To cite this article:

Wu, Y., Radev, S. T., & Tuerlinckx, F. (in press). Testing and improving the robustness of amortized bayesian inference for cognitive models. *Psychological Methods*.

Testing and Improving the Robustness of Amortized Bayesian Inference for Cognitive Models

Yufei Wu[1,*], Stefan T. Radev[2], and Francis Tuerlinckx[1]

[1]University of Leuven, Belgium

[2]Rensselaer Polytechnic Institute, US

[*]*Corresponding author:*yufei.wu@kuleuven.be

Abstract

Contaminant observations often cause problems when estimating the parameters of cognitive models. In this study, we tested and improved the robustness of parameter estimation using amortized Bayesian inference. We conducted systematic analyses in two settings: a toy example (i.e., a normal distribution with an unknown mean) and a popular cognitive model, the Drift Diffusion Model. First, we studied the stylized sensitivity curve and the breakdown point of the estimators. Next, we proposed a simple data augmentation approach that incorporated a contamination distribution into the data-generating process during training to train robust estimators. We examined several robust estimators with different contamination distributions, and evaluated their performance and cost in terms of accuracy and efficiency loss relative to a standard estimator. Introducing contaminants from a Cauchy distribution during training significantly increases the robustness of the neural density estimator, as measured by bounded sensitivity functions and a substantially higher breakdown point. Overall, the proposed method is straightforward and practical to implement, with broad applicability in fields where outlier detection or removal is challenging.

Keywords: outlier, robust estimation, amortized Bayesian inference, drift diffusion model, Cauchy distribution

Introduction

Identifying and appropriately handling contaminants is both unavoidable and essential for drawing accurate conclusions from quantitative data. Contaminants are generated by processes different from the process under study. For example, in behavioral science, careless responding is a major source of contamination (Maniaci & Rogge, 2014). On the one hand, contamination can lead to outliers or extreme values that deviate significantly from

the majority of data points (Aggarwal, 2017; Hawkins, 1980). Commencing with Huber's foundational work (1964) on M-estimators for estimating the mean of a univariate normal distribution, the field of robust statistics has generated several methods for detecting and dealing with outliers (Donoho & Huber, 1982; Hawkins, 1980; Tukey, 1979; Hampel et al., 2005). On the other hand, contaminants can also have numerical values that are difficult to distinguish from the majority, making accurate estimates more challenging (Maronna et al., 2006).

Some statistical and computational models in psychology are particularly vulnerable to the presence of contaminants due to the nature of their underlying assumptions. One example is the widely used Drift Diffusion Model (DDM; Ratcliff, 1978) that explains patterns in choice reaction time data from fast decision-making tasks. In the most basic DDM variant, a key parameter is the non-decision time (T_{er}) capturing the time taken for stimulus encoding and motor execution. By design, T_{er} is estimated to be lower than the shortest reaction time in the data set. Thus, a short outlier can distort the estimate of T_{er} and consequently the other parameters that jointly determine the process with T_{er}. As a result, mitigating the impact of contaminants has been a persistent challenge in DDM estimation (Ratcliff, 1993; Ratcliff & Rouder, 1998; Ratcliff & Tuerlinckx, 2002; Myers et al., 2022).

Various sophisticated methods have been proposed to reduce the impact of outliers in fitting of different models (Maronna et al., 2006; Sumarni et al., 2017; Aggarwal, 2017). When models are fit using maximum likelihood estimation (MLE) or Bayesian methods (e.g., MCMC sampling, Kruschke, 2011), robust methods often require modifying the loss or the likelihood function of the model. This introduces two key challenges. First, modifying the loss function can be impractical for many model users. Second, not all models have an explicitly available loss function or likelihood (Palestro et al., 2018).

In this paper, we studied robustness within an amortized Bayesian inference (ABI, Radev et al., 2020) framework, a simulation-based approach combined with neural networks. ABI leverages simulated training data to optimize generative neural networks that encode structural and functional knowledge about the simulation model and its parameters. By recasting the costly posterior sampling task as forward passes through a trained neural network, ABI achieves nearly instantaneous (i.e., *amortized*) parameter inference for new data sets (Gonçalves et al., 2020; Radev et al., 2020). Throughout this paper, we refer to these neural networks as neural posterior estimators (NPEs). Crucially, ABI methods bypass the model likelihood (hence their alternative designation as "likelihood-free") and avoid expensive MCMC sampling or likelihood approximations altogether, making them an attractive and versatile addition to the modeler's toolkit (Schmitt et al., 2024a).

Nevertheless, amortized methods are more susceptible to estimation errors than their non-amortized counterparts (e.g., MCMC) in the presence of *model misspecification* (Schmitt et al., 2021). The presence of contaminants can be viewed as a special case of model misspecification (Schmitt et al., 2021), and it is a serious obstacle for the adoption of amort-

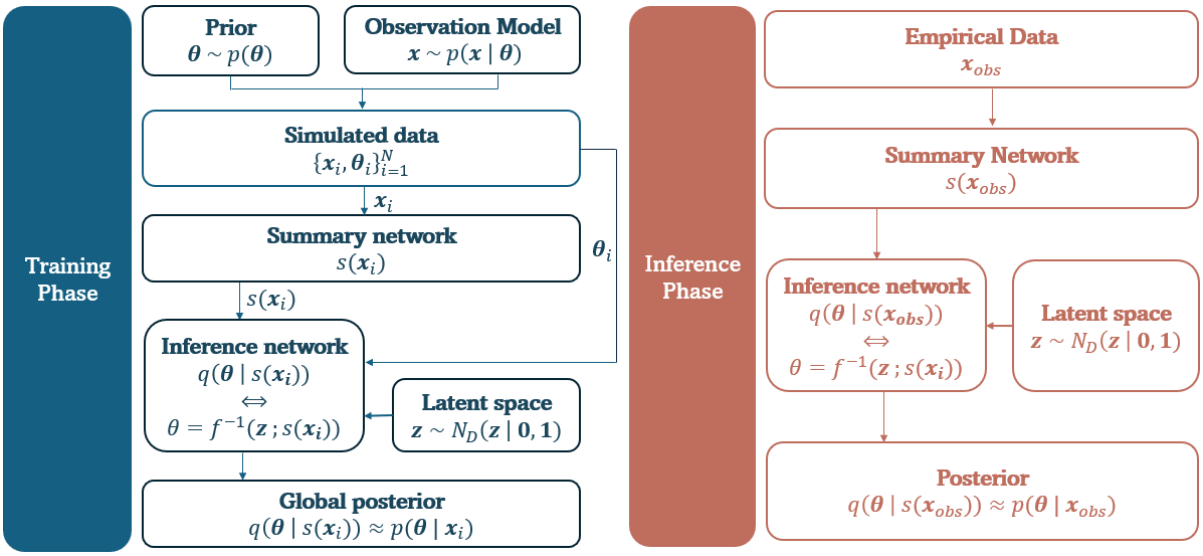


Figure 1: A basic amortized Bayesian workflow. Parameters and data are simulated from a prior and an observation model. The simulations are used as training data for the summary and inference networks that jointly learn the posterior. Once the networks are trained, they can instantly sample from the approximate posterior given any new data.

ized methods across behavioral and cognitive sciences, where computational models are nearly always approximations and data tends to be noisy, unpredictable, and corrupted in unexpected ways. Although previous work has focused on inducing *robust* amortized estimators (Ward et al., 2022; Huang et al., 2024; Siahkoohi et al., 2023; Kelly et al., 2024; Elsemüller et al., 2025), these methods either require non-trivial modifications to the straightforward simulation-based training or sacrifice the amortization property.

Thus, in this study, we investigated the influence of contaminants and proposed a simple data augmentation approach to enhance the robustness of ABI in cognitive modeling. To clarify the use of ABI, we compared amortized and traditional inference methods and conducted a systematic empirical investigation into the meaning of learned summary statistics. The univariate normal distribution and DDM served as test beds to demonstrate our robust methodology. Our approach can be extended to a wide range of other stochastic models of cognition, whether likelihood-based or simulation-based.

Amortized Bayesian Inference

Amortized methods utilize model simulations to train specialized neural networks that learn to compress data of varying sizes and sample from the posterior of model parameters given *any* data set compatible with the model (Schmitt et al., 2021).

A simple amortized workflow is depicted in Figure 1. During the training phase, a prior $\boldsymbol{\theta} \sim p(\boldsymbol{\theta})$ and a generative model $\boldsymbol{x} \sim p(\boldsymbol{x} \mid \boldsymbol{\theta})$ are defined, where $\boldsymbol{\theta} \in \mathbb{R}^D$. Subsequently, parameters are simulated from the prior and passed to the cognitive model to generate synthetic data sets. The simulated parameters and data sets $\{\boldsymbol{x}_i, \boldsymbol{\theta}_i\}_{i=1}^I$ are used to train two neural networks: a summary and an inference network. The summary

network transforms each data set $\boldsymbol{x}_i$ into fixed-size *approximately sufficient* summary statistics $s(\boldsymbol{x}_i)$, where $s(\cdot)$ represents the transformation applied by the summary network. Simultaneously, the inference network learns to approximate the true posterior of model parameters given $\boldsymbol{x}$ as accurately as possible:

$$q(\boldsymbol{\theta} \,|\, s(\boldsymbol{x})) \approx p(\boldsymbol{\theta} \,|\, \boldsymbol{x}), \tag{1}$$

Approximating complex posterior distributions involves sampling from high-dimensional conditional probability distributions. Recent advancements in generative deep learning provide a wide range of expressive algorithms to address this challenge (e.g., Kobyzev et al., 2020; Lipman et al., 2022; Yang et al., 2023). For instance, normalizing flows (Kobyzev et al., 2020) learn an invertible transformation f between the complex target distribution and a predefined simple latent distribution $\boldsymbol{z}$ (e.g., a spherical Gaussian), such that sampling from $\boldsymbol{z}$ and applying the inverse f^{-1} yields samples from the approximate posterior:

$$\boldsymbol{\theta} \sim q(\boldsymbol{\theta} \mid s(\boldsymbol{x})) \Longleftrightarrow \boldsymbol{\theta} = f^{-1}(\boldsymbol{z};\, s(\boldsymbol{x})) \text{ with } \boldsymbol{z} \sim \mathcal{N}_D(\boldsymbol{z} \,|\, \mathbf{0}, \mathbf{I}), \tag{2}$$

where f is an invertible function parameterized by a conditional invertible network (Ardizzone et al., 2019; Radev et al., 2020). The summary and inference networks are jointly optimized to ensure that the approximate posterior corresponds as closely as possible to the true posterior. The parameters of the neural networks are learned by minimizing the Kullback-Leibler (KL) divergence between the true and the approximate posterior for any data set $\boldsymbol{x}$ sampled from the prior predictive distribution $p(\boldsymbol{x})$ (for more details, please refer to Kullback & Leibler, 1951; Radev et al., 2020):

$$(f^*, s^*) = \underset{f,s}{\operatorname{argmin}}\, \mathbb{E}_{p(\boldsymbol{x})}\Big[D_{\mathrm{KL}}\big(p(\boldsymbol{\theta} \mid \boldsymbol{x}) \,||\, q(\boldsymbol{\theta} \mid s(\boldsymbol{x}))\big)\Big]. \tag{3}$$

Upon convergence, the neural networks acquire an approximate representation of the full posterior. In practice, we define a prior distribution broad enough to generate a realistic range of empirical data. The inference phase (see Figure 1) is then straightforward: the observed data $\boldsymbol{x}^{(obs)}$ is fed into the networks, and the corresponding posterior can be inferred without any overhead.

More recent amortized algorithms employ multi-step continuous-time models, such as flow-matching (Dax et al., 2023), diffusion models (Simons et al., 2023), and few-step consistency models (Schmitt et al., 2024b). In this study, we focused on normalizing flows, the simplest family of deep probabilistic models that works sufficiently well in practice.

Drift Diffusion Model

The DDM is by far the most popular cognitive model for fitting choice reaction time data in terms of neurocognitively plausible parameters (Ratcliff, 1978; Ratcliff et al., 2016; Usher & McClelland, 2001). In choice reaction tasks, participants are asked to make

binary decisions, such as determining the moving direction (left or right) of a collection of random dots (Mulder et al., 2012), or classifying words and non-words (Ratcliff et al., 2004). Performance is then compared across different conditions (e.g., primed vs. non-primed) or participant demographics (e.g., younger vs. older adults).

The core assumptions of DDM are illustrated in Figure 2. Participants accumulate evidence about a certain decision after stimulus onset, and when the evidence reaches one of the two boundaries: 0 or a, a decision is made, and a is known as the boundary separation. The starting point of this process is denoted as $z \in [0, 1]$. If $z > 0.5$ ($z < 0.5$), then there is a bias towards the upper (lower) boundary. The drift rate v represents the average rate of evidence accumulation under a specific condition. The stochastic differential equation (SDE) provides a precise description of this diffusion process:

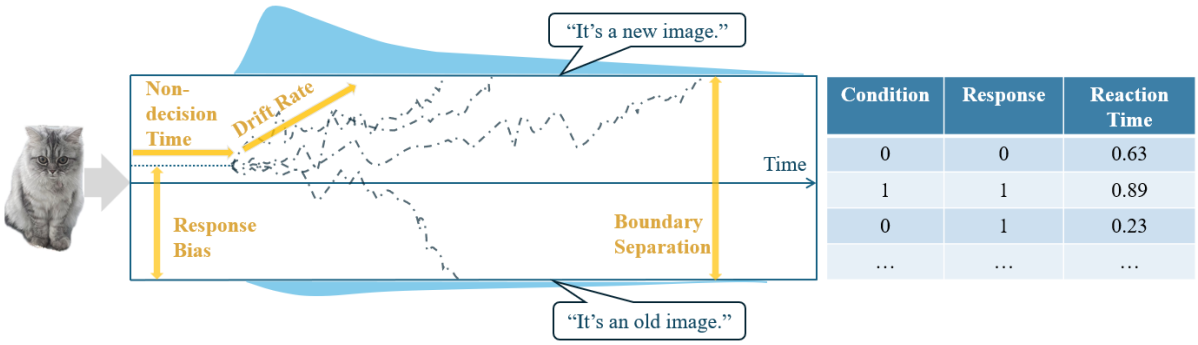
$$dX_t = v\,dt + \sigma\,dW_t, \tag{4}$$

for $0 < X_t < a$. Equation 4 captures the random evolution of accumulated evidence X over time t, with v, the deterministic drift, and dW_t, the differential of Brownian motion (or Wiener process). When $X_t \geq a$ or $X_t \leq 0$, the diffusion process stops and the decision is made. Finally, the non-decision time T_{er} accounts for processes unrelated to decision formation. These parameters together determine the distribution of reaction times and choices.

Numerous variants of the standard DDM have been proposed (see e.g., Ratcliff & Tuerlinckx, 2002; Ratcliff & Rouder, 1998; Ratcliff et al., 2016; Wieschen et al., 2020). This paper focuses on the standard version. The reasons for this choice are threefold. First, the added parameters, such as trial-to-trial variability parameters in v and T_{er}, often cause estimation problems (see e.g., Lerche & Voss, 2016; Tillman et al., 2020). Second, the standard DDM is an interesting case to study robustness, given the strong dependence of T_{er} on the minimum response time. Third, its analytical tractability enables us to better understand the mechanisms underlying learned summary statistics in ABI, effectively unpacking the learning process's black box.

Although DDM is known to be sensitive to outliers and various fitting methods exist for this model (Ratcliff & Tuerlinckx, 2002; Wagenmakers et al., 2007; Voss & Voss, 2008; Ahn et al., 2017), no study has systematically investigated the impact of contaminants in DDM estimation(Ratcliff & Tuerlinckx, 2002; Lerche et al., 2017). One reason for this gap is that large-scale simulation studies can be highly time-consuming with existing methods, while the instantaneous posterior sampling with ABI is extremely efficient (e.g., Von Krause et al., 2022).

There is no universal solution to the problem of contamination. A common approach is to set cutoffs for reaction times (standard thresholds include reaction times longer than 100ms or 200ms and shorter than 2.5s or 3s) and discard trials that fall outside these predetermined limits (Myers et al., 2022). Alternative methods to mitigate the impact of outliers include applying cutoffs based on either the standard deviation or the



Condition	Response	Reaction Time
0	0	0.63
1	1	0.89
0	1	0.23
…	…	…

Figure 2: A graphical illustration of the drift diffusion process and the resulting reaction time data in a hypothetical visual recognition memory task. Participants view an image and judge whether it is old (i.e., previously seen) or new (i.e., previously unseen). The rows in the resulting data table correspond to individual trials of the experiment, with conditions and responses coded as 0 (old image) and 1 (new image).

interquartile range (Ratcliff, 1993). While being less arbitrary, these methods can reduce information gain and introduce their own biases (Ratcliff, 1993; Ulrich & Miller, 1994). More challenges arise when masked contaminants are present in data sets (Ratcliff, 1993), making the DDM an ideal test bed for evaluating our data augmentation approach within the framework of ABI.

Parameter Recovery Study for the Drift Diffusion Model

To offer insights into the performance and inner workings of end-to-end amortized inference, we compared estimates from (a) ABI, implemented in the Python package `BayesFlow` (Radev et al., 2023), version 1.1.4, (b) MCMC sampling, using the R library `JAGS` (Plummer et al., 2003) and its `dwiener` module (Wabersich & Vandekerckhove, 2014), and (c) EZ diffusion (Wagenmakers et al., 2007). As the golden standard for fully Bayesian estimation, MCMC sampling serves as a benchmark to evaluate ABI's performance. EZ diffusion helps to understand the learned summary statistics in ABI, because its estimation relies on three *sufficient summary statistics*. It works only when the model assumes $z = 0.5$ and provides an analytical solution to the other key DDM parameters, v, a, and T_{er}. EZ diffusion uses the following sufficient summary statistics: M_{RT}, the mean reaction time of correct responses; V_{RT}, the variance of reaction time for correct responses; and P_c, the proportion of correct response (for details, please refer to Wagenmakers et al., 2007). We estimated the DDM parameters with BayesFlow, JAGS, and the EZ-diffusion to investigate (a) the validity of our amortized Bayesian setup for recovering the parameters compared to simpler alternatives; and (b) the relationship between the summary statistics learned by the summary network and the simple statistics used by EZ-diffusion.

For simulation, we assumed a minimal DDM with $z = 0.5$ and a single drift rate v. Ground-truth parameters (v, a, T_{er}) were sampled from their corresponding prior

Parameters	Distribution
v (Drift rate)	$U(0.2, 2)$
a (Boundary separation)	$U(0.5, 2.5)$
T_{er} (Non-decision time)	Gamma$(1.5, 0.2)$

Table 1: Prior distributions for the parameter recovery study. Gamma$(1.5, 0.2)$ denotes a gamma distribution with shape parameter 1.5 and scale parameter 0.2.

(Table 1), with the within-trial noise variance σ^2 in the diffusion process fixed at 1. The prior was selected to align with the conservative parameter range suitable for EZ diffusion, based on a systematic review of DDM parameters. (Wagenmakers et al., 2007; Tran et al., 2021). Each data set contained 200 observations, and every observation i (with $i = 1, \ldots, 200$) consisted of a pair of a reaction time x_i and choice y_i, jointly simulated from the Wiener process, $(x_i, y_i) \sim \text{Wiener}(v, a, z = 0.5, T_{er})$.

For ABI, we defined the neural network architecture based on model and data requirements. Given the DDM's IID trial assumption, we used a permutation-invariant Set Transformer (Lee et al., 2019) as the summary network, with three outputs matching the EZ diffusion method's sufficient statistics. The inference network had six affine coupling layers. NPE training ran for 100 epochs (1000 iterations per epoch, batch size 32) using the Adam optimizer (initial LR $= 5 \times 10^{-4}$, cosine decay schedule). The training regime was consistent with typical NPE training regimes for stochastic cognitive models (Schumacher & Radev, 2023), and the loss value converged by the end of the training.

After training, 500 new data sets were simulated to assess parameter recovery. For Bayesflow, simulated data were passed through the trained summary and inference networks to generate posterior samples. JAGS was run with four chains, each using 1000 adaptation steps, 1000 burn-in steps, and 1000 sampling iterations. To make results comparable, the prior in JAGS was the same as the prior in BayesFlow (Table 1). Convergence was assessed via $\hat{R}$ statistics (Gelman & Rubin, 1992), with $\hat{R} < 1.01$ indicating acceptable convergence. For the EZ diffusion model, parameters were derived using the three sufficient summary statistics. Out of 500 data sets, 471 converged in JAGS; the remaining 29 data set was excluded from all methods to maintain comparability. No convergence issue occurred in BayesFlow or EZ diffusion. The parameter recovery plots for all three methods can be found in Figure A1, and the simulation-based calibration shows that the estimation from Bayesflow was unbiased (see Figure A2).

Performance metrics are reported in Table 2. The root mean squared error (RMSE) and the average marginal posterior standard deviation (SD) across 500 data sets measured the accuracy and precision of three estimation methods. The RMSE is defined as: $\text{RMSE}_\theta = \sqrt{\frac{1}{n}\sum(\hat{\theta}_i - \theta_i)^2}$, and its variability was measured by the SD of the mean absolute error $|\hat{\theta}_i - \theta|$ across 500 data sets. We also reported the spread of individual marginal posterior SD values as a measure of posterior contraction.

As expected, BayesFlow and JAGS showed highly similar accuracy and precision across

all parameters, with negligible differences in RMSE, average posterior SD, and variability. EZ diffusion was slightly less accurate than these two methods.

	RMSE			Posterior SD		
	EZ	BayesFlow	JAGS	EZ	BayesFlow	JAGS
v	0.136 (0.088)	0.113 (0.073)	0.115 (0.074)	N/A	0.112 (0.030)	0.111 (0.030)
a	0.128 (0.087)	0.076 (0.051)	0.081 (0.051)	N/A	0.076 (0.031)	0.068 (0.028)
T_{er}	0.045 (0.031)	0.014 (0.010)	0.014 (0.010)	N/A	0.015 (0.008)	0.012 (0.006)

Table 2: Performance comparison between methods based on RMSE and posterior SD. The $RMSE$ and the posterior SD were calculated from estimates of 500 data sets using three distinct methods. For both metrics, the value reported in parentheses represents the standard deviation of that metric's values across the 500 data sets. Since EZ diffusion only provides point estimates, no standard deviation is available for this method.

Interpreting the Learned Summary Statistics

Given the agreement between estimation in EZ-diffusion and BayesFlow, it is reasonable to assume that the three learned summary statistics $\boldsymbol{s}_B$ should contain the same information as the sufficient summary statistics in EZ diffusion, denoted as $\boldsymbol{s}_{EZ} = (M_{RT}, V_{RT}, P_c)$ (see Figure A3 in Appendix for a direct mapping).

To explore their relationship, we retrieved $\boldsymbol{s}_B$ and $\boldsymbol{s}_{EZ}$ for the 500 data sets used above, and built a multivariate random forest (Segal & Xiao, 2011, implemented in the R package `MultivariateRandomForest`). It is a non-parametric machine learning algorithm that captures complex non-linear relationships and makes predictions (Segal & Xiao, 2011). We trained a multivariate random forest with 250 sets of $(\boldsymbol{s}_B, \boldsymbol{s}_{EZ})$ and used the remaining 250 $\boldsymbol{s}_B$ as a test set to predict the corresponding $\boldsymbol{s}_{EZ}$.

As shown in Figure 3, the correlation coefficient between the true $\boldsymbol{s}_{EZ}$ and $\hat{\boldsymbol{s}}_{EZ}$ predicted by $\boldsymbol{s}_{BF}$ was higher than 0.9. This empirical result is not surprising (Radev et al., 2020). Yet, it convincingly demonstrates that summary networks learn a representation of the data that aligns with the information used by analytical methods.

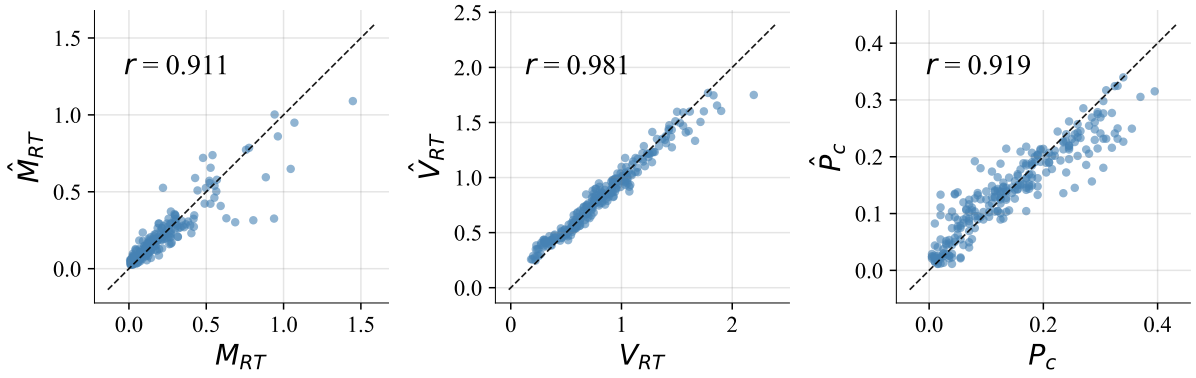


Figure 3: Predicted sufficient summary statistics $\hat{\boldsymbol{s}}_{EZ}$ from $\boldsymbol{s}_B$. The x-axis are the observed values of M_{RT}, V_{RT}, and P_c, and the y-axis is the $\hat{M}_{RT}$, $\hat{V}_{RT}$ and $\hat{P}_c$ predicted by $\boldsymbol{s}_B$

.

The Impact of Outliers in Amortized Bayesian Inference

Next, we assessed the robustness of ABI with the stylized sensitivity curve (SSC, Andrews et al., 1972) and the breakdown point (BP; Donoho & Huber, 1982). To the best of our knowledge, this is the first study to apply these in the context of ABI.

The Stylized Sensitivity Curve

When fitting parametric models, the data are assumed to follow a distribution $F_{\boldsymbol{\theta}}$ with unknown parameters $\boldsymbol{\theta}$. Bayesian methods return an approximation to the posterior $p(\boldsymbol{\theta} \mid \boldsymbol{x})$, which can be summarized into a point estimate $\hat{\boldsymbol{\theta}}$ (e.g., the posterior mean), and the dependence of the estimator on $\boldsymbol{x}$ is denoted as $\hat{\boldsymbol{\theta}}(\boldsymbol{x})$. If the sample size n becomes very large (i.e., $n \to \infty$), the information in the sample $\boldsymbol{x}$ converges to the information provided by the distribution F, and thus $\hat{\boldsymbol{\theta}}(F) \to \boldsymbol{\theta}$, assuming the estimator is consistent. However, contamination may be present. Denoting the contamination distribution as G and the (small) fraction of contaminants as $\epsilon \in [0, 1]$, we can define the *contaminated distribution* $F_{\boldsymbol{\theta}}^{\epsilon,G}$ as follows:

$$F_{\boldsymbol{\theta}}^{\epsilon,G} = (1 - \epsilon)F + \epsilon\, G. \tag{5}$$

In the study of the influence of outliers, the contaminant distribution is usually defined as a point mass $G \equiv \delta_{x^c}$, with

$$\delta_{x^c} = \begin{cases} 1 & \text{if } x = x^c \\ 0 & \text{otherwise,} \end{cases} \tag{6}$$

where x^c is the contaminant. Thus, we can define the influence function (IF; Maronna et al., 2006) for an estimator $F(\hat{\boldsymbol{\theta}})$ and a point mass contaminant distribution at x^c as:

$$\mathrm{IF}_{\hat{\boldsymbol{\theta}}}(x^c, F) = \lim_{\epsilon \to 0} \frac{\hat{\boldsymbol{\theta}}\big((1 - \epsilon)F + \epsilon\delta_{x^c}\big) - \hat{\boldsymbol{\theta}}(F)}{\epsilon}. \tag{7}$$

The IF indicates how much influence a contaminant x^c exerts on the estimate $\hat{\boldsymbol{\theta}}$ asymptotically. The IF can be traced for a range of x^c values, resulting in an *influence curve* where the x-axis depicts the value of x^c, and the y-axis depicts the influence of x^c on $\hat{\boldsymbol{\theta}}$.

Finding a closed-form expression for the IF of an estimator of an arbitrary F is hard in practice. Instead, we can compute the stylized sensitivity curve (SSC, Andrews et al., 1972) to approximate IF through simulations. The bias a contaminant x^c introduces to an estimate is given by $\hat{\boldsymbol{\theta}}(\boldsymbol{x}, x^c) - \hat{\boldsymbol{\theta}}(\boldsymbol{x})$, where $\hat{\boldsymbol{\theta}}(\boldsymbol{x}, x^c)$ is the estimate calculated with the original sample $\boldsymbol{x}$ plus the contaminant x^c (and thus having n+1 as sample size [1]). Since the bias inherently depends on a specific finite sample $\boldsymbol{x}$, using a single simulated sample would introduce random variability. To mitigate this and obtain a more reliable

[1] While dividing this bias with $\frac{1}{(n+1)}$ in large samples yields the asymptotic bias of x^c, we omit this denominator to allow for sample size dependence.

representation, *pseudo data* are being used. Pseudo data consist of the quantiles of the underlying distribution F, defined as $x_i = F^{-1}\left(\frac{(i-0.5)}{n}\right)$. When an analytical form for F is unavailable, these quantiles are found by simulating a large number of observations from the observation model to accurately approximate F, and then computing the empirical quantiles. This approach ensures a highly representative base sample, free from single-sample randomness, and is called "stylized"(Andrews et al., 1972).

Furthermore, as the underlying distribution F depends on the true parameter $\boldsymbol{\theta}$, the precise shape of the SSC0 depends on these parameters. To obtain a reliable estimate of SSC across plausible parameter spaces, one can average the SSCs over B distinct sets of pseudo data, each generated based on different $F_{\boldsymbol{\theta}}$, with $\boldsymbol{\theta}$ drawn from a prior or varying across a plausible range:

$$\overline{SSC}(x^c) = \frac{1}{B}\sum_{b=1}^{B}\left[\hat{\boldsymbol{\theta}}(\boldsymbol{x}^b, x^c) - \hat{\boldsymbol{\theta}}(\boldsymbol{x}^b)\right], \tag{8}$$

with $\boldsymbol{x}^b$ being the bth pseudo data. By varying the value of x^c, we can plot an expected SSC that illustrates the relationship between the x^c value and the impact it has on estimates.

The Breakdown Point

The BP is the minimum amount of contamination that can lead an estimator to produce extremely aberrant values (Donoho & Huber, 1982). Assume that Θ is parameter space for the parameter vector $\boldsymbol{\theta}$ (i.e., $\boldsymbol{\theta} \in \Theta$). The asymptotic contamination BP, denoted $\epsilon^\star$, for an estimator $\hat{\boldsymbol{\theta}}$ at the distribution F is defined as the largest $\epsilon^\star \in [0, 1]$ such that for $\epsilon < \epsilon^\star$, $\hat{\boldsymbol{\theta}}\big((1-\epsilon)F + \epsilon G\big)$ remains bounded away from the boundary of Θ, regardless of the contaminating distribution G (Maronna et al., 2006). What this means is that the contamination should not drive the estimated parameter $\hat{\boldsymbol{\theta}}$ to the boundary of Θ (or to infinity if Θ is unbounded for some or all of its components). A high BP means that $\epsilon^\star$ is large and thus that $100\epsilon^\star$ of the sample can be contaminated while leaving the estimate still bounded. For example, the BP of the sample mean (as an estimator of the population mean) is 0%, while the BP of the median is 50%.

In a practical setting with a finite sample size n, we define a fraction $p = \frac{m}{n}$, so that $m = p \cdot n$ observations in the sample $\boldsymbol{x}$ are replaced by x^c (without loss of generality, we replace the first m entries, as the models in this paper assume IID observations). The contaminated sample can be denoted as $\boldsymbol{x}^{c(p)}$. This leads to an estimate $\hat{\boldsymbol{\theta}}(\boldsymbol{x}^{c(p)})$. Again, the sample average over B samples can be taken: $\overline{\hat{\boldsymbol{\theta}}}(x^c, p)$. The BP is then the fraction p^* for which all fractions $p < p^*$ lead to an average estimate that is still acceptable: $|\overline{\hat{\boldsymbol{\theta}}}(x^c, p) - \overline{\hat{\boldsymbol{\theta}}}| < \Delta$ (where Δ is some tolerance). Importantly, in this paper, we only studied point contamination (i.e., $G = \delta_{x^c}$), resulting in an approximate version of BP.

The Impact of Outliers in Amortized Estimation of the Mean of a Normal Distribution

We first studied the SSC and BP in a toy example, the estimation of the mean μ, where n IID samples were drawn from:

$$x_1, x_2, \ldots, x_n \overset{\text{IID}}{\sim} \mathcal{N}(\mu, 1), \tag{9}$$

so that $\boldsymbol{x} = \{x_1, \ldots, x_n\}$. The parameters $\{\mu_j\}_{j=1}^{J}$ were sampled from a prior $\mathcal{N}(0, 1)$, and for each μ_j, 10 to 100 samples from $\mathcal{N}(\mu_j, 1)$ were generated. The neural network architecture consisted of a Deep Set network with two outputs (Zaheer et al., 2017), and an inference network comprising two affine coupling layers. The training ran for 10 epochs (4000 iterations per epoch, batch size 32) using the Adam optimizer (initial LR $= 5 \times 10^{-4}$, cosine decay schedule). After training, parameter recovery was performed to confirm that the estimated posterior mean aligned almost perfectly with the analytical posterior mean (Gelman et al., 2013).

Stylized Sensitivity Curve of the Amortized Estimator of the Mean

We assessed the contaminant's impact using the SSC for sample sizes $n \in \{10, 20, 100\}$, with x^c systematically from –100 to 100 in steps of 1. For each x^c, 100 parameters and 100 data sets (1000 observations per data set) were simulated. From each of these larger data sets, we derived smaller pseudo data sets $\boldsymbol{x}$ by extracting n quantiles using the aforementioned $\frac{i-0.5}{n}$ rule (with $i = 1, \ldots, n$). We then introduced the contaminant x^c to these pseudo data sets and computed the average change in estimated μ.

This toy example allows an analytical expression for SSC. Since the prior and likelihood are both normally distributed, the difference in posterior mean between $\boldsymbol{x}^c = (\boldsymbol{x}, x^c)$ and $\boldsymbol{x}$ is:

$$\begin{aligned}
\hat{\mu}(\boldsymbol{x}, x^c) - \hat{\mu}(\boldsymbol{x}) &= \frac{1}{n+2}(x_1 + \ldots + x_n + x^c) - \frac{1}{n+1}(x_1 + \ldots + x_n) \\
&= \frac{1}{n+2}x^c + \frac{1}{n+2}\sum_{i=1}^{n} x_i - \frac{1}{n+1}\sum_{i=1}^{n} x_i.
\end{aligned}$$

This demonstrates a linear relationship between x^c and the theoretical SSC, with slope $\frac{1}{n+2}$. Since the prior is centered around zero, the intercept also takes the value of zero. The theoretical SSC and that obtained from ABI are shown in the left panel of Figure 4. This indicates that the neural estimator behaves identically to the analytical posterior mean in the presence of an outlier.

Breakdown Point of Amortized Mean Estimator

We estimate the BP using an extreme point contaminant $x^c = -100$, replacing a fraction $p \cdot n$ of observations to create contaminated samples $\boldsymbol{x}^{c(p)}$. For each p and $n \in 10, 20, 100$, we simulate 500 data sets and compute the average estimate $\overline{\hat{\mu}}(\boldsymbol{x}^{c(p)})$, which is then plotted against p. Again, the BP is available analytically for this simple example:

$$\begin{aligned}\hat{\mu}(\boldsymbol{x}^{c(p)}) &= \frac{1}{n+1}(p \cdot n \cdot x^c + \sum_{i=n-p\cdot n}^{n} x_i) \\ &= \frac{n}{n+1} p x^c + \frac{1}{n+1} \sum_{i=n-p\cdot n}^{n} x_i,\end{aligned}$$

such that averaging over the data $\boldsymbol{x}$ and the parameter μ yields:

$$\overline{\hat{\mu}}(\boldsymbol{x}^c, p) = \frac{n}{n+1} p x^c$$

Thus, even for the smallest fraction $p = \frac{1}{n}$, the estimate can be made arbitrarily large by choosing an appropriate value for x^c.

As can be seen from the right panel in Figure 4, the ABI estimation of μ depended linearly on the fraction of outliers in the sample (as also our derivation has shown). Therefore, it is no surprise that the results from the theory and ABI align closely.

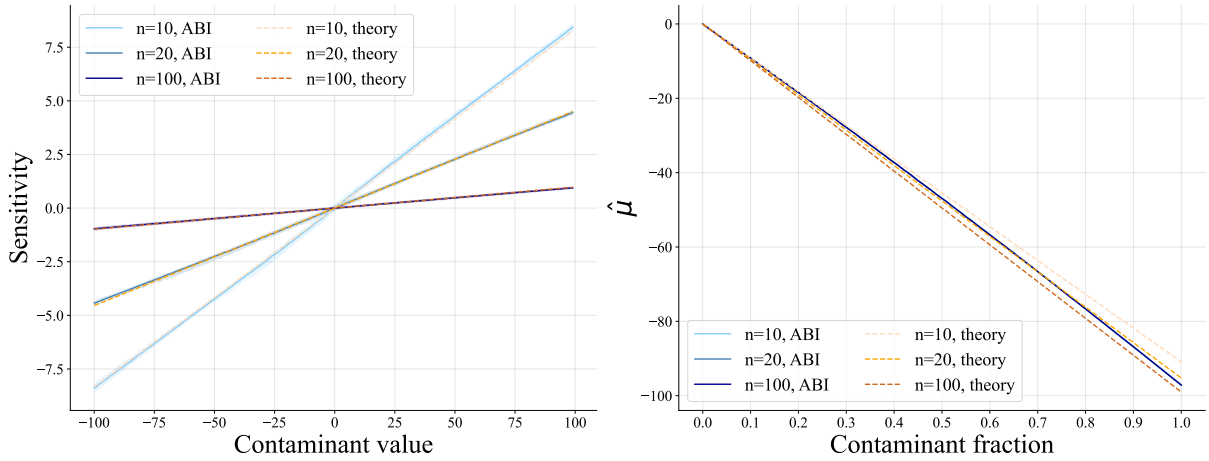


Figure 4: The SSC and BP for the μ estimator when $n \in \{10, 20, 100\}$. The left panel displays the SSC for estimating μ. For reasons of comparison, both the theoretical and the ABI-based SSC are shown. The shaded areas around the ABI-based SSC lines indicate the 2.5% to 97.5% percentiles of the influence. The right panel shows the average estimate of μ with a fraction of p outliers (i.e., copies of $x^c = -100$) inserted into the sample, that is, $\overline{\hat{\mu}}(\boldsymbol{x}^c, p)$.

The Impact of Outliers in Amortized DDM Parameter Estimation

Specification of the ABI for Estimating DDM Parameters

We simulated choice reaction time data to train the NPE, using the priors listed in Table A1. The observation model comprised two conditions with opposite drift rates (v_1, v_2), while other parameters remained fixed in every trial. The number of trials n in each data set varied from 100 to 1000, allowing us, after training, to make inferences on data sets with a wide range of trial numbers.

The summary network was a Set Transformer (Lee et al., 2019) that output 12 summary statistics, while the inference network is a neural spline flow composed of six coupling layers (Durkan et al., 2019). The training process consisted of 100 epochs (1000 iterations/ epoch, batch size of 32) using the Adam optimizer (initial LR $= 5 \times 10^{-4}$, cosine decay schedule). The loss value converged successfully by the end of the training. Next, we simulated an additional 500 data sets and performed parameter recovery (see row (a) in Figure A4).

Stylized Sensitivity Curve of Amortized DDM Estimator

We introduced contaminants in the choice reaction time data as a pair (x^c, y^c). For each contaminant, the response $y^c \sim$ Bernoulli(0.5). This randomization served two purposes: (a) it isolated the effect of x^c by averaging out choice influence, and (b) it mirrored the real-world scenario where outliers often arise from random guesses.

To evaluate our estimator, we systematically varied x^c from 0.01 to 25 seconds in 0.05-second increments. For each x^c value, we generated 100 representative pseudo data sets, each containing 200 trials (100 trials/condition). Our simulation process involved two main steps: (a) parameter and data set simulation: We first simulated 100 parameter sets (v_1, v_2, a, z, T_{er}), then simulated 100 full data sets, each consisting of 5000 trials (2500 trials/ condition). (b) Pseudo-data set generation from quantiles: For each simulated full data set, we calculated the error response rates (e_1 and e_2) for both conditions. These rates allowed us to calculate the approximate number of error trials needed for our pseudo-data sets (rounded to the nearest integer, i.e., $100 \times e_1$ and $100 \times e_2$). We then constructed the 200-trial pseudo-data sets by extracting specific quantiles from the simulated DDM response distributions. For the error trials, we took the first up to the $(100 \times e_1)$th quantile from the error reaction time distribution for condition 1, and similarly for condition 2. For the correct trials, we took the first up to the $(100 - 100 \times e_1)$th quantile from the correct reaction time distribution for condition 1, and similarly for condition 2. Lastly, a contaminating pair (x^c, y^c) was added to each of these pseudo-data sets, specifically within condition 1.

Row (a) in Figure 5 shows the SSC for the DDM parameters. To provide a more fine-grained analysis in the short outlier value range, as they are expected to be particularly detrimental, we also plot the SSC of x^c ranging from 0 to 1 second, with an increment

of 0.01 seconds as in row (b) of Figure 5. While the solid blue lines represent the SSC averaged over 100 pseudo-data sets, the shaded areas indicate the 2.5% to 97.5% percentiles of the influence. The results showed that even a single contaminant can substantially affect DDM parameter estimates.

When examining the SSC, short contaminants ($< 1s$) consistently led to two primary effects: first, an underestimated non-decision time T_{er}, and second, an overestimated boundary separation a (see row (b) in Figure 5). This is likely because an underestimated T_{er} prolongs the "decision time", which means more information is needed to reach a decision. While these patterns remained consistent across all 100 pseudo-data sets, the contaminant's impact on drift rates and starting point varied, possibly due to the random sampling of y^c. Longer contaminants ($> 1s$) have less of an effect on T_{er}. While they still affect other parameters—particularly the drift rate v of the first condition—since the altered trial extends the mean reaction time of the first condition, again implying slower evidence accumulation. Overall, both short and long outliers significantly distort DDM estimates.

Breakdown Point of Amortized DDM Estimator

To explore the BP of the NPE, we introduced a fraction p of extreme contaminants x^c into the first condition, assigning random responses $y^c \sim \text{Bernoulli}(0.5)$. For each p, 500 data sets (100 observations per condition per data set) were simulated and fitted. We assessed the estimates by comparing the average posterior mean from perturbed data sets to that from the original data set. When the estimate converges to the prior mean (Table A1), shown as the red dashed line in rows (c) and (d) in Figure 5, the estimator is considered to produce reasonable estimates; otherwise, it is considered to have broken down.

Rows (c) and (d) in Figure 5 show the BP results for short outliers ($x^c = 0.01s$) and long outliers ($x^c = 20s$), respectively. With more $x^c = 0.01s$ in the first condition, the drift rates in both conditions and the boundary separation approached zero. With more $x^c = 20s$ in the first condition, v_1 was estimated to be near zero, v_2 becomes more extreme, and the boundary separation approaches the upper boundary of its prior. In both cases, the estimation became severely distorted starting at $p^* = \frac{1}{n}$, which was identified as the BP. Notably, multiple short outliers even yielded negative posterior means for T_{er}, exceeding its valid range (row (c) in Figure 5). These results highlight the estimator's high sensitivity to contamination.

Robust Amortized Bayesian Inference: Training with Perturbed Data

Our results show that estimators trained on data strictly conforming to model assumptions were highly sensitive to contamination. However, robustness can be improved by including contaminants in the training data. Throughout the remainder of this paper, we refer to

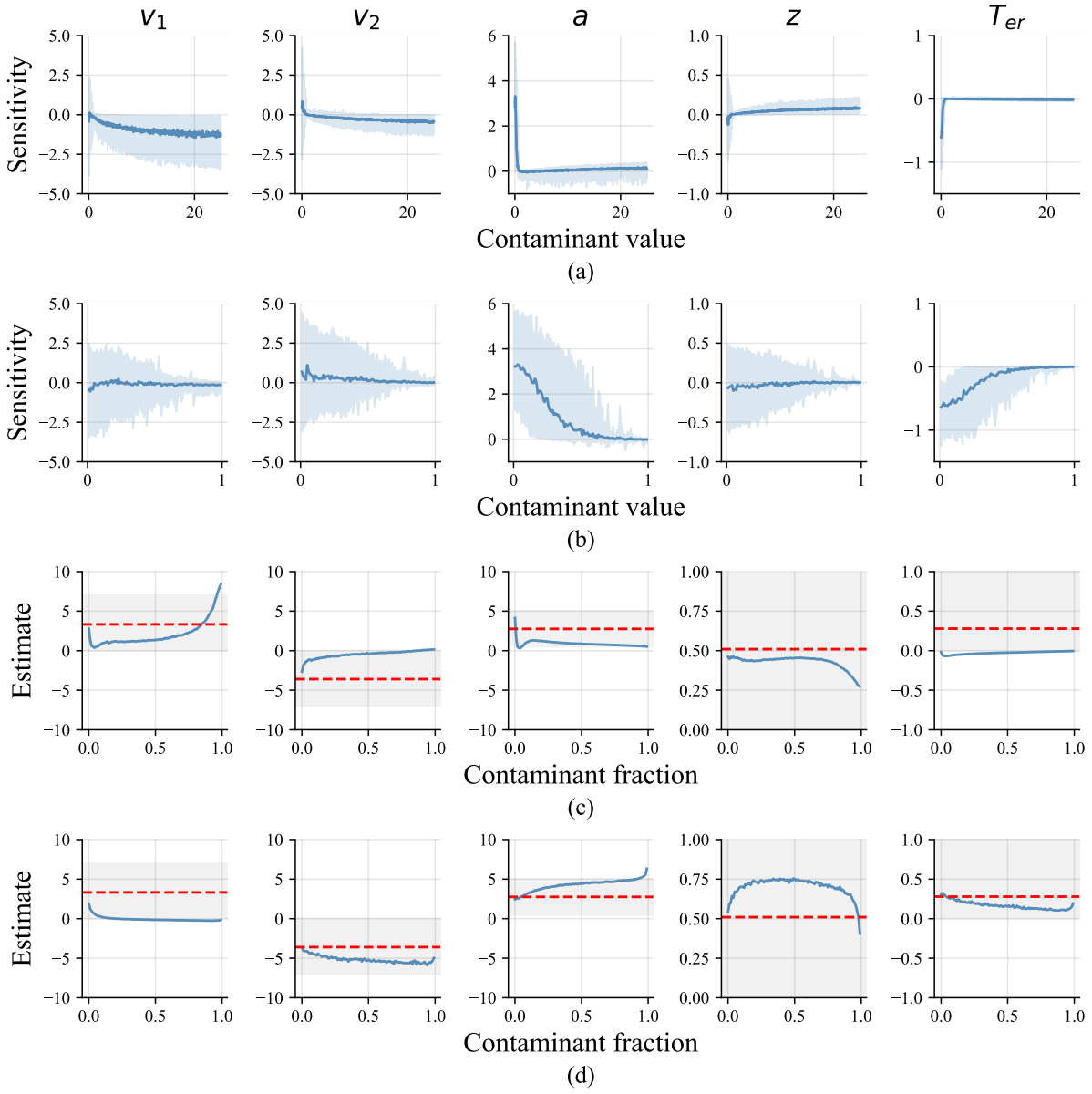


Figure 5: SSC and BP of amortized DDM estimation. Row (a) shows the SSC when x^c ranges from 0.01s to 25s. Row (b) zooms in on the SSC when $x^c = 0.01 \sim 1s$, with a higher resolution. Row (c) shows the BP of the estimator when $x^c = 0.01s$, while row (d) shows that with large outliers, namely $x^c = 20s$. The true posterior mean across 1000 data sets is the prior mean, which is indicated with the red dashed line. The grey area marks the prior density for this parameter. The deviation between the estimates of perturbed data sets (in the blue line) and the red dashed line represents the systematic bias introduced by a certain fraction of outliers.

the estimator trained on clean data as the *standard estimator* and the one trained with contaminated data as the *robust estimator*. While this approach is conceptually simple, key questions remain, such as (a) whether it truly enhances robustness and (b) how different contamination types affect parameter estimates. We address these questions using both a toy example and the DDM.

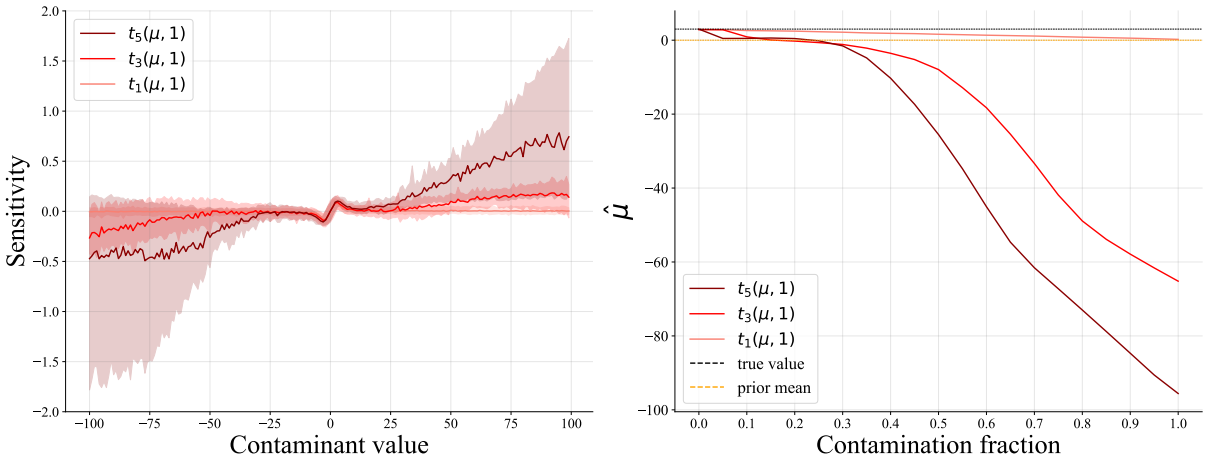


Figure 6: SSC and BP of robust neural estimators of μ trained with different degrees of freedom ν.

Robust Amortized Estimation of the Mean of a Normal Distribution

Training data of robust estimators $\boldsymbol{x}^c$ in the toy example came from a contaminated observation model:

$$\boldsymbol{x}^c \sim \begin{cases} \mathcal{N}(\mu, 1), & \text{with probability } 1 - \pi \\ t_\nu(\mu, 1), & \text{with probability } \pi, \end{cases} \tag{10}$$

where $t_\nu(\mu, 1)$ refers to a t distribution with location μ, scale 1, and degrees of freedom ν, and π is a predefined contamination probability that we have set to 0.1. In the remainder, we train three estimators with $\nu \in \{1, 3, 5\}$, respectively.

Three robust estimators were trained using the same settings as the standard estimator. After training, 500 new data sets were simulated to evaluate parameter recovery for each estimator.

Robustness of the estimators was evaluated using SSC and BP, following the same procedure as for the standard estimator ($n = 20$). Results are shown in the left panel of Figure 6, to be compared with Figure 4 (note the differing y-axis scales). The shaded areas around SSCs indicate the 2.5% to 97.5% percentiles of the influence in the 500 data sets. Near the origin, the SSC shows a linear dependence on x^c; for more extreme values, the influence is down-weighted. The t_1 robust estimator effectively suppressed influence across the entire range, while t_3 and t_5 estimators show increased sensitivity to extreme outliers—likely due to extrapolation, as such values were absent during training. They also exhibited greater variability in the sensitivity across 100 pseudo data sets, as evidenced by the significantly wider shaded areas for the t_3 and t_5 compared to the t_1 estimator. Table 3 lists the proportion of regular and far outliers (per definition in Tukey (1977)) for each contamination distribution. Under 10% contamination, the t_1 distribution produces substantially more regular and far outliers than the others.

Interestingly, the behavior of the neural robust t_1 estimator aligned with Tukey's

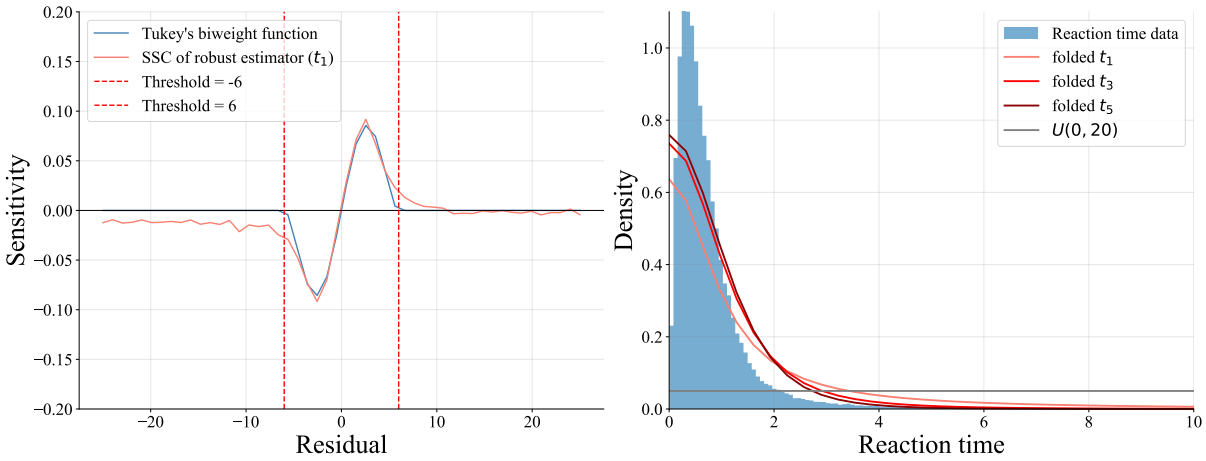


Figure 7: Tukey's biweight function and reaction time data. The left panel overlays the SSC of Tukey's biweight loss function and that of the robust μ estimator with t_1. The right panel shows the histogram of reaction times generated by the DDM for 500 prior predictive realizations, overlaid with the densities of the four contamination distributions.

biweight loss function from robust statistics (Tukey, 1979). The IF of Tukey's biweight loss function is:

$$\text{IF}_{\hat{\mu}}(x^c, F) = \begin{cases} (1 - (x^c/k)^2)^2, & \text{if } |x^c| \leq k \\ 0 & \text{if } |x^c| > k, \end{cases} \tag{11}$$

where k is a tuning constant or a cutoff value that ensures the estimator is insensitive to contaminants in absolute value larger than k. When we overlaied the graph of the robust t_1 estimator's SSC with Tukey's biweight loss function (using $k = 6$), as seen in the left panel of Figure 7, their significant overlap suggested the NPE functioned similarly to traditional robust estimators developed under the MLE framework.

We also assessed the BP of the robust estimators by contaminating data sets with increasing fractions p of outliers ($x^c = -100$) and computing the average estimate $\hat{\mu}$ across 500 data sets per estimator. To obtain meaningful BP comparisons, we set the true mean of the validation data sets to be $\mu = 3$ rather than using a prior centered at zero. Otherwise, reversion to the prior mean during breakdown would go undetected. An estimator that is robust to contaminants should show an average estimated mean $\hat{\mu}$ close to the true mean $\mu = 3$. As shown in the right panel of Figure 6, the t_1-trained estimator demonstrates the greatest resistance, with robustness decreasing as ν increases. When all values were replaced with $x^c = -100$, the estimates converged to zero (i.e., the prior mean), indicating the network no longer extracted information from the data.

Overall, assuming a contaminated observation model when training the NPE significantly enhanced the robustness in estimating μ, and t_1 performed best among our candidate contamination distributions.

Estimator		Regular outliers $Q_3 + 1.5 \cdot IQR$	%	Far outliers $Q_3 + 3 \cdot IQR$	%
Robust	normal + 10% t_1	2.828	2.585	4.949	1.269
	normal + 10% t_3	2.734	1.280	4.785	0.174
	normal + 10% t_5	2.719	1.008	4.758	0.051
Standard	normal	2.698	0.698	4.721	0.0002

Table 3: Percentage of regular and far outliers under three contaminated simulators (with $\pi = .1$) and the standard normal simulator. Regular outliers are data points beyond 1.5 times the interquartile range (IQR) from the lower and upper quartiles (i.e., Q_1 and Q_3, respectively), while far outliers are those beyond 3 times the IQR from the quartiles (Tukey, 1977). This table shows the percentage of regular and far outliers in different simulators, along with the corresponding thresholds. Only the upper boundaries are shown because the distributions we use in simulators are symmetric.

Robust Amortized DDM Estimation

For robust DDM estimation, we produced contaminated data $(\boldsymbol{x}^c, \boldsymbol{y}^c)$ as follows:

$$(\boldsymbol{x}^c, \boldsymbol{y}^c) \sim \begin{cases} \text{Wiener}(v, a, z, T_{er}), & \text{with probability } 1 - \pi \\ G, & \text{with probability } \pi \end{cases} \tag{12}$$

with $\pi = 0.1$. We tested four different contamination distributions G: x^c was drawn from a folded-t_ν distribution (i.e., the absolute value of t distributed random variable) or from a uniform distribution $U(0, 20)$:

$$x_i^c \sim \begin{cases} \text{folded-}t_1 & \text{or} \\ \text{folded-}t_3 & \text{or} \\ \text{folded-}t_5 & \text{or} \\ U(0, 20). \end{cases}$$

The corresponding response was also contaminated $y_i^c \sim \text{Bern}(0.5)$, implying random guessing.

The right panel of Figure 7 shows the histogram of perfectly clean reaction time data (for a variety of parameter vectors; in blue) overlaid with the density of the different contamination distributions.

Four robust estimators were trained using the exact sample sizes, network architecture, and training epochs as the standard DDM estimator. All robust estimators converged and accurately recovered parameters (see row (b) to (e) in Figure A4).

We then evaluated the performance of robust DDM estimators using the SSC. Overall, Figure 8 shows that all four robust estimators demonstrate resistance to both short and long outliers (Figure 8 should be compared to Figure 5). Among all the t-based robust estimators, a lower ν resulted in: (a) average sensitivity closer to zero, indicating higher robustness, (b) lower variability in sensitivity across 100 pseudo-data sets. This means

that a lower ν led to reduced sensitivity to contaminants, especially when x^c was large. Compared to the robust estimator with folded t_1, the estimator with $U(0, 20)$ showed greater sensitivity towards short outliers between $0 - 1$s, and the variability in estimated difference was higher (Figure 9). This is possibly because the t-based robust estimator assigned higher weights to short contaminants than the $U(0, 20)$ robust estimator (see the right panel in Figure 7). Yet, interestingly, the $U(0, 20)$ robust estimator exhibited resistance to out-of-training x^c values (e.g., 20s to 25s), suggesting it effectively learned to filter out large values.

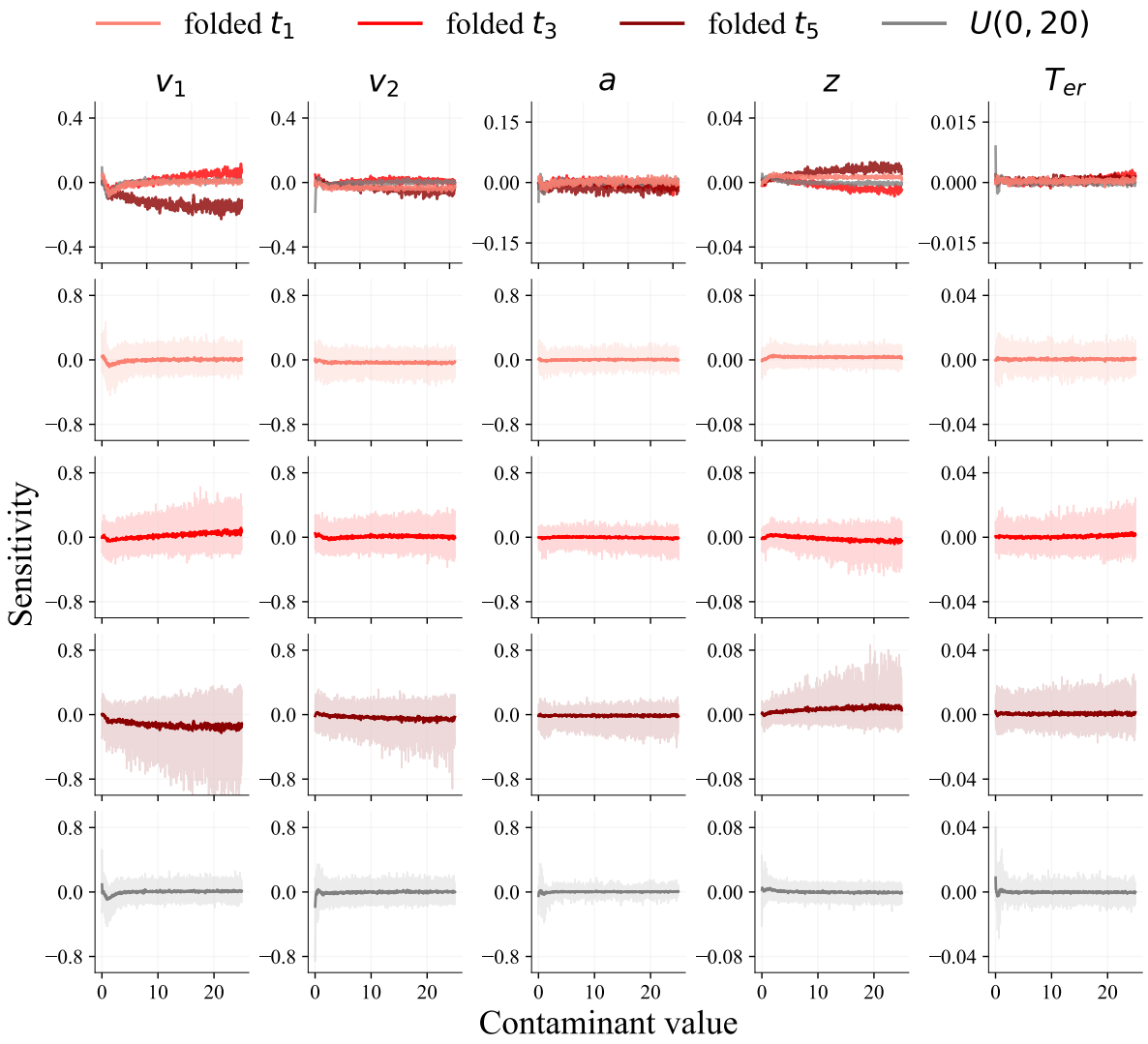


Figure 8: The SSC of robust DDM estimators with x^c ranges from 0 to 25s. The first row provides a comparative overview of the SSC for four distinct robust estimators. The last four rows show the variability in sensitivity across the 100 pseudo-data sets in four robust estimators, respectively.

The breakdown properties of the robust estimators are further illustrated in Figure 10, using $x^c = 0.01$s and $x^c = 20$s across contamination levels $p \in [1/100, 100/100]$ in the first condition. When $x^c = 0.01s$, the uniform estimator broke down earlier than the t-based estimators (see the upper panel in Figure 10). This difference arose because the folded-t distributions assigned more weight near zero than the $U(0, 20)$. Within the t-based robust

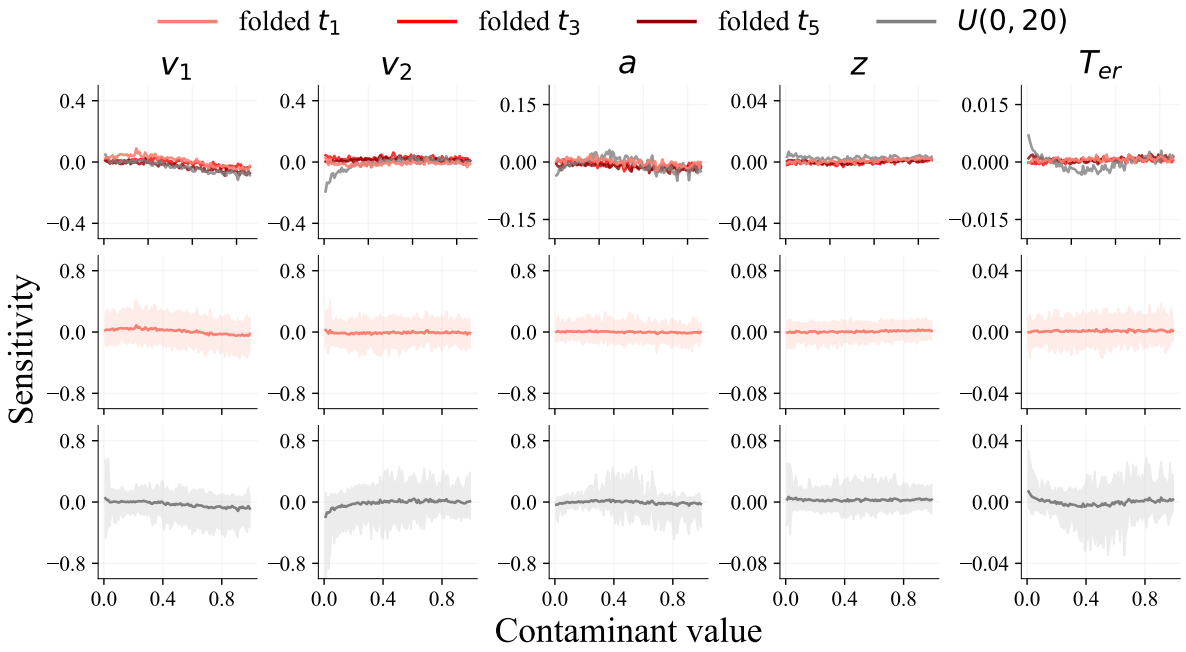


Figure 9: The SSC of robust DDM estimators with x^c ranges from 0 to 1s. The first row provides a comparative overview of the SSC for four distinct robust estimators. The second row is the variability in sensitivity across 100 pseudo-data sets in the robust estimator with folded t_1, and the third row shows that in the $U(0, 20)$ estimator. The plots for the folded t_3 and t_5 robust estimators are omitted, as they are highly similar to the folded t_1.

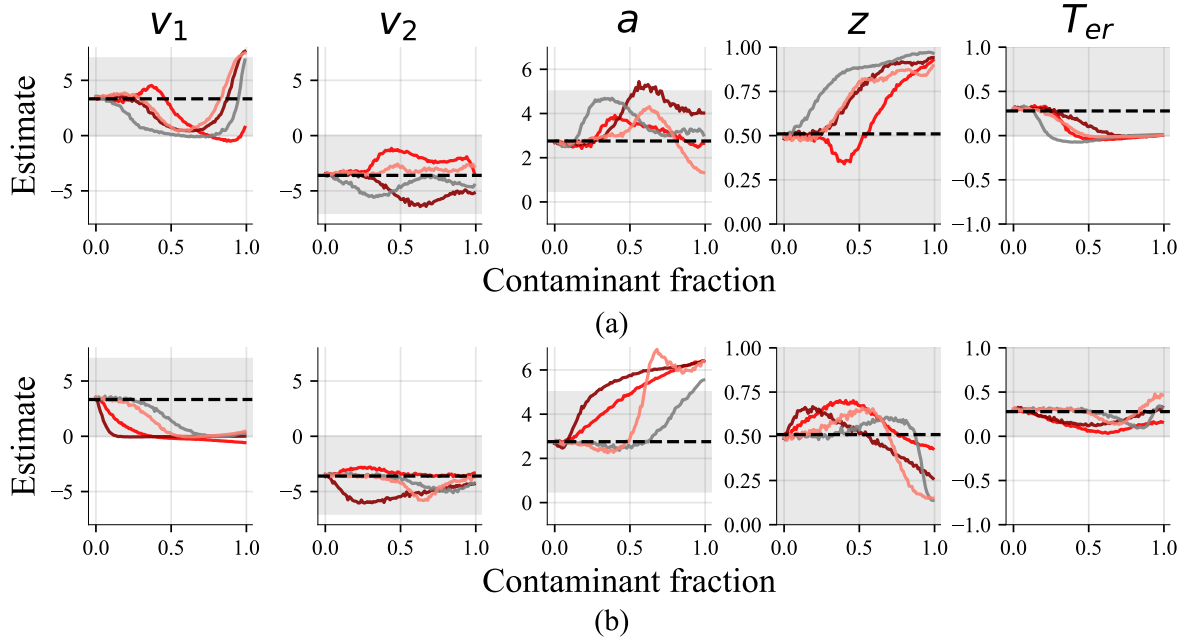


Figure 10: BP of robust DDM estimators (applied to data sets with two conditions and 100 trials per condition). Panel (a) displays the BP of the estimator when $x^c = 0.01s$, while panel (b) shows the BP with large outliers ($x^c = 20s$). The true posterior mean across 1000 data sets is indicated with the black dashed line. The grey area marks the prior density for this parameter. The deviation between the estimates of perturbed data sets and the black dashed line is the systematic bias that a certain fraction of outliers brings.

estimators, there was no significant difference in BP, and the average estimate deviated from the prior mean when there was around 25% contamination. This was because the weights that folded- t_1, t_3, and t_5 assigned to short contaminants were similar. When $x^c = 20s$, the robust estimator with $U(0, 20)$ exhibited the highest BP. Among all the t-based estimators, the lower the ν, the higher the BP, as lower ν indicates higher weights on long contaminants. Overall, for DDM, the t_1-based estimator offered the highest BP (approximately 25%) in both short and long point contamination scenarios.

In summary, training with a contaminated observation model improved the robustness of DDM estimators. The choice of contamination distribution had a critical impact on robustness. For t-based robust estimators, lower ν led to higher robustness and less variability in the influence from one contaminant, especially for large outliers. Meanwhile, the t_1-based estimator generally offered the most robust performance across both short and long point contamination scenarios. The $U(0, 20)$-based estimator showed higher sensitivity to short contaminants, while exhibiting resistance to large, out-of-training contaminants. It broke down early for short outliers but displayed the highest BP for long ones. Based on these results, the t_1-based robust estimator is considered the best candidate for contamination distribution, and additional exploratory simulations are presented in the Appendix, varying the contamination probability π under the t_1 distribution.

The Cost of Robustness

It is well understood that any robust approach comes with costs (Maronna et al., 2006). For example, robust M-estimators often exhibit efficiency losses, which refer to a higher asymptotic variance of the robust estimator compared to the standard estimator in uncontaminated data estimation (Hampel et al., 2005). We explored the cost of the robust NPE, evaluating (for the jth parameter θ_j) the accuracy and efficiency of a robust estimator $\hat{\theta}_j^R$ compared to a standard estimator $\hat{\theta}_j^S$.

As a measure of accuracy, we computed the ratio of mean absolute error (MAE) as follows:

$$\text{MAE Ratio}_j^{S,R} = \frac{1}{B}\sum_{b=1}^{B}\frac{|\hat{\theta}_j^S(\boldsymbol{x}_b) - \theta_{j(b)}|}{|\hat{\theta}_j^R(\boldsymbol{x}_b) - \theta_{j(b)}|}, \tag{13}$$

where $\boldsymbol{x}_b$ is the bth data set simulated from the standard model, $\theta_{j(b)}$ is the true jth parameter value for bth data set, and B is the total number of data sets.

For efficiency loss, we used the ratio of average posterior variance:

$$\text{Posterior Variance Ratio}_j^{S,R} = \frac{1}{B}\sum_{b=1}^{B}\frac{\text{var}(\hat{\theta}_j^S \mid \boldsymbol{x}_b)}{\text{var}(\hat{\theta}_j^R \mid \boldsymbol{x}_b)}, \tag{14}$$

where $\text{var}(\hat{\theta}_j \mid \boldsymbol{x}_b)$ is the variance of the marginal posterior distribution of the jth parameter based on the bth simulated data set $\boldsymbol{x}_b$.

The Cost of Robustness in the Estimation of the Mean

We simulated 20,000 clean data sets ($n = 20$) and fit them using three robust estimators, comparing each to the standard estimator. The MAE ratio between t_1, t_3, t_5 and the standard estimator is 0.999, 0.998, 0.999, respectively, and the posterior variance ratio is 0.909, 0.943, and 0.968, respectively. The MAE ratios were close to 1, indicating minimal accuracy loss (0.1–0.2%). However, variance ratios revealed notable efficiency loss, with the t_1-based estimator showing the largest (9.0%). This aligned with the efficiency loss observed in M-estimators for finite samples ($n = 20$) when estimating μ in a univariate normal distribution (Wu, 1985). For example, Hampel, Huber, and Tukey's biweight ($k = 6$) M-estimators exhibited efficiency losses of 4.8%, 17.0%, and 11.2% compared to the optimal least-squares estimator, respectively. Our robust estimators with t_1, t_3, and t_5 showed 9.0%, 5.5%, and 3.3% efficiency losses compared to a standard neural μ estimator, respectively. This highlights the t_1 estimator's strong robustness with reasonable efficiency. As ν increases, efficiency improves—illustrating the trade-off between robustness and efficiency, consistent with SSC/BP results in Figure 6.

The Cost of Robustness in DDM Estimation

We simulated 10000 data sets ($n = 100$) and calculated the MAE ratios and posterior variance ratios compared to the standard estimator. The results are shown in Tables 4 and 5, respectively. As expected, the accuracies were lower for all the robust estimators (e.g., depending on the parameter, about 12% to 24% loss in accuracy for the robust estimator based on the t_1 distribution) compared to the standard one. No systematic relationship between ν and the performance of the estimators was detected. The robust estimator with $U(0, 20)$ exhibited less accuracy loss (from around 6% to 15% for the five parameters).

All robust estimators showed moderate to high efficiency loss, with up to 43% for T_{er} in the folded-t_1 model. However, this loss is not directly comparable to that in estimating μ due to the DDM's greater complexity and parameter interdependence. The robustness–efficiency trade-off can be adjusted by lowering the contamination probability π during training, reducing efficiency loss (see Table A2,A3, and A4). However, as shown in Figure A6 and the Appendix, decreasing π also lowered the estimator's BP.

Table 4: MAE ratios of the four robust neural DDM estimators. Each row compares the robust estimator to the standard estimator.

Robust estimator	v_1	v_2	a	z	T_{er}
t_1	0.76	0.79	0.769	0.88	0.792
t_3	0.731	0.722	0.728	0.865	0.764
t_5	0.728	0.74	0.734	0.862	0.733
$U(0, 20)$	0.876	0.894	0.907	0.936	0.848

Table 5: Posterior variance ratios of the four robust neural DDM estimators. Each row compares the robust estimator to the standard estimator.

Robust estimator	v_1	v_2	a	z	T_{er}
t_1	0.613	0.596	0.621	0.732	0.571
t_3	0.637	0.569	0.634	0.694	0.581
t_5	0.618	0.569	0.672	0.713	0.568
$U(0, 20)$	0.723	0.672	0.723	0.756	0.636

Real Data Example

To demonstrate the practical utility of our robust approach, we applied it to the data set from Ratcliff & Rouder (1998), available as `rr98` in the `RTDists` R package (Singmann et al., 2022). We fit both the raw and outlier-cleaned versions of the data using standard and robust estimators to compare their estimates.

In this task, three participants were asked to decide whether the pixel arrays were "bright" or "dark". The experiment was a 33×2 within-subject design: brightness strength (0–100% white pixels across 33 levels) and instruction (speed vs. accuracy). This setup targeted the speed–accuracy trade-off in decision-making. Each participant completed approximately 4,000 trials per instruction condition. To simplify analysis, the 33 brightness levels are grouped into five bins. we fit separate DDMs for the speed and accuracy instructions as they may differentially affect boundary separation and other parameters. This follows the approach of Ratcliff and Rouder (1998) and Singmann (2022).

Figure A7 shows the proportion of "dark" responses across brightness bins, revealing a clear effect of stimulus strength on choice. We also plot RT quantiles (10%, 30%, 50%, 70%, 90%) by strength level and instruction condition. As expected, the speed condition yielded shorter RTs, highlighting substantial differences between conditions (Figure A8).

The standard estimator was trained on clean reaction time data, while the robust estimator used a folded-t_1 contamination model with a 10% contamination probability. Both used identical network architectures and training parameters as in the previous section. The number of observations per data set in the training data varied from 100 to 5,000 trials to maintain flexibility for data sets up to 4,000 trials. Training was performed on an NVIDIA Tesla V100-SXM2-32GB GPU, with each epoch taking 95 seconds and a total training time of 158 minutes per network.

Both raw and cleaned data sets were input to the two networks. Data cleaning followed the original procedure by Ratcliff and Rouder (1998), excluding trials with reaction times below 200ms or above 2500ms.

Posterior means and standard deviations for all conditions are shown in Figures 11 and 12. Under the speed instruction, fitting raw versus cleaned data with the standard estimator yielded notable differences, particularly for participants 1 and 2. In contrast, for the accuracy condition, the impact of data cleaning was minor. This is likely because

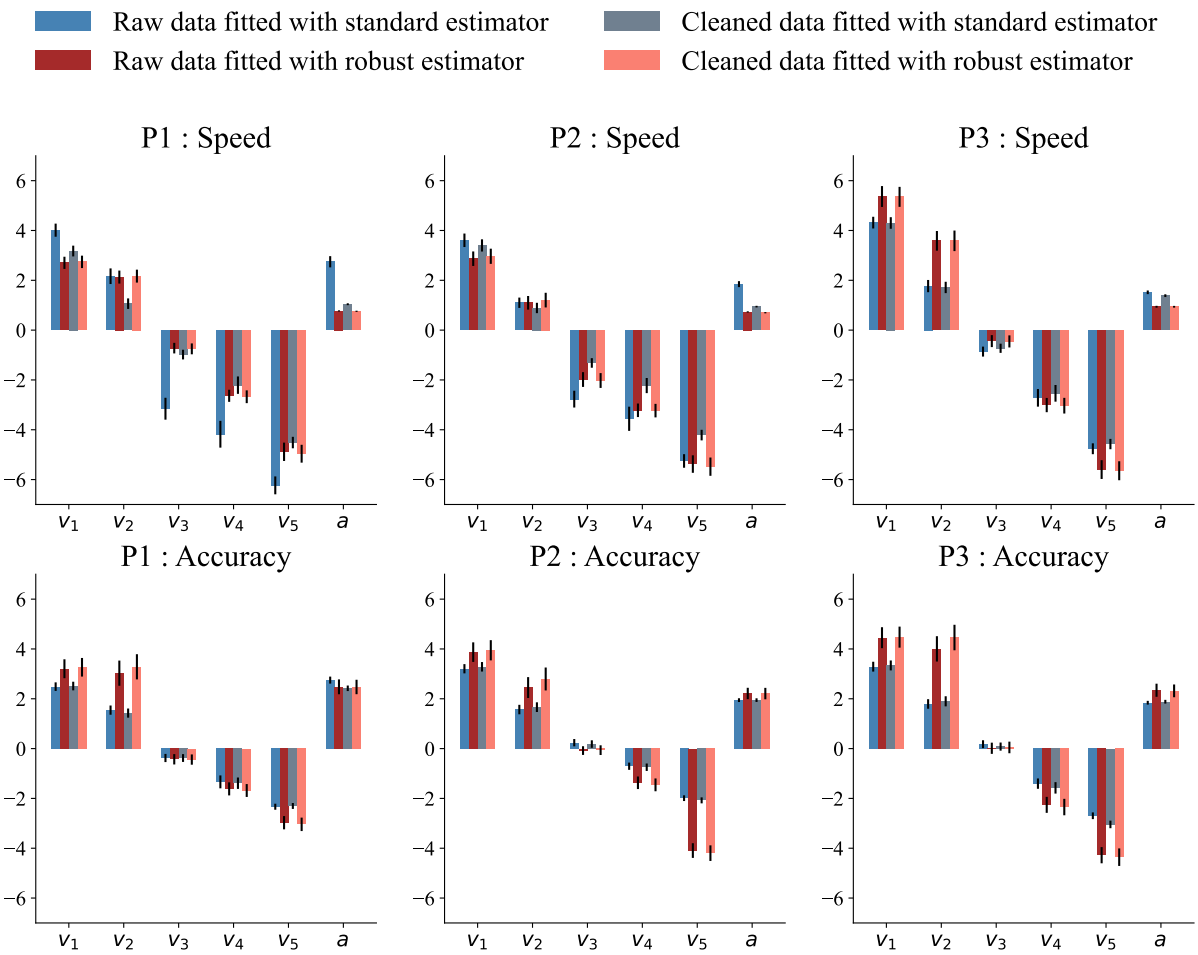


Figure 11: Inference of drift rate and boundary separation across different data sets and estimators. The y-axis represents the estimated values, while the x-axis shows the parameter names. The error bar represents one posterior standard deviation. These parameters are displayed in a single figure due to their similar scales.

most outliers in the accuracy condition are long (95% between 2.52s and 6.98s), which may reflect genuine decision hesitations rather than contamination. Long outliers generally have less influence on parameter estimates than short ones. Under the speed condition, outliers are predominantly short (95% between 0.123s and 0.199s), which strongly affects estimation. In contrast, the robust estimator produced consistent results regardless of whether raw or cleaned data were used.

Since real data lack ground-truth parameters, we assessed how well the estimates reflect experimental design effects. Parameters v_1 to v_5 correspond to conditions with increasing proportions of white pixels, with v_1 representing the lowest and v_5 the highest. Given that the "dark" response is the upper bound, we expected an ordinal trend: $v_1 > v_2 > v_3 > v_4 > v_5$ under both speed and accuracy instructions, with v_3 nearest to zero due to maximal perceptual noise. This expected pattern was successfully recovered by the robust estimator applied to raw data.

As for other parameters, in the speed condition, the standard estimator applied to raw data severely underestimated non-decision time for participants 1 and 2 due to short

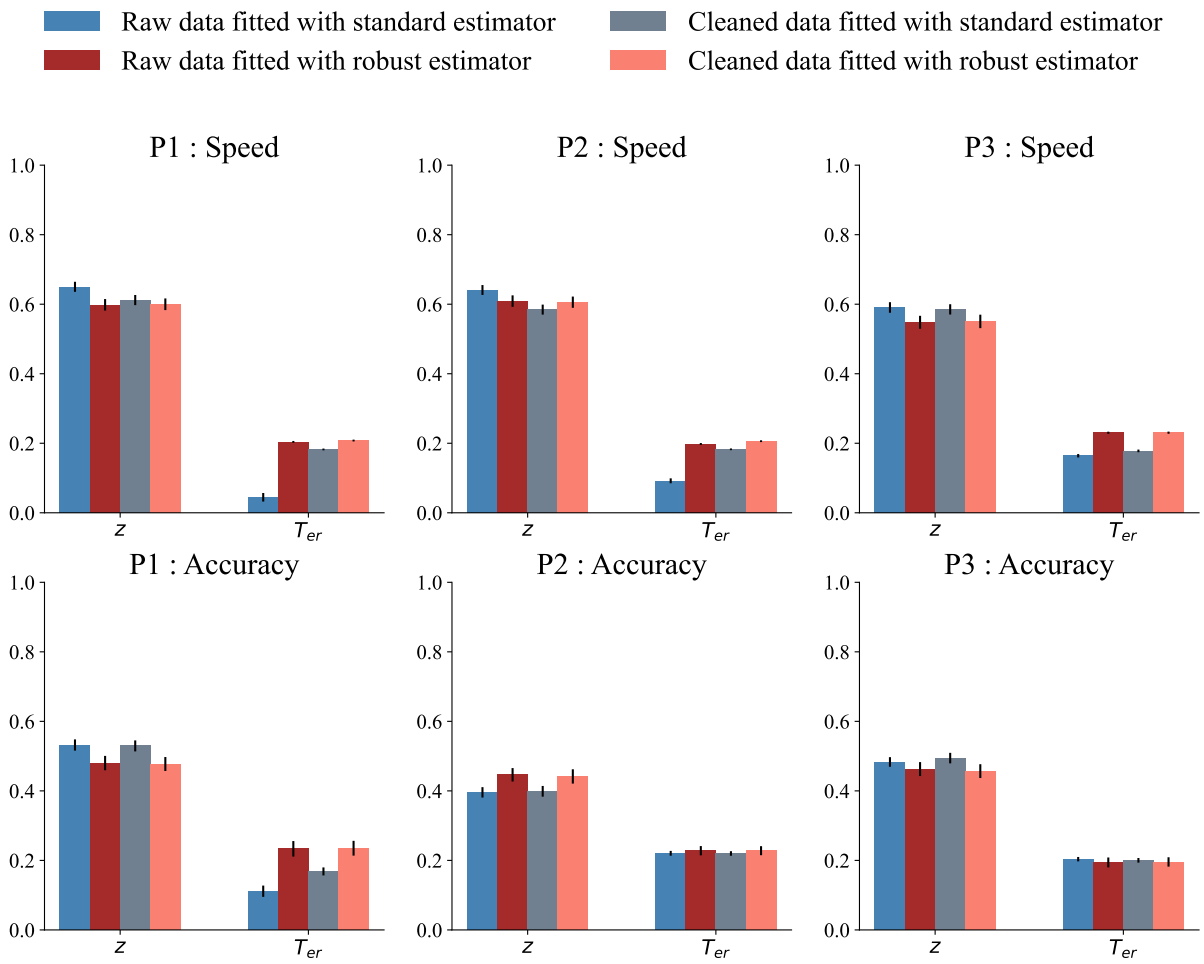


Figure 12: Inference of response bias and non-decision time across different data sets and estimators. The y-axis represents the estimated values, while the x-axis shows the parameter names. The error bar represents one standard deviation in the posterior distribution. These parameters are displayed in a single figure due to their similar scales.

outliers. This underestimation biased other parameters, notably inflating drift rates and boundary separation (see row (b) in Figure 5 and Figure 11). The bias in boundary separation was obvious when comparing speed and accuracy conditions. While speed instructions are typically associated with lower boundary separation (Ratcliff, 1978; Ratcliff & Rouder, 1998; Ratcliff et al., 2016), this expected pattern emerged only in the robust estimates—not in the standard estimator, which failed to capture the effect, potentially leading to incorrect conclusions.

Overall, the robust estimator effectively recovered expected patterns induced by experimental manipulations, such as the ordinal ranking of conditions, while mitigating biases from outliers. In contrast, the standard estimator was sensitive to outliers, often underestimating non-decision time and inflating other parameters, which can lead to misleading conclusions. Notably, the robust estimator under the ABI framework demonstrated strong performance and practical accessibility for real-world applications.

Discussion

This paper focused on testing and improving the robustness of ABI in the context of cognitive modeling. It provided the first systematic exploration of the robustness of ABI, combining theoretical analysis and empirical assessment. It also offered a straightforward method for robustifying ABI, increasing its reliability across a wide range of applications in psychology.

The parameter recovery study showed that ABI can perform posterior inference for stochastic cognitive models to the same level of accuracy and precision as MCMC sampling. Additionally, the comparison between learned summary statistics and sufficient summary statistics being used in EZ diffusion (Wagenmakers et al., 2007) uncovered the black box of neural networks. These findings position ABI as a method that can maintain interpretability while leveraging the flexibility and capacity of neural networks.

The impact of contaminants in ABI was then assessed with SSC and BP. When estimating μ in a univariate normal distribution, the SSC and BP in analytical posterior inference and ABI nicely coincided. This congruence is promising as it implies that ABI does not introduce unforeseen or erratic reactions to the presence of contaminants. In DDM estimates, our analysis showed that even a single contaminant led to bias, which pointing out the necessity of improving the robustness of ABI to ensure accuracy in contamination-prone data sets.

To enhance the robustness of ABI, a simple yet effective data augmentation method was proposed. By assuming a fraction of the training data (e.g., 10%) comes from the contamination distribution, the neural networks can learn to map contaminated data to the correct posterior. We examined contamination distributions based on t distributions with different degrees of freedom ν and a uniform distribution. In both toy example and DDM estimation, we found that assuming contamination arising from a t distribution with one degree of freedom (i.e., a Cauchy) largely improved robustness. Meanwhile, the robust μ estimator with Cauchy distribution exhibited the same SSC as Tukey's Biweight function (Tukey, 1979), an M-estimator that applies transformations on residuals when using the traditional least squares method. These results suggest that the data augmentation approach can achieve similar effects to manually manipulating the loss function. This method is a very straightforward approach that addresses a crucial methodological gap by integrating robust statistics with amortized inference.

The cost of robust ABI was also explored. Compared to a standard estimator, a robust estimator exhibited higher efficiency loss (i.e., higher posterior variance) when applied to uncontaminated data. Nevertheless, the robust neural μ estimators with Cauchy distribution can be considered as highly robust and efficient compared to its counterparts in traditional robust statistics (Hampel et al., 2005; Tukey, 1979; Huber, 1964). Yet, evaluating the efficiency loss of the robust DDM estimator is complicated due to the absence of a direct comparative reference. Additionally, we found the same robustness-efficiency

trade-off in ABI as in robust statistics (Hampel et al., 2005): the higher the robustness, the larger the efficiency losses. The cost of robustness in ABI aligned closely with principles from robust statistics, paving the way to integrate these two research domains.

Lastly, we applied the robust ABI to a real data set and found that fitting raw choice reaction time data to the robust estimator with a Cauchy distribution yielded accurate estimates. This method holds significant potential for application in research areas where contamination detection or removal is particularly challenging.

At the same time, we acknowledge the limitations of current work. In our two examples, we primarily used $t-$based robust estimators, but the choice of contamination distribution is flexible and should be tailored to the specific model and data. This choice requires the researcher to have a general understanding of the potential range of contaminants. Moreover, although the Cauchy distribution is a good contamination distribution candidate when we have univariate data, specifying the contamination distribution in multi-dimensional data is more challenging as it involves accounting for the underlying correlation structure. Thus, it is worthwhile to study this robust approach in a multi-dimensional setting. Secondly, the probability π of one entry in the data set being contaminated during network training is determined by the researcher, which again requires knowledge about the empirical data set. A possible extension would involve treating π as a parameter that can vary across data sets during data simulation, allowing it to be estimated and used as an index of data quality. Such an approach can also lead to more efficient estimation for data sets that have a lower fraction of outliers. However, estimating π could be challenging in cases where the fraction of contaminants is small or when the contamination model differs significantly from the assumed one. Another possible future research direction concerns optimizing the neural architecture or post-training correction (Siahkoohi et al., 2023) with respect to the robustness of inference.

In conclusion, as ABI is a powerful method due to its high flexibility and low computational cost in inference, assessing and robustifying the method facilitates both methodological development and practical application. This work paves the way for a new era of robust and efficient Bayesian inference, extending the applicability of ABI to complex real-world data sets.

Author Notes

Prior Dissemination

Part of this work was presented at the Annual Meeting of the Psychometric Society (2024) in Prague, Czech Republic, the Winter Conference of the Interuniversity Graduate School of Psychometrics and Sociometrics (2024) in The Hague, Netherlands, and the workshop on Accelerating Statistical Inference and Experimental Design with Machine Learning (2025) in Cambridge, United Kingdom. A preprint of this article has been posted online on arXiv (https://arxiv.org/abs/2412.20586).

Corresponding Author

Correspondence concerning this article should be addressed to Yufei Wu.
Email: yufei.wu@kuleuven.be

Conflicts of Interest

The authors declare no conflict of interest.

Author Contributions

Yufei Wu: Methodology, software, data curation, formal analysis, writing—original draft. **Stefan T. Radev**: Resources, software, methodology, writing—reviewing and editing. **Francis Tuerlinckx**: Conceptualization, methodology, supervision, writing—reviewing and editing.

Data Availability

All data and code are available in the following repository: https://github.com/yufeiwu1011/Robust-Amortized-Bayesian-Inference

Funding

Yufei Wu and Francis Tuerlinckx are partially supported by a grant from the Research Council of KU Leuven (C14/23/062).

Acknowlegment

We thank Kristof Meers for his assistance in accelerating the simulation code for the diffusion model. We also thank Prof. Mia Hubert for her valuable remarks on our methodology.

References

Aggarwal, C. C. (2017). *Outlier Analysis*. Springer International Publishing. https://doi.org/10.1007/978-3-319-47578-3

Ahn, W.-Y., Haines, N., & Zhang, L. (2017). Revealing neurocomputational mechanisms of reinforcement learning and decision-making with the hBayesDM package. *Computational Psychiatry*, 1, 24–57. https://doi.org/doi:10.1162/CPSY_a_00002

Andrews, D. F., Bickel, P. J., Hampel, F. R., Huber, P. J., Rogers, W. H., & Tukey, J. W. (1972). *Robust estimates of location: Survey and advances* (1 ed.). Wiley Series in Probability and Statistics. Princeton University Press.

Ardizzone, L., Lüth, C., Kruse, J., Rother, C., & Köthe, U. (2019). *Guided Image Generation with Conditional Invertible Neural Networks*. http://arxiv.org/abs/1907.02392. arXiv:1907.02392 [cs]

Dax, M., Wildberger, J., Buchholz, S., Green, S. R., Macke, J. H., & Schölkopf, B. (2023). Flow matching for scalable simulation-based inference. *arXiv preprint arXiv:2305.17161*. https://arxiv.org/abs/2305.17161

Donoho, D. & Huber, P. (1982). The notion of breakdown point. *A Fesrschrift for Eric Lehmann*, 157–184. Wadsworth.

Durkan, C., Bekasov, A., Murray, I., & Papamakarios, G. (2019). Neural spline flows. *Advances in Neural Information Processing Systems*, 32.

Elsemüller, L., Pratz, V., von Krause, M., Voss, A., Bürkner, P.-C., & Radev, S. T. (2025). Does unsupervised domain adaptation improve the robustness of amortized bayesian inference? a systematic evaluation. *arXiv preprint arXiv:2502.04949*.

Gelman, A., Carlin, J. B., Stern, H. S., Dunson, D. B., Vehtari, A., & Rubin, D. B. (2013). *Bayesian Data Analysis* (third ed.). Chapman and Hall/CRC Texts in Statistical Science Series. CRC. https://stat.columbia.edu/~gelman/book/

Gelman, A. & Rubin, D. B. (1992). Inference from Iterative Simulation Using Multiple Sequences. *Statistical Science*, 7(4). https://doi.org/10.1214/ss/1177011136

Gonçalves, P. J., Lueckmann, J.-M., Deistler, M., Nonnenmacher, M., Öcal, K., Bassetto, G., Chintaluri, C., Podlaski, W. F., Haddad, S. A., Vogels, T. P., et al. (2020). Training deep neural density estimators to identify mechanistic models of neural dynamics. *eLife*, 9, e56261.

Hampel, F. R., Ronchetti, E. M., Rousseeuw, P. J., & Stahel, W. A. (2005). *Robust Statistics: The Approach Based on Influence Functions* (1 ed.). Wiley Series in Probability and Statistics. Wiley. https://doi.org/10.1002/9781118186435

Hawkins, D. (1980). *Identification of Outliers*. Monographs on Statistics and Applied Probability Ser. Springer Netherlands.

Huang, D., Bharti, A., Souza, A., Acerbi, L., & Kaski, S. (2024). Learning robust statistics for simulation-based inference under model misspecification. *Advances in Neural Information Processing Systems*, 36.

Huber, P. J. (1964). Robust estimation of a location parameter. *The Annals of Mathematical Statistics*, 35(1), 73–101. https://doi.org/10.1214/aoms/1177703732

Kelly, R., Nott, D. J., Frazier, D. T., Warne, D., & Drovandi, C. (2024). Misspecification-robust sequential neural likelihood for simulation-based inference. *Transactions on Machine Learning Research*, 2024(June), Article–number.

Kobyzev, I., Prince, S. J., & Brubaker, M. A. (2020). Normalizing flows: An introduction and review of current methods. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(11), 3964–3979.

Kruschke, J. K. (2011). *Doing Bayesian data analysis: a tutorial with R and BUGS*. Academic Press.

Kullback, S. & Leibler, R. A. (1951). On Information and Sufficiency. *The Annals of Mathematical Statistics*, 22(1), 79–86. https://doi.org/10.1214/aoms/1177729694

Lee, J., Lee, Y., Kim, J., Kosiorek, A., Choi, S., & Teh, Y. W. (2019). Set transformer: A framework for attention-based permutation-invariant neural networks. *Proceedings of the 36th International Conference on Machine Learning (ICML)*, volume 97, 3744–3753. https://proceedings.mlr.press/v97/lee19d.html

Lerche, V. & Voss, A. (2016). Model complexity in diffusion modeling: Benefits of making the model more parsimonious. *Frontiers in Psychology*, 7, 1324.

Lerche, V., Voss, A., & Nagler, M. (2017). How many trials are required for parameter estimation in diffusion modeling? a comparison of different optimization criteria. *Behavior Research Methods*, 49, 513–537.

Lipman, Y., Chen, R. T., Ben-Hamu, H., Nickel, M., & Le, M. (2022). Flow matching for generative modeling. *The Eleventh International Conference on Learning Representations*.

Maniaci, M. R. & Rogge, R. D. (2014). Caring about carelessness: Participant inattention and its effects on research. *Journal of Research in Personality*, 48, 61–83. https://doi.org/10.1016/j.jrp.2013.09.008

Maronna, R. A., Martin, R. D., & Yohai, V. J. (2006). *Robust Statistics: Theory and Methods* (1 ed.). Wiley Series in Probability and Statistics. Wiley. https://doi.org/10.1002/0470010940

Mulder, M. J., Wagenmakers, E.-J., Ratcliff, R., Boekel, W., & Forstmann, B. U. (2012). Bias in the Brain: A Diffusion Model Analysis of Prior Probability and Potential Payoff. *The Journal of Neuroscience*, 32(7), 2335–2343. https://doi.org/10.1523/JNEUROSCI.4156-11.2012

Myers, C. E., Interian, A., & Moustafa, A. A. (2022). A practical introduction to using the drift diffusion model of decision-making in cognitive psychology, neuroscience, and health sciences. *Frontiers in Psychology*, 13, 1039172. https://doi.org/10.3389/fpsyg.2022.1039172

Palestro, J. J., Sederberg, P. B., Osth, A. F., Van Zandt, T., & Turner, B. M. (2018). *Likelihood-free methods for cognitive science*. Springer.

Plummer, M. et al. (2003). Jags: A program for analysis of bayesian graphical models using gibbs sampling. *Proceedings of the 3rd International Workshop on Distributed Statistical Computing*, 1–10.

Radev, S. T., Mertens, U. K., Voss, A., Ardizzone, L., & Köthe, U. (2020). *Bayesflow: Learning complex stochastic models with invertible neural networks*. https://doi.org/10.48550/arXiv.2003.06281. arXiv preprint arXiv:2003.06281

Radev, S. T., Schmitt, M., Schumacher, L., Elsemüller, L., Pratz, V., Schälte, Y., Köthe, U., & Bürkner, P.-C. (2023). BayesFlow: Amortized Bayesian Workflows With Neural Networks. *Journal of Open Source Software*, 8(89), 5702. https://doi.org/10.21105/joss.05702

Ratcliff, R. (1978). A theory of memory retrieval. *Psychological Review*, 85(2), 59–108. https://doi.org/10.1037/0033-295X.85.2.59

Ratcliff, R. (1993). Methods for dealing with reaction time outliers. *Psychological Bulletin*, 114(3), 510–532. https://doi.org/10.1037/0033-2909.114.3.510

Ratcliff, R., Gomez, P., & McKoon, G. (2004). A Diffusion Model Account of the Lexical Decision Task. *Psychological Review*, 111(1), 159–182. https://doi.org/10.1037/0033-295X.111.1.159

Ratcliff, R. & Rouder, J. N. (1998). Modeling Response Times for Two-Choice Decisions. *Psychological Science*, 9(5), 347–356. https://doi.org/10.1111/1467-9280.00067

Ratcliff, R., Smith, P. L., Brown, S. D., & McKoon, G. (2016). Diffusion Decision Model: Current Issues and History. *Trends in Cognitive Sciences*, 20(4), 260–281. https://doi.org/10.1016/j.tics.2016.01.007

Ratcliff, R. & Tuerlinckx, F. (2002). Estimating parameters of the diffusion model: Approaches to dealing with contaminant reaction times and parameter variability. *Psychonomic Bulletin & Review*, 9(3), 438–481. https://doi.org/10.3758/BF03196302

Schmitt, M., Bürkner, P.-C., Köthe, U., & Radev, S. T. (2021). *Detecting Model Misspecification in Amortized Bayesian Inference with Neural Networks*. https://doi.org/10.48550/ARXIV.2112.08866. Version Number: 5

Schmitt, M., Li, C., Vehtari, A., Acerbi, L., Bürkner, P.-C., & Radev, S. T. (2024a). Amortized Bayesian workflow. *NeurIPS 2024 Workshop on Bayesian Decision-making and Uncertainty*.

Schmitt, M., Pratz, V., Köthe, U., Bürkner, P.-C., & Radev, S. T. (2024b). Consistency models for scalable and fast simulation-based inference. *Advances in Neural Information Processing Systems (NeurIPS)*.

Schumacher, L. & Radev, S. T. (2023). *Principled amortized bayesian workflow for cognitive modeling*. `https://bayesflow.org/stable-legacy/_examples/LCA_Model_Posterior_Estimation.html`

Segal, M. & Xiao, Y. (2011). Multivariate random forests. *WIREs Data Mining and Knowledge Discovery*, 1(1), 80–87. `https://doi.org/10.1002/widm.12`

Siahkoohi, A., Rizzuti, G., Orozco, R., & Herrmann, F. J. (2023). Reliable amortized variational inference with physics-based latent distribution correction. *Geophysics*, 88(3), R297–R322.

Simons, J., Sharrock, L., Liu, S., & Beaumont, M. (2023). Neural score estimation: Likelihood-free inference with conditional score based diffusion models. *Fifth Symposium on Advances in Approximate Bayesian Inference*.

Singmann, H. (2022). *Reanalysis of Ratcliff and Rouder (1998) with Diffusion Model and LBA*. `https://cran.r-project.org/web/packages/rtdists/vignettes/reanalysis_rr98.html#description-of-the-experiment`

Singmann, H., Brown, S., Gretton, M., Heathcote, A., Voss, A., Voss, J., & Terry, A. (2022). *rtdists: Response Time Distributions*. `https://cran.r-project.org/web/packages/rtdists/index.html`. R package version 0.11.5

Sumarni, C., Sadik, K., Notodiputro, K. A., & Sartono, B. (2017). Robustness of location estimators under t -distributions: a literature review. *IOP Conference Series: Earth and Environmental Science*, 58, 012015. `https://doi.org/10.1088/1755-1315/58/1/012015`

Talts, S., Betancourt, M., Simpson, D., Vehtari, A., & Gelman, A. (2020). *Validating Bayesian Inference Algorithms with Simulation-Based Calibration*. `http://arxiv.org/abs/1804.06788`. arXiv:1804.06788 [stat]

Tillman, G., Van Zandt, T., & Logan, G. D. (2020). Sequential sampling models without random between-trial variability: The racing diffusion model of speeded decision making. *Psychonomic Bulletin & Review*, 27(5), 911–936.

Tran, N.-H., Van Maanen, L., Heathcote, A., & Matzke, D. (2021). Systematic parameter reviews in cognitive modeling: Towards a robust and cumulative characterization of psychological processes in the diffusion decision model. *Frontiers in psychology*, 11, 608287.

Tukey, J. W. (1977). *Exploratory Data Analysis*. Addison-Wesley.

Tukey, J. W. (1979). Robust Techniques for the User. *Robustness in Statistics*, 103–106. Elsevier. https://doi.org/10.1016/B978-0-12-438150-6.50013-3

Ulrich, R. & Miller, J. (1994). Effects of truncation on reaction time analysis. *Journal of Experimental Psychology: General*, 123(1), 34–80. https://doi.org/10.1037/0096-3445.123.1.34

Usher, M. & McClelland, J. L. (2001). The time course of perceptual choice: The leaky, competing accumulator model. *Psychological Review*, 108(3), 550–592. https://doi.org/10.1037/0033-295X.108.3.550

Von Krause, M., Radev, S. T., & Voss, A. (2022). Mental speed is high until age 60 as revealed by analysis of over a million participants. *Nature Human Behaviour*, 6(5), 700–708. https://doi.org/10.1038/s41562-021-01282-7

Voss, A. & Voss, J. (2008). A fast numerical algorithm for the estimation of diffusion model parameters. *Journal of Mathematical Psychology*, 52(1), 1–9. https://doi.org/10.1016/j.jmp.2007.09.005

Wabersich, D. & Vandekerckhove, J. (2014). Extending JAGS: A tutorial on adding custom distributions to JAGS (with a diffusion model example). *Behavior Research Methods*, 46(1), 15–28. https://doi.org/10.3758/s13428-013-0369-3

Wagenmakers, E.-J., Van Der Maas, H. L. J., & Grasman, R. P. P. P. (2007). An EZ-diffusion model for response time and accuracy. *Psychonomic Bulletin & Review*, 14(1), 3–22. https://doi.org/10.3758/BF03194023

Ward, D., Cannon, P., Beaumont, M., Fasiolo, M., & Schmon, S. M. (2022). *Robust Neural Posterior Estimation and Statistical Model Criticism*. http://arxiv.org/abs/2210.06564. arXiv:2210.06564 [cs, stat]

Wieschen, E. M., Voss, A., & Radev, S. (2020). Jumping to Conclusion? A Lévy Flight Model of Decision Making. *The Quantitative Methods for Psychology*, 16(2), 120–132. https://doi.org/10.20982/tqmp.16.2.p120

Wu, L. L. (1985). Robust m-estimation of location and regression. *Sociological Methodology*, 15, 316–388.

Yang, L., Zhang, Z., Song, Y., Hong, S., Xu, R., Zhao, Y., Zhang, W., Cui, B., & Yang, M.-H. (2023). Diffusion models: A comprehensive survey of methods and applications. *ACM Computing Surveys*, 56(4), 1–39.

Zaheer, M., Kottur, S., Ravanbakhsh, S., Poczos, B., Salakhutdinov, R. R., & Smola, A. J. (2017). Deep sets. *Advances in neural information processing systems*, 30.

Appendix

Parameter Recovery Study for the Drift Diffusion Model

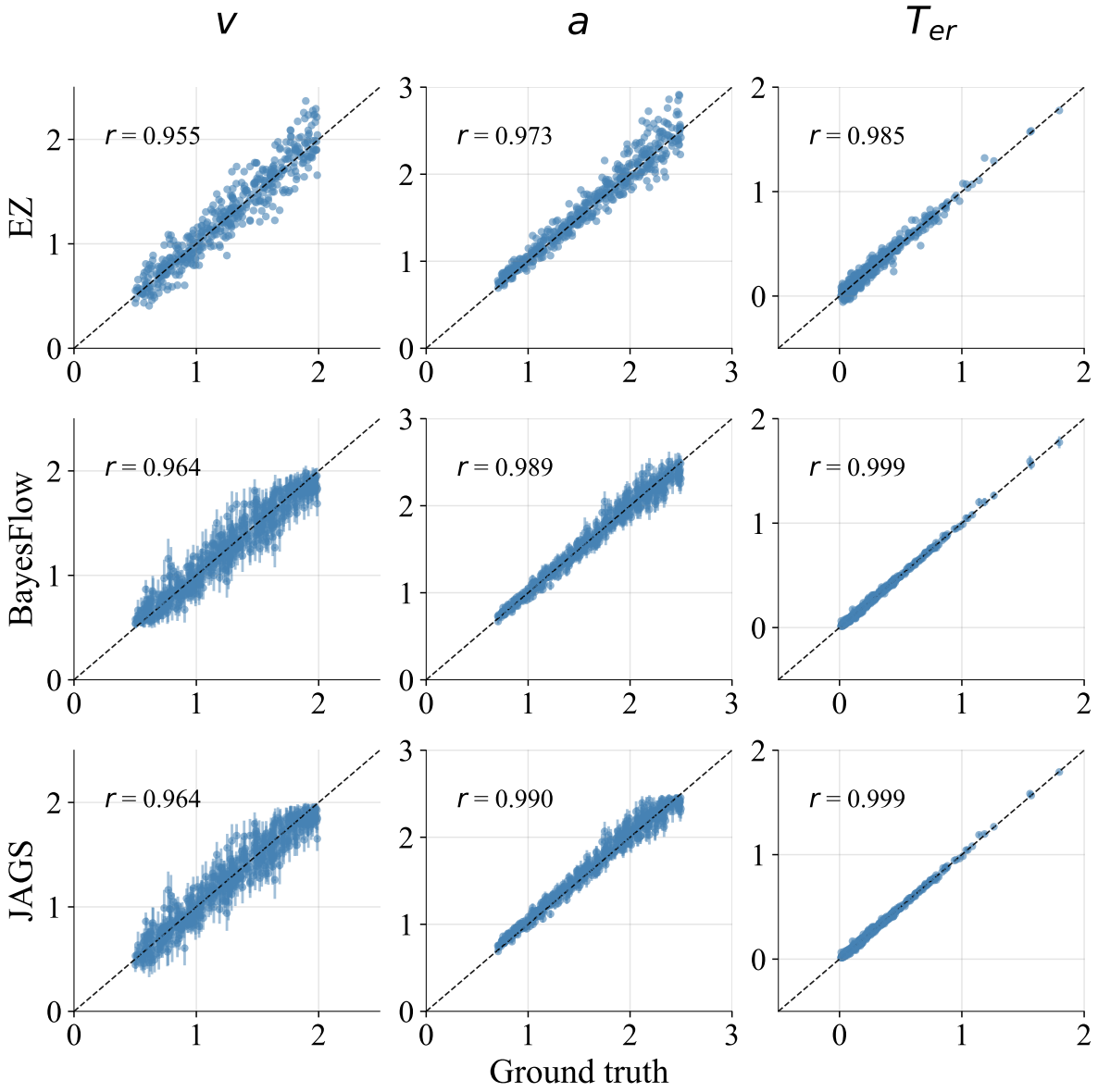


Figure A1: Parameter recovery of the minimal DDM with different estimators. The x- and y-axis show the ground truth and estimated values, respectively. The diagonal dashed lines indicate perfect recovery. Additionally, the Pearson's correlation coefficient r quantifies the alignment between the estimated and true values. The error bars for JAGS and BayesFlow show ± posterior standard deviation around the point estimates.

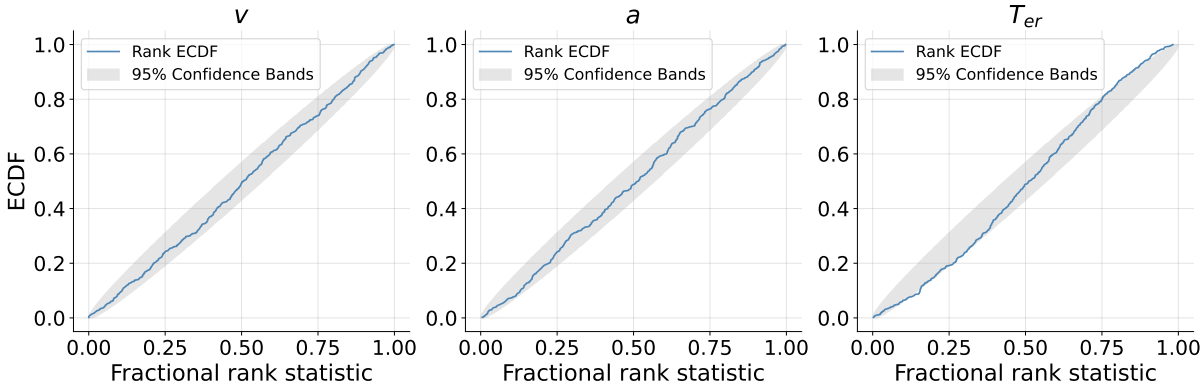


Figure A2: The empirical cumulative density function (ECDF) of rank statistics plotted against uniform ECDF. The posterior ranks of the prior draws distributed uniformly, indicating the NPE yielded correct posteriors (for details of rank statistics, please refer to Talts et al., 2020).

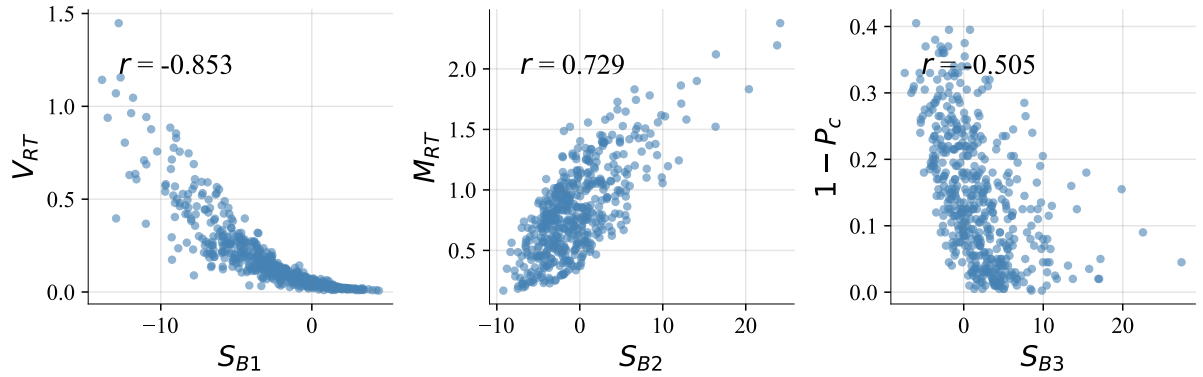


Figure A3: The non-linear relationship between $\boldsymbol{s}_B$ and $\boldsymbol{s}_{EZ}$. Three summary statistics from EZ diffusion estimation are plotted against three summary statistics learned by BayesFlow, respectively.

Priors and Parameter Recoveries in Neural DDM estimators

Parameters	Distribution
v_1 (Drift Rate 1)	$U(0, 7)$
v_2 (Drift Rate 2)	$U(-7, 0)$
a (Boundary Separation)	$U(0.1, 5)$
z (Starting Point)	$U(0.01, 0.99)$
T_{er} (Non-Decision Time)	Gamma$(1.5, 0.2)$

Table A1: Prior distributions over the DDM parameters. Gamma$(1.5, 0.2)$ denotes a gamma distribution with shape parameter 1.5 and scale parameter 0.2.

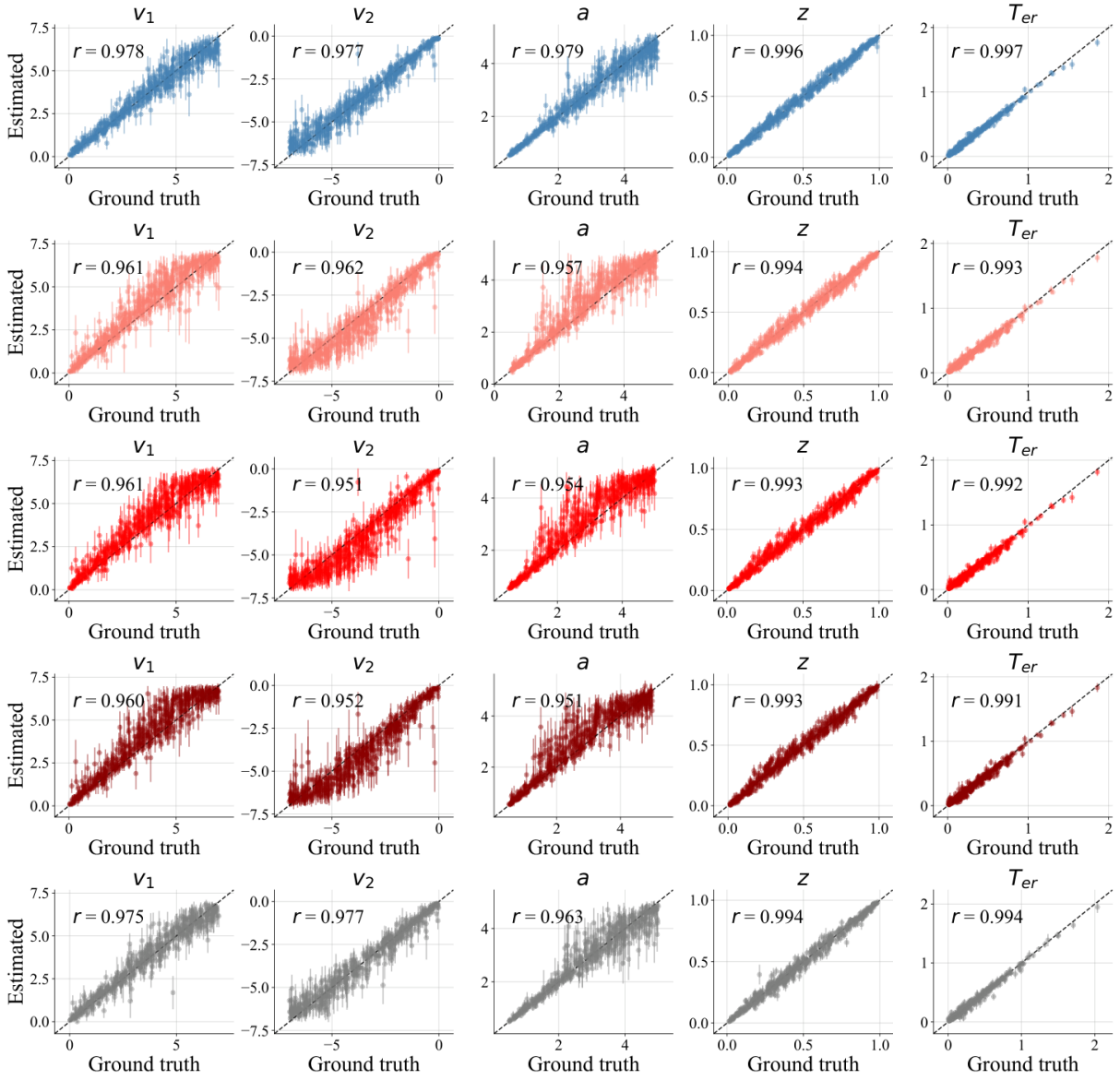


Figure A4: Parameter recovery of DDM estimators on 500 new data sets when the total trial number $n = 200$ (100 per condition). The 500 new data set were simulated from the standard DDM simulator without any contamination. From top to bottom, the estimators are: standard estimator, robust estimators with folded-t_1, folded-t_3, folded-t_5, and $U(0, 20)$, respectively.

Robust Estimators with Different Contamination Probability π

Through the simulations presented in the main body of this paper, we found t_1 distribution the optimal contamination distribution for training robust NPEs, with the contamination probability $\pi = 0.1$. We further explored the impact of π in both the toy example and the DDM. Specifically, we trained robust NPEs with $\pi \in \{0.01, 0.05, 0.10, 0.20\}$, respectively.

The network architectures were identical for all the μ estimators as specified in the main body text. Afterwards, we investigated the robustness of μ estimators with SSC and BP. As the left panel in Figure A5 shows, the SSCs show similar shapes for all the robust estimators. All of them were able to filter large contaminants out, and the larger the π, the lower the sensitivity. The BP plot (the right panel in Figure A5) reveals that the

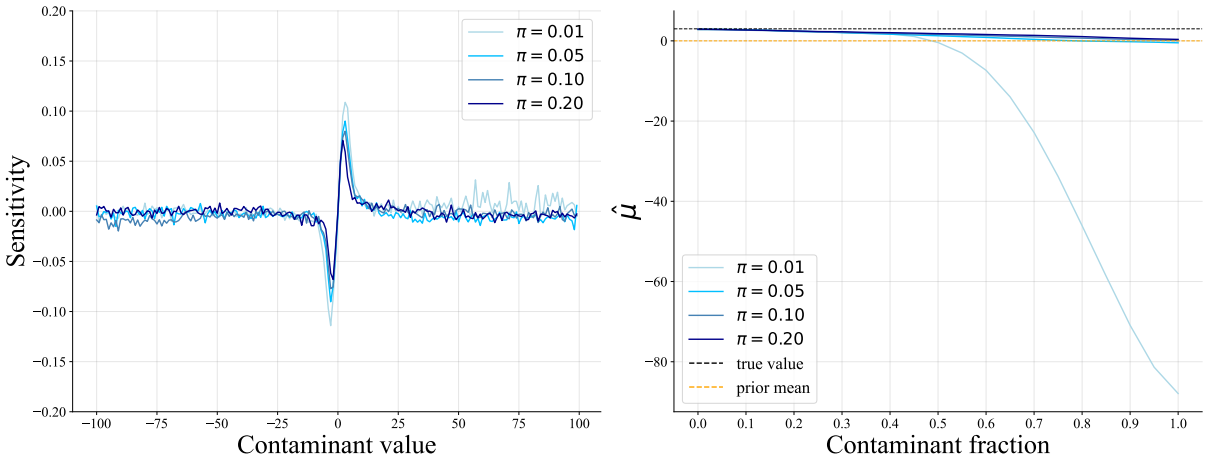


Figure A5: Performance of the t_1-based robust μ estimators with different contamination probability π. The left panel shows the SSC of the robust estimators with contaminants ranging from -100 to 100 (increasing by 1). The right plot shows the breakdown point with the contaminant value fixed at -100.

estimator with $\pi = 0.01$ had a lower BP around 40%, whereas the other robust estimators did not break down. Table A2 shows us that the MAE ratio and posterior variance ratio of robust estimators decreased with the increase of π, indicating that the accuracy and efficiency losses were lower for less robust estimators.

The network architectures and training settings were the same for all the DDM estimators as specified in the main body text. We investigated the SSC and BP of four robust DDM estimators and found that the higher the π, the less sensitive the estimator to a contaminant, especially when there was a short contaminant (see rows (a) and (b) in Figure A6). In terms of BP, the estimators with $\pi = 0.10$ and 0.20 showed a higher BP than the other two estimators for short contaminants ($x^c = 0.01s$), while the estimators with $\pi = 0.05$ and 0.10 showed a higher BP for long contaminants ($x^c = 20s$). The robustness and efficiency losses trade-off appeared again in the variance ratio of robust DDM estimators and standard DDM estimators: the higher the π, the smaller the variance ratio, thus larger efficiency losses.

To summarize, varying π affected the performance of robust estimators. While $\pi = 0.01$ can be too low and $\pi = 0.20$ is too high in terms of efficiency loss, a π from 0.05 to 0.10 allows us to balance these two properties, thus is more suitable to be used in practice.

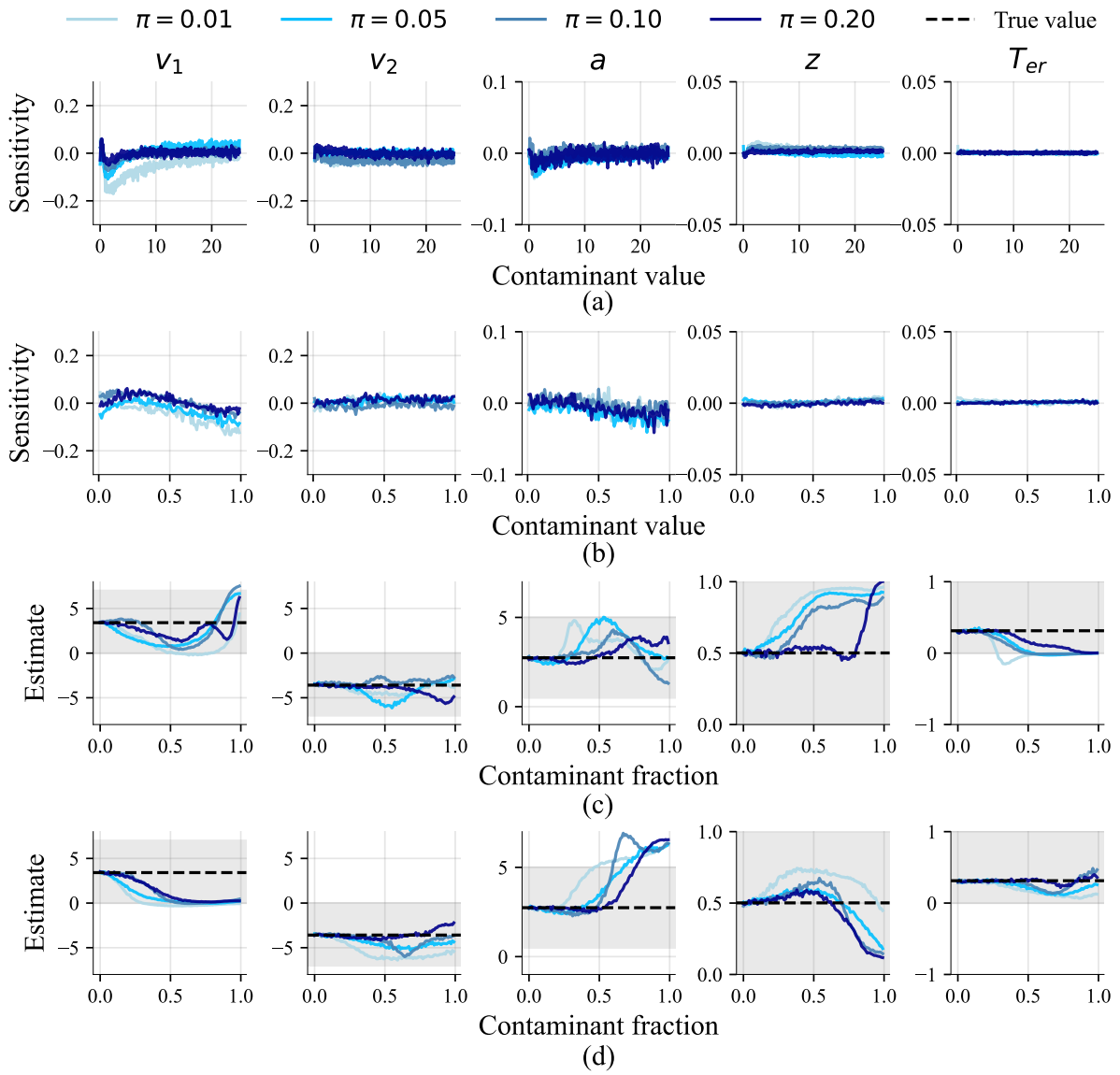


Figure A6: The DDM robust estimators with contamination distribution folded-t_1 and different contamination probability π. Row (a) shows the SSC of the robust estimators with outliers ranging from 0.01s to 25s (increasing by 0.05s). Row (b) shows the SSC with outliers ranging from 0.01s to 1s (increasing by 0.01s). Rows (c) and (d) display the BP plots with outlier values fixed at 0.01s and 20s, respectively.

Robust estimator	MAE Ratio	Posterior Variance Ratio
$\pi = 0.01$	1.003	0.981
$\pi = 0.05$	1.000	0.945
$\pi = 0.10$	0.996	0.913
$\pi = 0.20$	0.978	0.863

Table A2: MAE ratio and posterior variance ratio of location parameter estimators with different contamination probabilities π. This table shows the MAE ratio and the posterior variance ratio between a standard estimator and the robust estimator. In each row, the robust estimator has a t_1 as the contamination distribution and contamination probability π.

Robust estimator	v_1	v_2	a	z	T_{er}
$\pi = 0.01$	0.759	0.802	0.777	0.907	0.826
$\pi = 0.05$	0.73	0.732	0.735	0.892	0.797
$\pi = 0.10$	0.728	0.751	0.741	0.889	0.764
$\pi = 0.20$	0.875	0.907	0.916	0.965	0.884

Table A3: MAE ratio in DDM estimators with different π. This table shows the MAE ratio between a standard estimator and a robust estimator. In each row, the robust estimator has a folded-t_1 as the contamination distribution and a certain contamination probability π.

Robust estimator	v_1	v_2	a	z	T_{er}
$\pi = 0.01$	0.823	0.807	0.885	0.978	0.864
$\pi = 0.05$	0.720	0.699	0.74	0.819	0.71
$\pi = 0.10$	0.617	0.596	0.615	0.734	0.574
$\pi = 0.20$	0.436	0.517	0.531	0.574	0.492

Table A4: Posterior variance ratio of DDM estimators with different π. This table shows the posterior variance ratio in estimates between a standard estimator and a robust estimator. In each row, the robust estimator has a folded-t_1 as the contamination distribution and a certain contamination probability π.

Descriptive Statistics in the Real Data Example

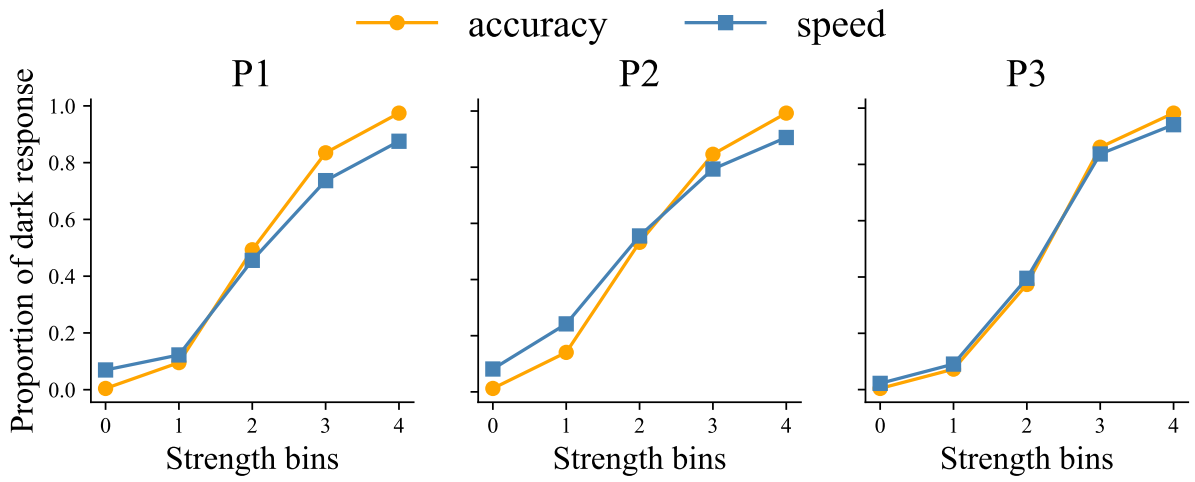


Figure A7: The relationship between proportion of "dark" responses and the brightness strength bins.

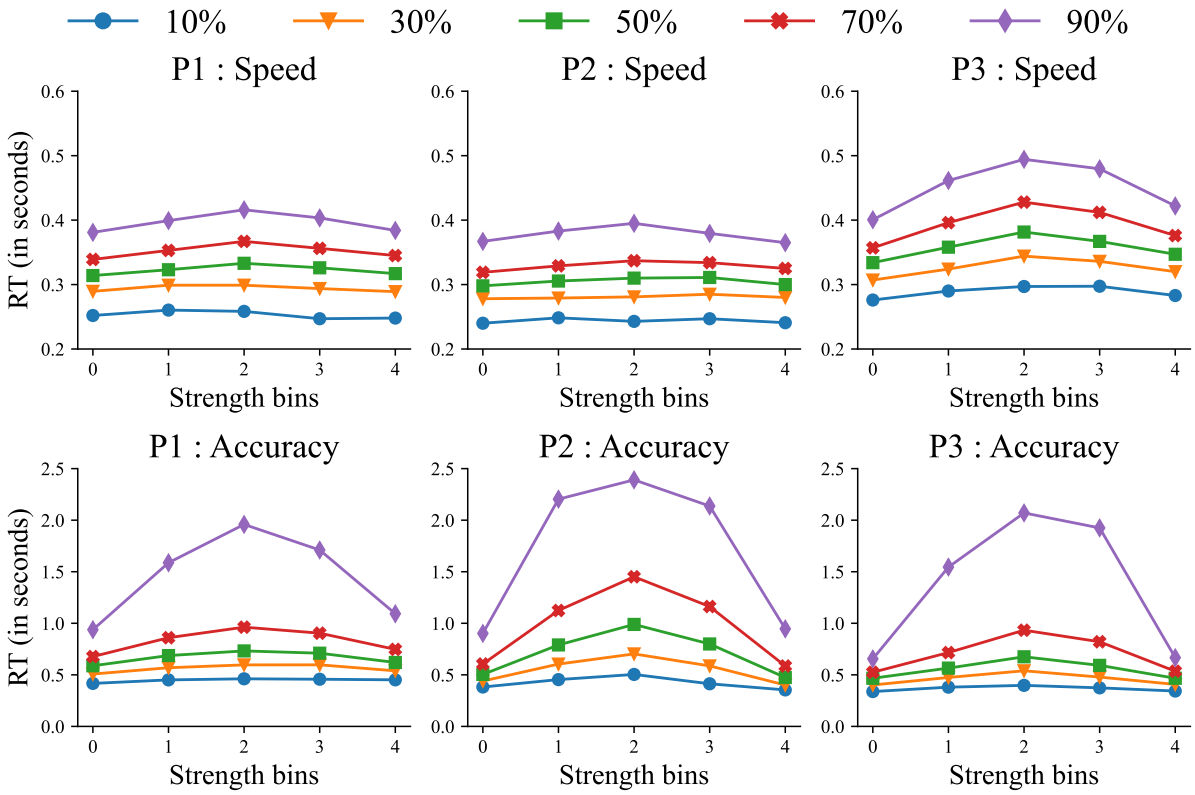


Figure A8: Quantile plot of reaction time. The upper panel shows the reaction time quantiles when participants were instructed to respond as fast as possible, while the lower panel displays the reaction time quantiles when participants were instructed to respond as accurately as possible.