Supplementary Material for "χ^2 Generative Adversarial Net"

Chenyang Tao, Liqun Chen, Ricardo Henao
Electrical & Computer Engineering
Duke University
Durham, NC 27708, USA
`chenyang.tao, liqun.chen, ricardo.henao@duke.edu`

Jianfeng Feng
Institute of Science and Technology for Brain-Inspired Intelligence
Fudan University
Shanghai, 200433, PR China
`jianfeng64@gmail.com`

Lawrence Carin
Electrical & Computer Engineering
Duke University
Durham, NC 27708, USA
`lcarin@duke.edu`

A Theorem 1

See our remarks in Section E.1 after Theorem 7.

B Proof of Lemma 2

Proof. Let us assume $\|f\|^2_{\mathcal{H}_X} = \|g\|^2_{\mathcal{H}_Y} = 1$. By slight abuse of notation, we have

$$\begin{aligned}
\langle f, V_{XY} g\rangle_{\mathcal{H}_Y} &= \langle g, C_{YY}^{-1/2} C_{XY} C_{XX}^{-1/2} f\rangle_{\mathcal{H}_Y} && (1)\\
&= \langle C_{YY}^{-1/2} g, C_{XY}(C_{XX}^{-1/2} f)\rangle_{\mathcal{H}_Y} && (2)\\
&= \text{cov}(C_{YY}^{-1/2} g, C_{XX}^{-1/2} f) && (3)\\
&= \text{corr}(C_{YY}^{-1/2} g, C_{XX}^{-1/2} f). && (4)
\end{aligned}$$

We have the last equality by

$$\begin{aligned}
\text{var}(C_{XX}^{-1/2} f(X)) &= \langle C_{XX}^{-1/2} f, C_{XX} C_{XX}^{-1/2} f\rangle_{\mathcal{H}_X} && (5)\\
&= \langle f, C_{XX}^{-1/2} C_{XX} C_{XX}^{-1/2} f\rangle_{\mathcal{H}_X} && (6)\\
&= \langle f, f\rangle_{\mathcal{H}_X} && (7)\\
&= \|f\|^2_{\mathcal{H}_X} = 1. && (8)
\end{aligned}$$

35^{th} International Conference on Machine Learning (ICML 2018), Stockholm, Sweden.

C Proof of Proposition 3

Proof. By definition, we have

$$\begin{aligned}
&\text{cov}(C_{XX}^{-1/2}e_i, C_{XX}^{-1/2}e_j) && (9)\\
&= \langle C_{XX}^{-1/2}e_i, C_{XX}(C_{XX}^{-1/2}e_j)\rangle_{\mathcal{H}_X} && (10)\\
&= \langle e_i, (C_{XX}^{-1/2}C_{XX}C_{XX}^{-1/2})e_j\rangle_{\mathcal{H}_X} && (11)\\
&= \langle e_i, e_j\rangle_{\mathcal{H}_X} && (12)\\
&= \delta_{ij}. && (13)
\end{aligned}$$

D Proof of Corollary 4

Proof. By the definition of Hilbert-Schmidt norm, we have

$$\begin{aligned}
\|V_{XY}\|_{\text{HS}}^2 &= \sum_i \sum_j \langle V_{XY}e_i, f_j\rangle_{\mathcal{H}} && (14)\\
&= \sum_i \sum_j \{\text{corr}(C_{XX}^{-1/2}e_i, C_{YY}^{-1/2}f_i)\}^2, && (15)
\end{aligned}$$

where we have used Lemma 2.

E Proof of Proposition 5

Lemma 6. Under the above notations, $\forall h \in L_Y^2 \triangleq L^2(\mu_{p,q}(y))$ and $j > 1$ we have

$$\text{corr}(e_j(X), h(Y)) = 0. \qquad (16)$$

Proof. Let

$$c_i : L_X^2 \to \mathbb{R}, i \in \mathbb{Z}_+ \qquad (17)$$

be the projection functions wrt $\mathcal{E}$, that is to say

$$f = \sum_{i=1}^{\infty} c_i(f)e_i, \forall f \in L_X^2. \qquad (18)$$

It is easy to see c_i takes the form

$$c_i(f) = \mathbb{E}_X[f(X)e_i(X)]. \qquad (19)$$

Recall that Y is binary ($\{-1, +1\}$) and all $h \in L_Y^2$ have zero mean. This implies that for all $h \in L_Y^2$, there exist a constant $K_h \in \mathbb{R}$ such that

$$h(y) = K_h y. \qquad (20)$$

With this insight, we only need to prove $\text{cov}(e_j(X), Y) = 0$ for all $j > 1$.

$$\begin{aligned}
&\text{cov}(Y, f(X) - c_1(f)e_1(X)) && (21)\\
&= \mathbb{E}_{X,Y}[Y(f(X) - c_1(f)e_1(X))] && (22)\\
&= \mathbb{E}_X[\mathbb{E}_{Y|X}[Y(f(X) - c_1(f)e_1(X))]] && (23)\\
&= \mathbb{E}_X[(f(X) - c_1(f)e_1(X))\mathbb{E}[Y|X]] && (24)\\
&= \mathbb{E}_X[f(X)\mathbb{E}[Y|X]] - c_1(f)\mathbb{E}_X[e_1(X)\mathbb{E}[Y|X]] && (25)\\
&= \mathbb{E}_X[f(X)\phi(X)] - c_1(f)\mathbb{E}_X[e_1(X)\phi(X)] && (26)\\
&\quad (\phi(x) = c_1(\phi)e_1(x)) &&\\
&= c_1(\phi)\mathbb{E}_X[f(X)e_1(X)] - c_1(f)c_1(\phi) && (27)\\
&= c_1(f)c_1(\phi) - c_1(f)c_1(\phi) && (28)\\
&= 0 && (29)
\end{aligned}$$

Now we can easily show $\mathrm{cov}(e_j(X), Y) = 0$ for $j > 1$. Let $f(x) = e_j(x)$ ($j > 1$), we have

$$\mathrm{cov}(Y, e_j(X)) = \mathrm{cov}(Y, e_j(X) - \underbrace{c_1(e_j)}_{=0}\, e_1(X)) = 0, \tag{30}$$

where we have use the fact $c_1(e_j) = \mathbb{E}_X[e_j(X)e_1(X)] = 0$ since e_1 and e_j are orthogonal to each other for $j > 1$. This concludes our proof. □

Proof of Proposition 5.

$(i) \Leftrightarrow (ii)$

This is a direct result of [5], Theorem 4 (included below as Theorem 7 for completeness).

$(iv) \Leftrightarrow (vi)$

Since $g_{\mathcal{X}}(x) = \frac{\phi(x)}{\|\phi(x)\|_\mu}$, we know $\mathrm{corr}(g_{\mathcal{X}}(X), Y) = \mathrm{corr}(\phi(X), Y)$. Recall from Lemma 4 that $\forall h \in L^2_Y$ have the form $h(y) = K_h y$ where $K_h \in \mathbb{R}$ is a constant, and notice

$$\mathrm{var}(Y) = \frac{1}{2} \cdot 1^2 + \frac{1}{2} \cdot (-1)^2 = 1,$$

so the identity mapping $\mathrm{Id}(y) = y$ is the orthonormal basis for L^2_Y. Since $g_{\mathcal{X}}(x)$ and $\mathrm{Id}(y)$ all have unit length in L^2 norm, we have

$$\begin{array}{c} \mathrm{corr}(g_{\mathcal{X}}(X), \mathrm{Id}(Y)) = \mathbb{E}_{X,Y}[g_{\mathcal{X}}(X)Y] \\ = \frac{1}{2}(\mathbb{E}_{X_1 \sim p}[g_{\mathcal{X}}(X_1)] - \mathbb{E}_{X_2 \sim q}[g_{\mathcal{X}}(X_2)]). \end{array} \tag{31}$$

Taking the square on each side of the above equation and swap $\mathrm{corr}(g_{\mathcal{X}}(X), Y)$ with $\mathrm{corr}(\phi(X), Y)$ prove the result.

$(i) \Leftrightarrow (iv)$

By the definition of $\|V_{XY}\|^2_{\mathrm{HS}}$, we have

$$\|V_{XY}\|^2_{\mathrm{HS}} = \sum_{i,j} \{\mathrm{corr}(e_i(X), f_j(Y))\}^2, \tag{32}$$

where $\mathcal{E} = \{e_i\}$ and $\mathcal{F} = \{f_j\}$ are respectively the CONS for L^2_X and L^2_Y. By Lemma 6, we know

$$\|V_{XY}\|^2_{\mathrm{HS}} = \{\mathrm{corr}(e_1(X), Y)\}^2. \tag{33}$$

We conclude the proof by noticing $e_1(x) \propto \phi(x)$ and $\mathrm{corr}(\cdot, \cdot)$ is invariant to affine transformations.

$(ii) \Leftrightarrow (iii)$

First note

$$\phi(x) = \frac{p(x) - q(x)}{p(x) + q(x)}. \tag{34}$$

From [13], Theorem 1 (included below for completeness), we know that

$$\begin{array}{r} D_{\mathcal{X}}(p, q) = \mathbb{E}_{X \sim p}[f_{\mathcal{X}}(X)] - \mathbb{E}_{X \sim q}[f_{\mathcal{X}}(X)] \\ = \int_{\mathcal{X}} f_{\mathcal{X}}(x)(p(x) - q(x))\,\mathrm{d}x, \end{array} \tag{35}$$

where

$$f_{\mathcal{X}}(x) = \frac{2\phi(x)}{D_{\mathcal{X}}(p, q)}. \tag{36}$$

This readily leads to

$$\frac{1}{2}\{D_{\mathcal{X}}(p, q)\}^2 = \int_{\mathcal{X}} \phi(x)(p(x) - q(x))\,\mathrm{d}x \tag{37}$$

We already know

$$\begin{array}{rcl} \mathrm{MI}_{\mathcal{X}} & = & \frac{1}{4}\left\{\int_{\mathcal{X}} g_{\mathcal{X}}(x)(p(x) - q(x))\,\mathrm{d}x\right\}^2 \\ & = & \frac{1}{4\|\phi(x)\|^2_\mu}\left\{\int_{\mathcal{X}} \phi(x)(p(x) - q(x))\,\mathrm{d}x\right\}^2, \end{array} \tag{38}$$

and plugin the integral proves the result.

$(iv) \Leftrightarrow (v)$

It is easy to show that $\phi(x) = 2\psi(x) - 1$, and we know $\mathrm{corr}(\cdot, \cdot)$ is invariant to the affine transformations. □

E.1 Normalized cross-covariance operator and χ^2 mutual information

In [5], the authors consider RKHS, and therefore the function spaces are denoted as $\mathcal{H}_\cdot$. $\mathbb{P}_{X \perp\!\!\!\perp Y}$ denotes the the joint of (X, Y) under independence.

Theorem 7. [[5], Theorem 4] If $(\mathcal{H}_X \otimes \mathcal{H}_Y) + \mathbb{R}$ is dense in $L^2(\mathbb{P}_{X \perp\!\!\!\perp Y})$, and V_{XY} is Hilbert-Schmidt, then we have

$$\|V_{YX}\|_{\text{HS}}^2 = \iint_{\mathcal{X}\times\mathcal{Y}} \left(\frac{p_{XY}(x,y)}{p_X(x)p_Y(y)} - 1\right)^2 p_X(x)p_Y(y)\,\mathrm{d}x\,\mathrm{d}y.$$

In our case, we consider $\mathcal{H} = L^2$, which satisfies the condition in the above Theorem.

F Complete algorithm for joint matching of multiple distributions

Without loss of generality, let us assume we are matching the distribution of M generators $\{G(z;\theta_m)\}_{m=1}^M$, where θ_m denotes the generator specific parameters. We denote all generator parameters collectively as Θ. Let $D(x;\omega) = \text{Softmax}(T(x;\omega)) \in [0,1]^M$ be the discriminator that describes the conditional distribution of the label given sample x. We denote μ_X and Σ_X respectively as the expectation and covariance of first the $(M-1)$-dimension of $D(x;\omega)$. This is because $D(x;\omega)$ is linearly dependent as $\sum_m [D(x;\omega)]_m = 1$, where we have used $[a]_k$ to denote the k-th dimension of a vector a. We continue to use $\Gamma(y)$ and Σ_Y for the label basis functions and the corresponding covariance described in Sec. 2.5, and use one-hot encoding $y \in \{0,1\}^M$ for the labels. The complete algorithm for multi-distribution matching χ^2-GAN is given in Algorithm SM 1, where we have used (Θ_t, ω_t) to denote the parameter values at iteration t. Note that apart from the cross-entropy loss to train the critic $D(x;\omega)$, we can also use a multi-output least squares regression similar to Algorithm 1 in the maintext. Instead of $Y \in \{\pm 1\}$, the output target for multi-output least squares regression becomes $(2y_m - \mathbf{1}_M)$ for a sample from the m-th distribution, where $\mathbf{1}_M$ denotes a vector of ones.

F.1 The covariance Σ_Y for $\Gamma(y)$

Proof. We want to compute $[\Sigma_Y]_{ij} = \mathbb{E}_Y[\gamma_i(Y)\gamma_j(Y)]$. For $i = j$, we have

$$\begin{aligned} [\Sigma_Y]_{ij} &= \frac{1}{M}\gamma_i(y_i)\gamma_j(y_i) + \frac{1}{M}\sum_{l\neq i}\gamma_i(y_l)\gamma_j(y_l) && (39)\\ &= \frac{1}{M} + \frac{M-1}{M}\frac{1}{(M-1)^2} = \frac{1}{M-1}. && (40)\end{aligned}$$

And for $i \neq j$, we have

$$\begin{aligned} [\Sigma_Y]_{ij} &= \frac{1}{M}\gamma_i(y_i)\gamma_j(y_i) + \frac{1}{M}\gamma_i(y_j)\gamma_j(y_j) + \frac{1}{M}\sum_{l\notin\{i,j\}}\gamma_i(y_l)\gamma_j(y_l) && (41)\\ &= -\frac{2}{M(M-1)} + \frac{M-2}{M}\frac{1}{(M-1)^2} = -\frac{1}{(M-1)^2}. && (42)\end{aligned}$$

This concludes our proof. □

F.2 Generalization for multiple critic functions

The use of multiple critic functions can be motivated from the following perspectives: 1) robustness, a single critic function may get trapped in a local optimum and stoping providing useful info to the generator; 2) efficiency, a group of simple critic functions may collectively offer the same discriminative power as one single sophisticated critic function, but much easier to optimize (parallelize). The potential gains by introducing additional critic functions have been investigated in the literature [4]. Algorithm SM 1 can be easily modified to work with multiple critic functions. One simply concatenates the output of critic functions as $\tilde{\Phi(x)}$, and then take care of the multicollinearity. More explicitly, we can compute the cross-correlation matrix C between the critic vector and $\Gamma(y)$, then

Algorithm SM 1: χ^2-GAN-Multi.

```
Input: batchsize $b$, decay $\rho$, learning rate $\delta$.
for $t = 1, 2, 3, \ldots$ do
    1. Sample minibatch $\{z_{m,i} \sim p(z)\}_{i=1}^{b}, m = 1, \cdots, M$
    2. Generate the samples: $x_{i,m} = G(z_{i,m}; \theta_{m,t-1})$
    3. Update the critic $D(x;\omega)$ to improve (minimize) the cross-entropy loss
           $-\sum_m \sum_i \log[D(x_{i,m};\omega)]_m$
    4. Update the mean and covariance estimates
           $d(x) = [D(x_{i,m};\omega_t)]_{1:(M-1)} \in \mathbb{R}^{M-1}$
           $d_{i,m} = d(x_{i,m}),\ \hat{\mu}_t = (1-\rho)\hat{\mu}_{t-1} + \frac{\rho}{Mb}\sum_{i,m} d_{i,m}$
           $\hat{\Sigma}_t = (1-\rho)\hat{\Sigma}_t + \frac{\rho}{Mb}\sum_{i,m}(d_{i,m} - \hat{\mu}_t)^T(d_{i,m} - \hat{\mu}_t)$
    5. Update the $\chi^2$ estimate
           $V(\Theta_t) = \frac{1}{Mb}\sum_{i,m} \hat{\Sigma}_t^{-1/2}\{d(G(z_{i,m};\theta_{m,t-1})) - \hat{\mu}_t\}\Gamma(y_{i,m})^T\Sigma_Y^{-1/2}$
           $V(\Theta_{t-1}) = (1-\rho)V_{t-1} + \rho V(\Theta_{t-1}),\ v(\Theta_{t-1}) = \|V(\Theta_{t-1})\|_{\text{Fro}}^2$
    6. Update the generators $\{G(z;\theta_m)\}_m$
           $\Theta_t = \Theta_{t-1} + \delta\,\text{GradClip}\,(\nabla_\Theta v_t(\Theta_{t-1}))$
end for
```

compute its singular-value decomposition $C = \tilde{U}S\tilde{V}^T$. Denoting U as the $(M-1)$ left-eigenvectors wrt non-vanishing singular vectors, and $\Phi(x)$ as the mean-centered, variance normalized $\tilde{\Phi}(x)$, then $U^T\Phi(x)$ gives the normalized critic $\Psi(x)$ used in V_{χ^2}.

G Connection between MMD and the χ^2 GAN objective

Recall that HSIC $= \|C_{XY}\|_{\text{HS}}^2$ is the unnormalized version of χ^2 objective $\|V_{XY}\|_{\text{HS}}^2$. We now prove the equivalence between MMD and HSIC in the generative modeling setting.

Proposition 8. For empirical distributions

$$\mu_n = n^{-1}\sum_{i=1}^{n}\delta_{(x_i,y_i)}, p_n = n^{-1}\sum_{i=1}^{n}\delta_{x_i}, q_n = n^{-1}\sum_{i=1}^{n}\delta_{y_i}, \tag{43}$$

we have

$$\text{HSIC}(\mu_n) = \frac{1}{4}\text{MMD}(p_n, q_n). \tag{44}$$

Proof.

$$\begin{aligned}\text{MMD}(p_n, q_n) &= \frac{1}{n^2}\left(\sum_{i,i'=1}^{n}\kappa(x_i, x_{i'}) - 2\sum_{i,j}\kappa(x_i, \tilde{x}_j) + \sum_{j,j'}\kappa(\tilde{x}_j, \tilde{x}_{j'})\right) &(45)\\ &= \frac{1}{n^2}\text{tr}(K_{p,q}^{(n)}L_n) &(46)\end{aligned}$$

$$\text{HSIC}(\mu_n) = \frac{1}{4n^2}\text{tr}(H_{2n}K_{p,q}^{(n)}H_{2n}H_{2n}L_nH_{2n}) \tag{47}$$

Since

$$\text{tr}(AB) = \text{tr}(BA), L_n = H_{2n} L_n H_{2n}, (H_{2n})^2 = H_{2n}, \tag{48}$$

we therefore have

$$\text{HSIC}(\mu_n) = \frac{1}{4n^2} \text{tr}(K_{p,q}^{(n)} L_n) = \frac{1}{4} \text{MMD}(p_n, q_n). \tag{49}$$

H Computing importance weights from critics of divergence-based models

We consider the f-GAN formulation [15], which optimizes the variational objective

$$V_f(D, G) = \mathbb{E}_{X \sim p_d}[D(X; \omega)] - \mathbb{E}_{X' \sim p_G}[f^*(D(X'; \omega))], \tag{50}$$

where $D(x; \omega)$ is the parameterized critic, $p_G(x)$ is implicitly defined by the parameterized generator $G(z; \theta)$ and latent variable $Z \sim p(z)$, $f(x)$ is the convex function that defines the f-divergence, and

$$f^*(u) \triangleq \sup_{x \in \text{supp} f} \{ux - f(x)\} \tag{51}$$

is the Fenchel conjugate of $f(x)$. When $\{D(x; \omega), \omega \in \Omega\}$ approximates arbitrary functions, the maximizer of (50) with generator fixed writes

$$D^*(x) = f' \left(\frac{p_d(x)}{p_G(x)} \right), \tag{52}$$

where $f'(x)$ is the derivative of $f(x)$ and we have the equality [14]

$$V_f(D^*, G) = \text{Div}_f(p_d \parallel p_G). \tag{53}$$

As such, to estimate the importance weight $w(x) = \frac{p_d(x)}{p_G(x)}$, we only need to take the inverse of $f'(x)$ wrt the learned critic $\hat{D}(x)$, i.e.

$$\hat{w}(x) = {f'}^{-1}(\hat{D}(x)). \tag{54}$$

For example, when optimizing the KL-divergence ($f(x) = -\log(x)$), then $f'(x) = -1/x$ and ${f'}^{-1}(x) = -1/x$, and therefore $w(x) = -1/D(x)$.

I Additional historical notes on correlation and independence metrics

The idea of maximal correlation measures of dependence at least dates back to the 1930s [8]. Later [17] set forth a list of desirable properties for a measure of statistical dependence, and showed that the maximal correlation measure

$$\sup_{f,g} \{\text{corr}(f(X), g(Y))\}, \tag{55}$$

where f and g are *Borel* measurable, satisfies such properties. The constraint on the function space later was relaxed from *Borel* spaces to L^2 spaces, and the resulting statistics was termed Φ^2 measure of dependency [11], which coincides with the χ^2 statistics we investigated here. [2]'s seminal work on kernel independence criteria, where they have used generalized kernel canonical correlation analysis to approximate Shannon mutual information, marks the beginning of the development of modern RKHS solutions to probabilistic problems. [6] derived the first rigorous RKHS independence criteria using the cross-covariance theory developed by [3]. This work is generalized by [5], establishing its connections to information theoretic measures. The works of Brownian correlations [19] can be shown to be equivalent to RKHS independence measures, which provides an alternative view from the perspective of stochastic process.

J Discussion on specific techniques reporting state-of-the-art results

In terms of unsupervised Inception Score on Cifar10, the GAN variants summarized in Table SM 1 have reported better results. DFM-GAN uses the reconstruction error of a denoising autoencoder as an energy regularization to encourage the generation of low energy (more plausible) samples. This leads to a biased generator that possibly cheats IS metric (unpublished results). PG-GAN follows

Model	IS
DFM-GAN [21]	$7.72 \pm .13$
PG-GAN [9]	8.80
SN-GAN [12]	$7.58 \pm .12$
SN-GAN ResNet [12]	$8.42 \pm .08$
OT-GAN [18]	$8.47 \pm .12$

Table SM 1: Unsupervised Inception Score on Cifar10 of GANs with specific techniques

a specific training strategy: progressively "grow" a low resolution generator into a high resolution generator. While reporting unprecedented results, PG-GAN can only be applied to datasets that can be hierarchically factorized, and requires careful tuning and extensive computational resource. SN-GAN advocates a novel solution to constrain the Lipschitz constant of discriminator, which is of paramount importance to Wasserstein GAN variants. We found integrating SN techniques into our χ^2 GAN further improves results (results not shown). OT-GAN proposed a adaptive adversarial OT game. However, the Skinkhorn algorithm used is very sensitive to the mini-batch size. The best result reported uses 8k samples per "mini"-batch, and OT-GAN failed to give competitive results (IS < 7) when more reasonable mini-batch size ($64 \sim 256$) were used.

K Detailed experimental setup

model architecture, batch size, learning rate, etc. We use gradient clip by norm 0.1 on Cifar and CelebA experiments.

K.1 Toy model

To make fair comparison with other GAN models, we follow the same toy model network architecture as `https://github.com/igul222/improved_wgan_training/`. We only implement our model under its framework, and keep all the parameters unchanged.

K.2 MNIST

The model architecture on MNIST task is shown in Table 2

Decoder z to X	Discriminator
Input latent code z	Input two 28×28 Gray Image
MLP output 1024, BN	5×5 conv. 32 ReLU, stride 2, BN
MLP output 3136, BN	5×5 conv. 64 ReLU, stride 2, BN
5×5 deconv. 64 ReLU, stride 2, BN	5×5 conv. 128 ReLU, stride 2, BN
	input z through MLP output 1024, ReLU
5×5 deconv. 1 ReLU, stride 2, sigmoid	MLP output 1

Table SM 2: Architecture of the models for χ^2 GAN on MNIST.

K.3 Cifar10

We use DCGAN framework and WGAN-GP ResNet framework for our Cifar generation task . The learning rate for both architecture is 10^{-4}.

K.4 CelebA

We use DCGAN framework for our face generation task. The learning rate is 10^{-4}.

K.5 ImageNet

The detailed architecture is shown in Table 3

Decoder z to X	Discriminator
Input z	Input X
	5 × 5 conv. 64 ReLU, stride 2, BN
MLP output 1024, lReLU, BN	5 × 5 conv. 128 ReLU, stride 2, BN
MLP output 8192, lReLU, BN	5 × 5 conv. 256 ReLU, stride 2, BN
	5 × 5 conv. 512 ReLU, stride 2, BN
5 × 5 deconv. 256 lReLU, stride 2, BN	Input z through MLP, output 2046, ReLU
5 × 5 deconv. 128 lReLU, stride 2, BN	concat two features from X and z
5 × 5 deconv. 64 lReLU, stride 2, BN	
5 × 5 deconv. 3 tanh, stride 2, BN	MLP output 1

Table SM 3: Architecture of the models for χ^2 GAN on ImageNet.

K.6 Edges2shoes

In the experiment, we used the model architecture from the Disco-GAN paper [10]. Following the practice of Disco-GAN, we also initialized model parameters by optimizing a reconstruction loss. This has been proven necessary due to the size of dataset, and we verified that χ^2 GAN's ability to translate image is not a direct result of this initialization.

L More experimental results

L.1 Generation result on cifar10

We give the unconditional generation result on ImageNet in Figure SM 1.

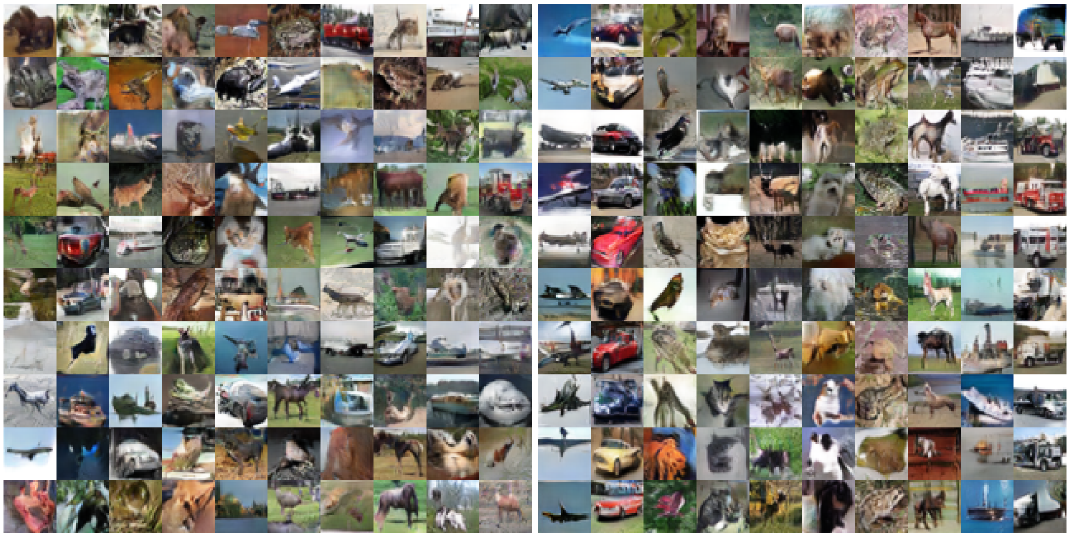

Figure SM 1: More generated images on Cifar-10. Left: unsupervised generation; right: supervised generation.

L.2 Generation result on CelebA

We give the unconditional generation result on standard CelebA (64 × 64) in Figure SM 2.

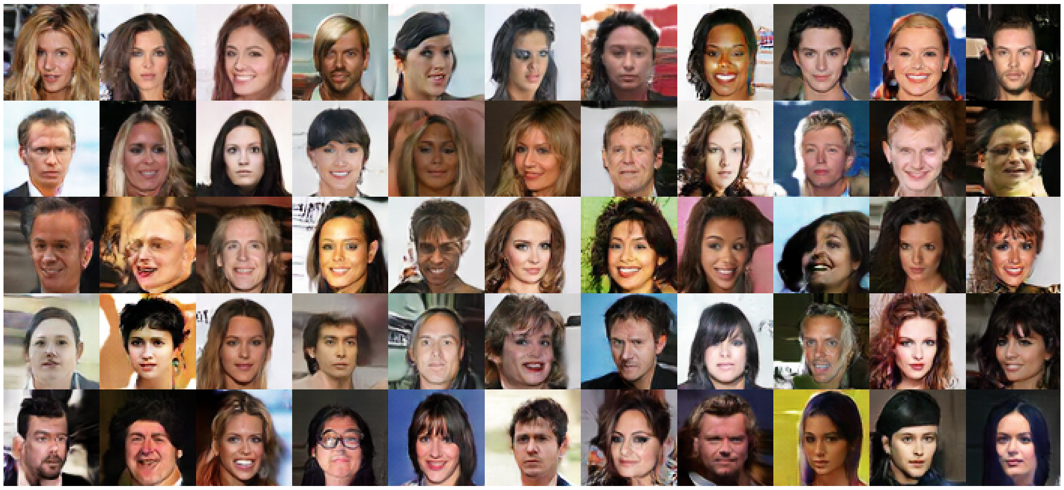

Figure SM 2: Generated images on CelebA

L.3 Generation result on high-resolution CelebA

We give the unconditional generation result on high-resolution CelebA (128×128) in Figure SM 3 to demonstrate χ^2-GAN's scalability wrt image resolution.

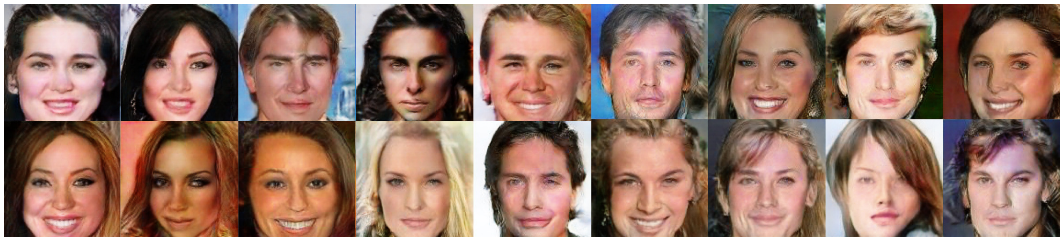

Figure SM 3: Generated high-resolution images on CelebA

L.4 Generation result on ImageNet

We give the unconditional generation result on ImageNet in Figure SM 4.

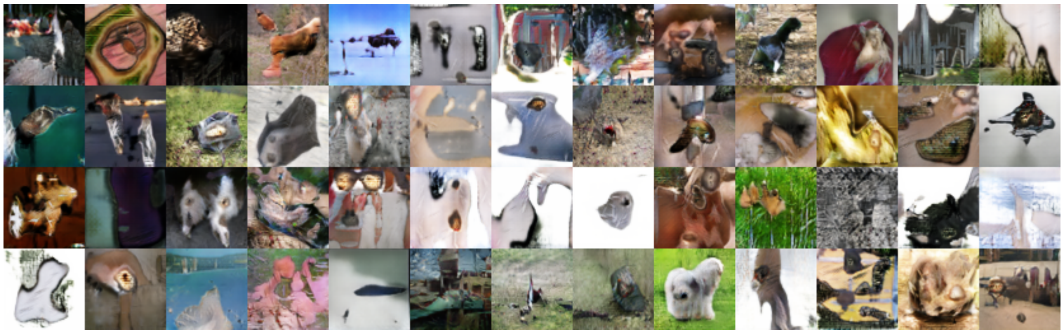

Figure SM 4: Generated images on ImageNet (unsupervised).

	25 Gaussians	CIFAR (CNN)	CIFAR (ResNet)	CelebA
Model parameters (million)	1.06M	9.69M	2.26M	9.75M
WGAN-GP per 100 iteration	1.93s	55.6s	46.5s	150.1s
χ^2GAN per 100 iteration	0.63s	13.7s	11.5s	26.5s
Speed-up	3.06	3.21	4.04	5.68

Table SM 4: Wall time comparison between WGAN-GP and χ^2 GAN.

L.5 Bag and shoe sketch dataset

In this experiment, we use a mixture of bag sketches and shoe sketches to verify the robustness and efficacy of χ^2 GAN. Generated samples from the learned distributions with DCGAN, WGAN-GP and χ^2 GAN are given in Figure 5. We extensively tuned hyperparameters and experimented with different stablizing architecture for DCGAN, but it is still unable to learn the target distribution. WGAN-GP was able to learn some features of the target distribution after significant tuning, but still outputs a significant portion of unrealistic images. Our χ^2 GAN quickly learns the distribution, generating realistic and diverse samples without any intervention. χ^2 GAN successfully learned the distribution for a wide range of different hyperparameters we experimented with.

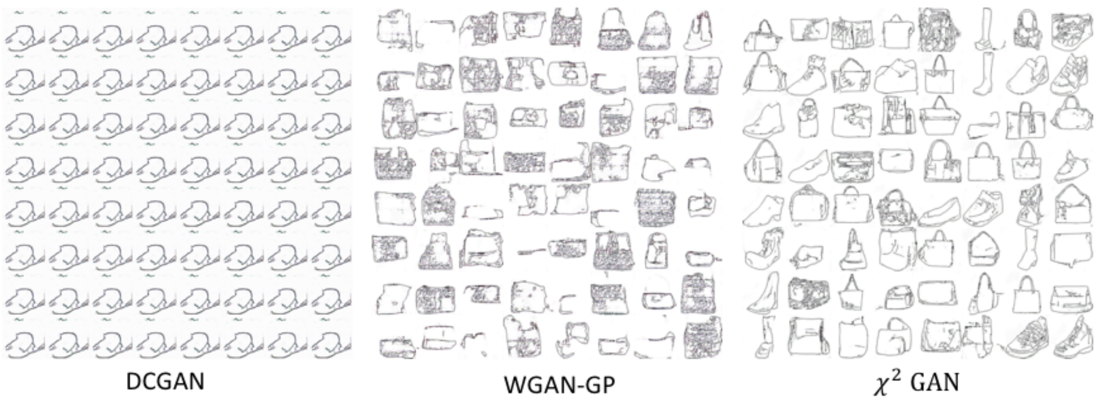

Figure SM 5: Learned distribution on mixture of shoe and bag sketches dataset (unsupervised).

M Computational efficiency

In Table SM 4 we further compare the training efficiency of WGAN-GP and χ^2 GAN. We report the average of per 100 iteration walltime for different network architectures used in this work. While giving better, or comparable results, χ^2 GAN generally achieves $3 \sim 6$ times speedup.

N Additional quantitative results for Cifar10

Table SM 5 summarizes FID scores on Cifar10 .

O Quantitative evaluation for importance resampling

In Figure SM 6 we provide quantitative results for importance resampling on Cifar10. In this experiment we vary the acceptance ratio, and accept the corresponding proportion of samples using the importance resampling algorithm described in main text. IS score improved as we reduce the acceptance ratio, visual inspection also confirmed the improvement of sample quality.

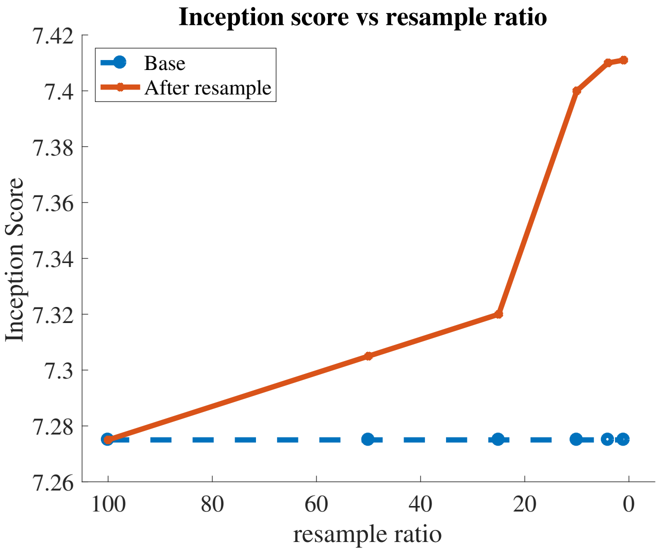

Figure SM 6: Inception score vs resample ratio

Model	FID
DCGAN [16]	55.7
WGAN † [1]	42.6
WGAN-GP † [7]	40.2
χ^2-GAN (ours)	**29.4**

Table SM 5: Unsupervised FID on CIFAR-10 (DCGAN architecture). † Results collected from [12].

P Additional evaluations on CelebA

To verify that generator trained with χ^2 GAN indeed learn the target distribution rather than remembering the training samples, we visually compare the generated samples with their closest neighbours from the training sample. We tried a few metrics to evaluate image similarity, and found *Structural Similarity* (SSIM) [20] correlates best with human judgement. As such, we use SSIM to characterise image similarity. See Figure SM 7 for the neighbour comparison results. While we can identify commonalities between the fakes and their neighbouring samples from the real, these samples do not look similar at all. We further validate the hypothesis that χ^2 GAN learns generalisable features via latent space interpolation. More specifically, we first draw two random samples, and then linearly interpolate the two samples in the latent space. We show some of the results in Figure 8.

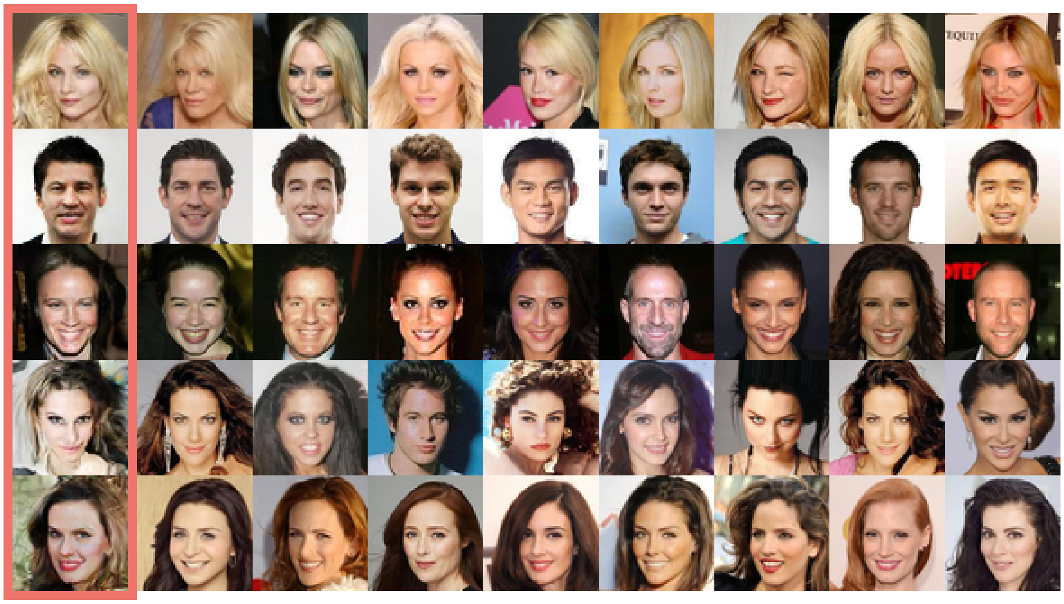

Figure SM 7: Nearest neighbours in the training set for generated samples on CelebA. First column: generated samples; other columns: training samples most similar to the fake sample.

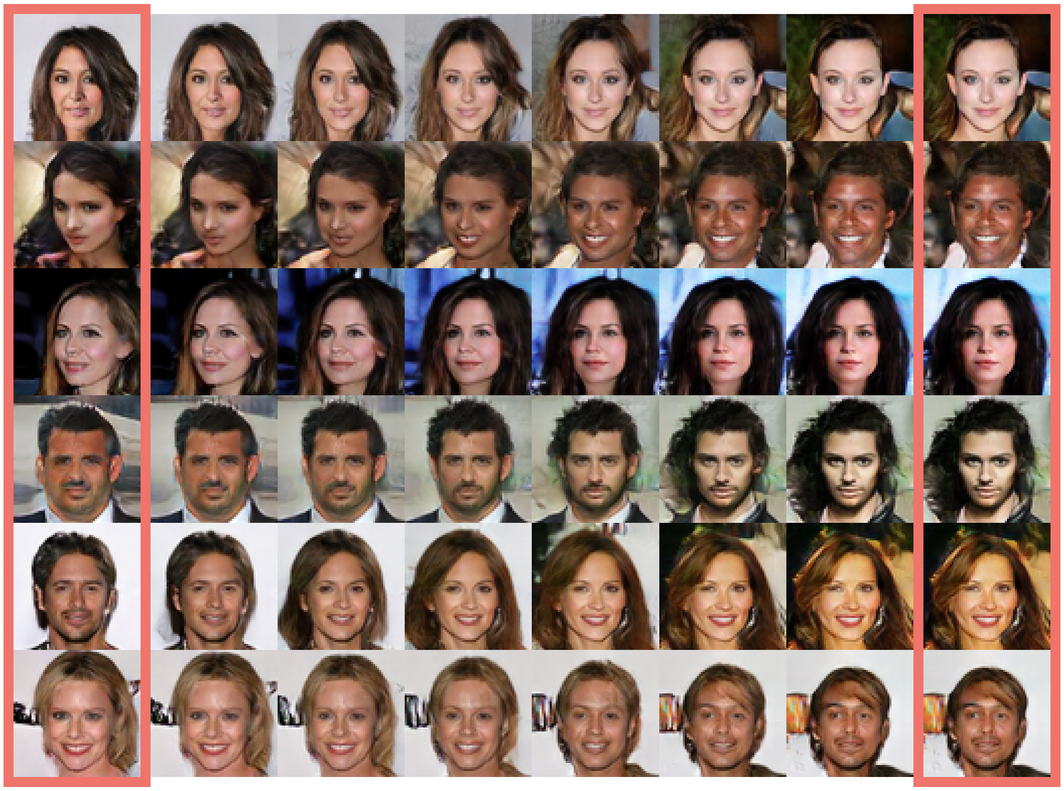

Figure SM 8: CelebA latent space interpolation. First and last column: generated samples; intermediate columns: linear interpolations in the latent space.

References

[1] Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein generative adversarial networks. In *ICML*, 2017.

[2] Francis R Bach and Michael I Jordan. Kernel independent component analysis. *The Journal of Machine Learning Research*, 3:1–48, 2003.

[3] Charles R Baker. Joint measures and cross-covariance operators. *Transactions of the American Mathematical Society*, 186:273–289, 1973.

[4] Ishan Durugkar, Ian Gemp, and Sridhar Mahadevan. Generative multi-adversarial networks. In *ICLR*, 2017.

[5] Kenji Fukumizu, Arthur Gretton, Xiaohai Sun, and Bernhard Schölkopf. Kernel measures of conditional dependence. In *NIPS*, 2007.

[6] Arthur Gretton, Ralf Herbrich, Alexander Smola, Olivier Bousquet, and Bernhard Schölkopf. Kernel methods for measuring independence. *The Journal of Machine Learning Research*, 6:2075–2129, 2005.

[7] Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron Courville. Improved training of Wasserstein GANs. *arXiv preprint arXiv:1704.00028*, 2017.

[8] H. O. Hirschfeld and J. Wishart. A Connection between Correlation and Contingency. *Mathematical Proceedings of The Cambridge Philosophical Society*, 31, 1935.

[9] Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive growing of GANs for improved quality, stability, and variation. In *ICLR*, 2018.

[10] Taeksoo Kim, Moonsu Cha, Hyunsoo Kim, Jungkwon Lee, and Jiwon Kim. Learning to discover cross-domain relations with generative adversarial networks. In *ICML*, 2017.

[11] Henry Oliver Lancaster. *Chi-Square Distribution*. Wiley Online Library, 1969.

[12] Takeru Miyato, Toshiki Kataoka, Masanori Koyama, and Yuichi Yoshida. Spectral normalization for generative adversarial networks. In *ICML Implicit Models Workshop*, 2017.

[13] Youssef Mroueh and Tom Sercu. Fisher GAN. In *NIPS*. 2017.

[14] XuanLong Nguyen, Martin J Wainwright, and Michael I Jordan. Estimating divergence functionals and the likelihood ratio by convex risk minimization. *IEEE Transactions on Information Theory*, 56(11):5847–5861, 2010.

[15] Sebastian Nowozin, Botond Cseke, and Ryota Tomioka. f-GAN: Training generative neural samplers using variational divergence minimization. In *NIPS*, 2016.

[16] Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. In *ICLR*, 2016.

[17] A. Rényi. On measures of dependence. *Acta Mathematica Academiae Scientiarum Hungarica*, 10(3-4):441–451, 1959.

[18] Tim Salimans, Han Zhang, Alec Radford, and Dimitris Metaxas. Improving gans using optimal transport. In *ICLR*, 2018.

[19] Gábor J Székely, Maria L Rizzo, et al. Brownian distance covariance. *The annals of applied statistics*, 3(4):1236–1265, 2009.

[20] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004.

[21] David Warde-Farley and Yoshua Bengio. Improving generative adversarial networks with denoising feature matching. In *ICLR*, 2017.