Conditional Normalising Flows for Interpretability

Valentyn Melnychuk

Master Thesis
MSc Data Science
Faculty of Mathematics, Informatics and Statistics
Ludwig-Maximilian University of Munich

Supervised by: PhD Candidate Gunnar König,
Univ.-Prof. Dr.-Ing. Moritz Grosse-Wentrup

March 22, 2021

Outline

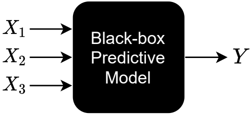
Outline

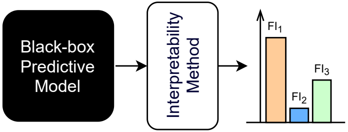
Intro – Interpretability

Feature importance (FI)

scores how much feature contributes to model's performance/prediction variance

Main focus of thesis: **post-hoc model-agnostic global feature importance**





Two stages of interpretability: fitting the model and inferring feature importances

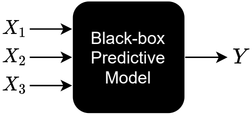
Intro – Interpretability

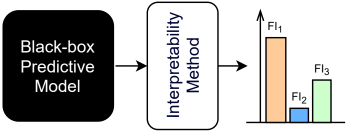
Feature importance (FI)

scores how much feature contributes to model's performance/prediction variance

Main focus of thesis: **post-hoc model-agnostic global feature importance**

- post-hoc – applied to fitted model





Two stages of interpretability: fitting the model and inferring feature importances

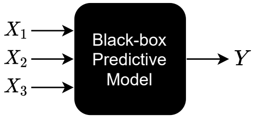
Intro – Interpretability

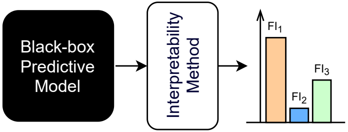
Feature importance (FI)

scores how much feature contributes to model's performance/prediction variance

Main focus of thesis: **post-hoc model-agnostic global feature importance**

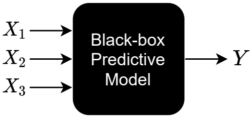
- post-hoc – applied to fitted model
- model-agnostic – not bound to specific model class





Two stages of interpretability: fitting the model and inferring feature importances

Intro – Interpretability

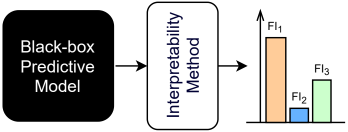


Feature importance (FI)

scores how much feature contributes to model's performance/prediction variance

Main focus of thesis: **post-hoc model-agnostic global feature importance**

- post-hoc – applied to fitted model
- model-agnostic – not bound to specific model class
- global – feature contribution to the overall performance (not to the individual prediction)



Two stages of interpretability: fitting the model and inferring feature importances

Intro – Perturbation-based feature importances

Feature importance is a **difference of generalisation risks** of original model and model with perturbed feature (**replacement variable**).

Permutation Feature Importance (PFI)

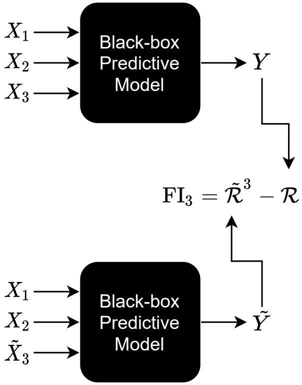
Replacement variable is sampled independently from marginal:

$$\tilde{X}_j \sim p(x_j)$$

Conditional Feature Importance (CFI)

Replacement variable is sampled conditionally on all the other features:

$$\tilde{X}_j \sim p(x_j / x_{-j})$$



Intro – Perturbation-based feature importances

Interpretation – destruction of relationship between feature and target:

- PFI has connection to interventional importance
- CFI estimates observational importance (ultimate importance of feature, if one knows values of all other features)

Relative Feature Importance (RFI) [König et al., 2020]

Replacement variable is sampled conditionally on **some subset** of features G, ranging from empty to full subset (also unused features):

$$\tilde{X}_j \sim p(x_j / x_G)$$

Intro – Importances via restricted models

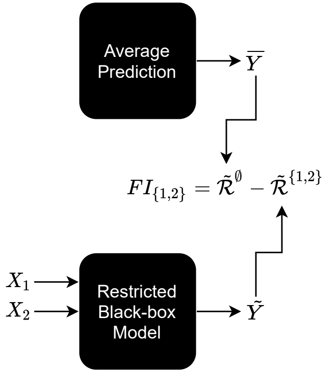
Other possibility to infer feature importance – use marginalised (restricted on set of features S) predictions of model:

$$h_S(x_S) = \mathbb{E}_{X_{\overline{S}} \sim p(x_{\overline{S}}/x_S)}\, h(x_S, X_{\overline{S}})$$

Then, individual / collective feature importance is defined as **reduction in risk over the average prediction**.

Intrinsically, to estimate restricted model, we also perform conditional sampling:

$$\tilde{X}_{\overline{S}} \sim p(x_{\overline{S}}/x_S)$$



Intro – Collective feature importances

Collective feature importances can be calculated for all the possible subsets $\rightarrow$ computational issues due to an exponential number of subsets

$\phi_1 \quad \phi_2 \quad \phi_3$

$$\mathrm{FI}_1 \approx \phi_1$$
$$\mathrm{FI}_2 \approx \phi_2$$
$$\mathrm{FI}_3 \approx \phi_3$$
$$\mathrm{FI}_{\{1,2\}} \approx \phi_1 + \phi_2$$
$$\mathrm{FI}_{\{2,3\}} \approx \phi_2 + \phi_3$$
$$\mathrm{FI}_{\{1,2,3\}} \approx \phi_1 + \phi_2 + \phi_3$$

Shapley Additive Global Importance (SAGE) [Covert et al., 2020]

Additive individual importances ϕ_j, which approximate conditional collective contributions of feature sets.

SAGE estimates require a linear number of evaluations.

Intro – Research Gap

Unrealistic assumptions

Replacement variable / set of variables are sampled from unrealistical conditional distributions:

Intro – Research Gap

Unrealistic assumptions

Replacement variable / set of variables are sampled from unrealistical conditional distributions:

- **RFI** experimented with multivariate Gaussian data & used conditional Gaussian distribution $\rightarrow$ complex distributions?

Intro – Research Gap

Unrealistic assumptions

Replacement variable / set of variables are sampled from unrealistical conditional distributions:

- **RFI** experimented with multivariate Gaussian data & used conditional Gaussian distribution $\rightarrow$ complex distributions?
- **SAGE** assumed features are independent and sampled from marginal distribution $\rightarrow$ unrealistic, off-manifold data generation!

Intro – Research Gap

Unrealistic assumptions

Replacement variable / set of variables are sampled from unrealistical conditional distributions:

- **RFI** experimented with multivariate Gaussian data & used conditional Gaussian distribution → complex distributions?
- **SAGE** assumed features are independent and sampled from marginal distribution → unrealistic, off-manifold data generation!

  Attempts to mitigate the problem for local SAGE (SHAP):

  - [Frye et al., 2020] – conditional VAE
  - [Aas et al., 2019] – conditional Gaussian, Gaussian copula, kernel estimates
  - [Mase et al., 2019] – selection of existing datapoints via similarity function

No empirical studies on how goodness-of-fit of an estimated sampler is related to FI inference.

Intro - Contribution

We propose to use **deep density estimator with tractable likelihood** for conditional sampling in global feature importance estimation:

Intro - Contribution

We propose to use **deep density estimator with tractable likelihood** for conditional sampling in global feature importance estimation:

- We utilise Conditional Normalising Flows (CNFs) and Mixture Density Networks (MDNs) (concurrent method)

Intro - Contribution

We propose to use **deep density estimator with tractable likelihood** for conditional sampling in global feature importance estimation:

- We utilise Conditional Normalising Flows (CNFs) and Mixture Density Networks (MDNs) (concurrent method)
- We empirically study, how goodness-of-fit and different distributional properties translates to the validity of estimated RFI and SAGE values, based on self-designed benchmark

Intro - Contribution

We propose to use **deep density estimator with tractable likelihood** for conditional sampling in global feature importance estimation:

- We utilise Conditional Normalising Flows (CNFs) and Mixture Density Networks (MDNs) (concurrent method)
- We empirically study, how goodness-of-fit and different distributional properties translates to the validity of estimated RFI and SAGE values, based on self-designed benchmark
- We provide the use case of deep samplers for detecting the influences of sensitive attributes

Intro - Contribution

We propose to use **deep density estimator with tractable likelihood** for conditional sampling in global feature importance estimation:

- We utilise Conditional Normalising Flows (CNFs) and Mixture Density Networks (MDNs) (concurrent method)
- We empirically study, how goodness-of-fit and different distributional properties translates to the validity of estimated RFI and SAGE values, based on self-designed benchmark
- We provide the use case of deep samplers for detecting the influences of sensitive attributes
- Code contributions: extension of RFI Python library (deep density estimators, synthetic benchmark)

Outline

Overview of Methods – Conditional Normalising Flow

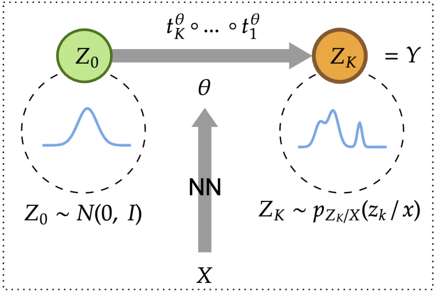
Conditional Normalising Flow (CNF) [Trippe and Turner, 2018, Winkler et al., 2019] – a series of invertable transformations, applied to base simple distribution.

Density (change of variables theorem):

$$f_\theta(y/x) = p_{Z_0}(z_0) \prod_{k=1}^{K} \left| \det \frac{dt_k}{dZ_{k-1}}(z_{k-1}/x) \right|^{-1}$$

Sampling:

$$\tilde{Y} = t_K \circ \ldots \circ t_1(\tilde{Z}_0) \quad \tilde{Z}_0 \sim N(0, I)$$



Parameters of transformations (radial, affine) are dependent on context X via neural network.

Overview of Methods – Mixture Density Network

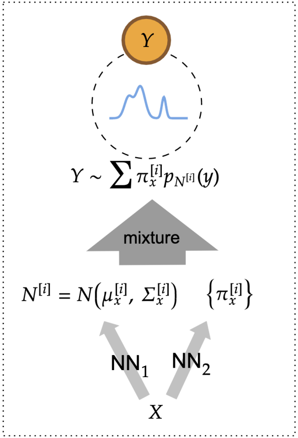
Mixture Density Network (MDN) [Bishop, 1994] – mixture of multivariate normal distributions (components) and categorical distribution.

Density:

$$f_\theta(y/x) = \sum_{i=1}^{C} \pi_x^{[i]} p_{N_x^{[i]}}(y)$$

Sampling:

$$\tilde{Y} \sim N(\mu_x^{[\tilde{c}]}, \Sigma_x^{[\tilde{c}]}) \quad \tilde{c} \sim \mathrm{Cat}(\pi_x^{[i]})$$



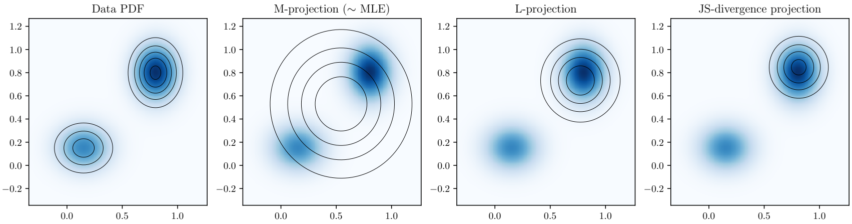
Mixture parameters are dependent on context X via neural networks.

Overview of Methods – Goodness-of-fit & Sampling

Maximum likelihood estimation (MLE) is equivalent to the minimization of KL-divergence between the data generating p and model's f_θ distributions (M-projection):

$$\arg\min_{\theta\in\Theta} \mathrm{KL}(p||f_\theta) \approx \arg\max_{\theta\in\Theta} \frac{1}{n}\sum_{i=1}^{n} \log f_\theta(y^{(i)}/x^{(i)})$$

Under misspecified model class it underestimates the real support of distribution $\rightarrow$ results in unrealistic sampling



Overview of Methods – Goodness-of-fit & Sampling

Why do we need tractable density $f_\theta(y/x)$ of estimated model for sampling? $\rightarrow$
It allows to effectively compute goodness-of-fit (GoF) and do model selection.

We utilised GoF metrics, evaluated on test subset:

- **Negative log-likelihood** – unbounded, good values could correspond to visually bad sample in high dimensions [Theis et al., 2015]
- **Hellinger distance** – bounded between 0 and 1, requires the knowledge of p
- **Kullback-Leibler divergence** – lower-bounded with 0, requires the knowledge of p
- **Jensen-Shannon divergence** – symmetrical, bounded between 0 and $\log 2$, requires the knowledge of p

Outline

Evaluation Benchmark – Aim & Dimensions

Aim of empirical study:

- check goodness-of-fit of estimators in different non-Gaussian scenarios
- evaluate, how goodness-of-fit contributes to feature importance validity
- study the influence of sensitive attributes

Evaluation Benchmark – Aim & Dimensions

Aim of empirical study:

- check goodness-of-fit of estimators in different non-Gaussian scenarios
- evaluate, how goodness-of-fit contributes to feature importance validity
- study the influence of sensitive attributes

Dimensions of evaluation:

- Synthetic / Semi-synthetic / Real datasets with known causal structure models (SCMs) or causal graphs (DAGs)
  - size of data-generating causal model (# of edges / # of nodes)
  - training subset size
- Density estimators (MDN, CNF, Conditional Gaussian distribution (CondGauss))
- Predictive models (Linear Regression, Random Forest, LightGBM Regressor) and risks (MSE, MAE)

Evaluation Benchmark – Benchmark Design

In principle, it is possible to evaluate FI of all triplets (target – feature of interest – context) $\rightarrow$ exponential number of evaluations & intractable GT values

Evaluation Benchmark – Benchmark Design

In principle, it is possible to evaluate FI of all triplets (target – feature of interest – context) $\rightarrow$ exponential number of evaluations & intractable GT values

Ground-truth values of FIs

- RFI can be approximately found for Linear Regression & MSE risk with multivariate Gaussian data (derivation in thesis)
- SAGE values are intractable even for simple models / risks [Van den Broeck et al., 2021].

Evaluation Benchmark – Benchmark Design

In principle, it is possible to evaluate FI of all triplets (target – feature of interest – context) $\rightarrow$ exponential number of evaluations & intractable GT values

Ground-truth values of FIs

- RFI can be approximately found for Linear Regression & MSE risk with multivariate Gaussian data (derivation in thesis)
- SAGE values are intractable even for simple models / risks [Van den Broeck et al., 2021].

Monte-Carlo estimate, by sampling from **true conditional distribution**?

- expressiveness versus tractability issue [Vergari et al., 2020] $\rightarrow$ either too simple causal SCM or intractable conditional distributions
- even approximate inference for SCMs – an open research question

Evaluation Benchmark – Benchmark Design

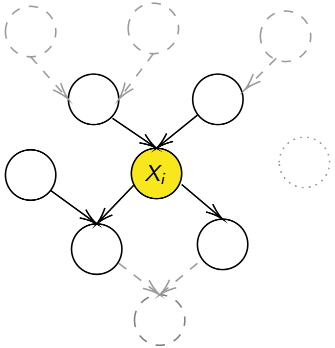
We propose to use feature selection concepts:

- *Strongly relevant features* – always conditionally dependent on target
- *Weakly relevant features* – can be independent of target, conditionally on some context
- *Irrelevant features* – always independent of target

Set of strongly relevant features = Markov Blanket of target

Markov blanket (MB) of node

Set of parents, children and parents of children of a node in causal DAG.



Target variable X_i (yellow), MB(X_i) – bold nodes, non-MB(X_i) - hatched/dotted nodes.

Evaluation Benchmark – Benchmark Design

Thus, we have a **quadratic number of evaluations**, depending on the causal DAG number of nodes.

For each possible target in causal graph we evaluate (1) RFIs and (2) SAGE of all training features:

- (1) RFIs of weakly-/irrelevant features, conditioned on strongly relevant, should be close to 0 (for correct predictor and correct sampler)
- (1) RFIs of strongly relevant features, conditioned on weakly-/irrelevant features, should be non zero and higher, than ones in the first case (and depend on the level of noise of structural assignments)
- (2) SAGE values should be zero for irrelevant features and non-zero for strongly relevant (for correct predictor and correct sampler)

Evaluation Benchmark – RFI Results / Synthetic datasets

Goodness-of-fit and conditioning size:

- variance of deep estimation increases with conditioning size
- deep estimators are always preferred for heavy tailed distributions
- no universally best estimator (no free lunch)
- sometimes – hard to notice superiority of estimator with NLL

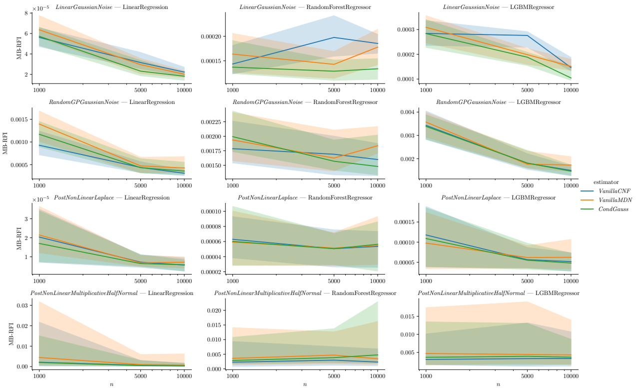
Goodness-of-fit and train size:

- Conditional Gauss can outperform deep estimators in low-data regimes on non-linear benchmarks

Evaluation Benchmark – RFI Results / Synthetic datasets

RFI values of weakly relevant features and **train size**:

- under correctly specified model (*LinearGaussianNoise* – LinearRegression) or for flexible enough LGBMRegressor RFI values indeed decrease.
- values for a limited model (RandomForestRegressor) stays relatively the same (see middle column)

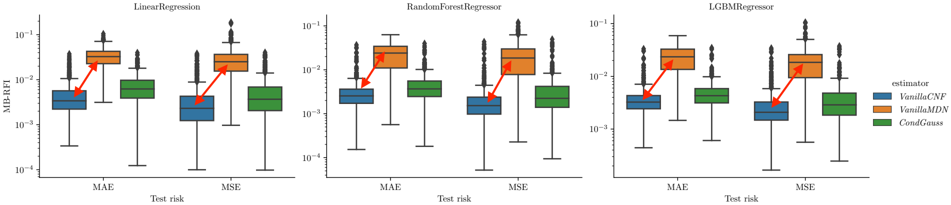


Median RFIs for MSE risk with respect to different train sizes. Columns – predictive models; rows – data-generating SCMs. Note, that y-axes are not shared across the figure.

Evaluation Benchmark – RFI Results / Semi-synthetic dataset

RFI values of weakly relevant features on SynTReN generator:

- substantial difference between estimated values for CNF and MDN (GoF ranking: CNF $>$ MDN $>$ Conditional Gaussian) $\rightarrow$ negative log-likelihood can be misleading with low amount of data

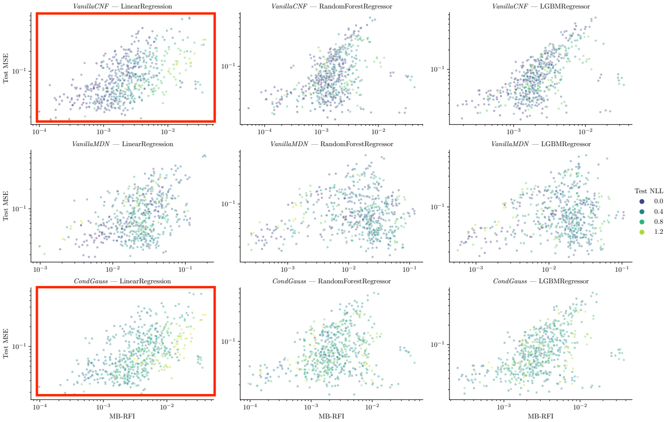


Box plots with RFIs for weakly relevant features on SynTReN datasets. x-axis – test risks (MAE, MSE), y-axis – RFI values. Note, that y-axes are log-scaled and not shared across the figure.

Evaluation Benchmark – RFI Results / Semi-synthetic dataset

Correlation between weakly relevant RFIs and test loss (SynTReN generator):

- high correlation between test risk and RFIs of weakly relevant features $\rightarrow$ lower predictive risk means lower RFI
- also, a correlation between values of test NLL and RFIs of weakly relevant features (for Linear Regression)

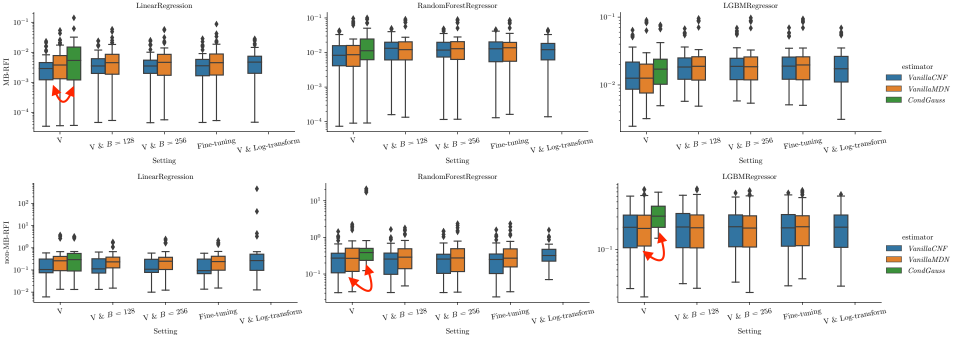


Scatter plot of RFI values of weakly relevant features and test MSE risk for SynTReN dataset. Rows – different density estimators, columns – predictive models.

Evaluation Benchmark – RFI Results / Real dataset

Deep estimators enhancements on Sachs-2005 (reduced batch size, fine-tuning, adding log-transformation to CNF):

- GoF: fine-tuning – the best improvment of CNFs, batch size of 128 – MDNs
- RFI values: enhancements don't substantially change estimates (difference only with Conditional Gaussian, which overestimated the values)

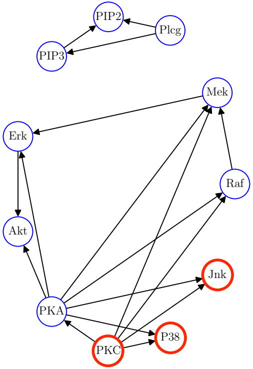


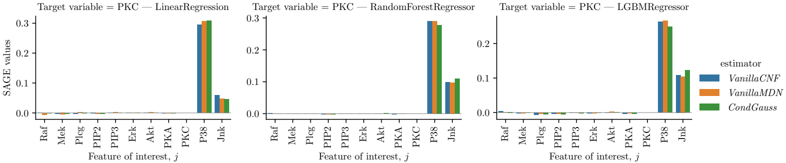
Box plots with RFIs for weakly (top) and strongly (bottom) features on Sachs-2005 datasets. x-axis – different density estimators, y-axis – RFI values. Note, that y-axes are log-scaled and not shared across the figure.

Evaluation Benchmark – SAGE Results / Real dataset

Main findings of SAGE estimation on Sachs-2005 dataset:

- deep estimators outperform Conditional Gaussian distribution
- estimated importances mirror the ground-truth connections of data-generating DAG



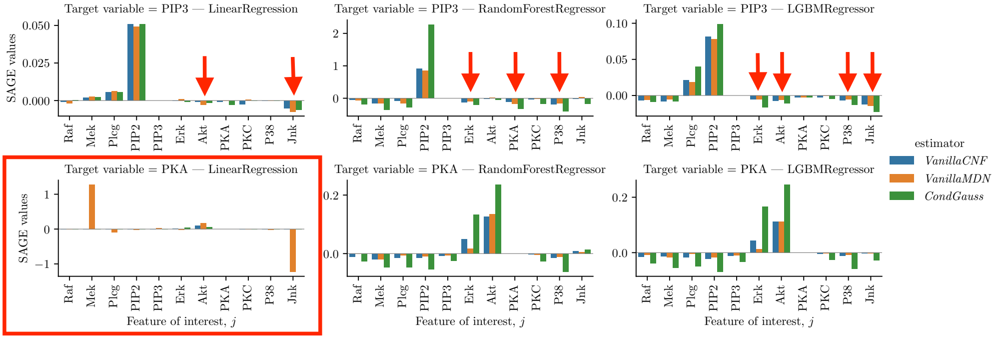


Estimated SAGE values of MAE risk for target '*PKC*'.

Evaluation Benchmark – SAGE Results / Real dataset

Main findings of SAGE estimation on Sachs-2005 dataset:

- SAGE values of weakly relevant features are sometimes below zero → support underestimation increases with dimensionality
- MDNs were less numerically stable and produced extreme values



Estimated SAGE values of MAE risk for targets '*PIP3*' (top row) and '*PKA*' (bottom row).

Evaluation Benchmark – Sensitive Attributes Use Case

Census Income dataset from UCI library [Dua and Graff, 2017]

- Prediction task: predict whether income exceeds $50K/year based on census data.
- Features: 8 Categorical + 4 Continuous.
- Sensitive attributes: '*Age*', '*Race*', '*Sex*'.
- Predictive model – LightGBM classifier.

Evaluation Benchmark – Sensitive Attributes Use Case

Census Income dataset from UCI library [Dua and Graff, 2017]

- Prediction task: predict whether income exceeds $50K/year based on census data.
- Features: 8 Categorical + 4 Continuous.
- Sensitive attributes: '*Age*', '*Race*', '*Sex*'.
- Predictive model – LightGBM classifier.

Aim – detect influence of sensitive features for two types of models:

- model with sensitive attributes
- 3 models, ignoring sensitive attributes

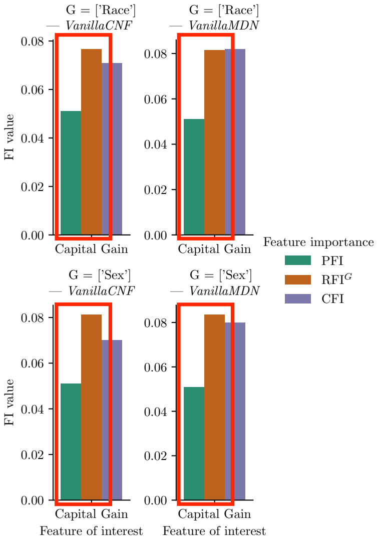
By comparing PFI and RFI (conditionally on sensitive features G), we can reason about direct / indirect influence of sensitive information for both models.

Evaluation Benchmark – Sensitive Attributes Use Case

Discovered issue: '*Capital Gain*' is a mixed-type variable $p(x = 0.0) > 0$:

- suspiciously high test log-likelihood
- CFI and RFI are higher, than PFI $\rightarrow$ empirical distribution substantially differs from estimated

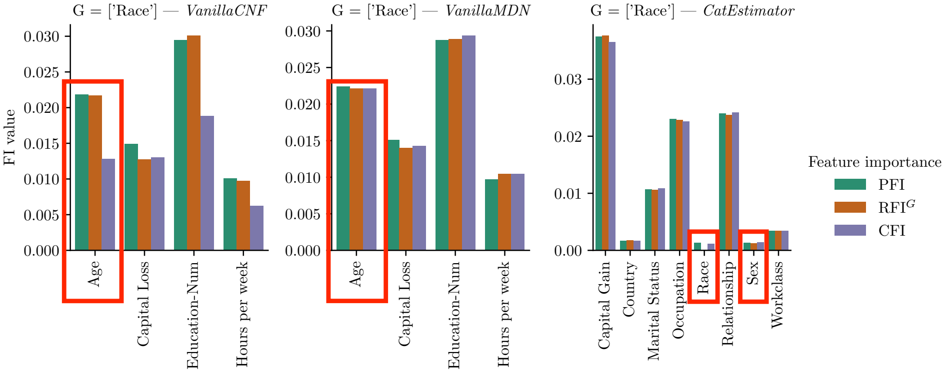
Ultimately, we used a 50-bins discretizer and treated this feature as categorical.



Evaluation Benchmark – Sensitive Attributes Use Case

Notable findings:

- PFIs of sensitive features were close to zero, when used as training features (max 2% for '*Age*').



Evaluation Benchmark – Sensitive Attributes Use Case

Other findings:

- after excluding sensitive features, test accuracy dropped maximally on 0.8%
- feature importances were almost the same between two types of classifiers
- negligible differences between PFI and RFI for the majority of features

Main conclusion: we do not observe any leakage of sensitive attributes via other features if we include or even exclude them from training.

Outline

Summary

- Deep density estimators should be preferred for heavy-tailed / multimodal / heteroscedastic distributions → they produce more realistic FI values
- CNFs are more numerically stable than MDNs (especially for heavy-tailed distributions)
- SAGE estimates could be wrongly negative, as the underestimation of real support increases with dimensionality
- One wants to use a sampler with the best GoF. But often, there is no need to spend too much computational power to fine-tune density estimators → estimated FIs are roughly similar
- Mixed-variable density estimation is an open issue, which causes incorrect FIs

Outline

References I

Aas, K., Jullum, M., and Løland, A. (2019).
Explaining individual predictions when features are dependent: More accurate approximations to shapley values.
arXiv preprint arXiv:1903.10464.

Bishop, C. M. (1994).
Mixture density networks.

Blundell, C., Cornebise, J., Kavukcuoglu, K., and Wierstra, D. (2015).
Weight uncertainty in neural network.
In *International Conference on Machine Learning*, pages 1613–1622. PMLR.

Covert, I., Lundberg, S., and Lee, S.-I. (2020).
Understanding global feature contributions with additive importance measures.

Dua, D. and Graff, C. (2017).
UCI machine learning repository.

Frye, C., de Mijolla, D., Begley, T., Cowton, L., Stanley, M., and Feige, I. (2020).
Shapley explainability on the data manifold.
arXiv preprint arXiv:2006.01272.

König, G., Molnar, C., Bischl, B., and Grosse-Wentrup, M. (2020).
Relative feature importance.
arXiv preprint arXiv:2007.08283.

Lachapelle, S., Brouillard, P., Deleu, T., and Lacoste-Julien, S. (2019).
Gradient-based neural dag learning.
arXiv preprint arXiv:1906.02226.

References II

Mase, M., Owen, A. B., and Seiler, B. (2019).
Explaining black box decisions by shapley cohort refinement.
arXiv preprint arXiv:1911.00467.

Mirza, M. and Osindero, S. (2014).
Conditional generative adversarial nets.
arXiv preprint arXiv:1411.1784.

Rothfuss, J., Ferreira, F., Boehm, S., Walther, S., Ulrich, M., Asfour, T., and Krause, A. (2019a).
Noise regularization for conditional density estimation.
arXiv preprint arXiv:1907.08982.

Rothfuss, J., Ferreira, F., Walther, S., and Ulrich, M. (2019b).
Conditional density estimation with neural networks: Best practices and benchmarks.
arXiv preprint arXiv:1903.00954.

Sohn, K., Lee, H., and Yan, X. (2015).
Learning structured output representation using deep conditional generative models.
Advances in neural information processing systems, 28:3483–3491.

Theis, L., Oord, A. v. d., and Bethge, M. (2015).
A note on the evaluation of generative models.
arXiv preprint arXiv:1511.01844.

Trippe, B. L. and Turner, R. E. (2018).
Conditional density estimation with bayesian normalising flows.
arXiv preprint arXiv:1802.04908.

References III

Van den Broeck, G., Lykov, A., Schleich, M., and Suciu, D. (2021).
On the tractability of shap explanations.

Vergari, A., Choi, Y., Peharz, R., and Van den Broeck, G. (2020).
Probabilistic circuits: Representations, inference, learning and applications.
In *Tutorial at the The 34th AAAI Conference on Artificial Intelligence.*

Winkler, C., Worrall, D., Hoogeboom, E., and Welling, M. (2019).
Learning likelihoods with conditional normalizing flows.
arXiv preprint arXiv:1912.00042.

Outline

Miscellaneous – Deep Conditional Density Estimators

Comparison of SOTA deep conditional density estimators

Parametric model	Tractable density	Exact Sampling	Tractable CDF	Tractable quantile function
Latent variable NNs (cVAE [Sohn et al., 2015], cGAN [Mirza and Osindero, 2014])	–	+	–	–
Bayesian NNs [Blundell et al., 2015]	–	+	–	–
Mixture Density Networks (MDNs) [Bishop, 1994]	+	+	+	–
Conditional Normalising Flow (CNFs) [Trippe and Turner, 2018, Winkler et al., 2019]	**+**	**+**	**+**	**+**

Additional advantages of MDNs and CNFs:

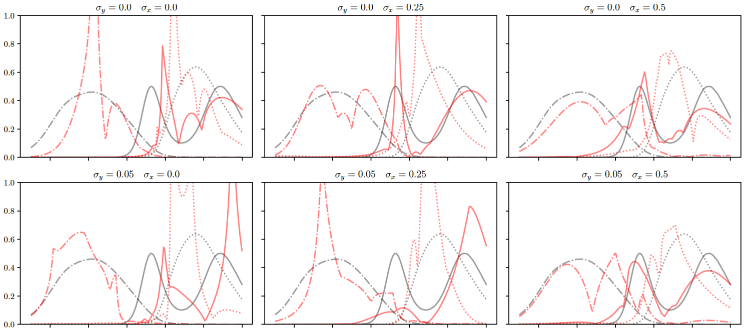
- few interpretable parameters, which control the complexity of distribution
- barely tuning needed due to noise regularisation [Rothfuss et al., 2019a]
- evaluated for tabular UCI benchmark datasets [Rothfuss et al., 2019a]

Miscellaneous – Noise Regularisation

In the context of CDE: it is unclear what kind of inductive bias to choose?

Possible solutions:

- [Trippe and Turner, 2018] – putting priors on latent features and NN weights + variational inference → need to know a reasonable prior
- [Rothfuss et al., 2019b] – **noise regularisation** (adding Gaussian noise to dependent and context variables)



Miscellaneous – Datasets

Generators / datasets, used for causal structure learning [Lachapelle et al., 2019] also fit to the needs of RFI/SAGE evaluation:

1. **4 synthetic SCM generators**: linear with additive Gaussian noise, non-linear with additive Gaussian noise, post non-linear with additive Laplace noise and multiplicative with Half-Normal noise.
2. **SynTReN generator** produces simulated gene expression data, that approximates experimental data.
3. **Sachs-2005** – real dataset, measures the expression level of different proteins and phospholipids in human cells.

Miscellaneous – Synthetic SCM dataset generators

1. *LinearGaussianNoise*.

$$X_j/\mathrm{Pa}_{X_j} \sim w_j^T \mathrm{Pa}_{X_j} + 0.2N(0,\sigma_j^2) \quad \sigma_j^2 \sim U[1,2], \quad w_{ij} \sim U[0,1]$$

2. *RandomGPGaussianNoise*.

$$X_j/\mathrm{Pa}_{X_j} \sim f_j(\mathrm{Pa}_{X_j}) + 0.2N(0,\sigma_j^2); \quad f_j \sim \mathcal{GP}(0, k(X, X')) \quad \sigma_j^2 \sim U[1,2]$$

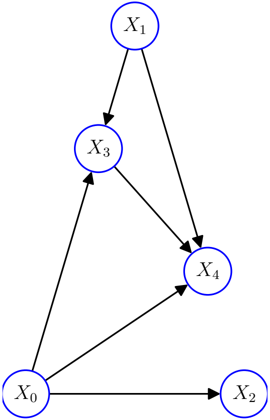
3. *PostNonLinearLaplace*.

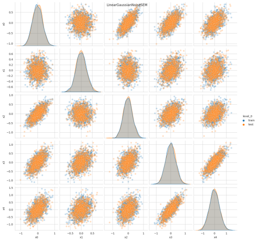
$$X_j/\mathrm{Pa}_{X_j} \sim \sigma(f_j(\mathrm{Pa}_{X_j}) + \mathrm{Laplace}(0, l_j)) \quad f_j \sim \mathcal{GP}(0, k(X, X'))$$

4. *PostNonLinearMultiplicativeHalfNormal*.

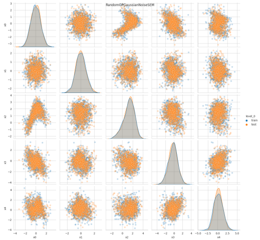
$$X_j/\mathrm{Pa}_{X_j} \sim \exp\left(\log\left(\sum \mathrm{Pa}_{X_j}\right) + |N(0,\sigma_j^2)|\right) \quad \sigma_j^2 \sim U[0,1]$$

Miscellaneous – Synthetic SCM dataset generators

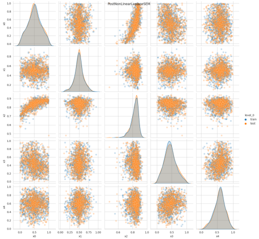




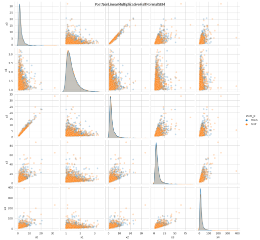
LinearGaussianNoise



RandomGPGaussianNoise



PostNonLinearLaplace



PostNonLinearMultiplicativeHalfNormal