Diffeomorphic Counterfactuals and Generative Models

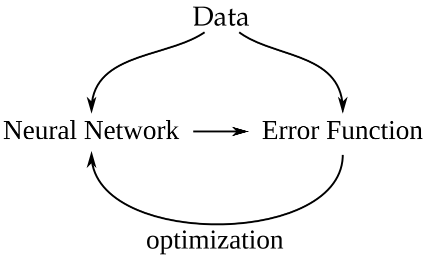
Jan E. Gerken

CHALMERS
UNIVERSITY OF TECHNOLOGY

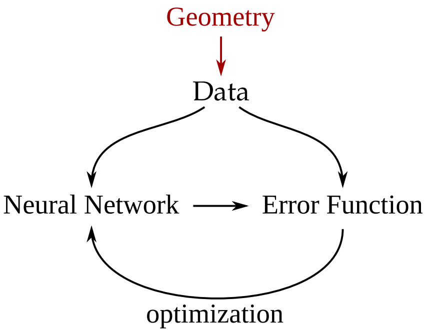
WASP
WALLENBERG AI, AUTONOMOUS SYSTEMS AND SOFTWARE PROGRAM

Seminar talk at
Electronics and Telecommunications Research Institute
Daejeon, Korea

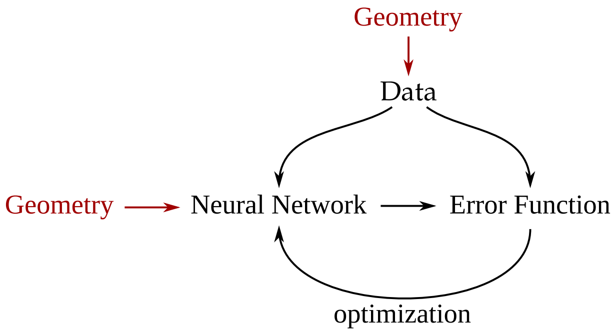
Based on joint work with
Ann-Kathrin Dombrowski, Klaus-Robert Müller and Pan Kessel

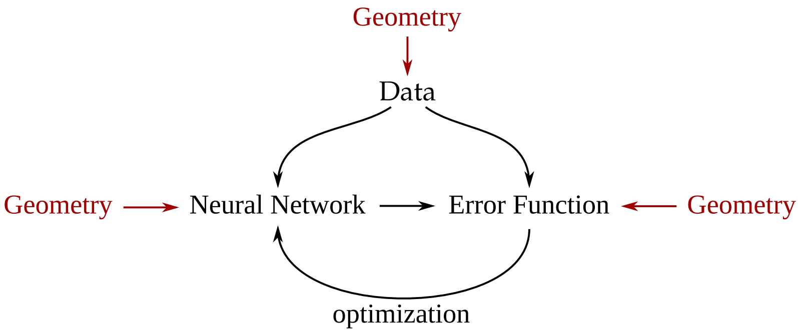

Data
Neural Network
Error Function
optimization

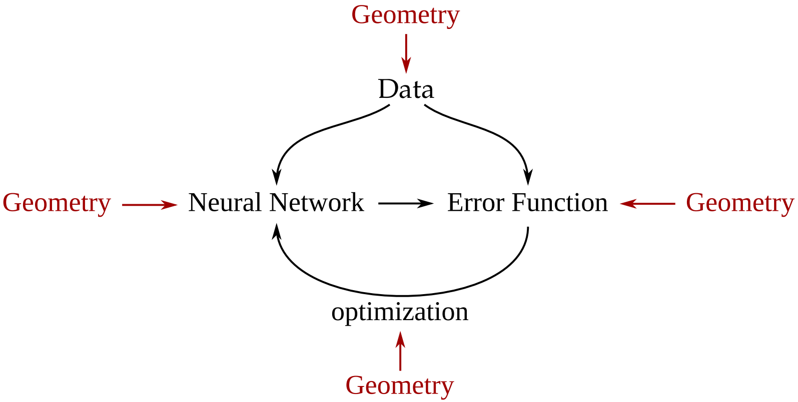
Geometric Deep Learning

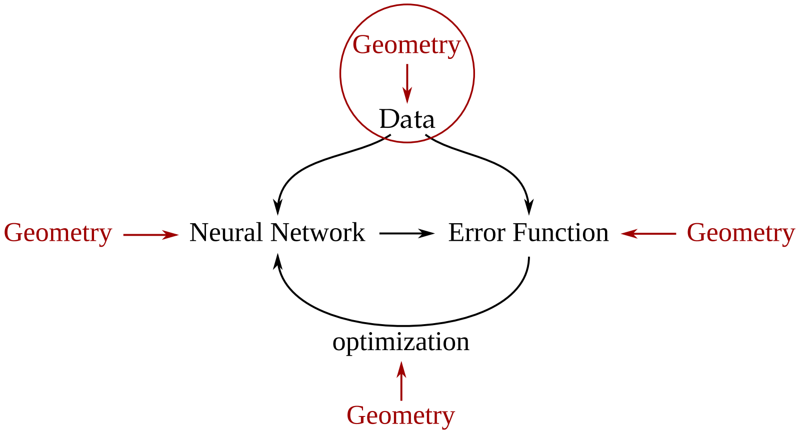


Geometric Deep Learning

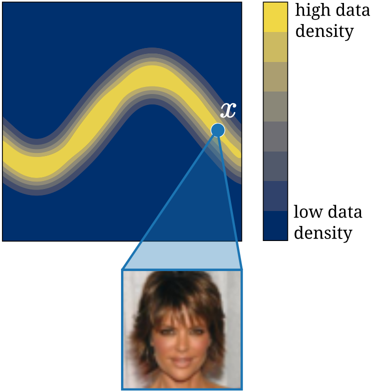



Geometry
Data
Geometry
Neural Network
Error Function
Geometry
optimization


Geometry
Data
Geometry
Neural Network
Error Function
Geometry
optimization
Geometry


Geometry
Data
Geometry
Neural Network
Error Function
Geometry
optimization
Geometry

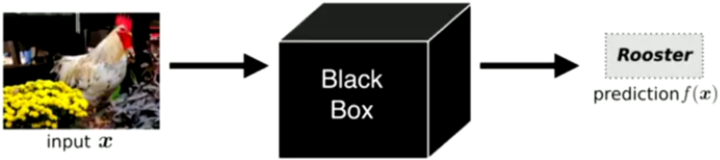
Geometry of the training data



- Manifold Hypothesis: Data lies on low-dim. submanifold of high-dim. input space
- E.g. MNIST pictures lie on $\sim$ 30-dim. submanifold of $28 \times 28 = 784$ dim. input space

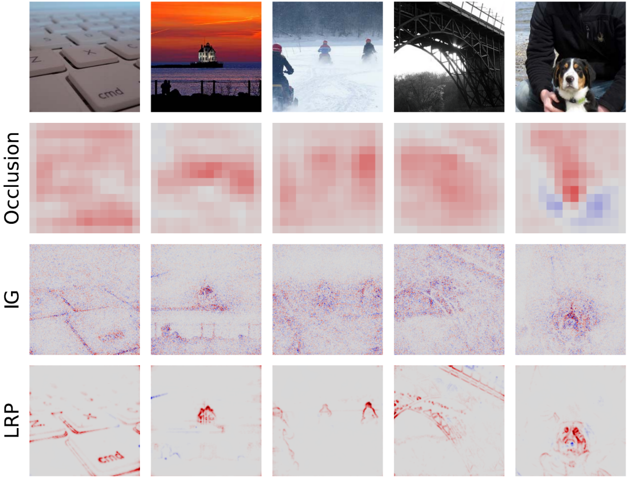
Explainable AI

Explainable AI (XAI)



- Neural network classifiers lack inherent interpretability
- This is in contrast to more traditional methods like linear- or physical models
- For safety-critical applications this poses a serious challenge in practice
- Research progress can also be impeded
- Seek explanations which provide insight into the neural network decisions

Saliency maps

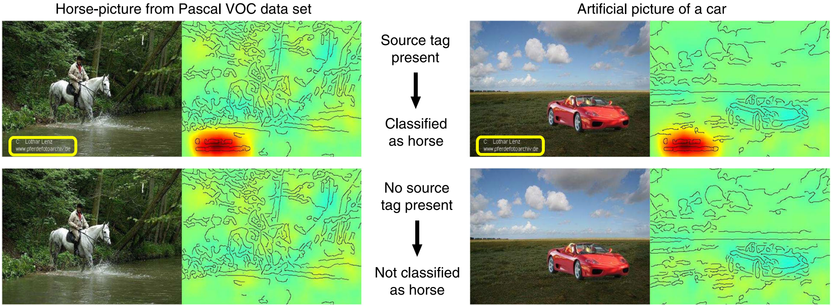


[Samek et al. 2021]

- Highlight areas in the input which were relevant for the classification
- Various different techniques exist
- Often, some form of gradient of output w.r.t. input is used

Clever Hans Effect

[Lapuschkin et al. 2019]



- Model uses spurious correlations in dataset (watermark) to make decision
- Term borrowed from psychology: Humans or animals react to cues given unconsciously by experiment leaders

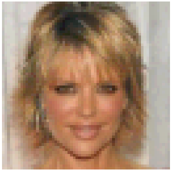
Clever Hans

Counterfactuals and adversarial examples

Counterfactual explanations

- ***Counterfactual of a sample***: Data point close to original but with different classification
- Difference between original and counterfactual reveals features which led to classification
- Example from CelebA dataset, classified as not-blonde:

original x



counterfactual x'

$|x - x'|$

Counterfactuals vs. Adversarial Examples

- To change classification, naively optimize target class k of classifier:

  For a classifier $f : X \to [0,1]^C$, compute

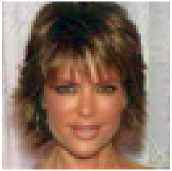
  $$\operatorname*{argmax}_x f_k(x)$$

  approximately by gradient ascent

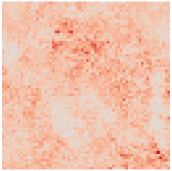
  $$x^{(t+1)} = x^{(t)} + \eta \frac{\partial f_k}{\partial x}(x^{(t)})$$

- Problem: No semantic changes in the image, have obtained ***adversarial example***

original x
blonde $p \approx 0.01$

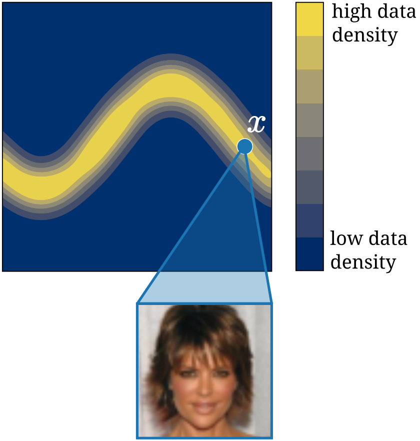


adversarial example x'
blonde $p \approx 0.99$



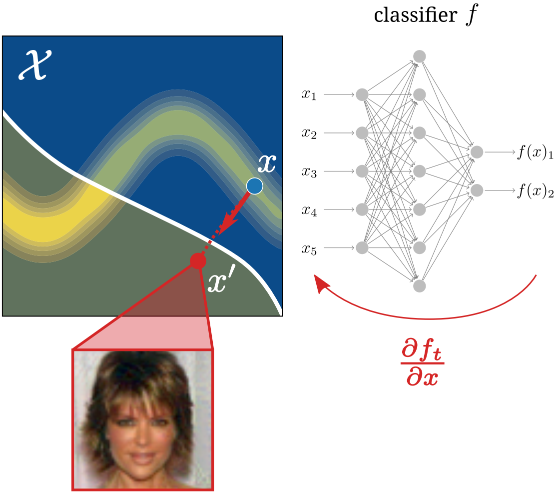
$\propto |x - x'|$

Manifold Hypothesis



- Assume that data lies on a low-dimensional submanifold of high-dimensional input space
- E.g. MNIST pictures lie on $\sim$ 30-dimensional submanifold of $28 \times 28 = 784$ dimensional input space

Adversarial Examples



$$x^{(i+1)} = x^{(i)} + \eta \frac{\partial f_t}{\partial x}(x^{(i)})$$

Normalizing Flows

Recap: Generative models

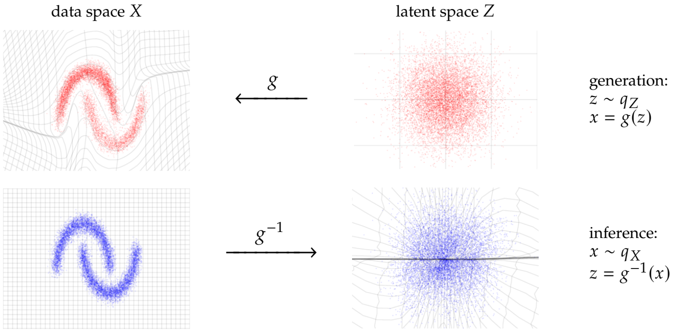
- Task: Given a dataset $\{x_i | i = 1, \ldots, N\}$, generate samples from the underlying distribution.
- One approach: Take samples from a simple latent (e.g. uniform or Gaussian) distribution and map them to samples of the target distribution
$$z \in Z = \mathbb{R}^{n_{\text{latent}}} \sim \mathcal{N}(0, 1) \quad \Rightarrow \quad G(z) \in X = \mathbb{R}^{n_{\text{data}}} \sim p_{\text{data}}$$
- The latent space is often much lower-dimensional than the data space

Normalizing flows

- Generative model g which maps ***base space*** Z to data space X ***bijectively***, i.e. it is a ***diffeomorphism***
- Probability distribution q_Z in Z is simple, e.g. uniform or normal
- Probability distribution q_X in X is given by change of variables

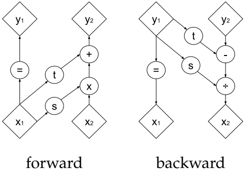
$$q_X(x) = q_Z(g^{-1}(x)) \left| \det \frac{\partial z}{\partial x} \right|$$

- Train via maximum likelihood: $\mathcal{L} = -\mathbb{E}_{x \sim p_{\text{data}}}[\log q_X]$



- Normalizing flow is a neural network with bijective building blocks
- Network needs to be easily invertible and have a tractable Jacobian determinant
- RealNVP uses ***affine coupling layers***

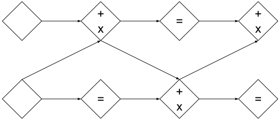
$$
\begin{aligned}
y_{1:d} &= x_{1:d} \\
y_{d+1:D} &= x_{d+1:D} \odot \exp(s(x_{1:d})) + t(x_{1:d}) \\
s, t &: \mathbb{R}^d \rightarrow \mathbb{R}^{D-d} \qquad \text{(deep CNNs)}
\end{aligned}
$$



- The Jacobian is given by

$$
\frac{\partial y}{\partial x^T} = \begin{bmatrix} 1_d & 0 \\ \frac{\partial y_{d+1:D}}{\partial x^T_{1:d}} & \mathrm{diag}\left(\exp(s(x_{1:d}))\right) \end{bmatrix} \quad \Rightarrow \quad \left|\frac{\partial y}{\partial x^T}\right| = \exp\left(\sum_j s(x_{1:d})_j\right)
$$

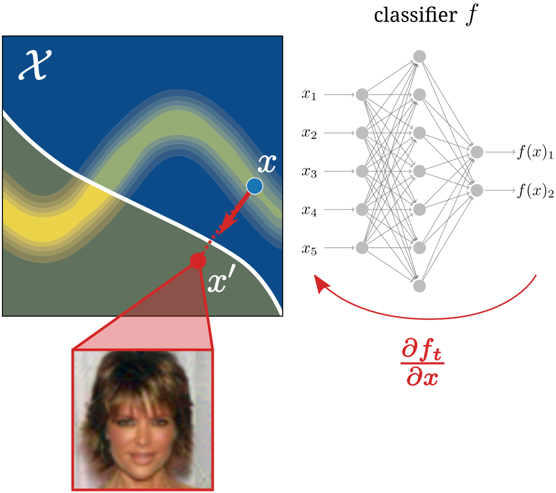
- Alternate the parts which are modified from layer to layer



- RealNVP uses multi-scale architecture

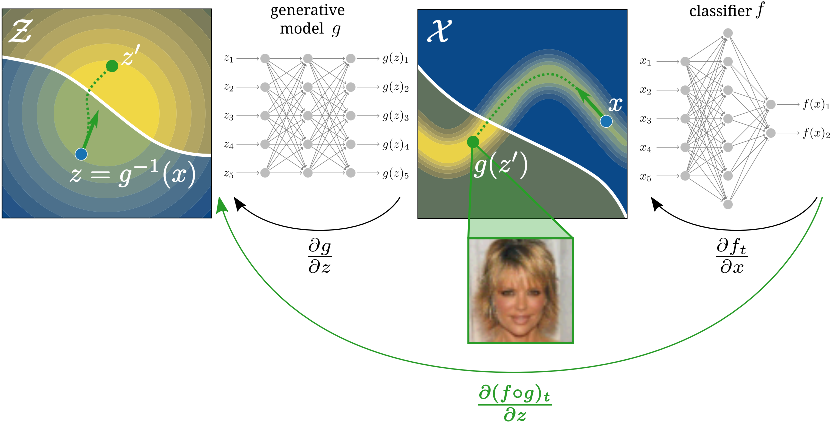
Diffeomorphic Counterfactuals

Recall: Adversarial Examples



$$x^{(i+1)} = x^{(i)} + \eta \frac{\partial f_t}{\partial x}(x^{(i)})$$

Diffeomorphic Counterfactuals



$$z^{(i+1)} = z^{(i)} + \lambda \frac{\partial (f \circ g)_t}{\partial z}(z^{(i)})$$

Gradient ascent in base space

- Gradient ascent in Z for class k of the classifier f with learning rate λ:

$$z^{(t+1)} = z^{(t)} + \lambda \frac{\partial (f \circ g)_k}{\partial z}(z^{(t)})$$

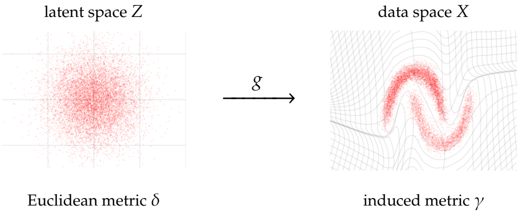
- Using change-of-variable under the flow:

Theorem

Gradient ascent in the base space Z is given by

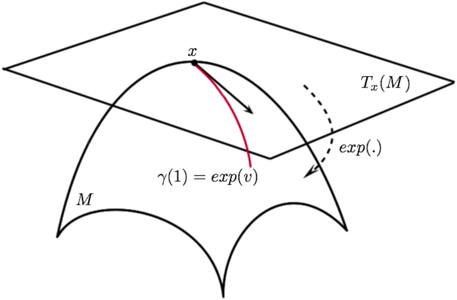
$$x^{(t+1)} = x^{(t)} + \lambda\, \boldsymbol{\gamma^{-1}(x^{(t)})}\, \frac{\partial f_k}{\partial x}(x^{(t)}) + \mathcal{O}(\lambda^2)$$

where $\gamma^{-1}(x) = \left(\frac{\partial g}{\partial z}\frac{\partial g}{\partial z}^T\right)(g^{-1}(x))$ is the inverse of the induced metric on X from Z under the flow g.



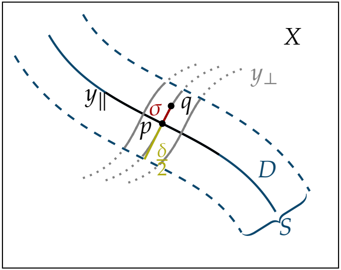
Normal coordinates

- ***Normal coordinates*** are a diffeomorphism-invariant way to construct coordinates in a neighborhood of a point x on a manifold M
- Consider $v \in T_x M$ and the affinely parametrized geodesic γ with
    1. $\gamma(0) = x$
    2. $\gamma'(0) = v$
- The ***exponential map*** maps v to the point $\gamma(1)$



- Choose a basis of $T_x M$
- Normal coordinates assign to a point $y = \exp(v)$ in a neighborhood of x the coordinates of v in the chosen basis

Data coordinates



- Assume that data lies in a region $S = \mathrm{supp}(p)$ around data manifold D, in data coordinates x^α
$$S_x = \left\{ x_D + x_\delta \,\middle|\, x_D \in D_x,\ x_\delta^\alpha \in \left(-\frac{\delta}{2}, \frac{\delta}{2}\right) \right\}$$
with $\delta \ll 1$.

- Define normal coordinates y^μ in a neighborhood of D by
  - Choose coordinates $y_\parallel$ on D and for each $p \in D$ a basis $\{n_i\}$ of $T_p D_\perp$
  - Construct affinely parametrized geodesic $\sigma : [0,1] \to X$ with $\sigma(0) = p$, $\sigma(1) = q$ and $\sigma'(0) \in T_p D_\perp$
  - The coordinates of q are given by $y_\parallel$ and the components $y_\perp^i$ of $\sigma'(0)$ in the basis $\{n_i\}$
  - For sufficiently small neighborhoods, this is unique
  - Rescale $\{n_i\}$ so that S in y coordinates also has extension δ

Gradient ascent in y-coordinates

- By choosing $\{n_i\}$ orthogonal wrt γ, the inverse induced metric takes the form

$$\gamma^{\mu\nu}(y) = \begin{pmatrix} \gamma_D^{-1}(y) & & & \\ & \gamma_{\perp_1}^{-1} & & \\ & & \ddots & \\ & & & \gamma_{\perp_{N_X - N_D}}^{-1} \end{pmatrix}^{\mu\nu} .$$

- The gradient ascent update $g^\alpha(z^{(i+1)}) = g^\alpha(z^{(i)}) + \lambda\, \gamma^{\alpha\beta} \frac{\partial f_t}{\partial x^\beta} + \mathcal{O}(\lambda^2)$ becomes

$$\gamma^{\alpha\beta} \frac{\partial f_t}{\partial x^\beta} = \frac{\partial x^\alpha}{\partial y_{\|}^\mu} \gamma_D^{\mu\nu} \frac{\partial f_t}{\partial y_{\|}^\nu} + \frac{\partial x^\alpha}{\partial y_\perp^i} \gamma_{\perp_i}^{-1} \frac{\partial f_t}{\partial y_\perp^i}$$

- For $\gamma_{\perp_i}^{-1} \to 0$ and $\frac{\partial x}{\partial y_\perp}$ bounded we have

$$\gamma^{\alpha\beta} \frac{\partial f_t}{\partial x^\beta} \to \frac{\partial x^\alpha}{\partial y_{\|}^\mu} \gamma_D^{\mu\nu} \frac{\partial f_t}{\partial y_{\|}^\nu}$$

  and hence the update step points along the data manifold.

$\Rightarrow$ ***In this case, obtain counterfactuals, not adversarial examples!***

The induced metric for well-trained generative models

Theorem (Diffeomorphic Counterfactuals)

For $\epsilon \in (0,1)$ and g a normalizing flow with Kullback–Leibler divergence $\mathrm{KL}(p,q) < \epsilon$,

$$\gamma_{\perp_i}^{-1} \longrightarrow 0 \qquad \text{as} \qquad \delta \longrightarrow 0$$

for all $i \in \{1, \ldots, N_X - N_D\}$.

Theorem (Approximately Diffeomorphic Counterfactuals)

If $g : Z \longrightarrow X$ is a generative model with $D \subset g(Z)$ and image $g(Z)$ which extends in any non-singular orthogonal direction $y_\perp^i$ to regions outside of D of low probability $p(x) \ll 1$,

$$\gamma_{\perp_i}^{-1} \longrightarrow 0$$

for $\delta \longrightarrow 0$ for all non-singular orthogonal directions $y_\perp^i$.

$\Rightarrow$ ***For well-trained generative models, the gradient ascent update in Z stays on the data manifold***

Sketch of proof (flow-case)

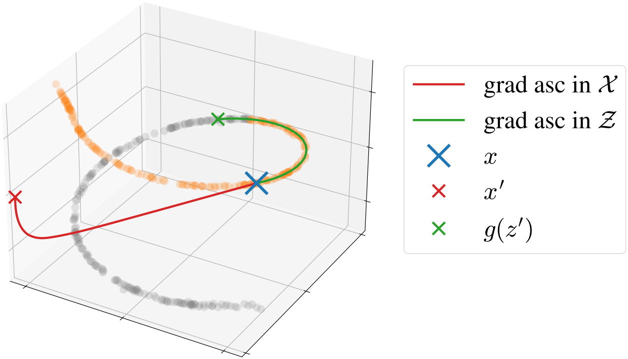
- For flows g with $\mathrm{KL}(p, q) < \epsilon$, almost all probability mass is concentrated in $S = \mathrm{supp}(p)$

$$
\begin{aligned}
0 < 1 - \epsilon < & \int_{S_x} q_X(x)\, \mathrm{d}x \\
= & \int_{S_x} q_Z(g^{-1}(x)) \left| \frac{\partial z^a}{\partial x^\alpha} \right| \mathrm{d}x \\
= & \int_{D_y} \sqrt{|\gamma_D|} \prod_{i=1}^{N_X - N_D} \int_{-\delta/2}^{\delta/2} \sqrt{|\gamma_{\perp_i}|}\, q_Z(z(y))\, \mathrm{d}y_\perp^i\, \mathrm{d}y_\parallel
\end{aligned}
$$

- When $\delta \to 0$, the integration domain shrinks to zero, but the value of the integral is bounded from below

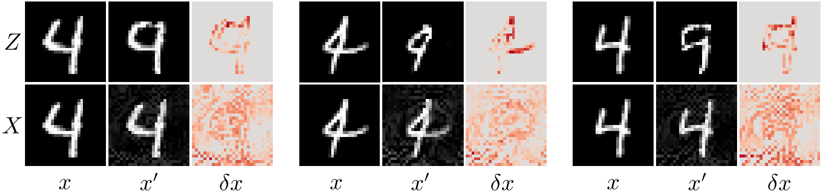
- Hence, the metric $\gamma_{\perp_i}$ has to diverge, i.e. $\gamma_{\perp_i}^{-1} \to 0$.

Toy example



Diffeomorphic Counterfactuals with MNIST

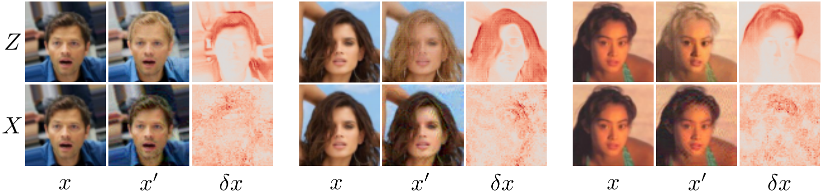
- Classifier: CNN with 10 classes (test accuracy: 99%)
- Flow: RealNVP
- Task: Change classification of 4 to 9



- Top row: Counterfactual computed in base space
- Bottom row: Adersarial example computed in data space

Diffeomorphic Counterfactuals with CelebA

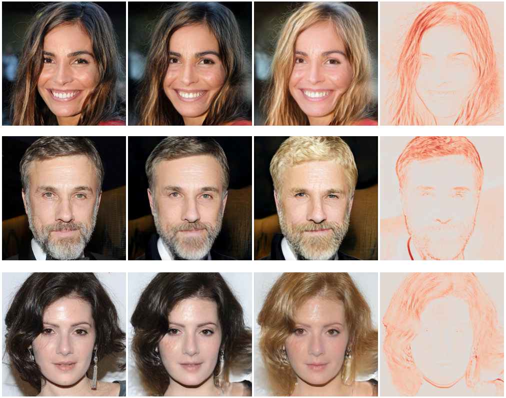
- Classifier: Binary CNN trained on ***blonde/not blonde*** attribute (test accuracy: 94%)
- Flow: Glow [Kingma et al. 2018]
- Task: Change classification from ***not blonde*** to ***blonde***



- Top row: Counterfactual computed in base space
- Bottom row: Adersarial example computed in data space

Approximately Diffeomorphic Counterfactuals with CelebA-HQ

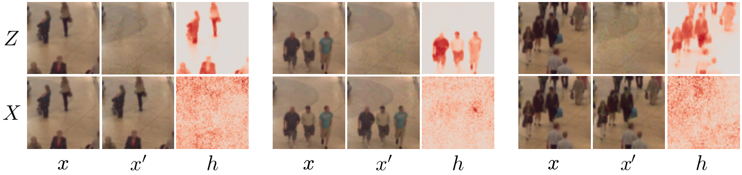
- For general generating models, inversion not exact ($\tilde{x} = g(z_0) \neq x_0$)
- Approximate diffeomorphic counterfactuals can be generated for high-dimensional datasets (1024×1024 pixels for CelebA-HQ)
- Use StyleGAN trained on CelebA-HQ [Karras et al. 2019]
- Use HyperStyle inversion of StyleGAN to find initial latent [Alaluf et al. 2022]



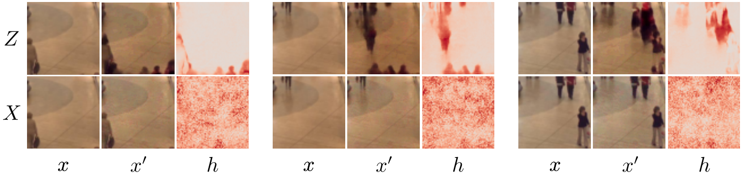
x $\tilde{x}$ $\tilde{x}'$ h

Diffeomorphic Counterfactuals for regression

- Consider crowd-counting dataset of mall images
- Count number of people in the image [Ribera et al. 2019]
- Flow: Glow [Kingma et al. 2018]
- Optimize for low number of people

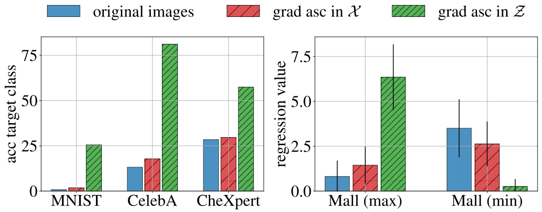


- Optimize for high number of people

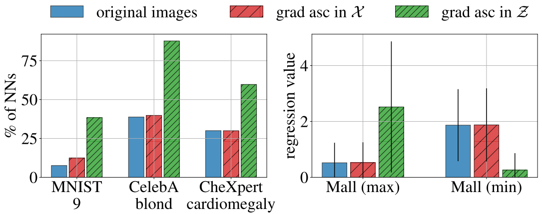


Quantitative evaluation

- Diffeomorphic Counterfactuals generalize to SVMs, adversarials do not



- The ground truth classes for the ten nearest neighbors match the target values of the counterfactuals more often for Diffeomorphic Counterfactuals than for adversarials

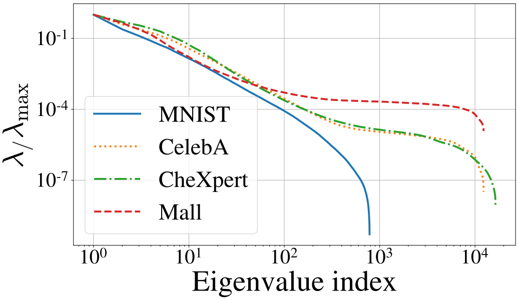


Dimension of the data manifold

- From induced metric, can infer tangent space of data manifold
- Perform singular value decomposition of the Jacobian $\frac{\partial g}{\partial z} = U\,\Sigma\,V$
- Rewrite the inverse induced metric as

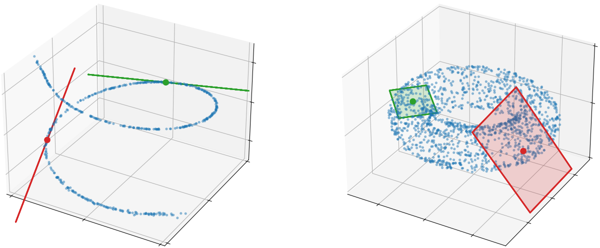
$$\gamma^{-1} = \frac{\partial g}{\partial z}\frac{\partial g}{\partial z}^{T} = U\,\Sigma^2\,U^T$$

- For N dimensional data manifold: N large singular values



Tagent space of data manifold

- The left-singular vectors of the Jacobian corresponding to large eigenvalues span the tangent space of the data manifold
- For toy data:



Conclusions

Summary

- Geometry can be used at many points along the deep learning pipeline
- Counterfactuals explain black-box classifiers by providing a realistic sample close to the original but with a different classification
- However, gradient ascent optimization of the target class leads to adversarial examples which lie off the data manifold
- Normalizing flows are bijective generative models
- Gradient ascent optimization in the base space of a normalizing flow stays on the data manifold and leads to counterfactuals
- For non-bijective generative models, this is still true approximately
- The reason is that the induced metric makes the learning rate in orthogonal directions small

Outlook

- Can one learn something about the invertibility of normalizing flows using this construction?
- The construction is very general, can it be applied to other problems where a neural network output needs to be optimized on a data manifold given by a generative model?

Diffeomorphic Counterfactuals with Generative Models

arXiv: 2206.05075

Accepted at IEEE PAMI

Thank you!

Appendix

Training normalizing flows

- Train by maximizing log-likelihood $\mathbb{E}_{x \sim p_{\text{data}}}[\log q_X]$ of the train data

$$\mathcal{L} = -\frac{1}{N} \sum_{i=1}^{N} \log q_X(x_i)$$

- Recall that this corresponds to minimizing the forward KL divergence of the data distribution p_{data} from q_X (Lecture 1)

$$\text{KL}(p_{\text{data}} || q_X) = \mathbb{E}_{x \sim p_{\text{data}}}[\log p_{\text{data}}(x) - \log q_X(x)]$$

- If we know p_{target} up to a constant

$$p_{\text{target}} = \frac{1}{Z} e^{-E(x)}$$

  can also train using reverse KL divergence

$$\begin{aligned} \text{KL}(q_X || p_{\text{target}}) &= \mathbb{E}_{x \sim q_X}[\log q_X(x) - \log p_{\text{data}}(x)] \\ &= \mathbb{E}_{x \sim q_X}[\log q_X(x)] + \mathbb{E}_{x \sim q_X}[E(x)] + \log Z \end{aligned}$$

  So the loss is

$$\mathcal{L} = \frac{1}{N} \sum_{i=1}^{N} \log(q_X(x_i)) + E(x_i)$$

  where the x_i are sampled from the flow (***self sampling***)

Gradient ascent in base space

- Gradient ascent in Z for class k of the classifier f with learning rate λ:

$$z^{(t+1)} = z^{(t)} + \lambda \frac{\partial (f \circ g)_k}{\partial z}(z^{(t)})$$

- Using change-of-variable under the flow:

$$\begin{aligned}
x^{(t+1)} = g(z^{(t+1)}) &= g\left(z^{(t)} + \lambda \frac{\partial (f \circ g)_k}{\partial z}\right) \\
&= g(z^{(t)}) + \lambda \sum_i \frac{\partial g}{\partial z_i} \frac{\partial (f \circ g)_k}{\partial z_i} + \mathcal{O}(\lambda^2) \\
&= g(z^{(t)}) + \lambda \sum_{i,j} \frac{\partial g}{\partial z_i} \frac{\partial g_j}{\partial z_i} \frac{\partial f_k}{\partial g_j} + \mathcal{O}(\lambda^2) \\
&= x^{(t)} + \lambda \, \gamma^{-1}(x^{(t)}) \frac{\partial f_k}{\partial x}(x^{(t)}) + \mathcal{O}(\lambda^2)
\end{aligned}$$

with $\gamma^{-1}(x) = (\frac{\partial g}{\partial z} \frac{\partial g}{\partial z}^T)(g^{-1}(x))$

Gradient ascent in base space

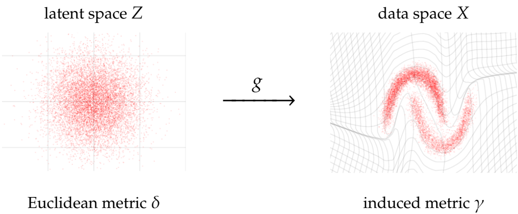
- When performing gradient ascent in X we do

$$x^{(t+1)} = x^{(t)} + \lambda \frac{\partial f_k}{\partial x}(x^{(t)})$$

  vs. gradient ascent in Z

$$x^{(t+1)} = x^{(t)} + \lambda \, \boldsymbol{\gamma^{-1}(x^{(t)})} \frac{\partial f_k}{\partial x}(x^{(t)}) + \mathcal{O}(\lambda^2)$$

- The additional term $\gamma^{-1} = \left(\frac{\partial g}{\partial z} \frac{\partial g}{\partial z}^T\right)$ is the inverse of the induced metric on X from Z



- Effectively, γ^{-1} aligns the gradient with the data manifold

Proof: diffeomorphic counterfactuals

- For flows g with $\mathrm{KL}(p_{\mathrm{data}}, q_X) < \epsilon$, almost all probability mass is concentrated in $S = \mathrm{supp}(p)$

$$\epsilon > \mathrm{KL}(p_{\mathrm{data}}, q_X) = \int_{S_x} p_{\mathrm{data}}(x) \ln\left(\frac{p_{\mathrm{data}}(x)}{q_X(x)}\right) \mathrm{d}x$$
$$\geq \int_{S_x} p_{\mathrm{data}}(x) \left(1 - \frac{q_X(x)}{p_{\mathrm{data}}(x)}\right) \mathrm{d}x = 1 - \int_{S_x} q_X(x)\, \mathrm{d}x$$

where we have used the inequality

$$\ln\left(\frac{1}{a}\right) \geq 1 - a \quad \Leftrightarrow \quad \ln(a) \leq a - 1$$

and therefore

$$0 < 1 - \epsilon < \int_{S_x} q_X(x)\, \mathrm{d}x = \int_{S_x} q_Z(g^{-1}(x)) \left|\frac{\partial z^a}{\partial x^\alpha}\right| \mathrm{d}x$$

Proof: diffeomorphic counterfactuals

- Evaluate the integral in y^α coordinates

$$1-\epsilon < \int_{S_x} q_Z(g^{-1}(x)) \left|\frac{\partial z^a}{\partial x^\alpha}\right| \mathrm{d}x = \int_{S_y} q_Z(g^{-1}(x(y))) \left|\frac{\partial z^a}{\partial x^\alpha}\right| \left|\frac{\partial x^\alpha}{\partial y^\mu}\right| \mathrm{d}y$$

Since

$$\gamma_{\mu\nu}(y) = \begin{pmatrix} \gamma_D(y) & & & \\ & \gamma_{\perp_1} & & \\ & & \ddots & \\ & & & \gamma_{\perp_{N_X - N_D}} \end{pmatrix}_{\mu\nu}$$

$$\Rightarrow \left|\frac{\partial z^a}{\partial x^\alpha}\right| \left|\frac{\partial x^\alpha}{\partial y^\mu}\right| = \sqrt{|\gamma_{\mu\nu}|} = \sqrt{|\gamma_D|} \prod_{i=1}^{N_X - N_D} \sqrt{|\gamma_{\perp_i}|}$$

we get

$$1-\epsilon < \int_{D_y} \sqrt{|\gamma_D|} \prod_{i=1}^{N_X-N_D} \int_{-\delta/2}^{\delta/2} \sqrt{|\gamma_{\perp_i}|}\, q_Z(z(y))\, \mathrm{d}y_\perp^i \, \mathrm{d}y_\parallel$$

- When $\delta \to 0$, the integration domain shrinks to zero, but the value of the integral is bounded from below
- Hence, the metric $\gamma_{\perp_i}$ has to diverge, i.e. $\gamma_{\perp_i}^{-1} \to 0$.