Causal Sensitivity Analysis: f-sensitivity through entropic value at risk computation

by

Matej Havelka

to obtain the degree of Master of Science
at the Delft University of Technology,
to be defended publicly on Wednesday June 26th, 2024 at 11:00 AM.

Student number:	5005914	
Project duration:	November 13th, 2023 – June 26th, 2024	
Thesis committee:	Prof. dr. ir. M. Reinders,	TU Delft, supervisor
	Dr. ir. J. Krijthe,	TU Delft, daily supervisor
	Dr. F. A. Oliehoek,	TU Delft, committee member

An electronic version of this thesis is available at http://repository.tudelft.nl/.

Causal Sensitivity Analysis: f-sensitivity through entropic value at risk computation

Matej Havelka

Abstract

The field of causal inference provides a variety of estimators that can be used to find the effect of a treatment on an outcome based on observational data. However, many of these estimators require the unconfoundedness assumption, stating that all relevant confounders are observed within the data. This assumption is quite strict and many real-life problems would violate it, due to confounders being too expensive, immoral, or abstract to observe. To weaken this assumption, causal sensitivity analysis attempts to quantify the level of hidden confounding and use it to create a possible bound on the causal effect. This study compares a well-established Marginal Sensitivity Model (MSM) with a newly proposed f-sensitivity model. Given f-sensitivity is relatively new, the current approaches for computation can be inefficient or non-deterministic. A new method of computing the f-sensitivity bound is proposed, which can lead to closed-form solutions in some estimator-specific cases. The differences between the MSM and f-sensitivity models are outlined and illustrated with examples, from which a set of guiding questions is created for researchers to decide between the two sensitivity models. All of the code required to reproduce this study is located in this GitHub repository.

1 Introduction

A lot of scientific fields rely on randomized controlled trials to gather data about the effect of a feature on an outcome. For example, measuring the effect a drug or some other treatment can have on the well-being of a patient. However, these randomized trials can be quite costly, sometimes unethical, or outright impossible to perform. This leaves the research community only with observational data, which requires a different approach to measure the effect, as non-causal associations might skew the results. The field of causal inference provides numerous methods for researchers to measure the effect of treatment on outcomes by adjusting for non-causal associations caused by other features, called confounders. For example, one might want to measure the effectiveness of a vaccine by comparing the average outcome on treated patients and the average non-treated (control) outcomes. This, however, could provide incorrect results in situations where younger people are less likely to get vaccinated, but tend to be more resilient to diseases. This would result in a disproportionate amount of younger people ending up in the control group, leading to a biased control outcome and effect measurement.

Within the field of causal inference, an important goal is to measure the average treatment effect (ATE), the expected difference in outcomes between treated and control individuals. There exist many estimators that provide such measurement, but they rely on the unconfoundedness assumption, which assumes that the provided data contains all the confounders that are required to adjust for bias in the ATE. This is rarely true in real-life, since some important variables might be too expensive, sensitive, or abstract to record (e.g., social status). To make causal inference estimators more general, Rosenbaum and Rubin (1983) created Sensitivity Analysis that changes the unconfoundedness assumption into a quantifiable metric, allowing estimators to be evaluated under varying degrees of hidden confounding.

The most well-known sensitivity model is called the Marginal Sensitivity Model (MSM) (Tan, 2006). This sensitivity model quantifies hidden confounding with a bound on the ratio between the observed distribution of treatment in the dataset and the true treatment distribution including a hidden confounder U. However, the MSM is sensitive to the distribution of the hidden

confounders and uses a similarity metric that produces counter-intuitive behaviour. Recently, a new sensitivity model was proposed by Jin et al. (2022) called f-sensitivity that creates an assumption about the bound on an f-divergence metric. F-sensitivity fixes the aforementioned issues with the MSM to a certain extent, but computing f-sensitivity can be tedious, as the algorithm proposed by the original authors is only an estimate of the bound, rather than a direct computation. In addition, the bound that f-sensitivity assumes can be harder to interpret and thus more difficult to use. Given these issues, the goal of this study is to create a comprehensive overview of the differences between the MSM and the new f-sensitivity, as well as propose a new approach to compute the f-sensitivity bound that comes closer to an efficient, closed-form solution.

Further background on the field of sensitivity analysis and relevant information for this study can be found in Section 2. Section 3 dives into the mathematical specifications of the MSM, f-sensitivity and sensitivity analysis as a whole, as well as proposing a new method of computing the f-sensitivity bounds using entropic value at risk. In Section 4 the MSM and f-sensitivity are compared and specific differences are highlighted using an experiment. Section 5 takes a look into the wider implications of this work on the field of causal inference, as well as possible future trajectories for research. Furthermore, Appendix A provides proofs for theorems found within the study, and Appendix B contains additional experiments that were tested, and provide further evidence for the claims made in the main text.

2 Relevant works

The accepted start of the field of causal inference and sensitivity analysis tends to be work done by Fisher (1958) and Cornfield et al. (1959), when the authors suggested that smoking might not be the real cause of lung cancer, but rather just a correlation caused by some DNA strain that makes people more likely to smoke, as well as likelier to get lung cancer. Rosenbaum and Rubin (1983) studied this idea and described an analytical way to measure the effect unobserved confounding has on the result of an estimator, further generalizing it in Rosenbaum (1987). However, this sensitivity model was based on the ratio of the true propensity (probability to be treated) distribution between two individuals, creating problems as all the measurements are relative to other individuals. To fix this, Tan (2006) introduced the now-called Marginal Sensitivity Model (MSM), which uses the same ratio, but between the true propensity score of an individual and the observed propensity of that individual.

The MSM became the standard when it comes to sensitivity analysis, and many extensions were made to generalize its applicability. The Generalized MSM proposed by Frauen et al. (2023) extends the MSM to continuous treatments and proposes extensions to different types of data, like longitudinal data. Zhang and Zhao (2024) point out that MSM assumes that the logit difference between the observed and full data propensity scores is uniformly bounded, and changes this assumption using L^2 analysis making the model more general. This model is called the L^2-*sensitivity*. Another extension created by Marmarelis et al. (2023) drops the assumption of discrete treatments and develops *Delta Sensitivity* that creates bounds with continuous treatments. A different model with continuous treatments is the *Curvature Sensitivity Model* developed by Melnychuk et al. (2024) that argues that regular MSM is reliant on point estimates of the counterfactuals, which makes the bounds unnatural. So they instead try to fit a curve on the counterfactual outcomes, creating a more natural and smooth bound. All of these extensions have the goal to apply the MSM on a wider selection of problems, but none of them change the initial assumption of the MSM.

As the MSM was used for a multitude of studies, some issues were brought to attention about the core assumptions of the MSM. This led to the development of different models with different assumptions. The X-mixture model formulated in Bonvini and Kennedy (2021) tries to find the worst-case scenario assuming that only some portion P of the population is affected by an unobserved confounder. The f-sensitivity model from Jin et al. (2022) uses the odds-ratio measure used by MSM and measures how far it is from being constant. This study takes a deeper look into f-sensitivity and how it compares to the widely used MSM.

While sensitivity analysis provides researchers with tools to test their results against possible unobserved confounding, there are other methods to adjust for hidden confounding directly during learning. CEVEA developed by Louizos et al. (2017) uses a variational au-

toencoder to encode the data that should embed some of the hidden confounding into the latent representation of the data. Work by Karlsson and Krijthe (2023) proposes a solution where a hidden confounder may be discovered by observing datasets of the same problem but from different locations. This allows researchers to find similarities between the data and label these similarities as possible hidden confounders. While these methods show promising results, this study focuses on methods that evaluate models on a quantifiable measure of hidden confounding, rather than trying to adjust for hidden confounding.

Some studies also extended the existing sensitivity analysis into different problem setups and fields. Recently, the field of causal reinforcement learning has been gaining traction. For an overview, see the survey by Zeng et al. (2023). Work by Kallus and Zhou (2018) extends sensitivity analysis into dynamic treatment regimes defined by Chakraborty and Moodie (2013). This is followed by Kallus and Zhou (2020) which adjusts the MSM for the reinforcement learning setting with global confounders. Kausik et al. (2023) introduces the model-based method that allows reinforcement learning agents to directly train in an environment based on the worst-case scenario defined by the MSM. These settings are interesting to study, but the f-sensitivity is a recent contribution and needs to be studied more before applied on such settings.

Lastly, work by Dorn et al. (2022) shows how the MSM can be expressed via conditional value at risk (CVaR). This paper proposes a similar approach, by showing how to relate f-sensitivity to entropic value at risk (EVaR) (Ahmadi Javid, 2012). For further information about value at risk, Ahmadi Javid (2012) provides a well structured introduction.

3 Methodology

Before diving into the specifications of the MSM and f-sensitivity, let us define some concepts from causal inference. The goal of causal inference within this study is to estimate the average treatment effect (ATE), defined as $E[Y|do(T = 1)] - E[Y|do(T = 0)]$. Other possible metrics to estimate are the Conditional ATE (CATE), defined as $E[Y|do(T = 1), X] - E[Y|do(T = 0), X]$, intuitively computing the ATE for a specific subgroup X. These definitions use the do-notation, defined by Pearl (2009), that indicates how the outcomes, also called counterfactuals, would behave if the individuals were to receive a certain treatment, possibly counter to the fact. The goal of sensitivity analysis is to create bounds on these metrics based on possible hidden confounding. Throughout this study the concern is about estimating the ATE. Additionally, all $x \in X$ are assumed to be discrete.

To measure the ATE, causal inference relies on three assumptions. The first one is the aforementioned unconfoundedness assumption, stating that the observed confounders are all the confounders that exist between this treatment and outcome. Using the potential outcome notation, which is explained in great detail in Zeng and Wang (2022), this assumption is denoted as $(Y(0), Y(1)) \perp T|X$, showing how potential outcome distributions $Y(0)$ and $Y(1)$ are independent of the treatment by adjusting for confounders X, meaning there can be no other confounders outside of X. The overlap assumption bounds the propensity score (probability of being treated), denoted as $e(x) = P(T = 1|X = x)$, such that $0 < e(x) < 1 \quad \forall x \in X$. This means that for every subpopulation $x \in X$ (with a non-zero probability) there are instances for both treated and control samples. The last assumption is the consistency assumption, stating that given there would be an intervention in the way treatments are assigned, the outcome distributions would remain the same.

The odds-ratio, defined as $OR(x, U) = \frac{e(x)/(1-e(x))}{e(x,U)/(1-e(x,U))}$, measures how far the propensity of subpopulation $x \in X$ deviates from the odds of treatment, if a new confounder U is introduced. Both $OR(x, U)$ and $OR(x, U)^{-1}$ tend to be used interchangeably, as $OR(x, U)^{-1}$ measures the ratio from the control perspective.

The purpose of sensitivity analysis is to quantify how the estimated ATE, from some causal estimator changes when some hidden confounding is introduced. It is not evaluating the estimator on how well it captures the causal link from the observed data, nor is it determining whether there are any hidden confounders in the data. To introduce some possible hidden confounding, sensitivity models use the observed dataset as a reference point, trying to create a set of possible distributions, resulting in the observed dataset. Secondly, it requires the causal estimator that can generate counterfactuals, which are used to measure the ATE in the dataset.

The following subsections 3.1 and 3.2 provide in-depth descriptions for the MSM and f-sensitivity models, respectively. Each model is described using a constraint programming formulation, as well as a custom algorithm to compute the bounds.

3.1 Marginal Sensitivity Model

The main idea behind MSM is to create the worst-case scenario for a model based on the data. This is done by defining a space of possible true distributions based on the odds-ratio, effectively creating a distance metric computed by measuring the highest and lowest ratios of propensity scores. The unconfoundedness assumption is replaced with an assumption that bounds the odds-ratio, defined as $\Gamma^{-1} \leqslant OR(x,U) \leqslant \Gamma \qquad \forall x \in X$, where Γ is the assumed bound. This assumption is said to hold for almost all $x \in X$, as with continuous features the ratio would be unbounded.

This leads to a set of distributions around the observed distribution $\mathrm{Q} = \{\mathcal{Q} : 1/\Gamma \leqslant OR_{\mathcal{Q}}(x,U) \leqslant \Gamma$ for all $x \in X\}$, from which the worst possible one needs to be selected. This selection can be done in many different ways. In this study, two main approaches are discussed: first the constraint programming computation, second the approximation using conditional value at risk. Before these implementations are discussed, it is worth pointing out what the inputs into an MSM are. The obvious input is the observed data, as those are required to form a range for the true distribution. Another input is some causal estimator that can predict an outcome based on confounders X and treatment T. There are countless solutions for the MSM that are estimator-specific, like Tan (2006), which provides a solution for the Inverse Probability Weighting (IPW) estimator. In the following implementations, the main assumption about the provided estimators is that they can predict counterfactuals for the data, such that each individual has predictions for both treated and control outcomes assigned.

Implementation. To create a bound on the ATE based on the MSM assumption, an algorithm is needed that creates a new distribution Q that is the worst-case true distribution found within the allowed range of the bound. Assuming that all $x \in X$ and $y \in Y$ are discrete, a new variable $\lambda(x,y)$ is introduced for each pair $(x,y) \in X \times Y$, symbolizing the value of the odds-ratio for samples containing x and y. Mathematically, this means that $e(x,y)/(1-e(x,y)) = \lambda(x,y)e(x)/(1-e(x))$. To work with this variable $\lambda(x,y)$, some constraints have to hold. First, it is important to ensure that in expectation, the observed dataset would be produced, or mathematically $E_X[\lambda(x,y)|y] = 1 \quad \forall x \in X$. This rescales the observed distribution while ensuring that the expectation of the new distribution remains the same as in the observed one. Now, to ensure that this new distribution falls within the MSM assumption, it is required to that $\Gamma^{-1} \leqslant \lambda(x,y) \leqslant \Gamma \quad \forall (x,y) \in X \times Y$ holds, as this bounds the possible new scale based on the assumed Γ. These two constraints allow finding the worst-case distribution. However, there is one final step which is to actually extract the newly observed ATE. To compute the ATE, there are 4 values needed, the highest and lowest expected outcome of the treated and control, denoted as μ_1^+ for highest expected outcome of the treated, μ_0^- as the lowest possible expected outcome for the control. With these values the bounds of the ATE can be computed as $\mathrm{ATE}^+ = \mu_1^+ - \mu_0^-$ and $\mathrm{ATE}^- = \mu_1^- - \mu_0^+$. These constraints can be put together into a constraints programming solver with a setup defined as follows:

$$
\begin{aligned}
\max_{\lambda} \quad & \sum_{x,y} yP(Y=y|X=x,T)e(x)\lambda(x,y)P(X=x) \\
\text{s.t.} \quad & \sum_{y} \lambda(x,y) = 1 && \forall x \in X \\
& \Gamma^{-1} \leqslant \lambda(x,y) \leqslant \Gamma && \forall x \in X, y \in Y
\end{aligned}
$$

While this constraint programming implementation works, the time it takes to solve grows with the number of (x,y) input pairs. This means that whatever the solver used to solve the formulation, it will have $|X| * |Y|$ variables, which might be computationally heavy depending on the data. This is why it is worth studying other possible approaches to compute, or at least approximate the bounds given by the MSM. This is exactly what Dorn et al. (2022) set out to do. The authors show how one can compute the MSM bounds using Conditional Value at Risk (CVaR), shown in Equation 1. CVaR is a risk measure that averages out the worst $1-\alpha$ percentile of the distribution. It is up to the user to define whether the worst part is going towards positive or negative infinity. Because CVaR is interested in percentiles, Dorn et al. (2022) claim it can

also be expressed using a quantile regressor $\hat{Q}_\tau(x,t)$, where the quantile $\tau = \frac{\Gamma}{1+\Gamma}$. With further mathematical properties Dorn et al. (2022) show that one can express the MSM bound directly based on observed data and quantile regressors, defined in Equation 2.

$$\mu_t^+(x) = \frac{1}{N}\sum_{i\in N}\Gamma^{-1}Y_i + (1-\Gamma^{-1})\text{CVaR}_\tau(x,t) \quad (1)$$

$$\begin{aligned}\mu_t^+ = \frac{1}{N}\sum_{i\in N}&\Gamma^{-1}Y_i + (1-\Gamma^{-1})[\hat{Q}_\tau(X_i,T_i) \\ &+\frac{1}{1-\tau}\{Y_i - \hat{Q}_\tau(X_i,T_i)\}]\end{aligned} \quad (2)$$

The efficiency of the CVaR approach depends on the type of the quantile regressor used. The publicly available implementations of quantile regression tend to use constraint programming. Scikit's implementation[1] also assumes linearity, which might damage the performance.

Issues. Using the odds-ratio as a similarity measure comes with certain issues. With continuous distributions the odds-ratio becomes unbounded (Jin et al., 2022), meaning that in theory there is no precise bound for the ratio. Additionally, odds-ratio might not be representative of the entire curve, as it only looks at the upper and lower bounds of the curve. This is visualized in Figure 1 which is taken from Jin et al. (2022) (Figure 1). This figure shows that similar curves might have different bounds, and thus might not be close in terms of MSM distance metric, while other curves might be quite different but have the same bounds.

A different issue is that the MSM does not take into account the distribution over the values of the hidden confounder. Because the constraint is supposed to hold for almost all $x \in X$, there might be a hidden confounder $U = 1$ that makes the odds-ratio increase substantially, even if $P(U = 1)$ would be low to almost zero. These issues are further discussed in Section 4.

3.2 F-sensitivity

F-sensitivity was built with the goal to fix the problem of unbounded odds-ratio. The idea is to switch from measuring similarity by bounding the odds-ratio and instead bound it with an f-divergence, which is a group of distance metrics between distributions. The intuition behind this is that instead of computing the bounds of the odds-ratio, it checks how far it is from being constant. For this study, instead of using the general form of f-sensitivity, the most common form using the Kullback-Leibler divergence with $f(t) = t\ln t$ is used.

Equation 3 shows the f-sensitivity assumption that bounds the divergence of the odds-ratio, in regards to the unobserved confounder. There are two assumptions, as one of them applies to treated group and the other to control. In this study, similarly as Jin et al. (2022), all equations are defined with the treated (first) definition, but all claims are applicable to both. This solves the problem with unbounded odds-ratio measure, as the integral scales everything by the probability distribution of U. It also utilizes a well-established similarity metric between distributions, fixing the unintended behaviours described by Figure 1.

$$\begin{aligned}\int f(OR(x,U))d\mathbf{P}_{U|X=x,T=1} \leqslant \rho \\ \int f(OR(x,U)^{-1})d\mathbf{P}_{U|X=x,T=0} \leqslant \rho\end{aligned} \quad (3)$$

By extending the assumption further, the work by Jin et al. (2022) introduces the set of possible distributions as $\mathrm{Q} = \{\mathcal{Q} : \frac{d\mathcal{Q}_X}{d\mathcal{P}_X}(x) = r(x), D_f(\mathcal{Q}_{Y|X}||\mathcal{P}_{Y|X}) \leqslant \rho\}$, where Q represents a possible distribution, $\mathcal{P}$ the observed distribution, $\frac{d\mathcal{Q}_X}{d\mathcal{P}_X}(x) = r(x)$ bounds the change of the possible distribution based on function $r(x)$ that is described later, and $D_f(\mathcal{Q}_{Y|X}||\mathcal{P}_{Y|X}) \leqslant \rho$ ensures that the f-sensitivity assumption from Equation 3 is satisfied. Same as with the MSM, one can either compute the constraints programming formulation, or try to use a different way to compute the f-sensitivity bound.

Implementation. The work that introduces f-sensitivity also provides both constraint programming and an approximation algorithm for computing the bound. The constraints programming formulation, defined in Equation 4, provides an exact solution based on rescaling the observed distribution, just as in the MSM formulation. In this formulation, however, the expected scale can shift a little based on the $r_1(x) =$

[1] https://scikit-learn.org/stable/auto$_e$*xamples/linear*$_m$*odel/plot*$_q$*uantile*$_r$*egression.html*

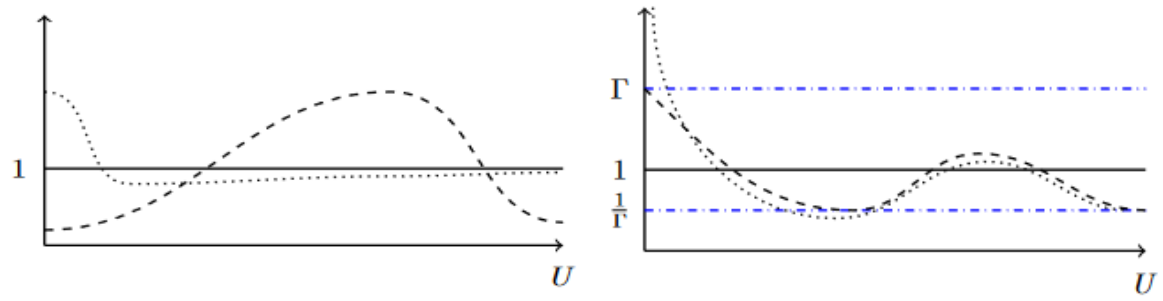

Figure 1: Left: examples of OR(x, U) that are quite different but have similar upper bounds. Right: examples of OR(x, U) that are similar but have drastically different upper bounds.

$\frac{(1-e(X))p_t}{e(X)(1-p_t)} = \frac{1}{r_0(x)}$ function that is observable from the data. This is because the f-sensitivity assumption only targets the posterior distribution $Y|X,T$ rather than the joint distribution Y,X, allowing for a shift in the distribution of X.

$$\begin{aligned} \max_{\lambda} & \sum_{x,y} yP(y|X=x,T)e(X=x)\lambda(x,y)P(X=x) \\ \text{s.t.} & \sum_{y} \lambda(x,y)P(y|X=x,T) = r(x) \quad \forall x \in X \\ & \sum_{y} f(\frac{\lambda(x,y)}{r(x)})P(y|X=x,T) \leqslant \rho \quad \forall x \in X \end{aligned} \tag{4}$$

While this formulation provides a neat way of explaining the point of f-sensitivity, same as the MSM formulation, its time complexity grows in terms of $|X|$ and $|Y|$. Additionally, because of the non-linearity of f, it requires a non-linear solver to obtain a solution. That is why the authors also provide an algorithm to approximate the f-sensitivity bound based on the lagrangian dual form. This algorithm is based on multitude approximations, and thus it might not always give the tightest bound. The algorithm can be found in Jin et al. (2022) (Algorithm 1). This algorithm is only an approximation of the bounds, and uses estimators to approximate nuisance parameters defined by the dual representation of the constraint programming formulation. It requires 3 estimators: one regular regression for propensity score estimation, one empirical risk estimator (ERM) to find the nuisance parameters (α, η) that minimize the lagrangian, and another regressor trying to predict the value of the lagrangian for an $x \in X$ without access to the outcome $y \in Y$. This means that the sharpness of the bound depends on the amount of data and how well-specified the used regression models are. It also means that the found bound is non-deterministic.

This study introduces a new way of computing the f-sensitivity bound. Starting from the set of possible distributions $\mathrm{Q} = \{\mathcal{Q} : \frac{d\mathcal{Q}_X}{d\mathcal{P}_X}(x) = r(x), D_{KL}(\mathcal{Q}_{Y|X}||\mathcal{P}_{Y|X}) \leqslant \rho\}$, using the relative entropy property of KL-divergence, set Q can be rewritten into $\mathrm{Q} = \{\mathcal{Q} : \frac{d\mathcal{Q}_X}{d\mathcal{P}_X}(x) = r(x), D_{KL}(\mathcal{Q}_{Y|X,T=0}||\mathcal{P}_{Y|X,T=0}) \leqslant \rho, D_{KL}(\mathcal{Q}_{Y|X,T=1}||\mathcal{P}_{Y|X,T=1}) \leqslant \rho\}$. This reformulation allows the application of the dual form of entropic value at risk (EVaR), leading to Theorem 1.

Theorem 1. *Bounds of f-sensitivity can be computed through the dual representation of the entropic value at risk (EVaR) resulting in*

$$\mu_t^+(x) = r_t(x) \inf_{z>0} \frac{1}{z} \ln \frac{M_{Y|X=x,T=t}(z)}{e^{-\rho}}$$
$$\mu_t^-(x) = r_t(x) \sup_{z<0} \frac{1}{z} \ln \frac{M_{Y|X=x,T=t}(z)}{e^{-\rho}}$$

Where $M_{Y|X=x,T=t}(z)$ is the moment-generating function of the $Y|X=x,T=t$ distribution, and ρ is the assumed sensitivity bound. $r_t(x)$ is defined as $\frac{(1-P(T=t|X=x))P(T=t)}{P(T=t|X=x)(1-P(T=t))}$.

Based on Theorem 1, whose proof can be found in Appendix A.1, finding the f-sensitivity bound becomes an issue of solving EVaR. The simplest approach would be to apply the definition of the moment-generating

function on the observed data combined with counterfactuals generated by an estimator that is being evaluated. This leads to to the moment generating function defined as $M_{Y|X=x,T=t}(z) = \frac{1}{N}\sum_{i=0}^{N} e^{zY_i}$ where N is the number of samples for which $X = x, T = t$ holds. Under this, moment generating function μ_t^+ becomes $r_t(x)\inf_{z>0}\frac{1}{z}\ln\frac{\sum_{i=0}^{N} e^{zY_i}}{Ne^{-\rho}}$, which can be solved using gradient descent, or other optimizing techniques.

Theorem 1 provides a solution to the bound computation, requiring the minimization of a function based on a single parameter z. While this does not necessarily make finding the bound more efficient than using the constraint programming solution, it provides a way to define a closed-form solution if an estimator is assumed. The following example shows how one can use this EVaR approach to compute the f-sensitivity bound.

Example 1. To ensure that Theorem 1 is correct and illustrate its use, a comparison between the results of the constraint programming and the EVaR approach is provided. Data is generated as follows:

$$\begin{aligned} U &\sim Bern(0.25) \\ X &\sim Bern(0.45) \\ T &\sim 0.5 + 0.25X - 0.3U \\ Y &\sim X + U + 2T - N(1, 0.1) \end{aligned} \tag{5}$$

This setting is meant to represent a general case, with hidden confounding U, confounder X, binary treatment T and outcome Y. Some important aspects are the total probability of being treated $P(T = 1) = 0.5P(X = 0, U = 0) + 0.2P(X = 0, U = 1) + 0.75P(X = 1, U = 0) + 0.45P(X = 1, U = 1) = 0.58$. A dataset of 30000 samples is generated to ensure that the results are not swayed by the lack of data.

With this setup, a Random Forest with its default parameters from the causalml[2] Python package is trained to generate the counterfactuals. There is no specific reasoning for using the random forest regressor, any estimator could have been used and the result would only change depending on the ATE predicted by the estimator. To ensure discreteness, each outcome was rounded to the first decimal number. With this newly generated dataset, the constraint programming and EVaR approaches are run on the same range of ρ. In this setup, the metric of interest is the ATE and the true ATE is equal to 2. However, because not all of the confounding is observed, the observable ATE shifts to $\text{ATE}_{obs} = 2 + E[U|do(T = 1)] - E[U|do(T = 0)] \approx 2 + 0.1347 - 0.4092 \approx 1.73$. The resulting plotted bounds of the ATE can be found in Figure 2.

As can be seen, both computation methods yield the same results. The predicted result does not seem to match with the true ATE, which is to be expected with a hidden confounder. The predicted ATE seem to match the theoretical observed ATE, given the outcomes were rounded to 1 decimal digit. The EVaR approach matches the solution from the constraint programming formulation, showing that the EVaR approach works. Further experiments to show the results of these 3 approaches can be found in Appendix B. Additionally, these results were also compared to the authors algorithm, however, the results obtained seem to be unstable. Due to the lack of access to the original implementation, it was implemented from scratch and it cannot be assured that these results are not the fault of the way it was implemented. These results can be found in Appendix B.2.

Now that it has been shown how to find a bound for the general case, let us introduce a closed-form solution that assumes that, to estimate the $Y|X, T$ distribution, an estimator fits a single normal distribution with learnable μ and σ. To find the closed-form solution, first a moment generating function needs to be defined. The normal distribution has a closed-form solution to its moment generating function given by $M_z(N(\mu, \sigma)) = e^{z\mu + \frac{1}{2}z^2\sigma^2}$. This leads to Theorem 2, which is expanded in Appendix A.2 with a proof extending it to a sum of Gaussians rather than a single normal distribution.

Theorem 2. *Assuming that the outcome follows a normal distribution $N(\mu, \sigma)$, the bounds of f-sensitivity can be expressed as:*

$$\mu_t^+(x) = r_t(x)[\sqrt{\rho\frac{\sigma^2}{2}} + \mu + \frac{\sqrt{2\rho}}{2}]$$
$$\mu_t^-(x) = r_t(x)[-\sqrt{\rho\frac{\sigma^2}{2}} + \mu - \frac{\sqrt{2\rho}}{2}]$$

Where the square root always yields the non-negative part of the possible solutions.

Theorem 2 provides a closed-form solution to the bounds of the f-sensitivity, since the exact parameters

[2] https://causalml.readthedocs.io/en/latest/about.html

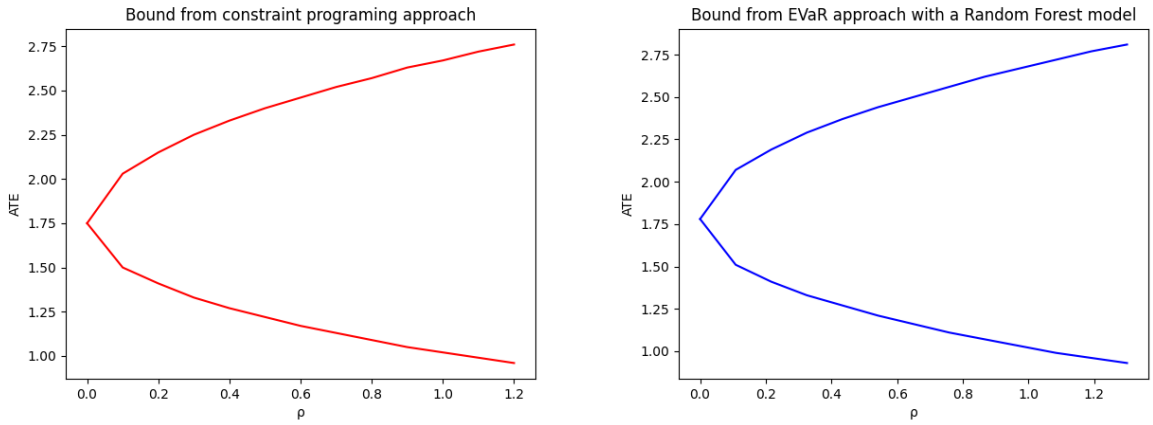

Figure 2: Bounds on the ATE computed by the two approaches for f-sensitivity. On the left is the constraint programming approach and on the right is the EVaR approach. Both approaches result in the same bound.

of the normal distribution are known. A sanity check for whether this holds is provided in Example 2.

Example 2. To show that Theorem 2 holds, a sanity check similar to Example 1 is provided. The experimental setup remains the same, following the distributions in Equation 5. The exact same dataset is used, but this time a Gaussian Mixture model[3] is used as the estimator with $k = 1$ to assume a single normal distribution. This estimator is used to approximate the outcome Y. A single normal distribution is fit for each possible X, T pair. The visualization of the results can be found in Figure 3, using both EVaR and the closed-form approach. The results slightly differ, as the normal distribution is unbounded, making its closed form also unbounded, while the EVaR approach will always stop at the maximum and minimum values observed in the data.

The results show that using the normal distribution does predict the ATE close to the observable ATE, but creates a larger spread as ρ increases, as it is not bounded by the observed distribution. Additionally, this allowed the computation to be done almost instantly, as the bounds follow the closed-form.

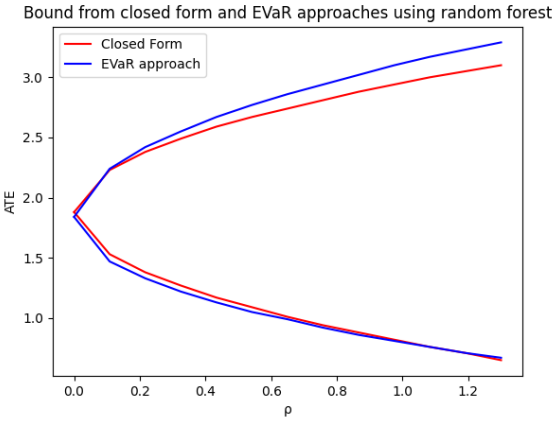

Figure 3: Bounds on the ATE computed by the EVaR (blue) and closed-form solution (red). The results are similar, as there are many samples which ensure that estimation is almost correct. However, as ρ increases, the difference also increases, since the closed-form solution works with a continuous unbounded distribution, while the EVaR approach assumes a discrete distribution bounded by the data.

[3]https://scikit-learn.org/stable/modules/generated/sklearn.mixture.GaussianMixture.html

4 Comparison between MSM and f-sensitivity

In both the MSM and the f-sensitivity model, the goal is to approximate possible bounds of a metric based on possible levels of unobserved confounding. This is done by creating a similarity metric between the observed and true propensity distribution and bounding it. However, it is worth to look at how these similarity measures differ and what reasons are there to pick one over the other.

As previously mentioned, the MSM uses the odds-ratio as a similarity measure. This effectively means that distributions are compared based on the maximum and minimum ratio of the two distributions. On the other hand, the f-sensitivity model uses the f-divergence metric, which is a well-established similarity metric, to compare the potential true distribution with the observed one. These two metrics differ and they can yield different results based on the distribution.

To show the behaviour of these similarity metrics, consider the simplest case of trying to select similar normal distributions to an observed standard normal distribution $N(0,1)$. While $e(x)/(1-e(x))$ will always be positive and, therefore, never follow a normal distribution, the goal is to visualize the behaviour of the similarity metric, which is easier with a normal distribution. The same experiment can be done with an exponential distribution, but it would not produce a comprehensive visualization.

First, using the similarity metric of maximum and minimum of the ratio between the observed distribution $N(0,1)$ and the true distribution $N(\mu,\sigma)$ would result in a in $\Gamma^{-1} \leqslant \frac{N(\mu,\sigma)}{N(0,1)} \leqslant \Gamma$, which can be expressed as $\Gamma^{-1} \leqslant \frac{1}{\sigma} e^{-1/2[(\frac{x-\mu}{\sigma})^2 - x^2]} \leqslant \Gamma$ for almost all $x \in \mathbb{R}$. This means that the inequality has to hold for all probable x. For this example, all x from -2.5 to 2.5 are considered, as this takes into account around 99% of cumulative probability of the observed x. With this setup, it is possible to find all the possible values of μ and σ that satisfy the inequality, resulting in a set of possible distributions that are considered similar to the observed standard normal distribution based on Γ.

Now, the same thing will be done in terms of KL-divergence. In this setting the constraint is $D_{KL}(N(0,1)||N(\mu,\sigma)) \leqslant \rho$. This can be expressed as $\frac{1}{2}(\frac{\mu^2}{\sigma^2} + \frac{1}{\sigma^2} - \ln\frac{1}{\sigma^2} - 1) \leqslant \rho$. However, as shown in Equation 3, f-sensitivity also uses the inverse of the odds-ratio, flipping the inequality into $\frac{1}{2}(\mu^2+\sigma^2-\ln\sigma^2-1) \leqslant \rho$. Again, it is possible to create a set of possible μ's and σ's that fulfill both conditions.

Using these two conditions, it is possible to plot the selected distributions on a grid where the x-axis represents the μ and the y-axis represents σ. Figure 4 shows how the selection of distributions differs between the two selection criteria. This illustrates how the measures can behave when using a similarity metric. While KL-divergence creates a convex ball around the observed distribution, the odds-ratio eventually creates a disconnected area. A disconnected area could be a problematic behaviour for any similarity metric, and it removes a lot of useful properties, like assuming that if two distributions are considered similar, all distributions between them are similar as well, that come with a convex result like the one from the KL-divergence.

This makes the bounded odds-ratio an odd choice when looking for a similarity metric, as it might induce some counterintuitive behaviour. However, it might be easier for researchers to determine what value of the odds-ratio is realistic, than to find a reasonable bound on the KL-divergence between the true and observed distribution.

Another difference comes from how the sensitivity models interact with the probability of the unobserved confounder. An interesting view to compare the MSM and the f-sensitivity model is through this distribution of the unobserved confounder. A question to answer would be, how the models react to the unobserved confounder being almost constant (e.g. $P(U=1) = 0.99$).

In the case of the MSM, the distribution of the hidden confounder lies within the computation of the observed propensity, as $e(x) = \sum_{u \in U} e(x,u)P(U = u|X = x)$. This implies that when U is almost constant, the observed propensity will also be almost equivalent to the true propensity score. However, because the MSM is based on the minimum and maximum of the odds-ratio, this unlikely event of U changing its value would be considered the worst-case scenario, effectively exploding the ratio. To show this, take an example where there are no confounders X, only the hidden binary confounder U such that $P(U = 1) = 0.99$. The true propensity score behaves as $e(U=1) = P(T=1|U=1) = 0.5$ and $e(U=0) = 0.1$. This means that the observed propensity score $P(T = 1) = e(U=1)P(U=1) + e(U=0)P(U=0) = 0.5 \cdot$

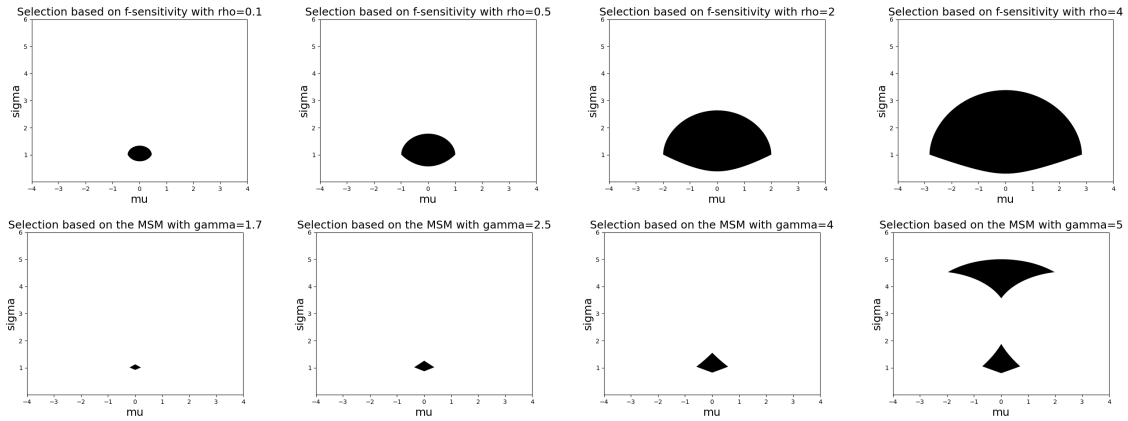

Figure 4: Selection of the plausible distributions based on the KL-criteria (top) and bounded odds-ratio (bottom). The observed distribution is a standard normal distribution and the black coloured area shows the possible distributions to pick from. In each setting, the assumed bound starts near no-unobserved confounding and goes to a high unobserved confounding setting. Each setting has the value assumed in the title of the graph. X-axis represents the possible mean μ of true distribution and Y-axis represents the standard deviation σ. The bounded odds-ratio checks for 99.99% of the observed $x \in X$.

$0.99 + 0.1 \cdot 0.01 = 0.496$. This results in the worst-case bound of $\Gamma = OR(U = 0) = \frac{P(T=1)/(1-P(T=1))}{e(U=0)/(1-e(U=0))} \approx 8.85$. A value of 8.85 could be considered quite high for an almost constant (U having the same value in 99% of the cases) experimental setup, however, this depends on the problem of the researcher applying the MSM. Before going into further details, let's first see how the f-sensitivity behaves under such conditions.

The f-sensitivity model deals with the hidden confounder distribution through the f-divergence computation. Simply put, the assumption includes the integral over the possible values of U and scales the result based on its distribution. To follow up on the almost constant example, with f-sensitivity the worst-case bound would be $\rho = f(OR(U = 0))P(T = 1|U = 0)P(U = 0) + f(OR(U = 1))P(T = 1|U = 1)P(U = 1) = 8.85 * \ln 8.85 * 0.1 * 0.01 + 0.984 * \ln 0.984 * 0.5 * 0.99 \approx 0.01$. In this case the odds ratio is almost constant, so the f-sensitivity records almost no hidden confounding.

To better illustrate this difference in behaviour, let the scenario remain the same, but plot how the sensitivity bounds behave in relation to $P(U = 1) = p$. The propensity scores remain the same at $e(U = 1) = 0.5$ and $e(U = 0) = 0.1$ and the observed propensity becomes variable $P(T = 1) = e(U = 1)p + e(U = 0)(1 - p)$. The plots for the MSM and f-sensitivity model can be found in Figure 5. While the two measurements cannot be compared outright, as each bound means something different as discussed previously, the interesting part is shown when p goes to 0 or 1. In MSM the bound keeps increasing, suggesting that indeed the MSM is sensitive to the hidden confounder distribution, while f-sensitivity is also sensitive to the distribution, but decreases the bound instead.

To summarise these observations, the similarity metric used by the MSM can create disconnected pockets of possible distributions, and the bound seems to increase as the hidden confounder becomes more constant. On the contrary, f-sensitivity uses a well-established similarity metric, and the bound keeps reducing when the hidden confounder becomes more constant.

5 Discussion

This work aimed to provide guidance and tools for causal sensitivity analysis by addressing two questions: First, how do the MSM and f-sensitivity models compare? Second, is there a way to compute f-sensitivity bounds through value at risk?

The answer to the first question consists of two aspects. The first is that the MSM uses the minimum and maximum of the odds ratio as a similarity metric, ex-

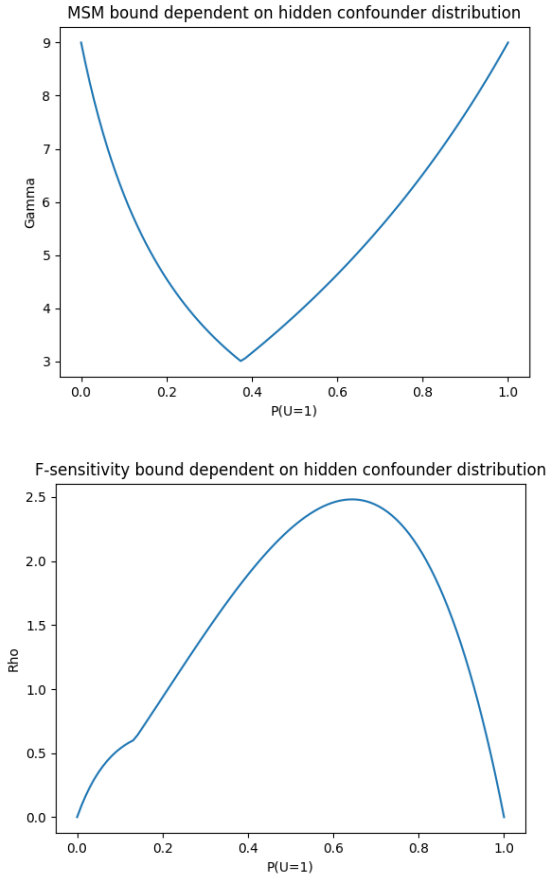

Figure 5: The evolution of the MSM (top) and f-sensitivity (bottom) bounds of the experiment described in Section 4. On the x-axis is the $P(U = 1)$, where U is a binary variable, so as it goes to 0 or 1 the hidden confounding becomes more constant. On the Y-axis you have the respective bounds for each model. As the hidden confounder becomes more constant, MSM bound increases, while f-sensitivity bound decreases, showing a difference between the interpretation of the bound. The sudden bump for f-sensitivity is defined by the selection between the treated and control f-divergence, as defined in Equation 3.

hibiting some counter-intuitive behavior when selecting possible distributions. In contrast, f-sensitivity uses a well-established similarity metric, but is harder to interpret as the bound does not directly relate to the propensity score like the MSM bound does. Therefore, there is no single answer to which approach is better, as both models can provide different information that is problem-specific. The MSM is sensitive to unlikely hidden confounders, which can be useful in scenarios where risking model failure is unacceptable, such as in medicine. The f-sensitivity model only considers hidden confounders that are likely to happen, better reflecting regular scenarios where some risk is allowed.

Regarding the second question, it has been shown that there is a way to find the bound of the f-sensitivity model through the entropic value at risk. Examples were provided: one where no model is assumed and one where a normal distribution is assumed, requiring no further approximations. This new way of computing the f-sensitivity bounds permits its wider applicability and presents an opportunity to study it in more computationally intensive settings like reinforcement learning.

Throughout this study, we aimed to ensure that the proposed method and implementation aligns with the original f-sensitivity model by providing simple examples in limited settings. However, real-world datasets could provide additional information about the differences in behaviour between the constraint programming and EVaR approaches. In addition, this would better reflect whether f-sensitivity is useful in real-life settings, or whether it takes too much time to compute or yields a wider bound. While synthetic data provides the opportunity to study the behaviour of sensitivity models with full access to all confounders, even the hidden ones, real-world data does not provide this, nor the true values for the counterfactuals. This limits how well can sensitivity models be evaluated on real-world data.

Additionally, the implementation provided in the GitHub repository could be improved. There was no effort put into tuning the learning rates or to stop computation once the EVaR converged to a maximum/minimum of the observed distribution. While these changes would improve the computing time, they would not alter the result or conclusions of this work. Similarly, the original code for f-sensitivity was not publicly accessible at the time of writing, and while

it was attempted to include it in the comparison, the results did not seem to behave the same as presented in the paper. Additionally, we attempted to ensure that the implementations were correct by comparing them with results observed in the literature. Given that this sensitivity model is relatively new, it is difficult to find many applications for comparison.

Within this study, only discrete datasets were considered, limiting the applicability of this method. To make the step towards continuous space, first a bounded space would need to be considered. It would be possible to use function estimation with a continuous feature space, however the output of such an estimator would need to be the outcome distribution, rather than just a scalar outcome. Such function estimators exist, and it would be interesting to further study how the bound behaves in such conditions.

It is also important to realize that this new approach works only with KL-divergence, as the dual representation of the entropic value at risk is created assuming KL-divergence. Further research can explore whether the method can be generalized to all f-divergence metrics and identify the properties when using f-sensitivity. A possible start would be to try to redefine f-sensitivity in terms of value at risk, or any other risk measure. This would permit the method to be solved with already existing solutions to risk analysis, as well as to generalize over other f-divergencies.

As mentioned before, many extensions of sensitivity models in reinforcement learning (RL) already exist, but they require fast computation of bounds for every existing state. With an efficient implementation of the EVaR approach, it is believed that f-sensitivity could also be extended and used as an evaluation metric that identifies the worst possible set of transitions minimizing the reward obtained by a trained policy.

Another important outlook is to create an approach for estimating what reasonable bounds would be on a problem-specific dataset. Currently, researchers find the ρ (or Γ) when the bound falls above a certain threshold of acceptable risk, which must be followed by a study into whether such ρ is realistic. If such an approach is found, sensitivity analysis can be fully integrated into the pipeline of causal inference without any additional input from experts. Possible, first step would be to analyse how the models behave when some of the hidden confounders would be hidden. This would effectively measure the strength of each confounder, which would then permit to put the found bound into perspective of the observed confounders. Other issues need to be taken into account, like joint confounders, or the fact that hiding confounders would only increase hidden confounding. One can also measure the strength as the integral over the difference between bounds found with and without a confounder.

Finally, sensitivity analysis has only been considered as a tool to evaluate trained models, but as shown in the literature, like Kausik et al. (2023) in RL setting, it can also be used during training as a form of regularization. It would be interesting to see whether it is possible to adapt our approach so that models, especially deep learning models, would be more aware of the distributions surrounding their learned distribution.

6 Conclusion

This work looked at the differences between the Marginal Sensitivity model and the new f-sensitivity model. While many similarities remain, and the goal does not change, the models behave differently under different conditions and results from both might provide valuable information about the range of possible causal effects when hidden confounding is introduced. This work also provided a new way of calculating f-sensitivity with an almost closed-form that can be extended for specific models. Hidden confounding is present everywhere, and causal sensitivity is needed to make sure our conclusions based on data are substantiated. The EVaR approach, presented and described in this study, takes us a step closer to a causal inference everyone can trust.

References

Ahmadi Javid, A. (2012). Entropic value-at-risk: A new coherent risk measure. *Journal of Optimization Theory and Applications*, 155.

Bonvini, M. and Kennedy, E. H. (2021). Sensitivity analysis via the proportion of unmeasured confounding. *Journal of the American Statistical Association*, 117(539):1540–1550.

Chakraborty, B. and Moodie, E. (2013). *Statistical Methods for Dynamic Treatment Regimes: Rein-*

forcement Learning, Causal Inference, and Personalized Medicine.

Cornfield, J., Haenszel, W., Hammond, E. C., Lilienfeld, A. M., Shimkin, M. B., and Wynder, E. L. (1959). Smoking and Lung Cancer: Recent Evidence and a Discussion of Some Questions. *JNCI: Journal of the National Cancer Institute*, 22(1):173–203.

Dorn, J., Guo, K., and Kallus, N. (2022). Doubly-valid/doubly-sharp sensitivity analysis for causal inference with unmeasured confounding.

Fisher, R. A. (1958). Cancer and smoking. *Nature*, 182(4635):596–596.

Frauen, D., Melnychuk, V., and Feuerriegel, S. (2023). Sharp bounds for generalized causal sensitivity analysis.

Jin, Y., Ren, Z., and Zhou, Z. (2022). Sensitivity analysis under the f-sensitivity models: a distributional robustness perspective.

Kallus, N. and Zhou, A. (2018). Confounding-robust policy improvement. In Bengio, S., Wallach, H., Larochelle, H., Grauman, K., Cesa-Bianchi, N., and Garnett, R., editors, *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc.

Kallus, N. and Zhou, A. (2020). Confounding-robust policy evaluation in infinite-horizon reinforcement learning.

Karlsson, R. K. A. and Krijthe, J. H. (2023). Detecting hidden confounding in observational data using multiple environments.

Kausik, C., Lu, Y., Tan, K., Makar, M., Wang, Y., and Tewari, A. (2023). Offline policy evaluation and optimization under confounding.

Louizos, C., Shalit, U., Mooij, J., Sontag, D., Zemel, R., and Welling, M. (2017). Causal effect inference with deep latent-variable models.

Marmarelis, M. G., Haddad, E., Jesson, A., Jahanshad, N., Galstyan, A., and Steeg, G. V. (2023). Partial identification of dose responses with hidden confounders.

Melnychuk, V., Frauen, D., and Feuerriegel, S. (2024). Partial counterfactual identification of continuous outcomes with a curvature sensitivity model.

Pearl, J. (2009). *Causality: Models, Reasoning and Inference*. Cambridge University Press, USA, 2nd edition.

Rosenbaum, P. R. (1987). Sensitivity analysis for certain permutation inferences in matched observational studies. *Biometrika*, 74(1):13–26.

Rosenbaum, P. R. and Rubin, D. B. (1983). Assessing sensitivity to an unobserved binary covariate in an observational study with binary outcome. *Journal of the Royal Statistical Society. Series B (Methodological)*, 45(2):212–218.

Tan, Z. (2006). A distributional approach for causal inference using propensity scores. *Journal of the American Statistical Association*, 101(476):1619–1637.

Zeng, J. and Wang, R. (2022). A survey of causal inference frameworks.

Zeng, Y., Cai, R., Sun, F., Huang, L., and Hao, Z. (2023). A survey on causal reinforcement learning.

Zhang, Y. and Zhao, Q. (2024). l^∞- and l^2-sensitivity analysis for causal inference with unmeasured confounding.

A Proofs

A.1 Proof of Theorem 1

Theorem 1. *Bounds of f-sensitivity can be computed through the dual representation of the entropic value at risk (EVaR) resulting in*

$$\mu_t^+(x) = r_t(x) \inf_{z>0} \frac{1}{z} \ln \frac{M_{Y|X=x,T=t}(z)}{e^{-\rho}}$$
$$\mu_t^-(x) = r_t(x) \sup_{z<0} \frac{1}{z} \ln \frac{M_{Y|X=x,T=t}(z)}{e^{-\rho}}$$

The proofs begins from the definition of the set of possible distributions from Jin et al. (2022). In this case, KL-divergence is used, which has the relative entropy property. The relative entropy property states that for any KL-divergence, it holds that $D_{KL}(p_1 Q_{X|T=1} + (1-p_1)Q_{X|T=0} || p_1 P_{X|T=1} + (1-p_1)P_{X|T=0}) \leqslant p_1 D_{KL}(Q_{X|T=1}||P_{X|T=1}) + (1-p_1) D_{KL}(Q_{X|T=0}||P_{X|T=0})$. This leads to a simplified form:

$$\begin{aligned}\mathbf{Q} &= \{\mathcal{Q} : \frac{d\mathcal{Q}_X}{d\mathcal{P}_X} = r(x), D_{KL}(\mathcal{Q}_{Y|X}||\mathcal{P}_{Y|X}) \leqslant \rho\} \\ &= \{\mathcal{Q} : \frac{d\mathcal{Q}_X}{d\mathcal{P}_X} = r(x), D_{KL}(\mathcal{Q}_{Y|X,T=1}||\mathcal{P}_{Y|X,T=1}) \leqslant \rho, \\ &D_{KL}(\mathcal{Q}_{Y|X,T=0}||\mathcal{P}_{Y|X,T=0}) \leqslant \rho\}\end{aligned}$$

Combining the dual formulation of EVaR from Ahmadi Javid (2012) defined in Equation 6 (Theorem 3.3 in authors paper), with the set of distributions $\mathbf{Q}$, where f-sensitivity tries to find the upper bound of the expected value, leads to $\sup_{\mathcal{Q}\in\mathbf{Q}}[E_{\mathcal{Q}}(Y|X,T)] = \text{EVaR}_{1-\alpha}(Y|X,T) = \text{EVaR}_{1-e^{-\rho}}(Y|X,T)$, equating the two. The set of possible solutions $\mathcal{Q}$ is defined as any distribution following $\mathcal{Q}_{Y|T=1} << \mathcal{P}_{Y|T=1}$, meaning that $\mathcal{Q}$ is not defined outside of $\mathcal{P}$, and $D_{KL}(\mathcal{Q}_{Y|T=1}||\mathcal{P}_{Y|T=1}) \leqslant \rho$, being the f-sensitivity assumption.

$$\text{EVaR}_{1-\alpha}(X) = \sup_{\mathcal{Q}} E_{\mathcal{Q}}(X) \tag{6}$$

Projecting this back into the primal, and forming a moment generating function based on the observed samples, leads to the solution defined in Equation 7.

$$\begin{aligned}\text{EVaR}_{1-\alpha}(Y) &= \inf_{z>0}[z^{-1}\ln(\frac{M_X(z)}{\alpha})] \\ &= \min_{z>0}[z^{-1}\ln(\frac{\frac{1}{N}\sum_{i=0}^{N} e^{zY_i}}{\alpha})] \\ &= \min_{z>0}[\frac{1}{z}\ln(\sum_{i=0}^{N} e^{zY_i}) - \frac{1}{z}\ln(N\alpha)]\end{aligned} \tag{7}$$

Where N is the number of samples, Y_i is the outcome of a sample, $\alpha = e^{-\rho}$ and z represents the moment of interest.

This, however, results in the upper bound for the input distributions, for the lower bound the equation changes to:

$$\max_{z<0}[\frac{1}{z}\ln(\sum_{i=0}^{N} e^{zY_i}) - \frac{1}{z}\ln(N\alpha)]$$

Lastly, f-sensitivity allows a shift in the distribution of confounders X through the $\frac{d\mathcal{Q}_X}{d\mathcal{P}_X} = r(x)$ constraint. This can be ensured by scaling the solution for every $x \in X$ by $r(x)$, resulting in Equations 8, which is equivalent to Theorem 1.

$$\begin{aligned}\mu_t^+(x) &= r_t(x) \inf_{z>0} \frac{1}{z} \ln \frac{M_{Y|X=x,T=t}(z)}{e^{-\rho}} \\ \mu_t^-(x) &= r_t(x) \sup_{z<0} \frac{1}{z} \ln \frac{M_{Y|X=x,T=t}(z)}{e^{-\rho}}\end{aligned} \tag{8}$$

A.2 Proof of Theorem 2

This proof works with the sum of Gaussians with a uniform weight rather than just a single Gaussian, to make the theorem more generic.

Theorem 2. *Assuming that the outcome follows a normal distribution $N(\mu, \sigma)$, the bounds of f-sensitivity can be expressed as:*

$$\mu_t^+(x) = r_t(x)[\sqrt{\rho\frac{\sigma^2}{2}} + \mu + \frac{\sqrt{2\rho}}{2}]$$
$$\mu_t^-(x) = r_t(x)[-\sqrt{\rho\frac{\sigma^2}{2}} + \mu - \frac{\sqrt{2\rho}}{2}]$$

Where the square root always yields the non-negative part of the possible solutions.

Assuming an estimator that fits a sum of Gaussians with uniform weights onto the outcome distribution defined as $\frac{1}{k}\sum_i^k N(\mu_i, \sigma_i)$, it is possible to define the moment generating function as:

$$\begin{aligned} M_z(Y|X=x, T=t) &= E[e^{zY|X=x,T=t}] \\ &= E[e^{\frac{z}{k}\sum_{i=0}^{k} N(\mu_i,\sigma_i)}] \\ &= E[\prod_{i=0}^{k} e^{\frac{z}{k}N(\mu_i,\sigma_i)}] \\ &= \prod_{i=0}^{k} E[e^{\frac{z}{k}N(\mu_i,\sigma_i)}] \\ &= \prod_{i=0}^{k} e^{\frac{z}{k}\mu_i + \frac{1}{2}\frac{z^2}{k^2}\sigma_i^2} \end{aligned} \tag{9}$$

Applying Theorem 1 means that a closed-form solution can now be found, shown in Equation 10.

$$\begin{aligned} \mu_t^+(x) &= r_t(x) \inf_{z>0} \frac{1}{z} \ln \frac{M_{Y|X=x,T=t}(z)}{e^{-\rho}} \\ &= r_t(x) \inf_{z>0} \frac{1}{z} (\ln \prod_{i=0}^{k} e^{\frac{z}{k}\mu_i + \frac{1}{2}\frac{z^2}{k^2}\sigma_i^2} + \rho) \\ &= r_t(x) \inf_{z>0} \frac{1}{z} (\sum_{i=0}^{k} \ln(e^{\frac{z}{k}\mu_i + \frac{1}{2}\frac{z^2}{k^2}\sigma_i^2}) + \rho) \\ &= r_t(x) \inf_{z>0} \frac{1}{z} (\rho + \sum_{i=0}^{k} \frac{z}{k}\mu_i + \frac{1}{2}\frac{z^2}{k^2}\sigma_i^2) \\ &= r_t(x) \inf_{z>0} [\frac{\rho}{z} + \sum_{i=0}^{k} \frac{1}{k}\mu_i + \frac{1}{2}\frac{z}{k^2}\sigma_i^2] \end{aligned} \tag{10}$$

This leads to a closed-form solution described in Equation 11. The result gives us the optimal z to use in order to find the maxima or minima (for μ_t^- the negative square result of the square root is taken).

$$\begin{aligned} &\inf_{z>0} [\frac{\rho}{z} + \sum_{i=0}^{k} \frac{1}{k}\mu_i + \frac{1}{2}\frac{z}{k^2}\sigma_i^2] \\ \implies &\frac{d}{dz}\frac{\rho}{z} + \sum_{i=0}^{k} \frac{1}{k}\mu_i + \frac{1}{2}\frac{z}{k^2}\sigma_i^2 = 0 \\ &-\frac{\rho}{z^2} + \sum_{i=0}^{k} \frac{1}{2}\frac{1}{k^2}\sigma_i^2 = 0 \\ &z = \sqrt{\frac{\rho}{\sum_{i=0}^{k} \frac{1}{2k^2}\sigma_i^2}} \end{aligned} \tag{11}$$

With all of this, the final solution can be found by computing $\mu_t^+(x) = \sqrt{\frac{\rho}{2k^2}\sum_{i=0}^{k}\sigma_i^2} + \sum_{j=0}^{k}[\frac{\mu_j}{k}] + \frac{\sqrt{2\rho}}{2k}$. The same can be applied to μ_t^-, but with a negative result of the square root. All of this leads to the extended form of Theorem 2 found below:

Theorem 3. *Assuming a sum of Gaussians with a uniform weight k, that fits set of parameters μ and σ represented as $Y|X = x, T = t \sim \frac{1}{k}\sum_{i=0}^{k} N(\mu_i(x,t), \sigma_i(x,t))$, the bounds of f-sensitivity can be expressed as:*

$$\mu_t^+(x) = r_t(x)[\sqrt{\rho \sum_{i=0}^{k} \frac{\sigma_i^2}{2k^2}} + \sum_{j=0}^{k} \frac{\mu_j}{k} + \frac{\sqrt{2\rho}}{2k}]$$

$$\mu_t^-(x) = r_t(x)[-\sqrt{\rho \sum_{i=0}^{k} \frac{\sigma_i^2}{2k^2}} + \sum_{j=0}^{k} \frac{\mu_j}{k} - \frac{\sqrt{2\rho}}{2k}]$$

Where the square root always yields the non-negative part of the possible solutions.

Setting $k = 1$ in the above theorem results in Theorem 2.

B Further experimentation

In this section more experiments are shown which were not considered in the main body, since they were not essential to the claims made. These results still reinforce the same conclusions, but might be in a more complex or general setting.

B.1 Reproduction of the authors results

To check whether the EVaR formulation matches the results of the Jin et al. (2022) formulation, the first attempt was to reproduce the numerical experiments in the paper. However, the authors provide the result of only one run in a normally distributed setting. In this setting, the metric in question is the Average Treatment effect on the Control (ATC) and not the ATE. The setting, which is described thoroughly in Section 5.1 (Jin et al., 2022) of the original paper contains uniformly distributed features, and thus the EVaR approach cannot be applied to it in its current form. However, similarly to the sum of Gaussians estimator, it is possible to create a closed-form solution if a normal distribution is assumed. To adjust for the non-determinism of the authors' algorithm, an area within 2 standard deviations of the average results is highlighted.

The comparison between the 2 results can be found in Figure 6. The results show that the theoretical EVaR result would match the true result, which makes sense given that we assumed the normal distribution. Additionally, the result from the authors is within 2 standard deviations, making it possible that the results would match, if ran on multiple seeds.

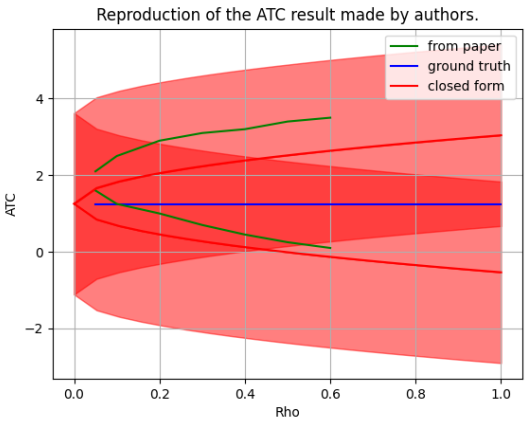

Figure 6: The comparison between the results of the closed-form approach through EVaR and the result given from the paper. The blue line indicates the true ATC, green line indicated the result from the paper and red the EVaR result.

B.2 Authors Algorithm

To further connect the EVaR approach with the work done by Jin et al. (2022), an attempt was made to compare the two. For this comparison, the same dataset as in Example 2 was used. However, as mentioned previously, the authors' algorithm is only approximating the bound, not computing it directly, and it requires 2 regressors and 1 empirical risk minimizer (ERM) for that approximation.

In this setting, each estimator is using linear regression, as the input is just a single binary variable. The first regressor is used to estimate the $r(x)$ function, while th second and third are used within the ERM to predict nuisance parameters α and η that minimize the lagrangian. The last regressor is then used to predict the value of the lagrangian based only on X. The authors' algorithm also splits the dataset into 3 independent datasets and predicts the bound for each, outputting the average bound. In this experiment, each regressor was given 1000 steps with learning rate set to 0.001 and each step training on the entire available dataset (one of the 3 splits). Because of the non-deterministic nature of this algorithm, it was ran 10 times on different seeds and the results were averaged. The results are shown in Figure 7.

The results do not match the expected base result for the case when $\rho = 0$. In that case, the bound should match the ATE observed by the Random Forest estimator, which was around 1.75. This is most likely due to wrong implementation of some part of the algorithm, or due to missspecification of the estimators used. The results are quite unstable which is most likely due to the lack of data. Further research needs to be done to check whether the provided implementation is giving the correct results, or if the results are correct and the algorithm just does not perform well in a simple setting with a binary confounder.

B.3 Range of experiments

In this experiment, the same setup was used as in Example 1, but with a range of possible values used for the set values. Equation 12 formulates this in terms of u_p, x_u, t_u, y_u variables. Each formulation had a dataset of 30000 samples generated and both the constraint

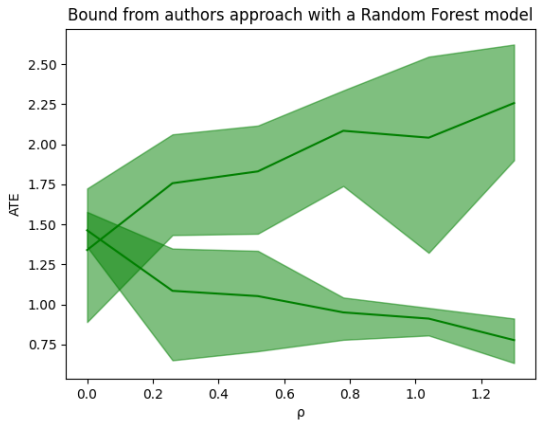

Figure 7: Results obtained from the authors' algorithm when ran on the setting from Example 1. The results seem unstable and do not match the observed ATE when no hidden confounding is assumed. The solid green line represents the average observed bound, and the faded area illustrates the area between the 5th and 95th percentiles of the observed bounds.

programming and the EVaR approaches were tested to see whether the results would match. Because of computing time, the range of each variable was limited as shown in the results in Table 1. Additionally, each setting was only tested on $\rho \in \{0, 0.5, 1\}$, and the highest absolute difference between the approaches was measured.

$$\begin{aligned} U &\sim Bern(u_p) \\ X &\sim Bern(x_u) \\ T &\sim 0.5 + 0.25X + t_u U \\ Y &\sim X + y_u U + 2T - N(1, 0.1) \end{aligned} \tag{12}$$

To show these results, Table 1 tracks the highest observed absolute difference between the two approaches. Each row looks at a specific variable and each column represents a possible value for that variable. The values within are the absolute difference between all the scenarios where that variable had the specific value assigned. These values are rounded to the nearest upper 3rd decimal digit, as both the Ipopt[4] solver used for the constraint programming setting and the learning rate used for gradient descent are set to 0.001. While there are some differences between the two methods, these are usually cause by the different tolerance levels of the solver used for the constraint programming definition and the tolerance that determines when gradient descent converged for the EVaR setting.

[4]https://coin-or.github.io/Ipopt/

	-0.5	-0.25	0.05	0.15	0.95
u_p	-	-	0.004	0.005	0.005
x_u	-	-	0.003	0.004	0.005
t_u	-	0.003	0.004	0.005	-
y_u	0.004	0.005	0.005	0.005	0.004

Table 1: Each row looks at a specific variable. Each column represents a different value a variable can take. Values within the table represent the absolute difference between the constraint programming approach and EVaR approach. Settings which are impossible, such as probabilities outside of [0, 1], are discarded and labeled with a dash.

B.4 Dimensionality experiments

The last set of experiments was with the dimension of confounders X. While in all other experiments X is represented by a single binary variable, in this experiment it is set to $k = 3$ binary variables. This leads to the formulation in Equation 13. This is to show that the approach works in more complex setting with multiple input variables. However, given each possible $x \in X$ is considered in both approaches, the time complexity of higher dimensions can be too much for a simple computer to compute.

$$\begin{aligned} U &\sim Bern(0.25) \\ X &\sim Bern(0.45)^k \\ T &\sim 0.5 + 0.25 \sum_i^k X_i/3 - 0.3U \\ Y &\sim \sum_i^k X_i/3 + U + 2T - N(1, 0.1) \end{aligned} \tag{13}$$

The results are shown in Figure 8. While the results mostly match, however the constraint programming approach took significantly longer to compute and failed to converge in some instances, creating these steps in the result as ρ becomes larger.

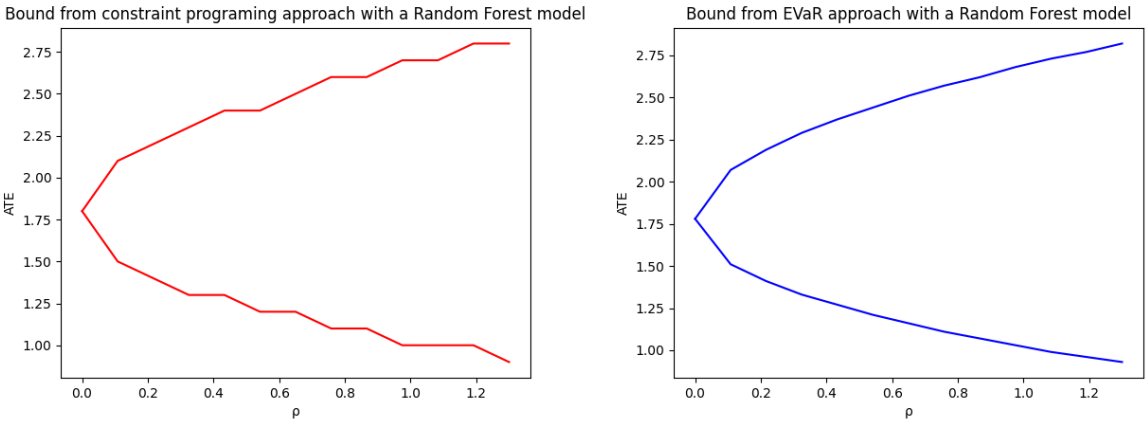

Figure 8: Results of the dimensionality experiment. In this setup there are 3 binary confounders instead of 1. The results of the constraint programming method (left) and the EVaR method (right) still match, but the constraint programming method takes longer and fails to converge in some instances, creating sudden steps.