A. Proofs

Proposition 3. *$\forall P, Q \in \mathcal{P}(\mathcal{X})$ such that $P \ll Q$, $\forall T \in L^\infty(Q)$ such that $\mathrm{im}(T) \subseteq \mathrm{dom}((f')^{-1})$, and $\forall \mathcal{R} \subseteq L^\infty_{\geq 0}(Q)$ such that $\{\mathbb{1}\} \subseteq \mathcal{R}$ we have:*

$$I_f(T; P, Q) \leq \mathcal{L}^{\mathcal{R}}_f(T; P, Q) \leq I_W(T; P, Q). \tag{22}$$

Proof. From Propositions 1, and that $\mathcal{R} \subseteq L^\infty_{\geq 0}(Q)$, we have:

$$I_f(T; P, Q) = \inf_{r \in L^\infty_{\geq 0}(Q)} \ell_f(T, r; P, Q) \leq \inf_{r \in \mathcal{R}} \ell_f(T, r; P, Q) = \mathcal{L}^{\mathcal{R}}_f(T; P, Q). \tag{27}$$

From Proposition 2 and that $\{\mathbb{1}\} \subseteq \mathcal{R}$, we have:

$$\mathcal{L}^{\mathcal{R}}_f(T; P, Q) = \inf_{r \in \mathcal{R}} \ell_f(T, r; P, Q) \leq \inf_{r \in \mathbb{1}} \ell_f(T, r; P, Q) = \mathcal{L}^{\mathcal{R}}_f(T; P, Q) \leq I_W(T; P, Q). \tag{28}$$

Combining the two inequalities completes the proof. □

Theorem 1. *For $\{\mathbb{1}\} \subseteq \mathcal{R} \subseteq L^\infty_{\geq 0}(Q)$, define*

$$D_{f,\mathcal{R}}(P \| Q) := \sup_{T \in \mathcal{F}} \mathcal{L}^{\mathcal{R}}_f(T; P, Q) \tag{23}$$

where $\mathcal{F} := \{T : \mathcal{X} \to \mathrm{dom}((f')^{-1}), T \in L^\infty(Q)\}$. Then

$$D_f(P \| Q) \leq D_{f,\mathcal{R}}(P \| Q) \leq \sup_{T \in \mathcal{F}} I_W(T; P, Q). \tag{24}$$

Proof. From Proposition 1, we have the following upper bound for $D_{f,\mathcal{R}}(P \| Q)$:

$$\begin{aligned} &\sup_{T \in \mathcal{F}} \inf_{r \in \mathcal{R}} \mathbb{E}_P[f(r)] + \mathbb{E}_P[T] - \mathbb{E}_Q[r \cdot T] \\ \leq\ &\sup_{T \in \mathcal{F}} \inf_{r \in \{\mathbb{1}\}} \mathbb{E}_P[f(r)] + \mathbb{E}_P[T] - \mathbb{E}_Q[r \cdot T] \\ =\ &\sup_{T \in \mathcal{F}} \mathbb{E}_P[T] - \mathbb{E}_Q[T] = \mathrm{IPM}_{\mathcal{F}}(P, Q), \end{aligned} \tag{29}$$

We also have the following lower bound for $D_{f,\mathcal{R}}(P \| Q)$:

$$\begin{aligned} &\sup_{T \in \mathcal{F}} \inf_{r \in \mathcal{R}} \mathbb{E}_P[f(r)] + \mathbb{E}_P[T] - \mathbb{E}_Q[r \cdot T] \\ \geq\ &\sup_{T \in \mathcal{F}} \inf_{r \in L^\infty_{\geq 0}(Q)} \mathbb{E}_P[f(r)] + \mathbb{E}_P[T] - \mathbb{E}_Q[r \cdot T] \\ =\ &\sup_{T \in \mathcal{F}} \mathbb{E}_P[T] - \mathbb{E}_Q[f_*(T)] = D_f(P \| Q). \end{aligned} \tag{30}$$

Therefore, $D_{f,\mathcal{R}}(P \| Q)$ is bounded between $D_f(P \| Q)$ and $\mathrm{IPM}_{\mathcal{F}}(P, Q)$ and thus it is a valid divergence over $\mathcal{P}(\mathcal{X})$. □

Theorem 2. *Let $f(u) = u \log u$ and $\mathcal{F}$ a set of real-valued bounded measurable functions on $\mathcal{X}$. For any fixed choice of P, Q, and $T \in \mathcal{F}$, we have*

$$\operatorname*{arg\,min}_{r \in \Delta(Q)} \mathbb{E}_Q[f(r)] + \mathbb{E}_P[T] - \mathbb{E}_Q[r \cdot T] = \frac{e^T}{\mathbb{E}_Q[e^T]} \tag{26}$$

Proof. Consider the following Lagrangian:

$$h(r, \lambda) := \mathbb{E}_Q[f(r)] - \mathbb{E}_Q[r \cdot T] + \lambda(\mathbb{E}_Q[r] - 1) \tag{31}$$

where $\lambda \in \mathbb{R}$ and we formalize the constraint $r \in \Delta(r)$ with $\mathbb{E}_Q[r] - 1 = 0$. Taking the functional derivative $\partial h / \partial r$ and setting it to zero, we have:

$$\begin{aligned} &f'(r)\,\mathrm{d}Q - T\,\mathrm{d}Q + \lambda \\ &= (\log r + 1)\,\mathrm{d}Q - T\,\mathrm{d}Q + \lambda = 0, \end{aligned} \tag{32}$$

so $r = \exp(T - (\lambda + 1))$. We can then apply the constraint $\mathbb{E}_Q[r] = 1$, where we solve $\lambda + 1 = \mathbb{E}_Q[e^T]$, and consequently the optimal $r = e^T / \mathbb{E}_Q[e^T] \in \Delta(Q)$. □

B. Example KL-WGAN Implementation in PyTorch

```
def get_kl_ratio(v):
    vn = torch.logsumexp(v.view(-1), dim=0) - torch.log(torch.tensor(v.size(0)).float())
    return torch.exp(v - vn)

def loss_kl_dis(dis_fake, dis_real, temp=1.0):
    """
    Critic loss for KL-WGAN.
    dis_fake, dis_real are the critic outputs for generated samples and real samples.
    temp is a hyperparameter that scales down the critic outputs.
    We use the hinge loss from BigGAN PyTorch implementation.
    """
    loss_real = torch.mean(F.relu(1. - dis_real))
    dis_fake_ratio = get_kl_ratio(dis_fake / temp)
    dis_fake = dis_fake * dis_fake_ratio
    loss_fake = torch.mean(F.relu(1. + dis_fake))
    return loss_real, loss_fake

def loss_kl_gen(dis_fake, temp=1.0):
    """
    Generator loss for KL-WGAN.
    dis_fake is the critic outputs for generated samples.
    temp is a hyperparameter that scales down the critic outputs.
    We use the hinge loss from BigGAN PyTorch implementation.
    """
    dis_fake_ratio = get_kl_ratio(dis_fake / temp)
    dis_fake = dis_fake * dis_fake_ratio
    loss = -torch.mean(dis_fake)
    return loss
```

C. Argument about χ^2-Divergences

We present a similar argument to Theorem 2 to χ^2-divergences, where $f(u) = (u-1)^2$.

Theorem 3. *Let $f(u) = (u-1)^2$ and $\mathcal{F}$ is a set of real-valued bounded measurable functions on $\mathcal{X}$. For any fixed choice of P, Q, and $T \in \mathcal{F}$ such that $T \geq 0, T - \mathbb{E}[T] + 2 \geq 0$, we have*

$$\arg\min_{r \in \Delta(Q)} \mathbb{E}_Q[f(r)] + \mathbb{E}_P[T] - \mathbb{E}_{Q_r}[T] = \frac{T - \mathbb{E}_Q[T] + 2}{2}$$

Proof. Consider the following Lagrangian:

$$h(r, \lambda) := \mathbb{E}_Q[f(r)] - \mathbb{E}_Q[r \cdot T] + \lambda(\mathbb{E}_Q[r] - 1) \tag{33}$$

where $\lambda \in \mathbb{R}$ and we formalize the constraint $r \in \Delta(r)$ with $\mathbb{E}_Q[r] - 1 = 0$. Taking the functional derivative $\partial h / \partial r$ and setting it to zero, we have:

$$\begin{aligned} & f'(r)\,\mathrm{d}Q - T\,\mathrm{d}Q + \lambda \\ = & \, 2r\,\mathrm{d}Q - T\,\mathrm{d}Q + \lambda = 0, \end{aligned} \tag{34}$$

so $r = (T - \lambda)/2$. We can then apply the constraint $\mathbb{E}_Q[r] = 1$, where we solve $\lambda = \mathbb{E}_Q[T] - 2$, and consequently the optimal $r = (T - \mathbb{E}_Q[T] + 2)/2 \in \Delta(Q)$. □

In practice, when the constraint $T - \mathbb{E}_Q[T] + 2 \geq 0$ is not true, then one could increase the values when T is small, using

$$\hat{T} = \max(T, c) + b \tag{35}$$

where b, c are some constants that satisfies $T(\hat{\boldsymbol{x}}) - \mathbb{E}_Q[\hat{T}] + 2 \geq 0$ for all $\boldsymbol{x} \in \mathcal{X}$. Similar to the KL case, we encourage higher weights to be assigned to higher quality samples.

If we plug in this optimal r, we obtain the following objective:

$$\mathbb{E}_P[T] - \mathbb{E}_Q[T] + \frac{1}{4}\mathbb{E}_Q[T^2] + \frac{1}{4}(\mathbb{E}_Q[T])^2 = \mathbb{E}_P[T] - \mathbb{E}_Q[T] - \frac{\mathrm{Var}_Q[T]}{4}. \tag{36}$$

Let us now consider $P = P_{\text{data}}$, $Q = \frac{P_{\text{data}}+G_\theta}{2}$, then the f-divergence corresponding to $f(u) = (u-1)^2$:

$$D_f(P\|Q) = \int_{\mathcal{X}} \frac{(P(\boldsymbol{x}) - Q(\boldsymbol{x}))^2}{\frac{P(\boldsymbol{x})+Q(\boldsymbol{x})}{2}} \,\mathrm{d}\boldsymbol{x}, \tag{37}$$

is the squared χ^2-distance between P and Q. So the objective becomes:

$$\min_\theta \max_\phi \mathbb{E}_{P_{\text{data}}}[D_\theta] - \mathbb{E}_{G_\theta}[D_\phi] - \mathrm{Var}_{M_\theta}[D_\phi], \tag{38}$$

where $M_\theta = (P_{\text{data}} + G_\theta)/2$ and we replace $T/2$ with D_ϕ. In comparison, the χ^2-GAN objective (Tao et al., 2018) for θ is:

$$\frac{(\mathbb{E}_{P_{\text{data}}}[D_\theta] - \mathbb{E}_{G_\theta}[D_\phi])^2}{\mathrm{Var}_{M_\theta}[D_\phi]}. \tag{39}$$

They do not exactly minimize χ^2-divergence, or a squared χ^2-divergence, but a normalized version of the 4-th power of it, hence the square term over $\mathbb{E}_{P_{\text{data}}}[D_\theta] - \mathbb{E}_{G_\theta}[D_\phi]$.

D. Additional Experimental Details

For 2d experiments, we consider the WGAN and KL-WGAN objectives with the same architecture and training procedure. Specifically, our generator is a 2 layer MLP with 100 neurons and LeakyReLU activations on each hidden layer, with a latent code dimension of 2; our discriminator is a 2 layer MLP with 100 neurons and LeakyReLU activations on each hidden layer. We use spectral normalization (Miyato et al., 2018) over the weights for the generators and consider the hinge loss in (Miyato et al., 2018). Each dataset contains 5,000 samples from the distribution, over which we train both models for 500 epochs with RMSProp (learning rate 0.2). The procedure for tabular experiments is identical except that we consider networks with 300 neurons in each hidden layer with a latent code dimension of 10. Dataset code is contained in https://github.com/kevin-w-li/deep-kexpfam.

E. Samples

We show uncurated samples from BigGAN trained with WGAN and KL-WGAN loss in Figures 6a and 6b.

(a) CelebA 64x64 samples trained with WGAN.

(b) CelebA 64x64 Samples trained with KL-WGAN.

(a) CIFAR samples trained with WGAN.

(b) CIFAR samples trained with KL-WGAN.