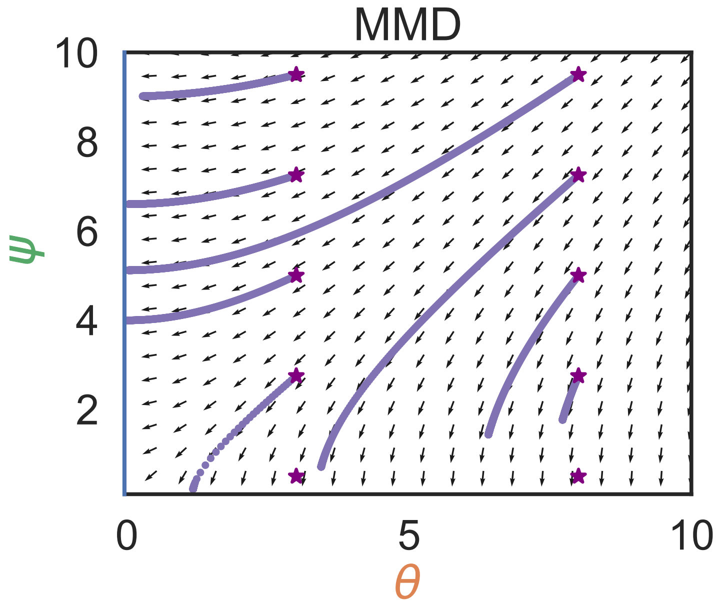


Non-smoothness of plain MMD GANs

Illustrative problem in $\mathbb{R}$, DiracGAN [Mescheder+ ICML-18]:

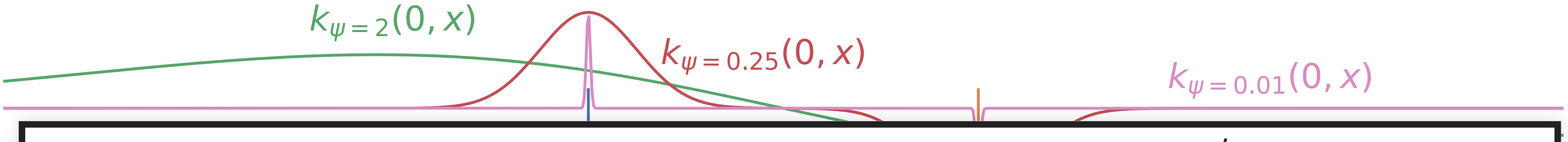






Non-smoothness of plain MMD GANs

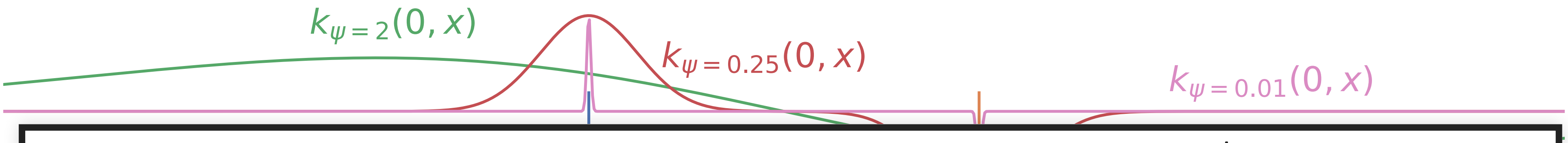
Illustrative problem in $\mathbb{R}$, DiracGAN [Mescheder+ ICML-18]:



- Just need to stay away from tiny bandwidths ψ

Non-smoothness of plain MMD GANs

Illustrative problem in $\mathbb{R}$, DiracGAN [Mescheder+ ICML-18]:

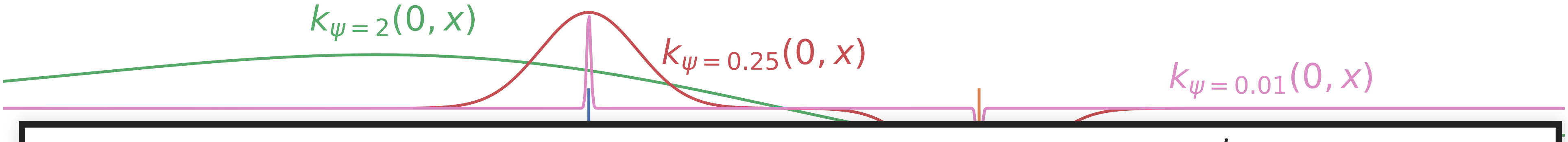


- Just need to stay away from tiny bandwidths ψ
- …deep kernel analogue is hard.

θ θ

Non-smoothness of plain MMD GANs

Illustrative problem in $\mathbb{R}$, DiracGAN [Mescheder+ ICML-18]:

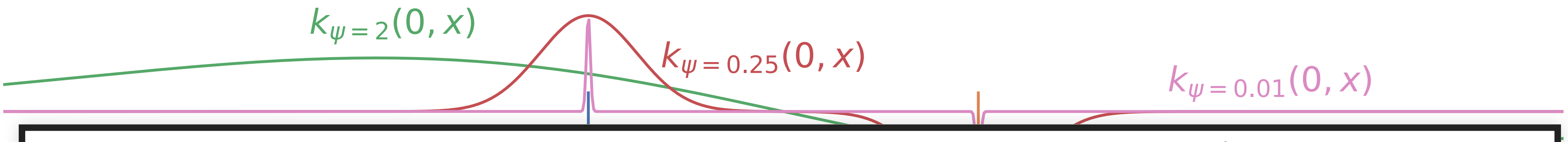


- Just need to stay away from tiny bandwidths ψ
- ...deep kernel analogue is hard.
- Instead, keep witness function from being too steep

θ θ

Non-smoothness of plain MMD GANs

Illustrative problem in $\mathbb{R}$, DiracGAN [Mescheder+ ICML-18]:

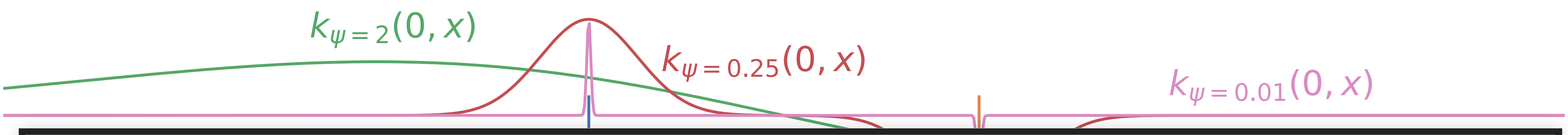


- Just need to stay away from tiny bandwidths ψ
- …deep kernel analogue is hard.
- Instead, keep witness function from being too steep
- $\sup_x \|\nabla f(x)\|$ would give Wasserstein
  - Nice distance, but hard to estimate

θ θ

Non-smoothness of plain MMD GANs

Illustrative problem in $\mathbb{R}$, DiracGAN [Mescheder+ ICML-18]:



- Just need to stay away from tiny bandwidths ψ
- ...deep kernel analogue is hard.
- Instead, keep witness function from being too steep
- $\sup_x \|\nabla f(x)\|$ would give Wasserstein
  - Nice distance, but hard to estimate
- Control $\|\nabla f(\tilde{X})\|$ *on average, near the data*
  - [Gulrajani+ NeurIPS-17 / Roth+ NeurIPS-17 / Mescheder+ ICML-18]

θ θ

MMD-GAN with gradient control

- If Ψ gives uniformly Lipschitz critics, $\mathcal{D}^{\Psi}_{\mathrm{MMD}}$ is smooth
- Original MMD-GAN paper [Li+ NeurIPS-17]: box constraint

MMD-GAN with gradient control

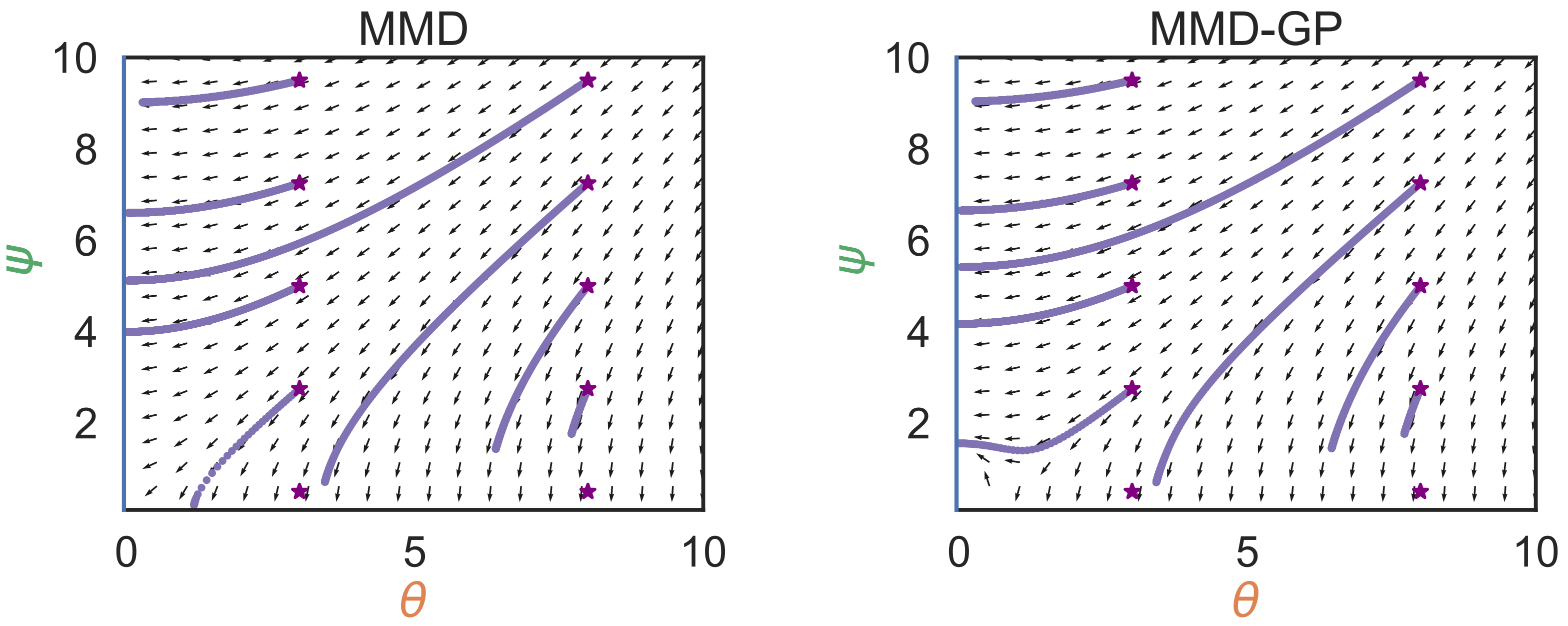
- If Ψ gives uniformly Lipschitz critics, $\mathcal{D}^{\Psi}_{\mathrm{MMD}}$ is smooth
- Original MMD-GAN paper [Li+ NeurIPS-17]: box constraint
- We [Bińkowski+ ICLR-18] used gradient penalty on critic instead

MMD-GAN with gradient control

- If Ψ gives uniformly Lipschitz critics, $\mathcal{D}^{\Psi}_{\mathrm{MMD}}$ is smooth
- Original MMD-GAN paper [Li+ NeurIPS-17]: box constraint
- We [Bińkowski+ ICLR-18] used gradient penalty on critic instead
  - Better in practice, but doesn't fix the Dirac problem...

MMD-GAN with gradient control

- If Ψ gives uniformly Lipschitz critics, $\mathcal{D}^{\Psi}_{\mathrm{MMD}}$ is smooth
- Original MMD-GAN paper [Li+ NeurIPS-17]: box constraint
- We [Bińkowski+ ICLR-18] used gradient penalty on critic instead
  - Better in practice, but doesn't fix the Dirac problem...



New distance: Scaled MMD

Want to ensure $\mathbb{E}_{\tilde{X} \sim \mathbb{S}}[\|\nabla f(\tilde{X})\|^2] \leq 1$

New distance: Scaled MMD

Want to ensure $\mathbb{E}_{\tilde{X} \sim \mathbb{S}}[\|\nabla f(\tilde{X})\|^2] \leq 1$

Can solve with $\langle \partial_i \phi(x), f \rangle_{\mathcal{H}} = \partial_i f(x)$...but too expensive!

New distance: Scaled MMD

Want to ensure $\mathbb{E}_{\tilde{X}\sim\mathbb{S}}[\|\nabla f(\tilde{X})\|^2] \leq 1$

Can solve with $\langle \partial_i \phi(x), f \rangle_{\mathcal{H}} = \partial_i f(x)$...but too expensive!

Guaranteed if $\|f\|_{\mathcal{H}} \leq \sigma_{\mathbb{S},k,\lambda}$

$$\sigma_{\mathbb{S},k,\lambda} := \left(\lambda + \mathop{\mathbb{E}}_{\tilde{X}\sim\mathbb{S}} \left[k(\tilde{X}, \tilde{X}) + [\nabla_1 \cdot \nabla_2 k](\tilde{X}, \tilde{X}) \right] \right)^{-\frac{1}{2}}$$

New distance: Scaled MMD

Want to ensure $\mathbb{E}_{\tilde{X}\sim\mathbb{S}}[\|\nabla f(\tilde{X})\|^2] \leq 1$

Can solve with $\langle \partial_i \phi(x), f \rangle_{\mathcal{H}} = \partial_i f(x)$...but too expensive!

Guaranteed if $\|f\|_{\mathcal{H}} \leq \sigma_{\mathbb{S},k,\lambda}$

$$\sigma_{\mathbb{S},k,\lambda} := \left(\lambda + \mathop{\mathbb{E}}_{\tilde{X}\sim\mathbb{S}} \left[k(\tilde{X}, \tilde{X}) + [\nabla_1 \cdot \nabla_2 k](\tilde{X}, \tilde{X}) \right] \right)^{-\frac{1}{2}}$$

Gives distance $\mathrm{SMMD}_{\mathbb{S},k,\lambda}(\mathbb{P}, \mathbb{Q}) = \sigma_{\mathbb{S},k,\lambda} \, \mathrm{MMD}_k(\mathbb{P}, \mathbb{Q})$

New distance: Scaled MMD

Want to ensure $\mathbb{E}_{\tilde{X}\sim\mathbb{S}}[\|\nabla f(\tilde{X})\|^2] \leq 1$

Can solve with $\langle \partial_i \phi(x), f \rangle_{\mathcal{H}} = \partial_i f(x)$...but too expensive!

Guaranteed if $\|f\|_{\mathcal{H}} \leq \sigma_{\mathbb{S},k,\lambda}$

$$\sigma_{\mathbb{S},k,\lambda} := \left(\lambda + \mathop{\mathbb{E}}_{\tilde{X}\sim\mathbb{S}} \left[k(\tilde{X}, \tilde{X}) + [\nabla_1 \cdot \nabla_2 k](\tilde{X}, \tilde{X}) \right] \right)^{-\frac{1}{2}}$$

Gives distance $\mathrm{SMMD}_{\mathbb{S},k,\lambda}(\mathbb{P}, \mathbb{Q}) = \sigma_{\mathbb{S},k,\lambda}\, \mathrm{MMD}_k(\mathbb{P}, \mathbb{Q})$

$$\mathcal{D}^{\Psi}_{\mathrm{MMD}} \text{ has } \mathcal{F} = \bigcup_{\psi\in\Psi} \left\{ f : \|f\|_{\mathcal{H}_\psi} \leq 1 \right\}$$

$$\mathcal{D}^{\mathbb{S},\Psi,\lambda}_{\mathrm{SMMD}} \text{ has } \mathcal{F} = \bigcup_{\psi\in\Psi} \left\{ f : \|f\|_{\mathcal{H}_\psi} \leq \sigma_{\mathbb{S},k,\lambda} \right\}$$

Deriving the Scaled MMD

$$\mathop{\mathbb{E}}_{\tilde{X}\sim\mathbb{S}}\left[\|\nabla f(\tilde{X})\|^2\right] \leq 1$$

Deriving the Scaled MMD

$$\mathop{\mathbb{E}}_{\tilde{X}\sim\mathbb{S}}[f(\tilde{X})^2] + \mathop{\mathbb{E}}_{\tilde{X}\sim\mathbb{S}}[\|\nabla f(\tilde{X})\|^2] + \lambda\|f\|_{\mathcal{H}}^2 \leq 1$$

Deriving the Scaled MMD

$$\mathop{\mathbb{E}}_{\tilde{X}\sim\mathbb{S}}[f(\tilde{X})^2] + \mathop{\mathbb{E}}_{\tilde{X}\sim\mathbb{S}}[\|\nabla f(\tilde{X})\|^2] + \lambda\|f\|_{\mathcal{H}}^2 \leq 1$$

$$\mathop{\mathbb{E}}_{\tilde{X}\sim\mathbb{S}}[f(\tilde{X})^2] = \left\langle f, \mathop{\mathbb{E}}_{\tilde{X}\sim\mathbb{S}}\left[k(\tilde{X},\cdot)\otimes k(\tilde{X},\cdot)\right] f \right\rangle$$

Deriving the Scaled MMD

$$\mathop{\mathbb{E}}_{\tilde{X}\sim\mathbb{S}}[f(\tilde{X})^2] + \mathop{\mathbb{E}}_{\tilde{X}\sim\mathbb{S}}[\|\nabla f(\tilde{X})\|^2] + \lambda\|f\|_{\mathcal{H}}^2 \leq 1$$

$$\mathop{\mathbb{E}}_{\tilde{X}\sim\mathbb{S}}[f(\tilde{X})^2] = \left\langle f, \mathop{\mathbb{E}}_{\tilde{X}\sim\mathbb{S}}\left[k(\tilde{X},\cdot)\otimes k(\tilde{X},\cdot)\right] f \right\rangle$$

$$\mathop{\mathbb{E}}_{\tilde{X}\sim\mathbb{S}}[\|\nabla f(\tilde{X})\|^2] = \left\langle f, \mathop{\mathbb{E}}_{\tilde{X}\sim\mathbb{S}}\left[\sum_{i=1}^{d}\partial_i k(\tilde{X},\cdot)\otimes \partial_i k(\tilde{X},\cdot)\right] f \right\rangle$$

Deriving the Scaled MMD

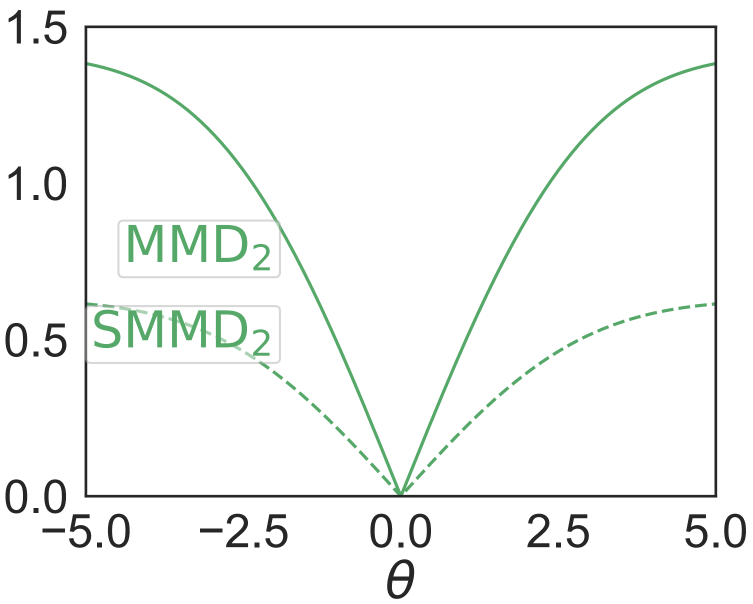
$$\mathop{\mathbb{E}}_{\tilde{X}\sim\mathbb{S}}[f(\tilde{X})^2] + \mathop{\mathbb{E}}_{\tilde{X}\sim\mathbb{S}}[\|\nabla f(\tilde{X})\|^2] + \lambda\|f\|_{\mathcal{H}}^2 \leq 1$$

$$\mathop{\mathbb{E}}_{\tilde{X}\sim\mathbb{S}}[f(\tilde{X})^2] = \left\langle f, \mathop{\mathbb{E}}_{\tilde{X}\sim\mathbb{S}}\left[k(\tilde{X},\cdot)\otimes k(\tilde{X},\cdot)\right] f \right\rangle$$

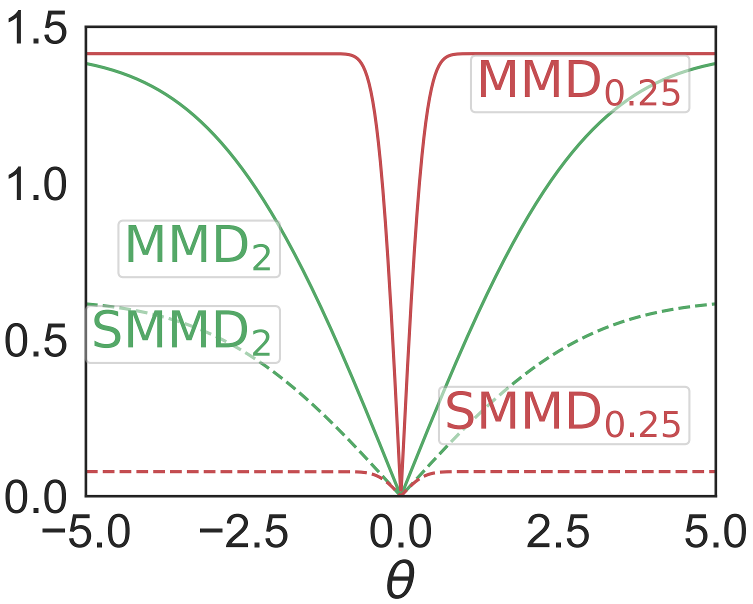
$$\mathop{\mathbb{E}}_{\tilde{X}\sim\mathbb{S}}[\|\nabla f(\tilde{X})\|^2] = \left\langle f, \mathop{\mathbb{E}}_{\tilde{X}\sim\mathbb{S}}\left[\sum_{i=1}^{d} \partial_i k(\tilde{X},\cdot)\otimes \partial_i k(\tilde{X},\cdot)\right] f \right\rangle$$

$$\langle f, D_\lambda f\rangle \leq \|D_\lambda\|\,\|f\|_{\mathcal{H}}^2 \leq \sigma_{\mathbb{S},k,\lambda}^{-2}\|f\|_{\mathcal{H}}^2$$

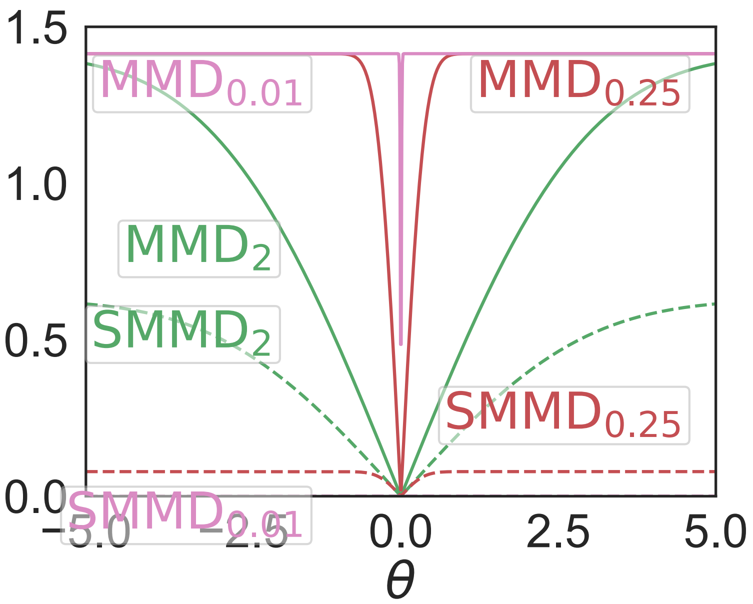
Smoothness of $\mathcal{D}_{\mathrm{SMMD}}$



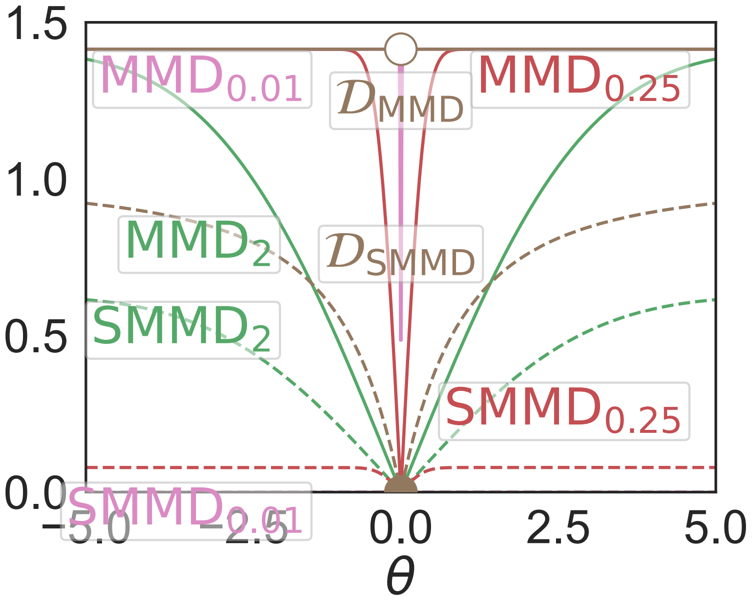
Smoothness of $\mathcal{D}_{\mathrm{SMMD}}$



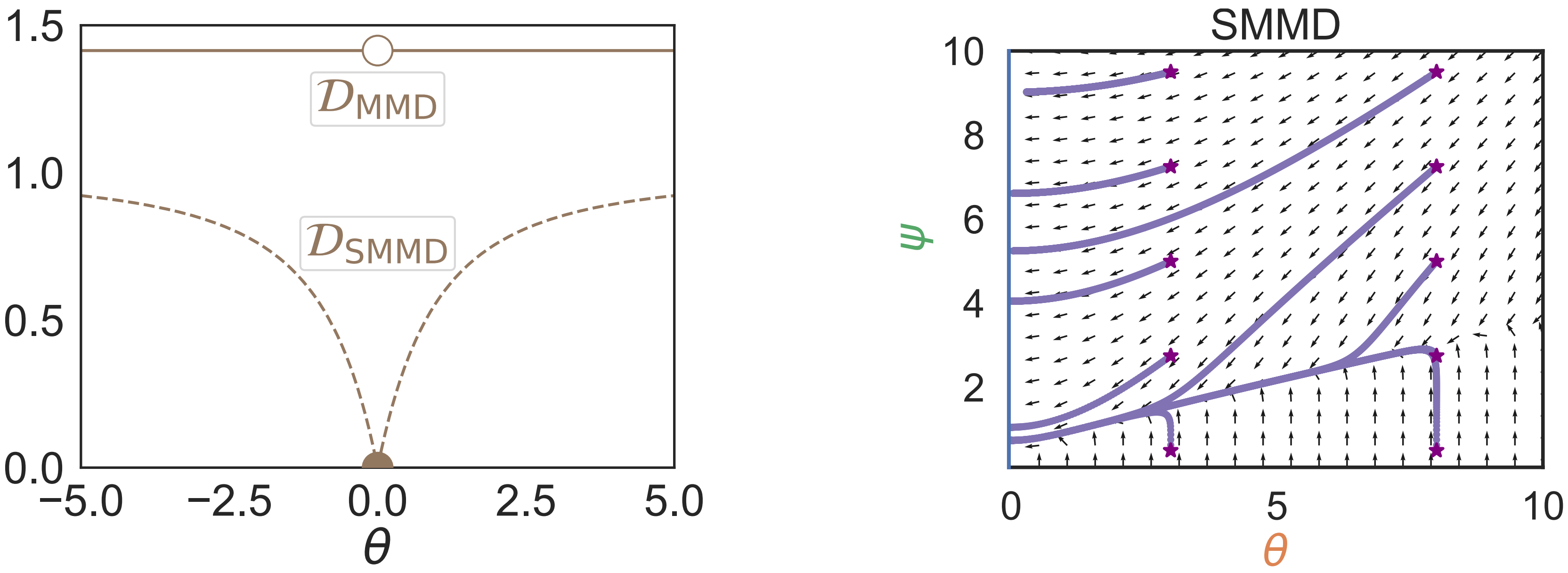
Smoothness of $\mathcal{D}_{\mathrm{SMMD}}$



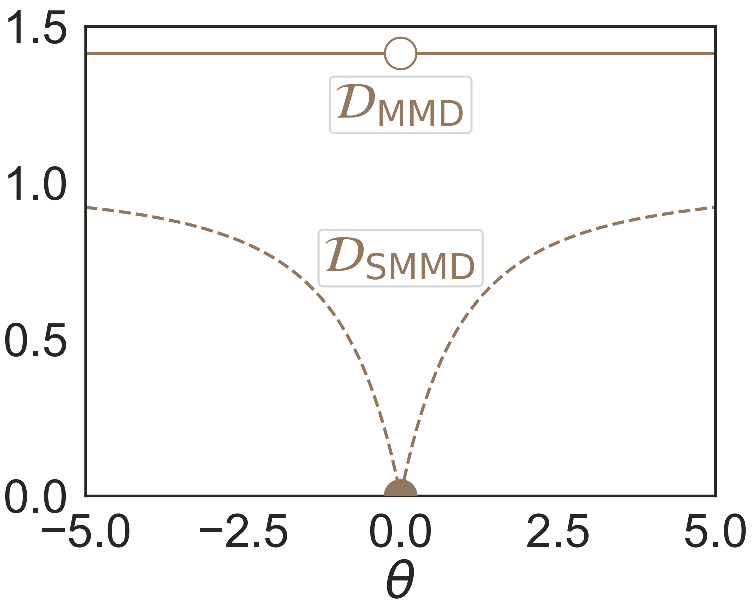
Smoothness of $\mathcal{D}_{\mathrm{SMMD}}$

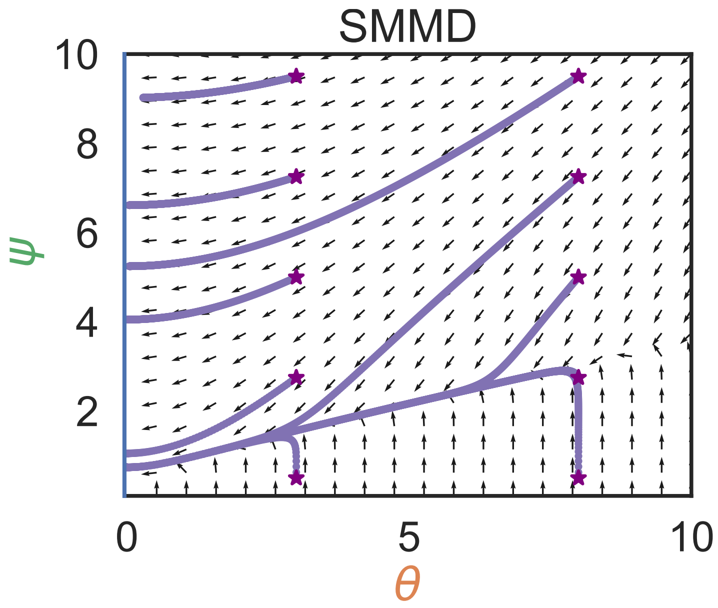


Smoothness of $\mathcal{D}_{\mathrm{SMMD}}$



Smoothness of $\mathcal{D}_{\text{SMMD}}$





Theorem: $\mathcal{D}_{\text{SMMD}}^{\mathbb{S},\Psi,\lambda}$ is continuous.

If $\mathbb{S}$ has a density; k_{top} is Gaussian/linear/...;

ϕ_ψ is fully-connected, Leaky-ReLU, non-increasing width;

all weights in Ψ have bounded condition number; then

$\mathcal{W}(\mathbb{Q}_n, \mathbb{P}) \to 0$ implies $\mathcal{D}_{\text{SMMD}}^{\mathbb{S},\Psi,\lambda}(\mathbb{Q}_n, \mathbb{P}) \to 0.$

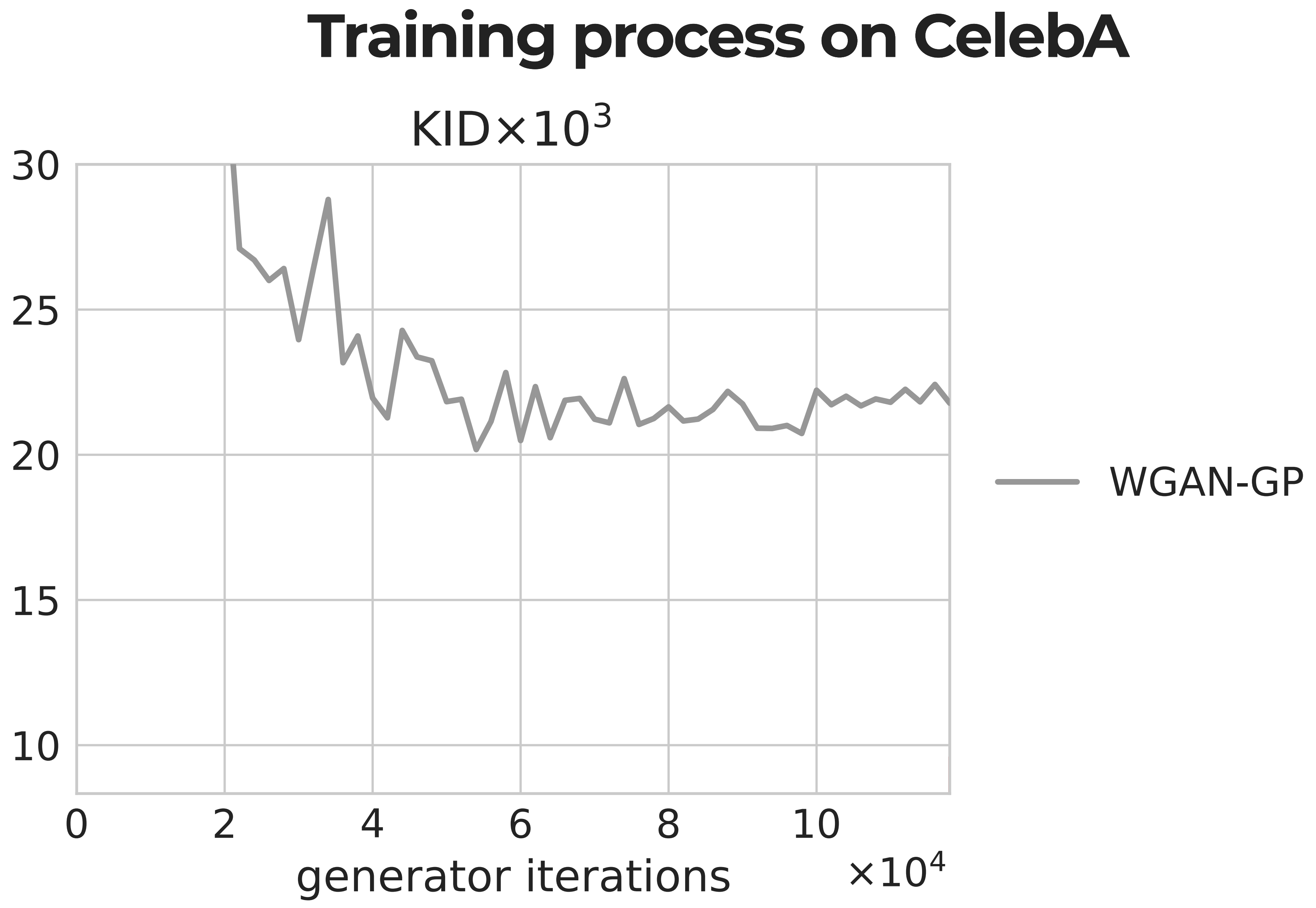
Results on 160×160 CelebA

SN-SMMD-GAN

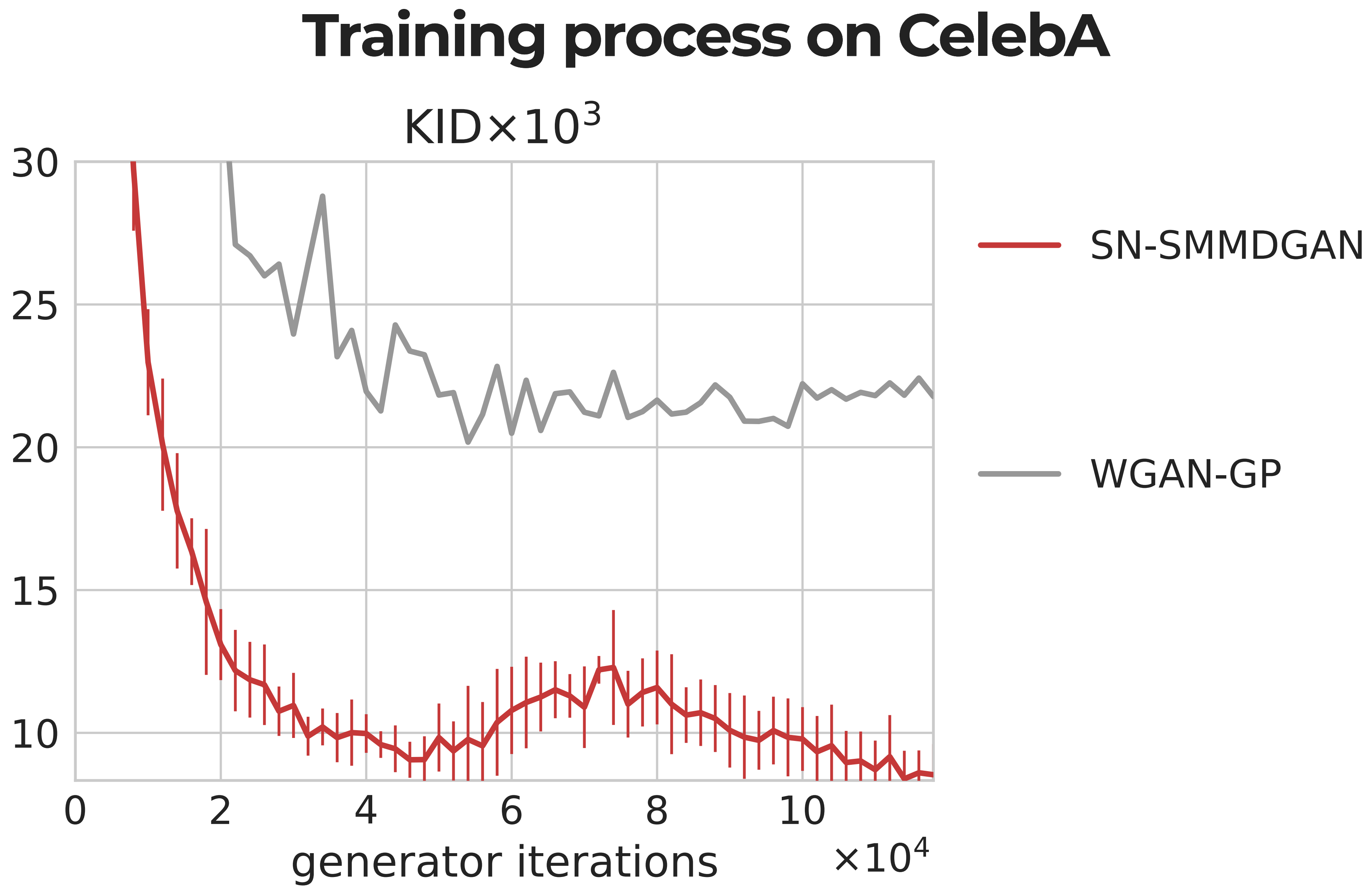
KID: 0.006

WGAN-GP

KID: 0.022

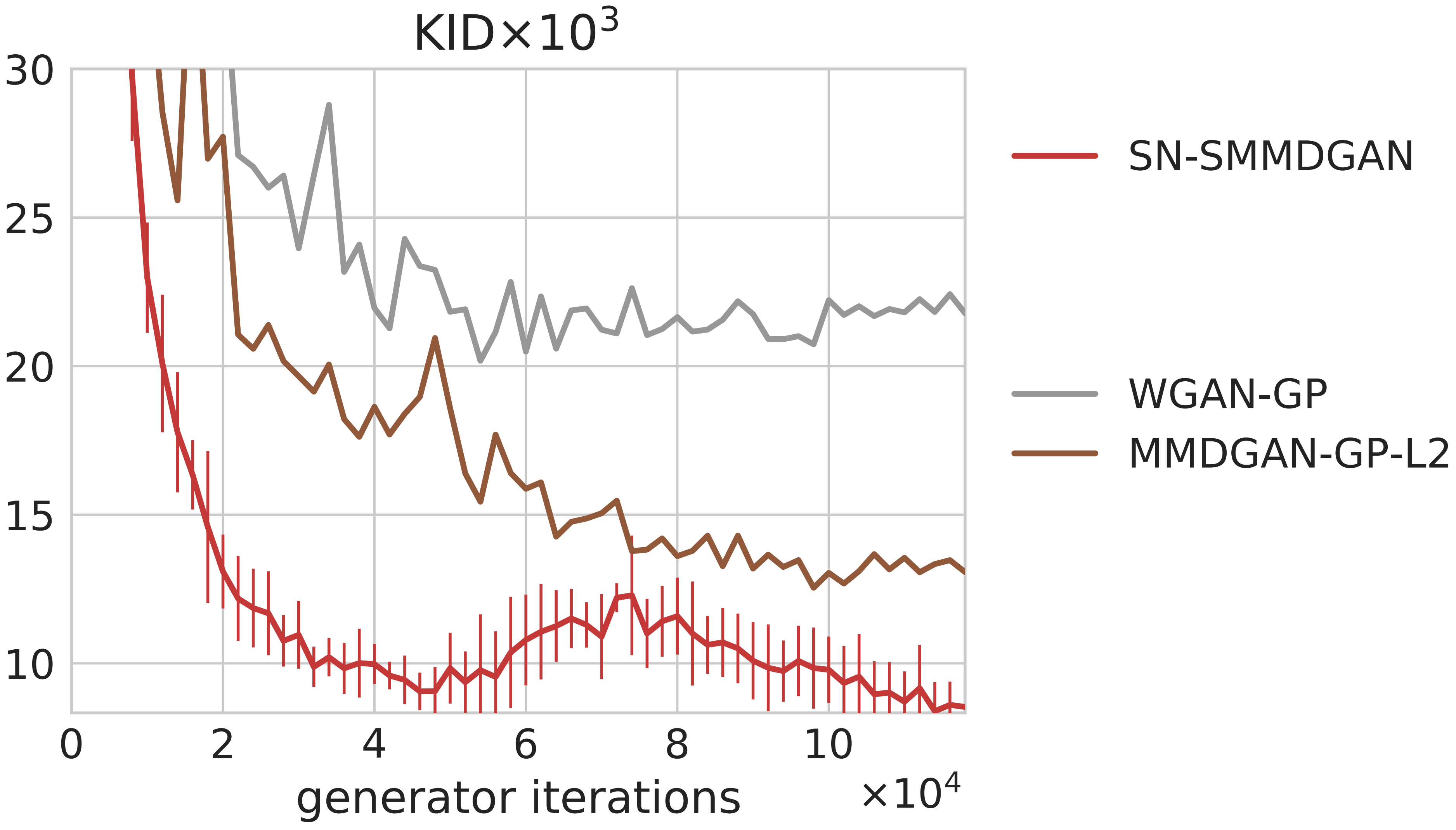


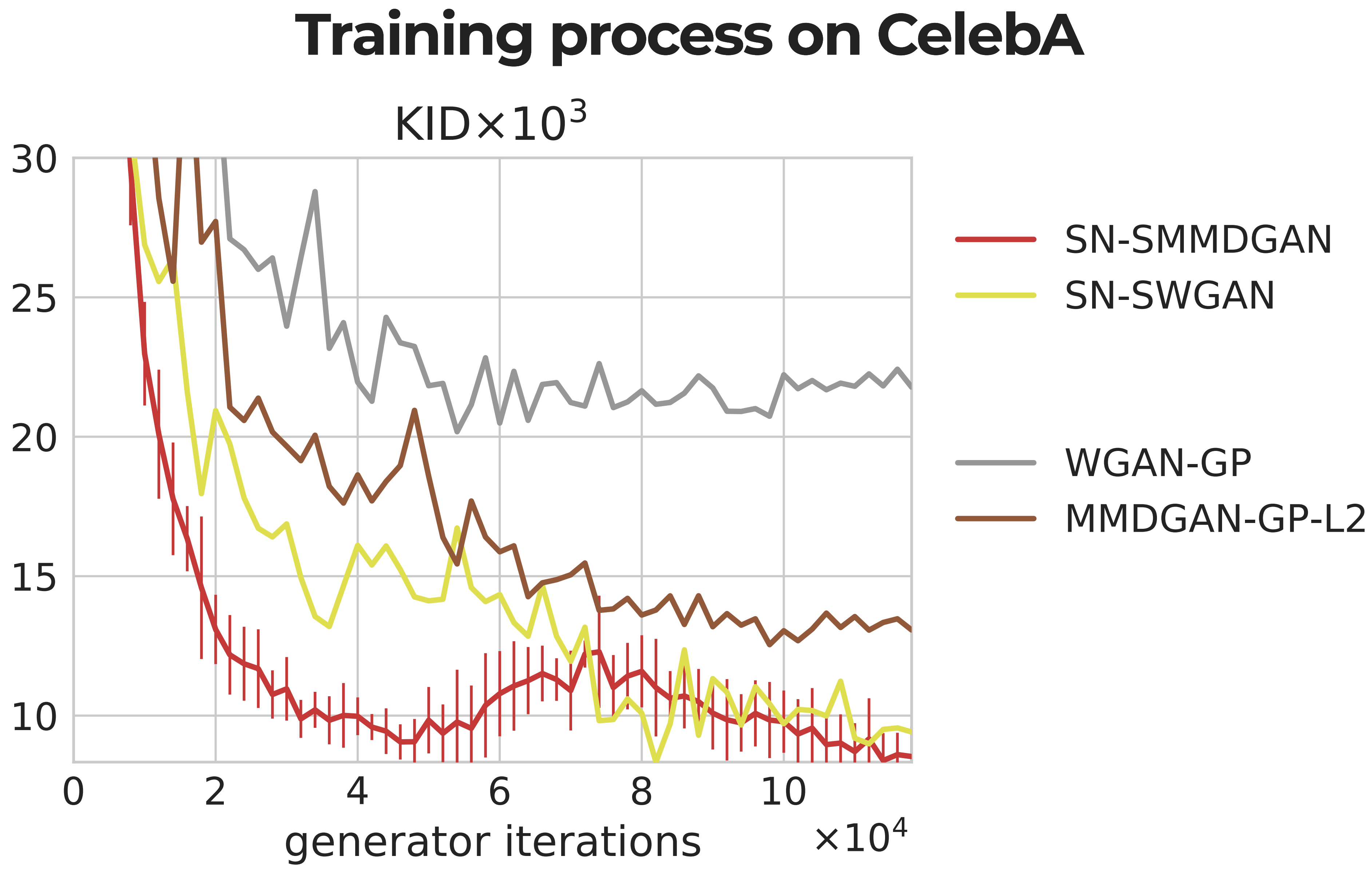
Training process on CelebA
KID×10^3
30
25
20
15
10
0
2
4
6
8
10
×10^4
generator iterations
WGAN-GP

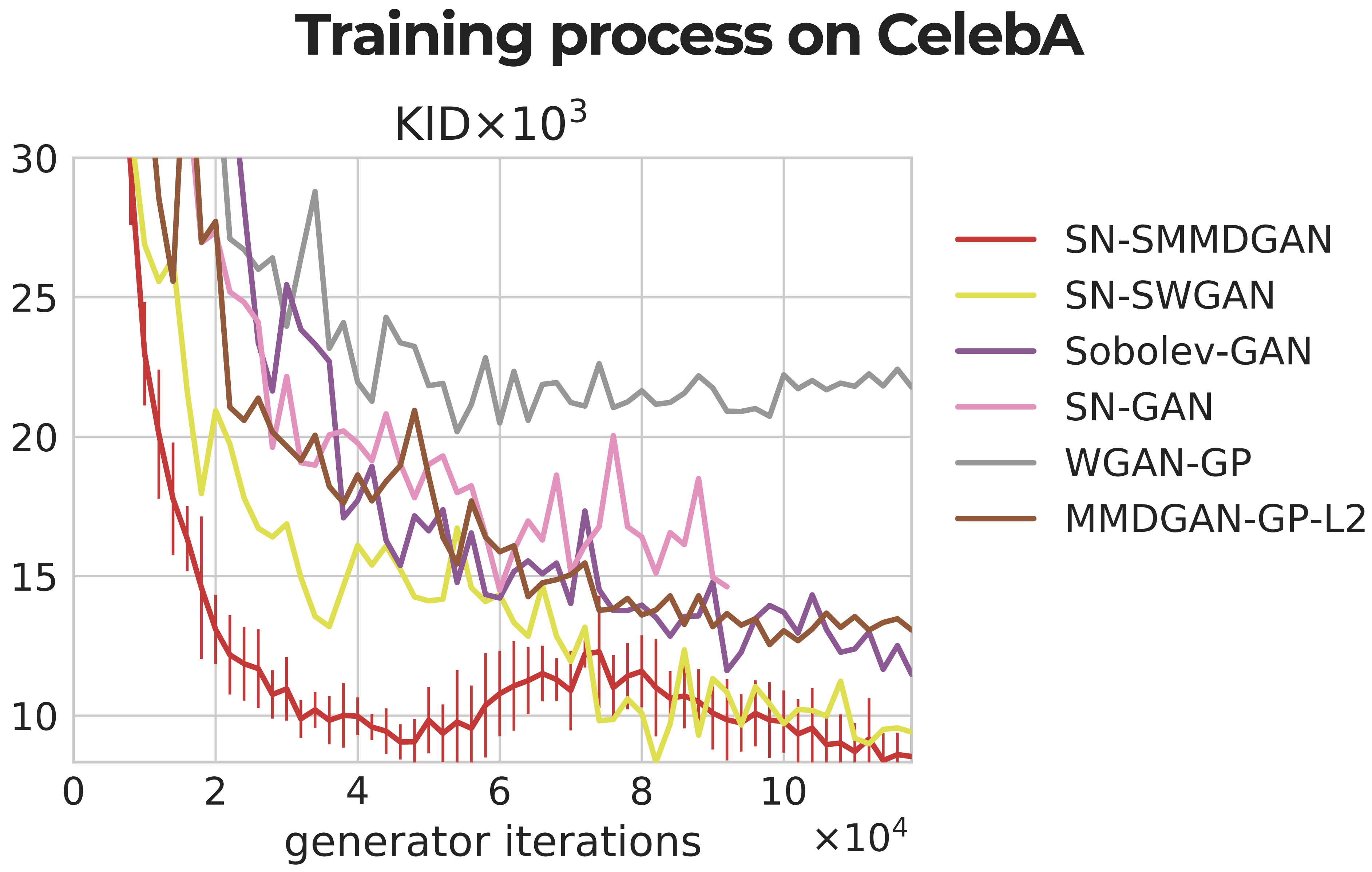


Training process on CelebA
KID×10^3
30
25
20
15
10
0
2
4
6
8
10
generator iterations
×10^4
SN-SMMDGAN
WGAN-GP

Training process on CelebA




Training process on CelebA
KID×10^3
30
25
20
15
10
0
2
4
6
8
10
generator iterations
×10^4
SN-SMMDGAN
SN-SWGAN
WGAN-GP
MMDGAN-GP-L2


Training process on CelebA
KID×10^3
30
25
20
15
10
0
2
4
6
8
10
generator iterations
×10^4
SN-SMMDGAN
SN-SWGAN
Sobolev-GAN
SN-GAN
WGAN-GP
MMDGAN-GP-L2

Evaluating generative models

Evaluating generative models

- Human evaluation: good at precision, bad at recall
- Likelihood: hard for GANs, maybe not right thing anyway
- Two-sample tests: always reject!

Evaluating generative models

- Human evaluation: good at precision, bad at recall
- Likelihood: hard for GANs, maybe not right thing anyway
- Two-sample tests: always reject!
- Most common: Fréchet Inception Distance, FID
  - Run pretrained featurizer on model and target
  - Model each as Gaussian; compute W_2

Evaluating generative models

- Human evaluation: good at precision, bad at recall
- Likelihood: hard for GANs, maybe not right thing anyway
- Two-sample tests: always reject!
- Most common: Fréchet Inception Distance, FID
  - Run pretrained featurizer on model and target
  - Model each as Gaussian; compute W_2
  - Strong bias, small variance: very misleading

Evaluating generative models

- Human evaluation: good at precision, bad at recall
- Likelihood: hard for GANs, maybe not right thing anyway
- Two-sample tests: always reject!
- Most common: Fréchet Inception Distance, FID
  - Run pretrained featurizer on model and target
  - Model each as Gaussian; compute W_2
  - Strong bias, small variance: very misleading
  - Simple examples where $\mathrm{FID}(\mathbb{Q}_1) > \mathrm{FID}(\mathbb{Q}_2)$ but $\widehat{\mathrm{FID}}(\hat{\mathbb{Q}}_1) < \widehat{\mathrm{FID}}(\hat{\mathbb{Q}}_2)$ for reasonable sample size

Evaluating generative models

- Human evaluation: good at precision, bad at recall
- Likelihood: hard for GANs, maybe not right thing anyway
- Two-sample tests: always reject!
- Most common: Fréchet Inception Distance, FID
  - Run pretrained featurizer on model and target
  - Model each as Gaussian; compute W_2
  - Strong bias, small variance: very misleading
  - Simple examples where $\mathrm{FID}(\mathbb{Q}_1) > \mathrm{FID}(\mathbb{Q}_2)$ but $\widehat{\mathrm{FID}}(\hat{\mathbb{Q}}_1) < \widehat{\mathrm{FID}}(\hat{\mathbb{Q}}_2)$ for reasonable sample size
- Our KID: MMD^2 instead. Unbiased, asymptotically normal

Recap

Combining a deep architecture with a kernel machine that takes the higher-level learned representation as input can be quite powerful.

— Y. Bengio & Y. LeCun (2007), "Scaling Learning Algorithms towards AI"

Recap

Combining a deep architecture with a kernel machine that takes the higher-level learned representation as input can be quite powerful.

— Y. Bengio & Y. LeCun (2007), "Scaling Learning Algorithms towards AI"

- Two-sample testing [ICLR-17, ICML-20]
  - Choose ψ to maximize power criterion
  - Exploit closed form of f^*_ψ for permutation testing
- Generative modeling with MMD GANs [ICLR-18, NeurIPS-18]
  - Need a smooth loss function for the generator
  - Better gradients for generator to follow (?)

Future uses of deep kernel distances

- Selective inference to avoid train/test split? Meta-testing?

Future uses of deep kernel distances

- Selective inference to avoid train/test split? Meta-testing?
- When $\mathbb{P} \neq \mathbb{Q}$, can we tell *how* they're different?
  - Methods so far: some mostly for low-d
  - Some look at points with large critic function

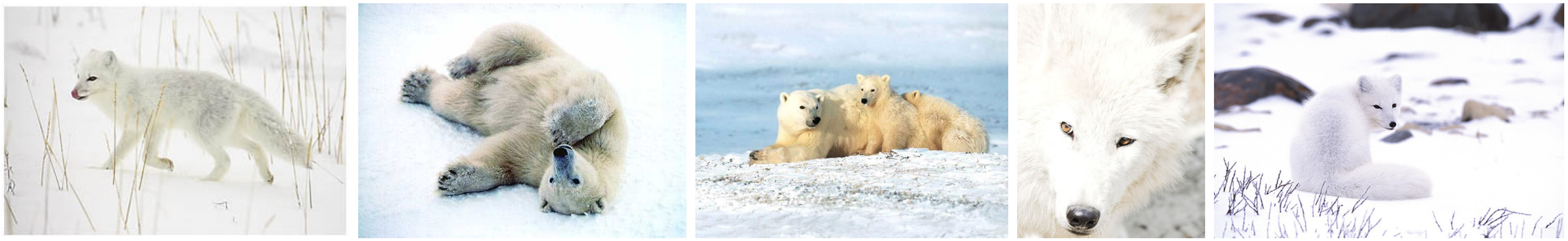
Future uses of deep kernel distances

- Selective inference to avoid train/test split? Meta-testing?
- When $\mathbb{P} \neq \mathbb{Q}$, can we tell *how* they're different?
  - Methods so far: some mostly for low-d
  - Some look at points with large critic function

Future uses of deep kernel distances

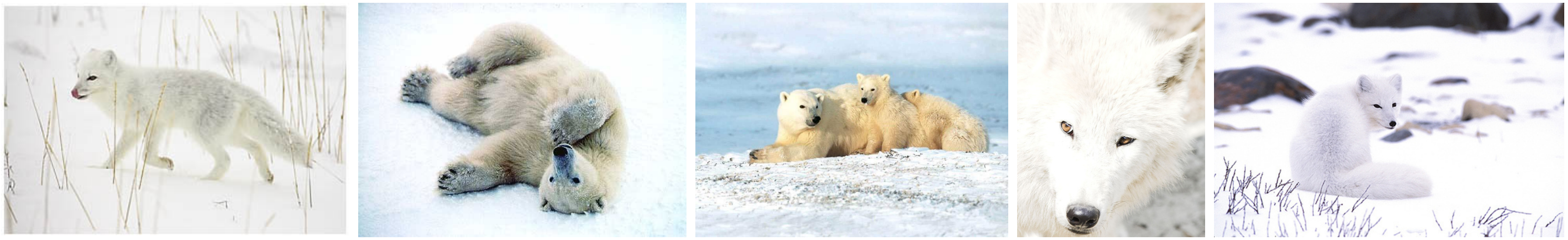
- Selective inference to avoid train/test split? Meta-testing?
- When $\mathbb{P} \neq \mathbb{Q}$, can we tell *how* they're different?
  - Methods so far: some mostly for low-d
  - Some look at points with large critic function



- Does model $\mathbb{Q}$ match dataset X (Stein testing)?

Future uses of deep kernel distances

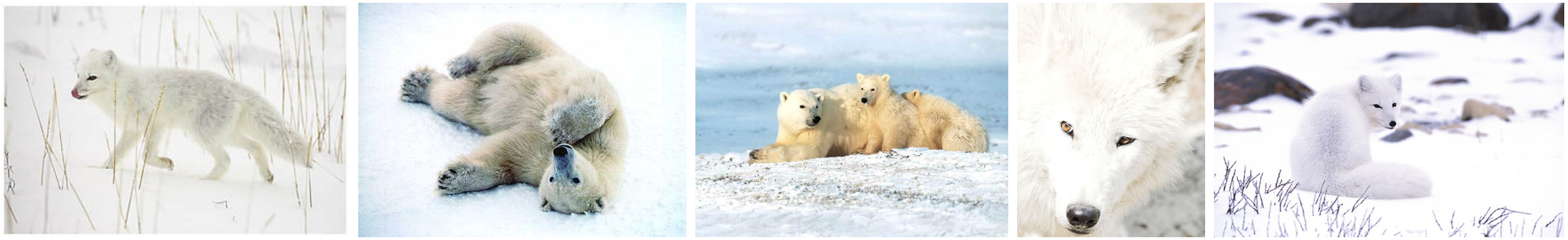
- Selective inference to avoid train/test split? Meta-testing?
- When $\mathbb{P} \neq \mathbb{Q}$, can we tell *how* they're different?
  - Methods so far: some mostly for low-d
  - Some look at points with large critic function



- Does model $\mathbb{Q}$ match dataset X (Stein testing)?
- Maximize deep dependence measure for unsupervised representation learning, as in contrastive learning

Future uses of deep kernel distances

- Selective inference to avoid train/test split? Meta-testing?
- When $\mathbb{P} \neq \mathbb{Q}$, can we tell *how* they're different?
  - Methods so far: some mostly for low-d
  - Some look at points with large critic function



- Does model $\mathbb{Q}$ match dataset X (Stein testing)?
- Maximize deep dependence measure for unsupervised representation learning, as in contrastive learning
- …