# Diffeomorphic Counterfactuals and Generative Models

Jan E. Gerken

Technische Universität Berlin

BIFOLD

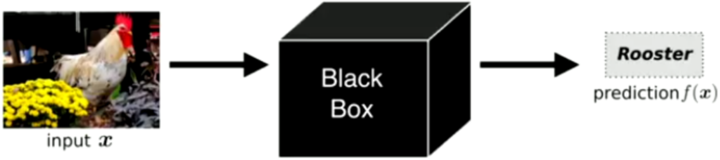
Geometric Deep Learning Seminar
Chalmers University of Technology

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

Published as a Contributed Talk at
2021 ICML Workshop on Invertible Neural Networks
Normalizing Flows, and Explicit Likelihood Models

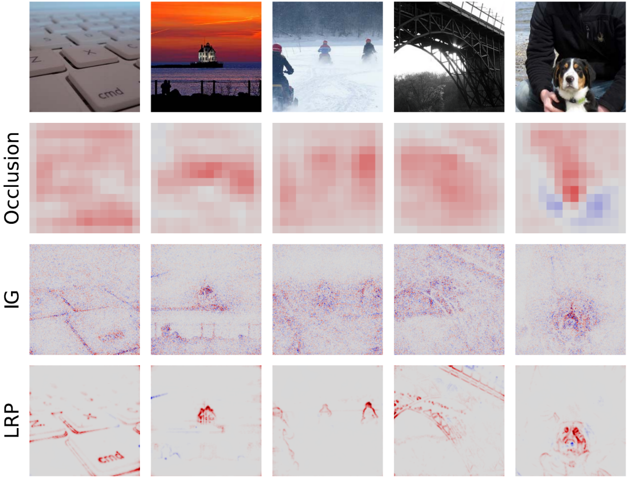
Journal Paper under review

# Explainable AI (XAI)



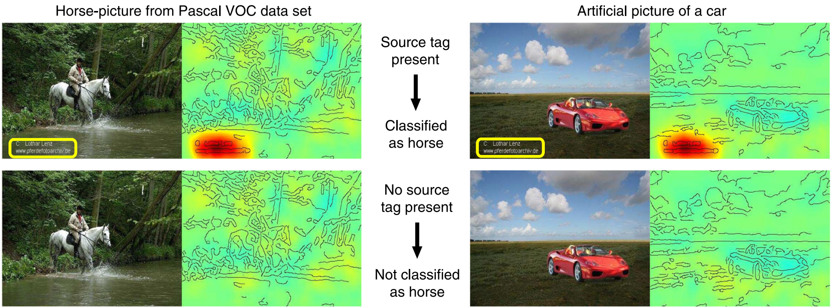

- Neural network classifiers lack inherent interpretability
- This is in contrast to more traditional methods like linear- or physical models
- For safety-critical applications this poses a serious challenge in practice
- Research progress can also be impeded
- Explanations which provide insight into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

# Saliency maps



# Clever Hans Effect

[Lapuschkin et al., Nat Commun 2019]

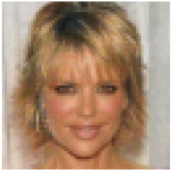


# Clever Hans

# 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 of classifier:

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

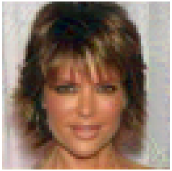
  $$\operatorname{argmax}_x f_k(x)$$

  approximately by gradient ascent

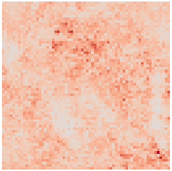
  $$x^{(t+1)} = x^{(t)} + \eta \frac{\partial f_k}{\partial x}(x^{(t)})$$

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

original $x$
blonde $p \approx 0.01$

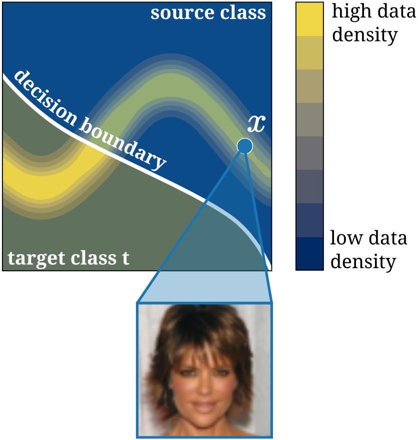


adversarial example $x'$
blonde $p \approx 0.99$



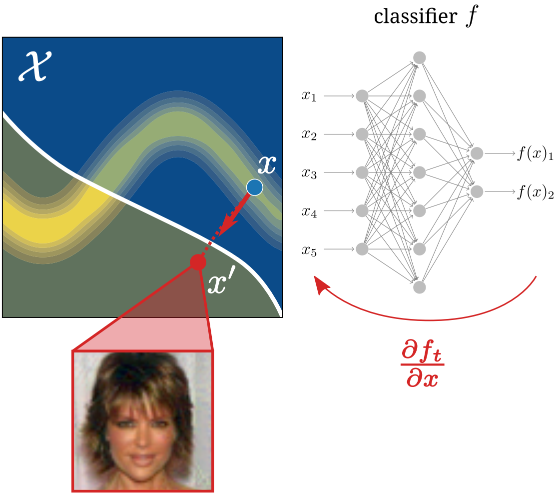
$\propto |x - x'|$

# 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



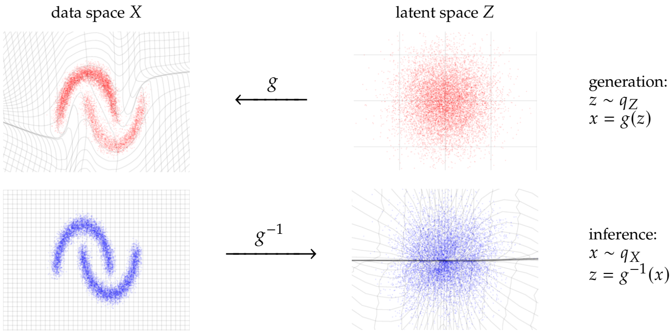

$$x^{(i+1)} = x^{(i)} + \eta \frac{\partial f_t}{\partial x}(x^{(i)})$$

# Normalizing flows

- 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 by maximizing log-likelihood $\log q_X$ of train data

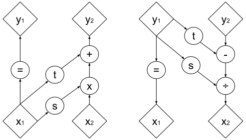


# RealNVP

[Dinh et al., ICLR 2017]

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

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

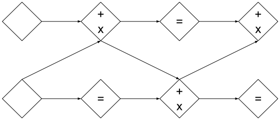



forward backward

- The Jacobian is given by

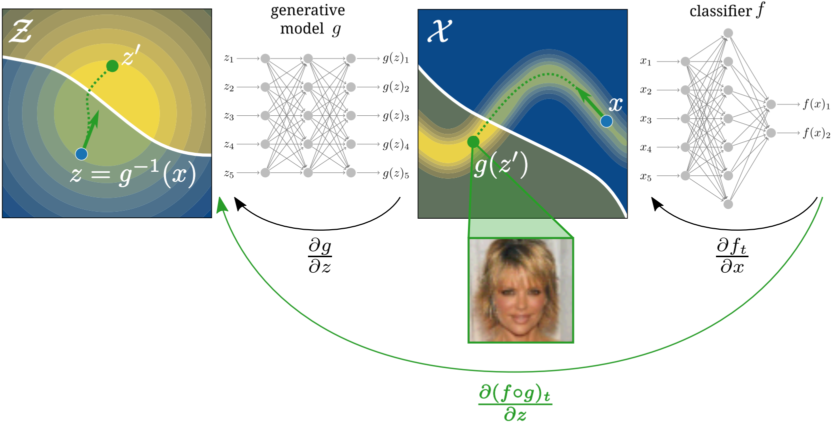
$$
\frac{\partial y}{\partial x^T} = \begin{bmatrix} 1_d & 0 \\ \frac{\partial y_{d+1:D}}{\partial x_{1:d}^T} & \operatorname{diag}\left(\exp(s(x_{1:d}))\right) \end{bmatrix} \quad \Rightarrow \quad \left|\frac{\partial y}{\partial x^T}\right| = \exp\left(\sum_j s(x_{1:d})_j\right)
$$

- Alternate the parts which are modified from layer to layer




- RealNVP uses multi-scale architecture

# Diffeomorphic Counterfactuals




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

# Gradient ascent in base space

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

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

- Using change-of-variable under the flow:

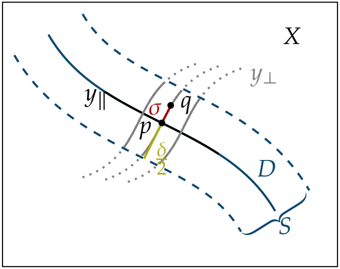
***Theorem***

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 $g$.

# Data coordinates




- Assume that data lies in a region $S = \mathrm{supp}(p)$ around data manifold $D$, in data coordinates $x^\alpha$

$$S_x = \left\{ x_D + x_\delta \,\middle|\, x_D \in D_x,\ x_\delta^\alpha \in \left(-\frac{\delta}{2}, \frac{\delta}{2}\right) \right\}$$

  with $\delta \ll 1$.

- Define normal coordinates $y^\mu$ 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 $\delta$

## Gradient ascent in $y$-coordinates

- By choosing $\{n_i\}$ orthogonal wrt $\gamma$, 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

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

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

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

  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 $\mathrm{KL}(p, q) < \epsilon$,

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

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

***Theorem (Approximately Diffeomorphic Counterfactuals)***

If $g : Z \to X$ 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 $p(x) \ll 1$,

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

for $\delta \to 0$ 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)

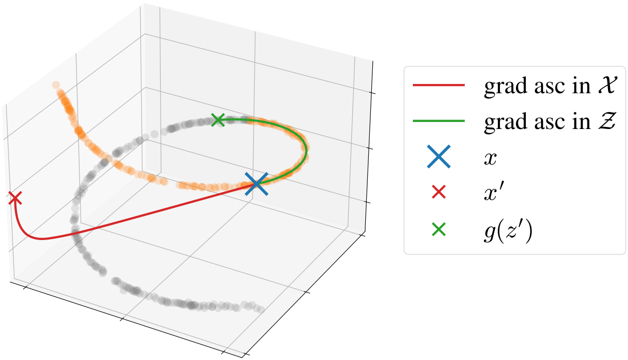
- For flows $g$ with $\mathrm{KL}(p,q) < \epsilon$, almost all probability mass is concentrated in $S = \mathrm{supp}(p)$

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

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

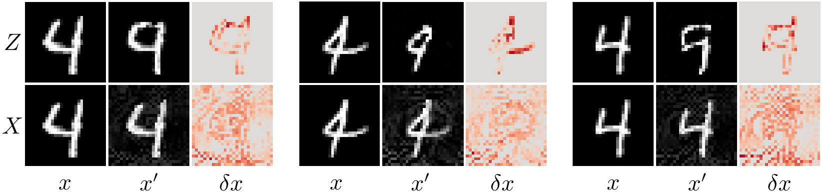
- Hence, the metric $\gamma_{\perp_i}$ has to diverge, i.e. $\gamma_{\perp_i}^{-1} \to 0$.

# Toy example



# Diffeomorphic Counterfactuals with MNIST

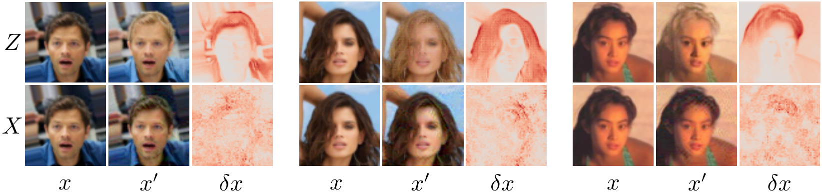
- Classifier: CNN with 10 classes (test accuracy: 99%)
- Flow: RealNVP [Dinh et al., ICLR 2017]
- Task: Change classification of 4 to 9




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

# Diffeomorphic Counterfactuals with CelebA

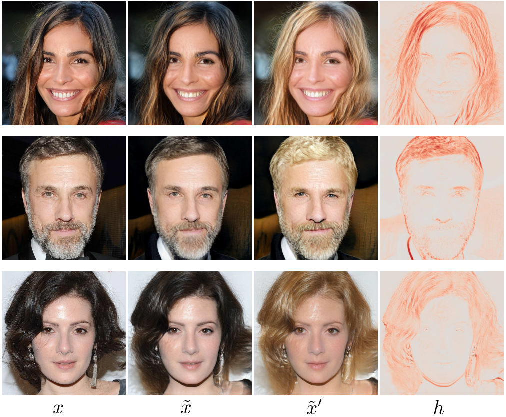
- Classifier: Binary CNN trained on ***blonde/not blonde*** attribute (test accuracy: 94%)
- Flow: Glow [Kingma et al., NeurIPS 2018]
- Task: Change classification from ***not blonde*** to ***blonde***




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

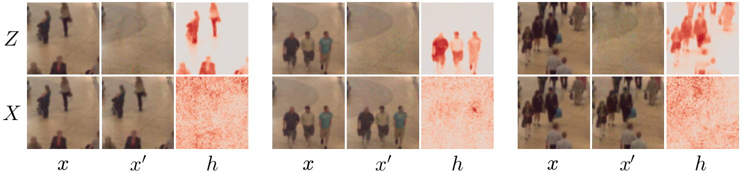
# Approximately Diffeomorphic Counterfactuals with CelebA-HQ

- 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 ($1024 \times 1024$ pixels for CelebA-HQ)
- Use StyleGAN trained on CelebA-HQ [Karras et al., IEEE/CVF 2019]
- Use HyperStyle inversion of StyleGAN to find initial latent [Alaluf et al., CVPR 2022]

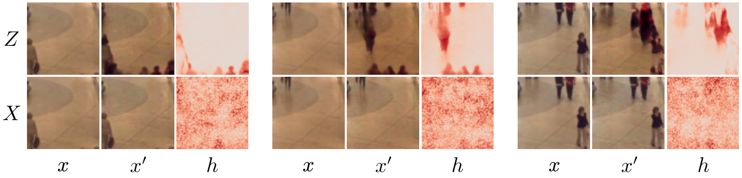


# Diffeomorphic Counterfactuals for regression

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

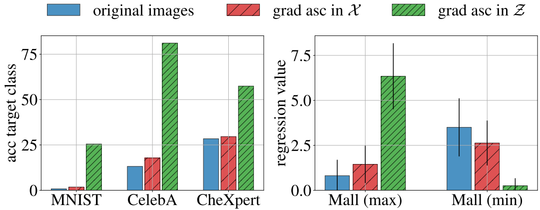



- Optimize for high number of people

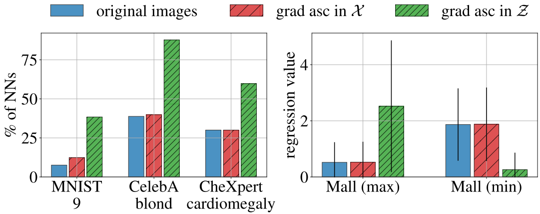


# Quantitative evaluation

- Diffeomorphic Counterfactuals generalize to SVMs, adversarials do not




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



# Conclusion

Summary

- 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?

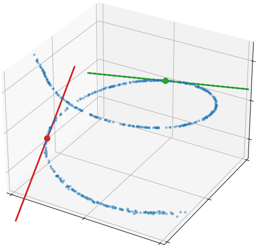
# Appendix

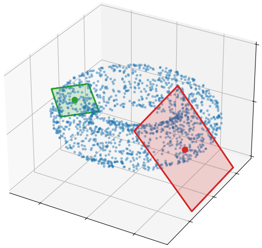
# Tangent space of data manifold

- 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

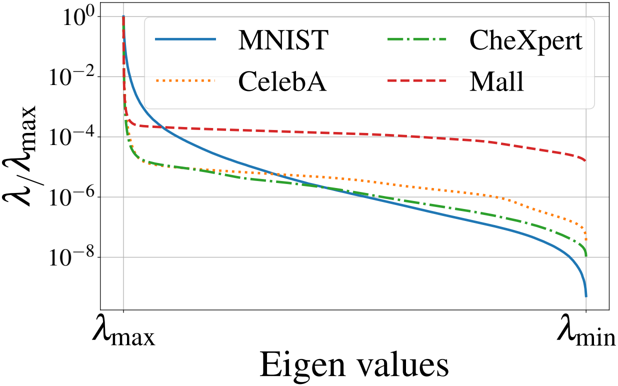
$$\gamma^{-1} = \frac{\partial g}{\partial z} \frac{\partial g}{\partial z}^T = U \Sigma^2 U^T$$

- For $N$ dimensional data manifold: $N$ large singular values
- Corresponding left-singular vectors span the tangent space of the data manifold
- For toy data:





# Eigenvalue spectrum of Jacobian



# Approximate Counterfactuals with VAEs

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