MORE PRACTICAL METHODS TO SAFEGUARD PRIVACY WHEN DISCLOSING STATISTICS BASED ON MICRODATA

Guillermo Andres Marquez Musso

ABSTRACT

Here, in this research report, we review methodologies of data perturbation via the application of noise onto regression estimates. Primarily, we compare two noise adding algorithms, the Raj Chetty & John Friedman's 'Maximum Observed (Local) Sensitivity' (MOS) disclosure algorithm, and the Cynthia Dwork & Jing Lei's 'Propose-Test-Release' (PTR) framework. The MOS releases noise infused univariate OLS slope coefficients and their standard errors, however it does not offer any privacy guarantee, being more akin to 'Statistical Disclosure Limitation' (SDL) perturbation methods. On the other hand, the PTR gives (ϵ,δ)-*differential privacy* but necessitates the use of robust estimation. We extend the practical applicability of both algorithms and test their accuracy to the true statistic in the context of Public-use Microdata from the U.S Census 2000 for the state of Alaska at the 5%-level; as well as a synthetic data set provided by Chetty & Friedman. We repeat this analysis with the (ϵ,δ)-*differential private* Gaussian Mechanism, and furthermore extend the case study to the bivariate problem. Our contributions are a proposed scaling of noise via the mean absolute deviation, and a variation on the PTR framework for releasing estimates based on small samples.

0 REFERENCES

[1] A. Narayanan, V. Shmatikov. Robust De-anonymization of Large Datasets (How to Break Anonymity of the Netflix Prize Dataset). 2008.

[2] R. Chetty, J. Friedman. A Practical Method To Reduce Privacy Loss When Disclosing Statistics Based on Small Samples. In 'AEA Papers and Proceedings', vol 109. Pages 414-420. 2019.

[3] M. Templ. Statistical Disclosure Control for Microdata: Methods and Applications in R. Pages 99-132. Springer, 2017.

[4] A. Smith. Privacy-preserving statistical estimation with optimal convergence rates. In 'Proceedings of the Forty-third Annual ACM Symposium on Theory of Computing, STOC'11. Pages 813-822. ACM, 2011.

[5] F. Aldà, B. Rubinstein. The Bernstein Mechanism: Function Release under Differential Privacy. 2016.

[6] M. Heikkilä, S. Kaski, S. Tarkoma, E. Lagerspetz, K. Shimizu, A. Honkela. Differentially private Bayesian learning on distributed data. In '31st Conference on Neural Information Processing Systems'. NIPS, 2017.

[7] C. Dwork, J. Lei. Differential privacy and robust statistics. In 'Proceedings of the 41th Annual ACM Symposium on Theory of Computing (STOC). ACM, 2009.

[8] C. Dwork, A. Roth. The Algorithmic Foundations of Differential Privacy (Foundations and Trends in Theoretical Computer Science). Page 17. Now Publishers Inc, 2014.

[9] Ibid, Page 19

[10] Ibid, Page 31

[11] Ibid

[12] Ibid, Page 53

[13] Ibid

[14] M. Koller, W. Stahel. Sharpening Wald-type Inference in Robust Regression for Small Samples. Seminar for Statistics. 2011.

[15] Y. Susanti, H. Pratiwi, S. Sulistijowati, T. Liana. M ESTIMATION, S ESTIMATION, AND MM ESTIMATION IN ROBUST REGRESSION. In 'International Journal of Pure and Applied Mathematics', vol91. Pages 349-360. 2014.

1 INTRODUCTION

Inadequate privacy policies and the very real possibility of data leakage undermines public trust and the willingness of individuals to provide personal information to data agencies. The public increasingly relies on public and private agencies to be discreet when disclosing data. And although there have been issues raised around the release of this data, be it the gaffes like NETFLIX's Recommendation Contest [1], or the very consequential EU's General Data Protection Regulations; it is also social scientists to whom this is of concern. Statistical estimates released by researchers are potential sources for the dissemination of confidential information.

In this report we focus our attention on the release of noise infused regression estimates for small samples, most particularly,

we look to the 2019 paper 'A PRACTICAL METHOD TO REDUCE PRIVACY LOSS WHEN DISCLOSING STATISTICS BASED ON SMALL SAMPLES' [2] written by Raj Chetty and John Friedman as the basis from whereon we build a differentially private release algorithm. The authors themselves state that their algorithm does not satisfy the criteria of differential privacy, and even their implementation resembles traditional statistical disclosure limitation (SDL) noise addition [3].[1]

Existing methods for large samples include Adam Smith's 'Widened Winzorised Mean' [4] who proposes a different robust estimator of location, the Winsorized mean. The most sophisticated approach we have found is Francesco Aldà & Benjamin Rubinstein's 'Berstein Mechanism' [5] for the release of functions whose training data consists of confidential information; it offers a weaker version of differential-privacy but bypasses much of the technical challenges of sensitivity analysis. Also, Antti Honkela et al. [6] have published a differentially private Bayesian linear regression estimator for genomic data.

All figures, tables, and algorithms are located in the appendix; however, since report focusses on econometric methods, we leave many of the technical definitions in text. We mention that all algorithms and all regression estimators (with exception of the SMDM estimator) were built on Python 3.6 and run on the Azure Notebooks Cloud server; and have henceforth been made publicly available on Github. Albeit of the complexity of implementing these algorithms through STATA, we were able to both wrangle data and perform regression estimation with greater efficiency, we were also able to include R's 'SMDM' robust estimator. More importantly, at that time there did not exist any publicly available implementation of Dwork & Lei's Propose-Test-Release (PTR) algorithm [7], nor did it appear to have been applied to any form of empirical research.[2] This is odd considering that it is a seminal piece in the literature of differentially private statistics.

[1] Existing Statistical Disclosure Limitation methods for continuous variables include adding noise, microaggregation, and shuffling data. Noise adding methods are comparatively simplistic compared to differential-privacy methods.

[2] To our understanding, this report includes the first application of this algorithm, a seminal piece in differential privacy literature, to any form of empirical data; excluding the original authors of course.

We find that our modified PTR algorithm performs better than the MOS algorithm at the cost of $1/10^{\text{th}}$ the ε, as well as being differentially private. We also find a robust-estimator, the SMDM-estimator, that returns very efficient Gaussian-noise infused estimates. And perhaps most interestingly, our proposed MAD algorithm overcomes the breakdown point of the PTR for small samples. In this report we keep our analysis 'short and sweet', since all the results are best apprehended visually in APPENDIX B.

1.1 DIFFERENTIAL PRIVACY

'Differential Privacy', first introduced in 2006 by Cynthia Dwork, is now the leading paradigm in the privacy-protecting perturbation of data. Intuitively speaking, it says any possible outcome of an analysis should be almost equally likely; independent of whether any individual opts in to, or opts out of, the data set. It offer an almost quantifiable guarantee that no risk is incurred by joining a statistical database.

The PTR algorithm[3] we use relies on (ε,δ)-*differential privacy*, a relaxation of the standard definition to allow for negligible δ.[4]

Def. [Distance Between Databases][8]:

The ℓ_1 norm of a database D is denoted $\|D\|_1$, and defined as $\|D\|_1 = \sum_{i=1}^{|N|}|x_i|$. Such that the distance between two databases, D, D', is $\|D - D'\|_1$

Def. [Differential Privacy][9]:

A randomised algorithm $\mathcal{M}$ with domain $\mathbb{N}^{|N|}$ gives (ε,δ)-*differential privacy* if, for all $S \subset \text{Range}(\mathcal{M})$ and for all $D, D' \in \mathbb{N}^{|N|}$ such that $\|D - D'\|_1 \leq 1$:

$$\mathbb{P}[\mathcal{M}(D) \in S] \leq e^{\varepsilon} \cdot \mathbb{P}[\mathcal{M}(D') \in S] + \delta$$

Def. [ℓ_1-sensitivity][10]:

The ℓ_1-sensitivity of a function f measures the magnitude by which a single individual's data can change the function f in the worst

[3] See APPENDIX A.3

[4]

case. It is therefore the uncertainty that must be introduced in the output in order to obscure the participation of a single individual. The ℓ_1-sensitivity of a function $f: \mathbb{N}^{|N|} \rightarrow \mathbb{R}^k$, for all $D, D' \in \mathbb{N}^{|N|}$, is:

$$\Delta f = \max_{\|D-D'\|_1=1} \|f(D) - f(D')\|_1$$

A key technical challenge in deriving differentially-private algorithms for regression estimates is the sensitivity of the regression. This is because the addition of one outliers to the dataset for an ordinary least-squares regression will dramatically change Δf. Even robust methods will break down if the same size is sufficiently small; as is often the case in empirical research.

For Chetty & Friedman, this value Δf is captured by $\frac{\chi}{n_i}$, the maximum observed sensitivity for all partitions divided by the partition size. However, they fail to include this parameter when scaling their noise. The reason this is, is because when using the Laplace noise addition mechanism, we require small Δf, since having larger Δf will flatten the distribution, and output large values of noise. This is clearly not desirable, so the workaround for Chetty & Friedman is to post-hoc scale noise between 0 and 1.

Also, Chetty & Friedman avoid discussing the appropriate scaling of Gaussian noise; whereby they apply the exact same post-hoc method over the standard normal distribution. However, in this report we follow the composition theorems for the Gaussian mechanism.

1.2 ADDITIVE NOISE MECHANISMS

Here we describe the Laplace and Gaussian noise adding mechanisms; it is worth noting that, unlike A. Smith, we avoid implementing the exponential mechanism.[5]

Def. [The Laplace Mechanism][11]:

Given any function $f: \mathbb{N}^{|N|} \rightarrow \mathbb{R}^k$, the Laplace mechanism is defined as:

$$\mathcal{M}_{\mathcal{L}}(D, f(\cdot), \varepsilon) = f(D) + (\mathcal{L}_1, \ldots, \mathcal{L}_k)$$

[5] The exponential mechanism is most often used for answering queries with utility functions

, where $\mathcal{L}_i$ are i.i.d. random variables drawn from $\text{Laplace}\left(0, b = \frac{\Delta f}{\varepsilon}\right)$;[6] in this report we denote the Laplace distribution as $\mathcal{L}(b)$; such that,

$$\mathcal{L}\left(b = \frac{\Delta f}{\varepsilon}\right) = \frac{1}{2\left(\frac{\Delta f}{\varepsilon}\right)} \exp\left(-\frac{|x|}{\frac{\Delta f}{\varepsilon}}\right)$$

Def. [ℓ_2-sensitivity][12]:

Before defining the mechanism for adding Gaussian noise, we define the ℓ_2-sensitivity of a function $f: \mathbb{N}^{|N|} \to \mathbb{R}^k$, for all $D, D' \in \mathbb{N}^{|N|}$, to be:

$$\Delta_2 f = \max_{\|D - D'\|_1 = 1} \|f(D) - f(D')\|_2$$

Def. [The Gaussian Mechanism][13]:

For $c^2 > 2\ln\left(\frac{1.25}{\delta}\right)$, the Gaussian Mechanism with parameter $\sigma \geq c \cdot \frac{\Delta_2 f}{\varepsilon}$ is (ε, δ)-*differentially private*:

$$\mathcal{N}\left(\sigma^2 = 2\ln\left(\frac{1.25}{\delta}\right)\left(\frac{\Delta_2 f}{\varepsilon}\right)^2\right) = \frac{1}{\sqrt{4\pi \ln\left(\frac{1.25}{\delta}\right)\left(\frac{\Delta_2 f}{\varepsilon}\right)^2}} \cdot \exp\left(-\frac{x^2}{4\ln\left(\frac{1.25}{\delta}\right)\left(\frac{\Delta_2 f}{\varepsilon}\right)^2}\right)$$

Whilst more useful for researchers downstream, draws from the Gaussian Mechanism will tend to be larger than their Laplacian counterpart;[7] this is because we require δ to be sufficiently (e.g. subpolynomially) small.

Def. [ε]:

It is routinely assumed that values of ε are within the unit interval $(0, 1)$. The intuition is that larger values of ε return a larger magnitude of random noise, as well as a weaker privacy guarantee.[8] When $\Delta f < \varepsilon$, which often arises in this report due to a

[6] see Appendix B. Figure 1.a

[7] see Appendix B. Figure 1.e

[8] It says, for example failing to be $(15, 0)$-*differentially private*, that there exists neighbouring databases and an output θ for which the ratio of probabilities of observing θ conditioned on the databases D or D' is large.

certain transformation of the data range, we produce extremely small, almost indistinguishable values of noise.

Def. [MOS mechanism]:

For the MOS algorithm we are to assume values of $\varepsilon \cdot 10^1$. This should output an excessively wasteful quantity of noise, as well as a very weak privacy guarantee, except that Chetty & Friedman remove ε and Δf from inside the noise additive mechanisms, i.e.:

$$\mathcal{L}_i = \sqrt{2} \cdot \frac{\Delta f}{\varepsilon \cdot 10^1} \cdot \mathcal{L}\left(\frac{1}{\sqrt{2}}\right)$$

, and;

$$\mathcal{N}_i = \sqrt{2} \cdot \frac{\Delta f}{\varepsilon \cdot 10^1} \cdot \mathcal{N}(1)$$

We compare the differences between using the standard differential privacy mechanisms and the MOS mechanisms in APPENDIX B.1. We see that by removing remove ε and Δf from the random draw, one preserves the scale of the distribution and thus draws from a very small range of values. Whereas including them into the distribution flattens the scale.

D. [Mean $\ell 2 - \text{Error}$]:

To compare the accuracy of each algorithm, we iterate each algorithm around 10 times and then calculate its mean squared error.

For all noise-infused outputs $\boldsymbol{\theta} = \hat{\beta} + \mathbf{z}$, with $\boldsymbol{\theta}, \mathbf{z} \in \mathbb{R}^N$, $\hat{\beta} \in \mathbb{R}^1$; we define the $\ell 2 - \text{Error}$ to be the Euclidean distance,

$$\text{E} = \text{d}\left(I\hat{\beta}, \boldsymbol{\theta}\right) = \left\| I\hat{\beta} - \boldsymbol{\theta} \right\|_2$$

For m iterations of whichever algorithm is specified, we denote the $\ell 2$-Error for each iteration as $\text{E}_m \in \text{E}$, and subsequently the Mean $\ell 2 - \text{Error}$ as,

$$\text{MSE} = \mathbb{E}[\text{E}] = \frac{1}{M} \sum_{m=1}^{M} \text{E}_m$$

2 THE PROBLEM

The goal is to release a statistic θ, i.e., the true statistic $\hat{\beta}$ plus noise z -- scaled by whichever algorithm we have chosen - within the context a very small dataset. Ideally, θ minimises privacy loss whilst preserving its statistical meaning and utility for further analysis.

We direct our attention to the linear regression model,

$$\mathbf{Y} = \mathbf{X}^T\boldsymbol{\beta} + \boldsymbol{\phi} \quad \text{w/ } \mathbf{Y}, \boldsymbol{\phi} \in \mathbb{R}^N; \ \mathbf{X} \in \mathbb{R}^{m \times N}; \ \boldsymbol{\beta} \in \mathbb{R}^m$$

The data set $D = \{(\mathbf{X}_i, y_i)_{i=1}^N\}$, denotes the empirically observed data aggregated across all cells, wherein we aim to estimate $\boldsymbol{\beta}$ -- albeit for most part we only consider the univariate problem with $m = 2$.

Furthermore, our analysis requires that we perform some level of partitioning on the dataset; i.e. $\lfloor N/p \rfloor$ distinct databases drawn from the original database.

For Chetty & Friedman, these partitions represent each individual Census tract for either the state or area level; and as such these partitions will greatly differ in size.

For Dwork & Lei, the Subsample & Aggregate algorithm takes these partitions to be of equal size, though we instead modify the algorithm to permit variations in size. Though I should point out, one could take each individual Census tract and perform this partition, at say $p = 2$, except that for very small tracts the Dwork & Lei noise-adding algorithm will tend to break down.

Fundamental to this comparison is the notion that we use regression estimators other than the standard ordinary least-squares to impute the scale of noise, due to the sensitivity problem.

Dwork & Lei use the most B-robust regression estimator $\mathbf{H}$, an M-estimator, whose output is defined as follows:

$$\hat{\beta} = \arg\min_{\beta} \left\{ \sum_{i=1}^{N} \frac{|y_i - \mathbf{X}_i^T\boldsymbol{\beta}|}{\|\mathbf{X}_i\|_2} \right\}$$

Instead, we build an ad-hoc MM-estimator, which combines both S and M estimation, in the hopes of reducing the influence of outliers in the dependent variable; it was only later on that we found another

variation on the MM-estimator for small sample sizes called SMDM, as described by Koller and Stahel [14] -- which we implemented into Python using R's 'lmrob' function. Regardless found both MM and SMDM to produce more or less the same utility.

2.1 REPLICATION PAPER

First of all, in this report we cannot make use of confidential data in our analysis for the very reason that we are making comparisons to true estimates. Chetty & Friedman did indeed apply their MOS methodology to their Opportunity Atlas project, however, to present their implementation of MOS they refer to a synthetic data set. Likewise, for the replication component of this report we present our analysis of the MOS method with this same dataset.

The most widely used approaches to obfuscate confidential information for microdata -- in social science -- are local cell suppression, wherein we omit cells for categorical data; and shuffling; wherein we switch values across different cells. These methods are, as we have discussed, fail to suffice. The authors make the case that these methods also affect the utility of the data for analysis, and that researchers should be instead wanting to perturb the output. We concern ourselves with this latter approach, since it coincides with our interest in producing differentially private estimates.

When implementing the MOS we discovered certain problem inherent to the algorithm, firstly the description of appropriate ε values were assumed to be in the set $\{1,2,..,10\}$, which is an order of magnitude of 1 greater than what is ordinarily used; i.e. since we are dividing the noise by ε, ordinary values would scale-up the noise by 10^1. We ignore this problem when analysing the MOS algorithm since it is not differentially private anyhow.

Secondly, the method by which we are supposed to scale the noise leaves little to be said about privacy; the instruction is that we first draw the random component from $\mathcal{L}\left(0,\frac{1}{\sqrt{2}}\right)$, and then scale by values which are released publicly, such that the output is prone to *post-processing* -- one can easily reconstruct the true statistic with an error of $\mathcal{L}\left(0,\frac{1}{\sqrt{2}}\right)$. In this sense the MOS is worse than

standard SDL noise adding methods, since we publicly release the non-random factor by which the noise is scaled.

In APPENDIX B.2. we see that Chetty & Friedman make use of MOS algorithm to output noise-infused statistics that are almost indistinguishable from the true statistic (at the aggregate level), for whichever dataset one is using. However we also find that our modified Propose-Test-Release algorithm and our Mean-Absolute-Deviation variation quickly converges to a similar Mean $\ell_2 - \text{Error}$ for ε close to 1, which is at most $1/10^{\text{th}}$ the amount of ε used in the MOS.

We move all discussion of comparing algorithms for our different datasets towards the end of the paper, since the results are more or less the same.

2.2 EXTENSION PAPER

The required extension component for this research report is, in effect, to take note of the MOS approach, particularly the inverted use of subsampling and aggregation, but actually provide a privacy guarantee found in differential privacy.

The 'Propose-Test-Release' algorithm outputs a noise infused estimate for any dataset which is subject to some level of partitioning, we first consider the case where the estimate we wish to output is the aggregate of all specified tract levels. The reason we pursue this line of thinking is because the PTR is prone to breakdown in smaller datasets, and later on we adapt the PTR to release the estimates at the tract level. This approach is similar to Chetty & Friedman's calculation of maximum observed sensitivity.

A slight technical challenge we found in pursuing the PTR algorithm, was that in transforming the range of data into the unit interval, we unwittingly created a dataset with much lower tolerance levels. Our workaround was to instead scale the noise using the regression residuals.

The basic intuition is that we are first computing a scale for the noise, which is based on the interquartile range of the $\hat{\beta}_i$'s -- where each $i \in \{1,2,\ldots,N\}$ is each partition. In this first computation we also check the sensitivity of the interquartile range against alternative $\hat{\beta}_{d,i}$ -- which are iteratively recomputed the estimates

for each partition. If any partition is too sensitive, it returns $\perp$.

We also infuse this scale with noise drawn from the relevant mechanism; one point of difference between our implementation and Dwork & Lei's is that we take the absolute value of the noise parameter, since the noise is treated as a power, any negative values would obliterate the scaling apparatus and force an output of $\perp$.

We use this scale as the tolerance level for comparing each $\hat{\beta}_i$ against its respective $\hat{\beta}_{d,i}$, by comparison we mean the absolute difference. If any $\hat{\beta}_i$ is too sensitive, we return $\perp$ for that partition; otherwise we divide the scale by a proportion of the sample size, and denote it to be the sensitivity, Δf. We then draw from the additive noise mechanism and add it to the true statistic.

The main deviation from Dwork & Lei -- and in general, the subsample & aggregate framework -- is that we work with data partitions of differing sizes, and as such every mention of the number of partitions, p, is replaced with the ratio N/n_i, where n_i is the size of partition p_i, and N is the aggregate sample size.

2.3 PUM DATA

We work with Public-Use Microdata (PUM) from the US Census 2000 for the state of Alaska at the 5% sample level. Our first analysis is a regression of Total Incomes $\geq$ 90$^{\text{th}}$-percentile, on an Individual's Age; we also remove any and all individuals whom are living in 'group quarters', or squatting:

$$\mathbf{Y}_{\textit{Total Income}_{90^{\text{th}}}} = \boldsymbol{\beta_0} + \boldsymbol{\beta_1}\mathbf{X}_{\textit{Age}} + \boldsymbol{\phi}$$

The total sample size for this first empirical dataset is $N_{\textit{Total Income}_{90^{\text{th}}}} = 2954$, which is partitioned into $p = 100$ groups of different sizes. Our second analysis regresses Total Incomes $\geq$ 99$^{\text{th}}$-percentile, on an Individual's Age and the Value of the residence in which they reside:

$$\mathbf{Y}_{\textit{Total Income}_{99^{\text{th}}}} = \boldsymbol{\beta_0} + \boldsymbol{\beta_1}\mathbf{X}_{\textit{Age}} + \boldsymbol{\beta_2}\mathbf{X}_{\textit{Value}} + \boldsymbol{\phi}$$

The total sample size for this second empirical dataset is $N_{Total\ Income_{99^{\text{th}}}} = 294$, which is partitioned into $p = 5$ groups of different sizes.[9] We make two modifications to the original dataset that are to be mentioned. Firstly, we extend the data-subset $d_{Value} \subset D$ to include individuals who rent. We estimate the value of these properties with the function V:

$$V(\text{rent}) = \frac{1000}{4.35} \cdot \text{rent}$$

Which first transforms monthly rent into weekly rent and then multiplies this by 10^2; e.g. \$800p/w $\rightarrow$ \$800,000. We believe this number to be a conservative estimate.

The second modification is to transform the range of each data-subset $d_k \subset D$ into the unit interval with the function U, which maps the interval $[a_k, b_k]$ into the interval $[0,1]$:

$$U(d_k) = \frac{d_k - a_k}{b_k - a_k}$$

, where $a_k = \arg\min d_k$ is the maximum value in data-subset d_k, and similarly $b_k = \arg\max d_k$. We do this because we wish to have a similar data environment as Chetty & Friedman who rank incomes. This conveniently creates an upper and lower bound for the dataset D, on which we can iterate the inclusion of outliers. In all other situations, we can make use of a robust estimators' vanishing point to define the bounds of the data.

3. DESCRIPTIVE ANALYSIS

The three case problems we consider are: the synthetic univariate problem which uses the synthetic data set; the empirical univariate problem, $\mathbf{Y}_{Total\ Income} = \boldsymbol{\beta_0} + \boldsymbol{\beta_1}\mathbf{X}_{Age} + \boldsymbol{\phi}$; and lastly the empirical bivariate problem; $\mathbf{Y}_{Total\ Income_{99^{\text{th}}}} = \boldsymbol{\beta_0} + \boldsymbol{\beta_1}\mathbf{X}_{Age} + \boldsymbol{\beta_2}\mathbf{X}_{Value} + \boldsymbol{\phi}$.

For each case problem, we analyse three algorithms: the MOS, PTR, and MAD; under the two additive noise frameworks: the Laplace mechanism, and the Gaussian mechanism.

[9] These specification for these 5 groups is the Area code for the Alaskan PUMs.

We compare the efficiency of each algorithm four different regression estimators: the OLS, the Winsorized OLS, the MM-estimator, and the SMDM-estimator.

The main finding ubiquitous to all case problems was that the difference between each regression estimator was the 'speed' at which each algorithm approached some specific MSE value for increasing ε. The robust, MM and SMDM, overall had much more efficient and accuracy outcomes than the non-robust methods.

In APPENDIX B.3 and B.4, we see that the PTR and MAD algorithms quickly approach $\mathrm{MSE} = 0$ in both the Laplacian and Gaussian mechanisms, for increasing ε. The bivariate problem, prone to outputting $\perp$ when using the PTR framework, will instead produce very efficient noise-infused estimates with the MAD algorithmic variation.

Our implementation of the Local-Release Framework[10] is extremely efficient when using the MAD algorithm, thus permitting the release of estimates based on extremely small samples, often without having to rely on the much more computationally expensive robust methods.

4. CONCLUSION

Building on Chetty & Friedman's innocent simplification of differential privacy techniques, we propose the first differentially private release mechanism for statistics based on small samples. This method outperforms Chetty & Friedman's MOS, which outperforms widely-used methods of disclosure limitation in both privacy loss and statistical bias.[11] Although we skipped over its application to other regression statistics, for example, differentially private standard errors; this was intentional, since these statistics passed all algorithms without outputting any $\perp$. Similar to the MOS algorithm, our modified PTR and MAD algorithms can be easily applied to any statistic of interest to social science.

[10] See APPENDIX A.5 and C

[11] Although we have also shown that the MOS is prone to post-processing, raising doubts to the claims made by Chetty & Friedman.

An extension to this work would be to analyse the changes in in a statistics' meaning after adding noise. Since the most arbitrary element in differential privacy is the specification of the ε parameter which is delegated to policy makers, it would be useful to have a metric to measure the loss in predictive power in every additional ε.

APPENDIX

A.1 MM-Estimator

The algorithm MM [16]

- Algorithm
  1. $\hat{\beta}$ using OLS
  2. S-estimator:
     a. OLS residuals $\boldsymbol{\phi} = \mathbf{Y} - \mathbf{X}^T\widehat{\boldsymbol{\beta}}$
     b. Scale:
        - if iteration $= 1$, do:
          - let $\text{MAD} = \text{median}(|\phi - \text{median}(\phi)|)$
          - $\hat{\sigma} = \frac{\text{MAD}}{\Phi^{-1}\left(\frac{3}{4}\right)}$[12]
        - else if iteration > 1, do:
          - let $K = 0.199$
          - $\hat{\sigma} = \sqrt{\frac{1}{N \cdot K} \cdot \sum_{i=1}^{N} w_i \phi_i^2}$
     c. Bisquare ratio $u = \frac{\phi}{\hat{\sigma}}$
     d. Weighted value:
        - let Tukey's biweight $c = 1.547$
        - if iteration $= 1$, do:
          - if $|u| \leq c$, do:
            - $w_i = \left(1 - \left(\frac{u}{c}\right)^2\right)^2$
          - else if $|u| > c$, do:
            - $w_i = 0$
        - else if iteration > 1, do:
          - if $|u| \leq c$, do:
            - $w_i = \frac{u^2}{2} - \frac{u^4}{2c^2} + \frac{u^6}{6c^2}$
          - else if $|u| > c$, do:

[12] Φ^{-1} is the reciprocal of the quantile function for the standard normal distribution

- $w_i = \frac{c^2}{6}$

e. $\hat{\beta}_S$ with weighted least-squares

f. iterate till convergence

3. S-estimator residuals $\boldsymbol{\phi} = \mathbf{Y} - \mathbf{X}^T\widehat{\boldsymbol{\beta}}_S$

4. $\hat{\sigma}_S = \sqrt{\frac{1}{N \cdot K} \cdot \sum_{i=1}^{N} w_i \phi_i^2}$

5. Now compute M-estimator with $\hat{\sigma}_S$:

a. Bisquare ratio $u = \frac{\phi}{\hat{\sigma}_S}$

b. Weighted value:

- let Tukey's biweight $c = 4.685$
- if $|u| \leq c$, do:
  - $w_i = \left(1 - \left(\frac{u}{c}\right)^2\right)^2$
- else if $|u| > c$, do:
  - $w_i = 0$

c. $\hat{\beta}_{MM}$ with weighted least-squares

d. iterate till convergence

A.2 MAXIMUM OBSERVED LOCAL SENSITIVITY

The algorithm MOS[13]

- Algorithm

1. For each partition, p_i:
   - $\hat{\beta}$ using OLS
   - $\hat{\beta}_d$ using OLS for each adjacent dataset D_d[14]
   - Local sensitivity:
     - $LS = \max\{\hat{\beta} - \hat{\beta}_d\} \;\; \forall d \in \left\{1, \ldots, 2^{\#(\hat{\beta})}\right\}$
2. MOS-Envelope:
   - $\chi = \max\{n_i \times LS_i\}$
     - where n_i is the size of partition p_i.

[13] We ignore the MOS calculation of standard errors because it is not pertinent to the research report. The reason this is is because the standard errors were not prone to the same technical problems as the covariates, and in fact passed through the PTR algorithm with ease.

[14] Where D_d is equivalent to D but with an additional point, (x_d, y_d) where $d \in \left\{1, \ldots, 2^{\#(\hat{\beta})}\right\}$. For example, $(x_d, y_d) \in \{(0,0), (0,1), (1,0), (1,1)\}$ for the univariate problem. Since we are working with variables bounded between $[0,1]$, these points represent the maximum and minimum values any combination of outliers could obtain.

3. Scaled noise:
   - $z_i = \frac{\chi\sqrt{2}}{\varepsilon \cdot n_i} \cdot \omega$
     - where $\omega \sim \mathcal{L}\left(0, \frac{1}{\sqrt{2}}\right)$, or $\omega \sim \mathcal{N}(0,1)$
     - and $\varepsilon \in \{1, 2, \dots, 10\}$
4. Infuse true statistics with noise:
   - $\theta_i = \hat{\beta}_i + z_i$
   - $\tilde{n} = n_i + z_i \cdot \frac{n_i}{\chi}$
5. Return θ, $\tilde{n}$, χ

A.3 PROPOSE-TEST-RELEASE FRAMEWORK

The algorithm PTR[15]

Here, $\delta = \frac{1}{2}\left(\frac{N}{p}\right)^{-\varepsilon \ln\left(\frac{N}{p}\right)}$, where p is the total number of partitions.[16]

Input $(\mathcal{R}, D, p, \varepsilon)$[17]

- Algorithm
  0. Define dataset D to be the aggregate of all census tracts p_i for some area
  1. Define true statistic $\hat{\beta}_t$ using OLS (or $\mathcal{R}$)
  2. In each partition, p_i:
     - $\hat{\beta}$ using $\mathcal{R}$ and store in B
     - $\hat{\beta}_d$ using $\mathcal{R}$ for each adjacent dataset D_d
  3. Compute bins $H_i(B) = \log_{1+\frac{1}{\ln n_i}}\left(\mathrm{IQR}(B)\right)$
  4. In each partition, p_i:
     - Replace $\hat{\beta}$ with $\hat{\beta}_d$ and store new B in B_d
     - Recompute bin for partition $H_i(B_d) = \log_{1+\frac{1}{\ln n_i}}\left(\mathrm{IQR}(B_d)\right)$
     - Repeat a-b for each $\hat{\beta}_d$:
       - If any $|H_i(B) - H_i(B_d)| > 1$:

[15] The description of this algorithm in Dwork & Lei's paper is much harder to decipher than should be. We believe our interpretation is the closest to the original intent of the authors. This may be why we could not find any papers that use this algorithm.

[16] We did not bother calculating δ's for scaled partition sizes because the size of δ is very small regardless.

[17] We denote $\mathcal{R}$ to be the regression estimator used. In this paper we consider OLS, Winsorized OLS, MM, and SMDM estimators

- Let $s_i = \perp$
- Else:
  - Let $s_i = \mathrm{IQR}(B) \cdot \left(1 + \frac{1}{\ln n_i}\right)^{|z_s|}$
    - If using Laplace mechanism:
      - $z_s \sim \mathcal{L}\left(\frac{1}{\varepsilon}\right)$
    - Else if using Gaussian mechanism:
      - $z_s \sim \mathcal{N}\left(\frac{2\ln(1.25/\delta)}{\varepsilon^2}\right)$

5. For each s_i:
   - If $s_i = 0$:
     - Let $h_i = \frac{1}{\sqrt{N}}$
   - Else if $s_i \neq \perp$:
     - let $h_i = s_i \cdot N^{-\frac{n_i}{2N}}$
6. In each partition, p_i:
   - Compute bins $J_i = \left|\hat{\beta} - \hat{\beta}_d\right|$ for all d
   - If any $J_i > h_i$:
     - Let $\theta_i = \perp$
   - Else:
     - Let $\theta_i = \hat{\beta}_t + z_i$
       - If using Laplace mechanism:
         - $z_i \sim \mathcal{L}\left(\frac{h_i}{\varepsilon}\right)$
       - Else if using Gaussian mechanism:
         - $z_i \sim \mathcal{N}\left(2\ln\left(\frac{1.25}{\delta}\right)\left(\frac{h_i}{\varepsilon}\right)^2\right)$
7. Return $\underset{\theta_i}{\operatorname{argmin}}\left\{\left\|\hat{\beta}_t - \theta_i\right\|_2\right\}$

The regression algorithm PTR, is approximately $\left((2^p + 3^p + 1)\varepsilon, (2p + 2^p)\delta\right)$-*differentially private*.

A.4 MEAN-ABSOLUTE-DEVIATION PTR FRAMEWORK

A variation of the PTR we discovered to be very effective for these kinds of small sample problems, appertained replacing the interquartile-range for the $\hat{\beta}$'s in each partition.[18]

Instead we propose using the median absolute deviation of the residuals for each partition and computing the $\ell 2$-norm.[19] This $\ell 2$-norm then denotes the sensitivity, Δf, of the $\hat{\beta}$'s in relation to their residuals.

The algorithm MAD[20]

- We replace steps 3-4 in algorithm PTR
  3. In each partition, p_i:
     - $\mathcal{R}_i$ residuals $\boldsymbol{\phi_i}$ and store in B
     - Let $\hat{\sigma}_i = \frac{\text{MAD}(\boldsymbol{\phi_i})}{\Phi^{-1}\left(\frac{3}{4}\right)}$
     - Compute bins $H_i(B) = \log_{1+\frac{1}{\ln n_i}}(\hat{\sigma}_i)$
  4. In each partition, p_i:
     - Store each $\mathcal{R}_d$ residual $\boldsymbol{\phi}_d$ in B_d
     - Let $\hat{\sigma}_d = \frac{\text{MAD}(\boldsymbol{\phi}_d)}{\Phi^{-1}\left(\frac{3}{4}\right)}$ for each $\boldsymbol{\phi}_d$
     - Recompute bin for partition $H_i(B_d) = \log_{1+\frac{1}{\ln n_i}}(\hat{\sigma}_d)$
     - If any $|H_i(B) - H_i(B_d)| > 1$:
       - Let $s_i = \perp$
     - Else:
       - Let $s_i = \|\hat{\sigma}\|_2 \cdot \left(1 + \frac{1}{\ln n_i}\right)^{|z_s|}$
         - If using Laplace mechanism:
           - $z_s \sim \mathcal{L}\left(\frac{1}{\varepsilon}\right)$
         - Else if using Gaussian mechanism:
           - $z_s \sim \mathcal{N}\left(\frac{2\ln(1.25/\delta)}{\varepsilon^2}\right)$

[18] The downside is that this algorithm is many times more computationally expensive than the PTR algorithm. We only bothered to iterate it twice when calculating the MSE.

[19] We considered using a geometric median approach for calculating the mean absolute deviation, however, we are considering only partitions of differing sizes, such that our residual vectors are of different dimensions.

[20] We urge readers to avoid confusing the MAD algorithm with our usage of "MAD" in calculating the mean-squared-error of the residuals.

A.5 LOCAL PTR FRAMEWORK

The final contribution in this report is the proposal of the Local Propose-Test-Release framework; which we would use to release estimates for each tract, rather than the aggregate; and thus allows us to bypass the problems of further partitioning small samples.

The algorithm LPTR

- We replace steps 6-7 in algorithm PTR
  6. In each partition, p_i:
     - Compute bins $J_i = \left|\hat{\beta} - \hat{\beta}_d\right|$ for all d
     - If any $J_i > h_i$:
       - Let $\theta_i = \perp$
     - Else:
       - Let $\theta_i = \hat{\beta}_i + z_i$
         - If using Laplace mechanism:
           - $z_i \sim \mathcal{L}\left(\frac{h_i}{\varepsilon}\right)$
         - Else if using Gaussian mechanism:
           - $z_i \sim \mathcal{N}\left(2\ln\left(\frac{1.25}{\delta}\right)\left(\frac{h_i}{\varepsilon}\right)^2\right)$
  7. Return $\boldsymbol{\theta}$
     - where $\theta_i \in \boldsymbol{\theta}$

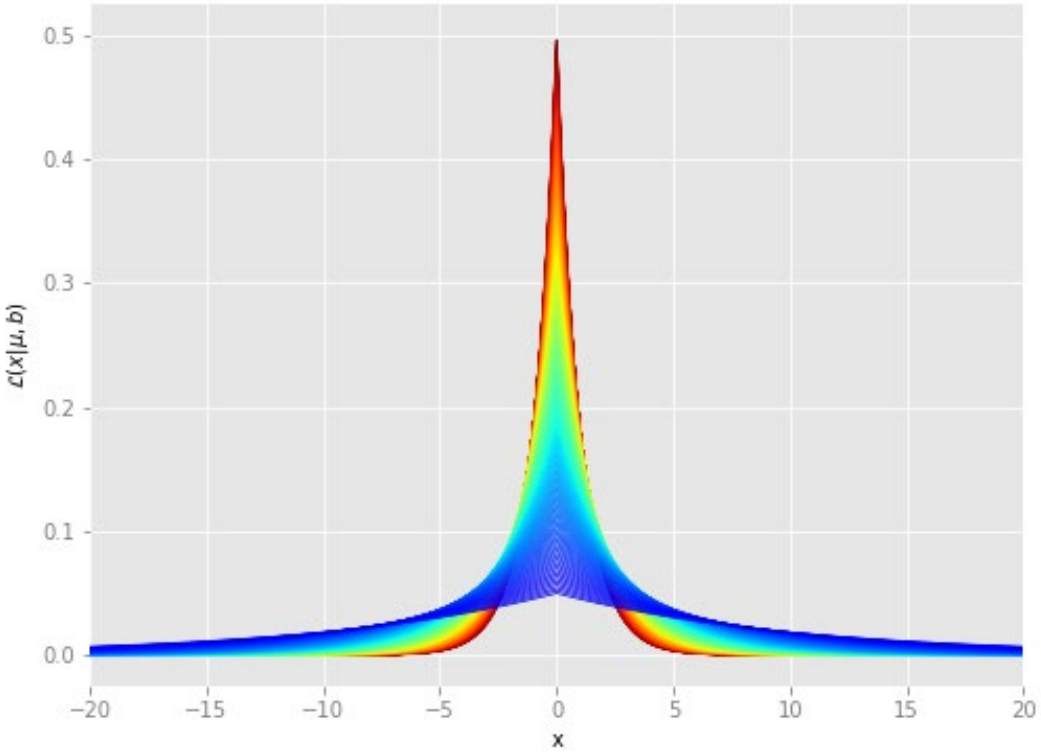
In this report we don't bother measuring the mean squared error of each tract estimate, and instead tabulate a count of non-$\perp$ outputs for each of the datasets we are analysing.[21]

[21] See APPENDIX C

B.1 : ADDITIVE NOISE MECHANISMS

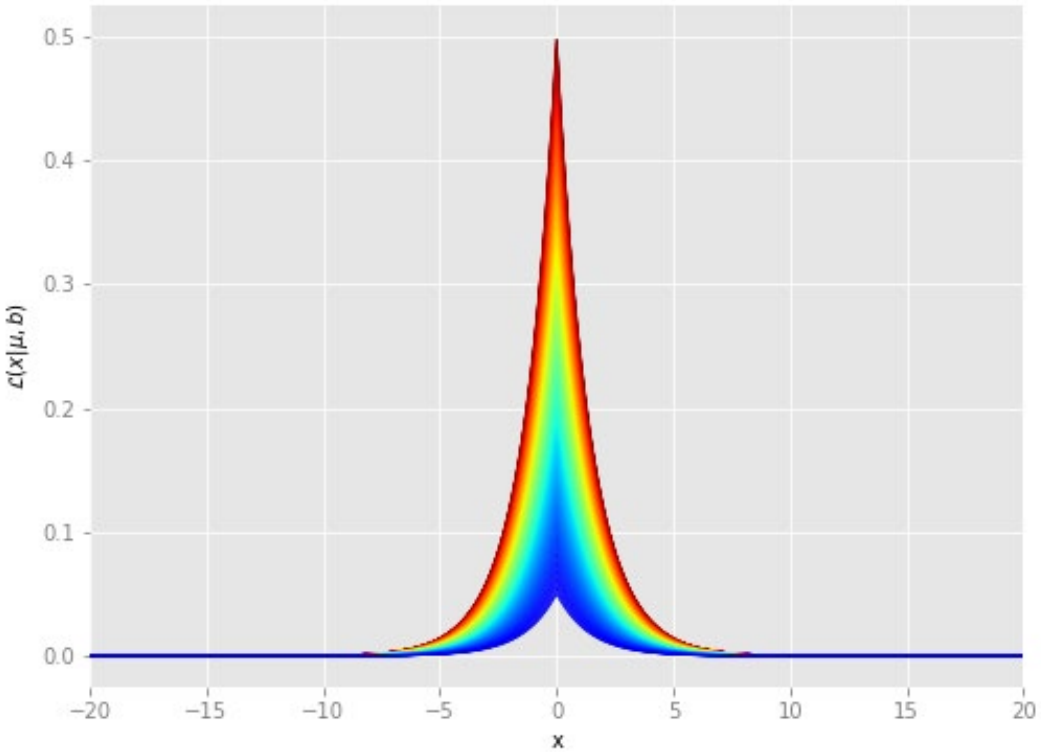
a. LAPLACE MECHANISM - effect of including Δf and ε

$\mathcal{L}(0, \Delta f/\varepsilon)$; $\Delta f = 1$



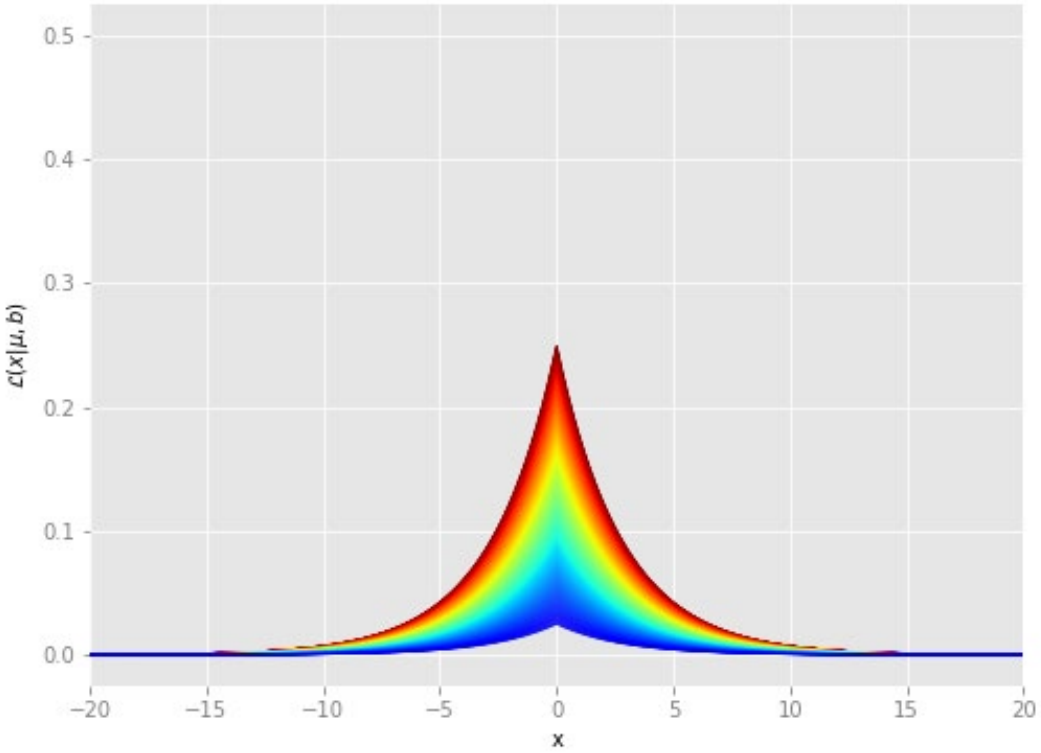
b. MOS LAPLACE MECHANISM - effect of excluding ε

$\mathcal{L}(0, 1/\sqrt{2}) \times \Delta f\sqrt{2}/(\varepsilon \cdot 10^{1)}$; $\Delta f = 1$



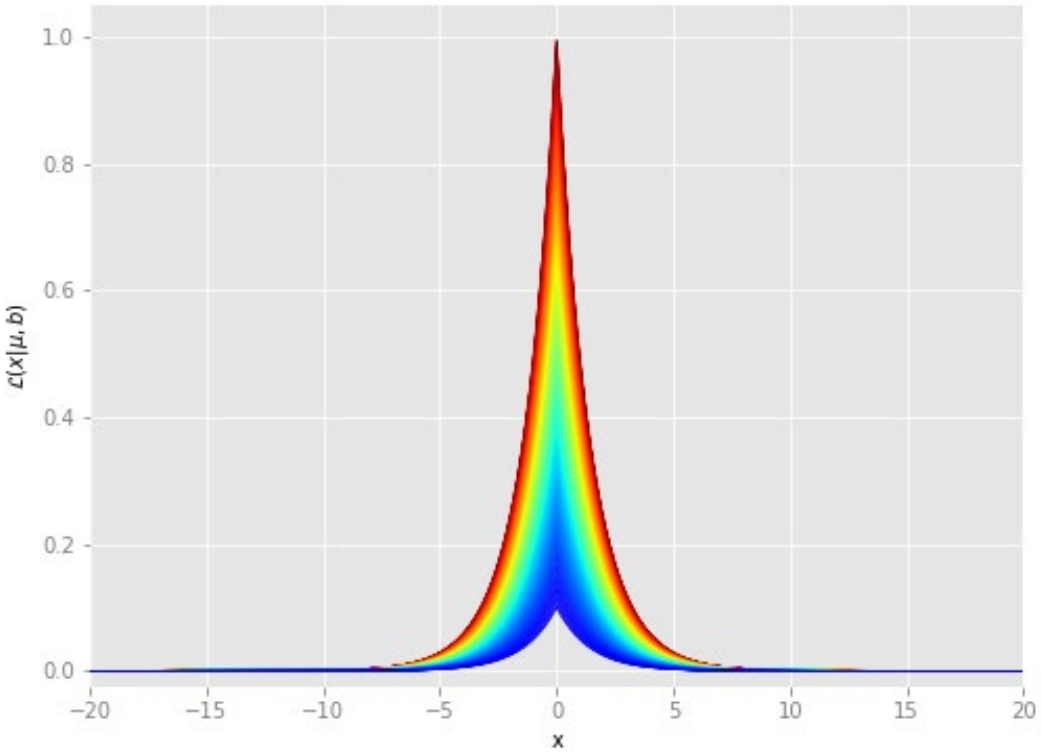
c. MOS LAPLACE MECHANISM – effect of including Δf

$$\mathcal{L}(0, \Delta f/\sqrt{2}) \times \sqrt{2}/(\varepsilon \cdot 10^{1)} \quad ;\Delta f = 2$$



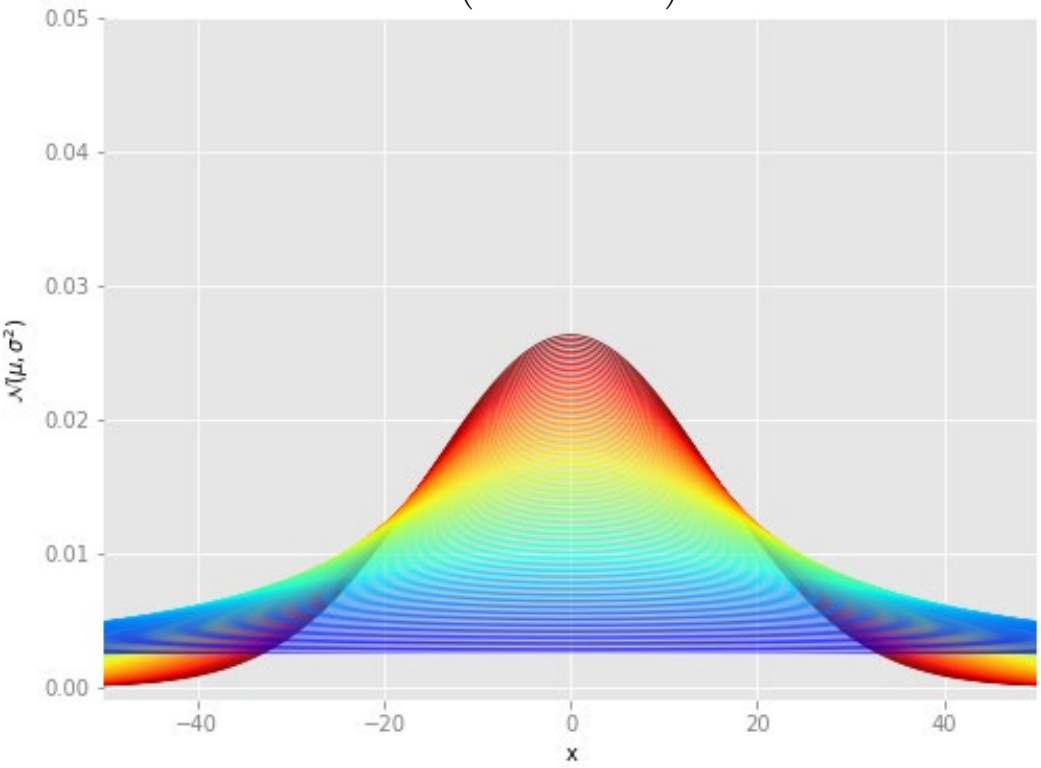
d. MOS LAPLACE MECHANISM – effect of excluding both Δf and ε

$$\mathcal{L}(0, 1/\sqrt{2}) \times \Delta f\sqrt{2}/(\varepsilon \cdot 10^{1)} \quad ;\Delta f = 2$$



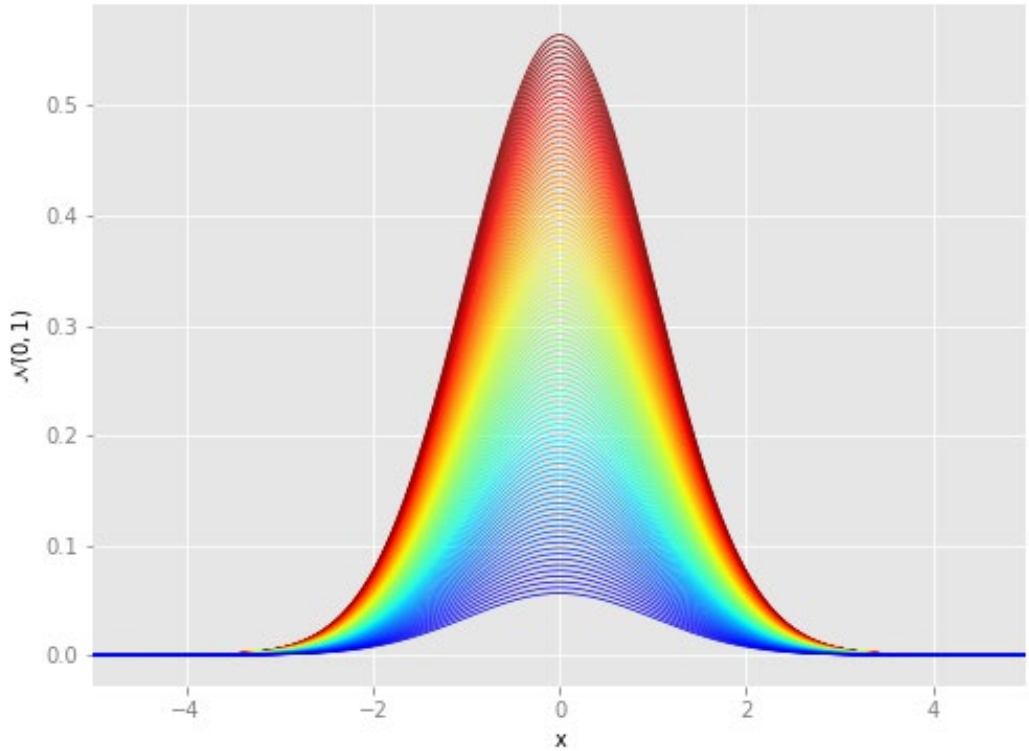
e. GAUSSIAN MECHANISM

$$\mathcal{N}\left(0, \ln\left(\frac{1.25}{\delta}\right)^2 \left(\frac{\Delta f}{\varepsilon}\right)^2\right)$$

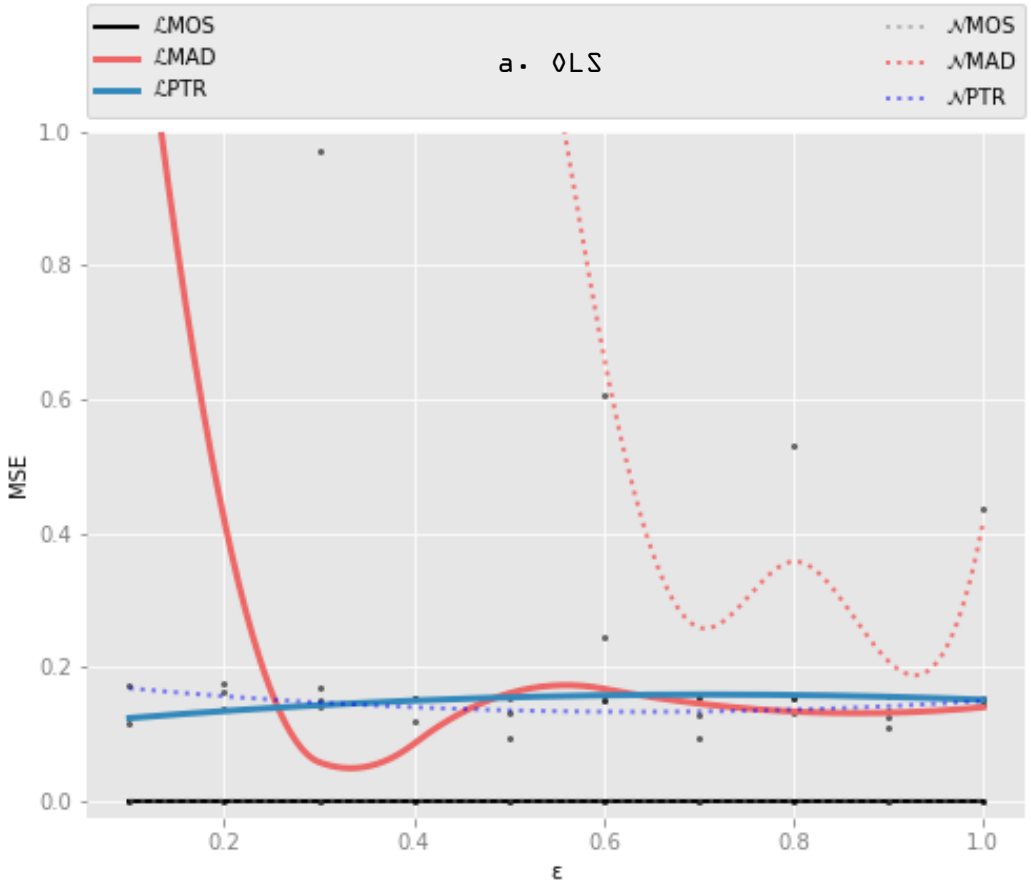


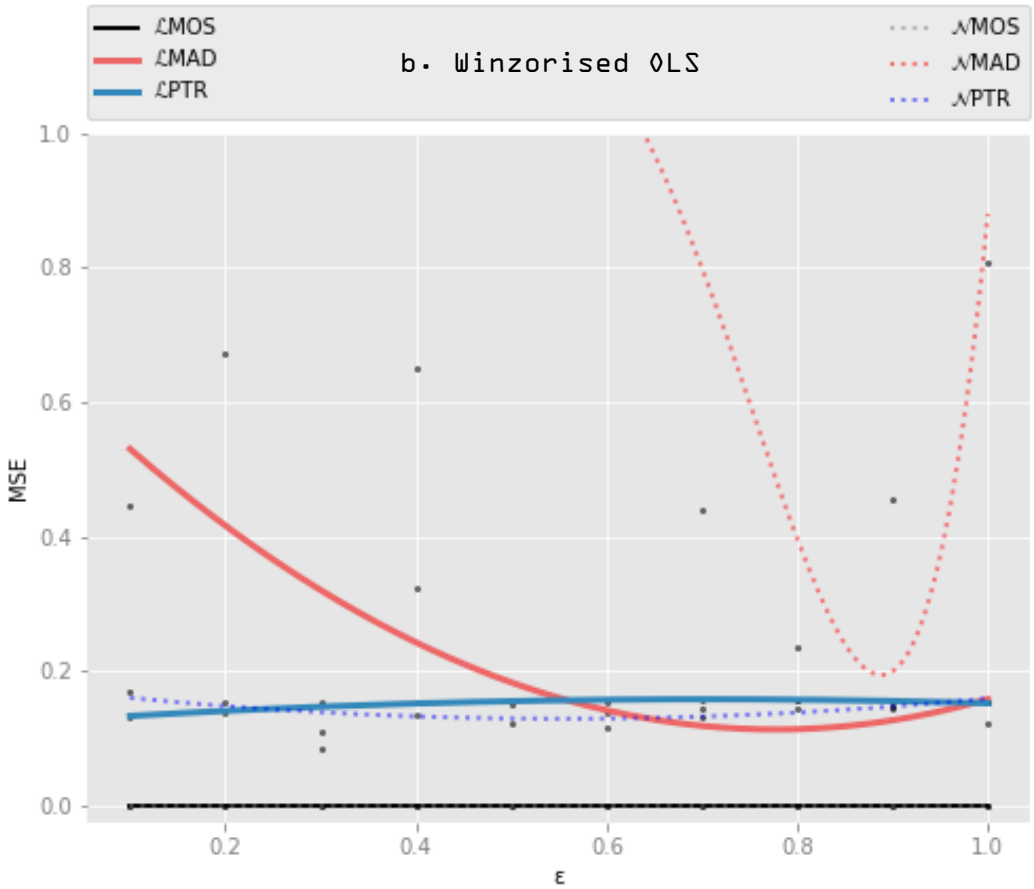
f. MOS GAUSSIAN MECHANISM - scaled standard normal

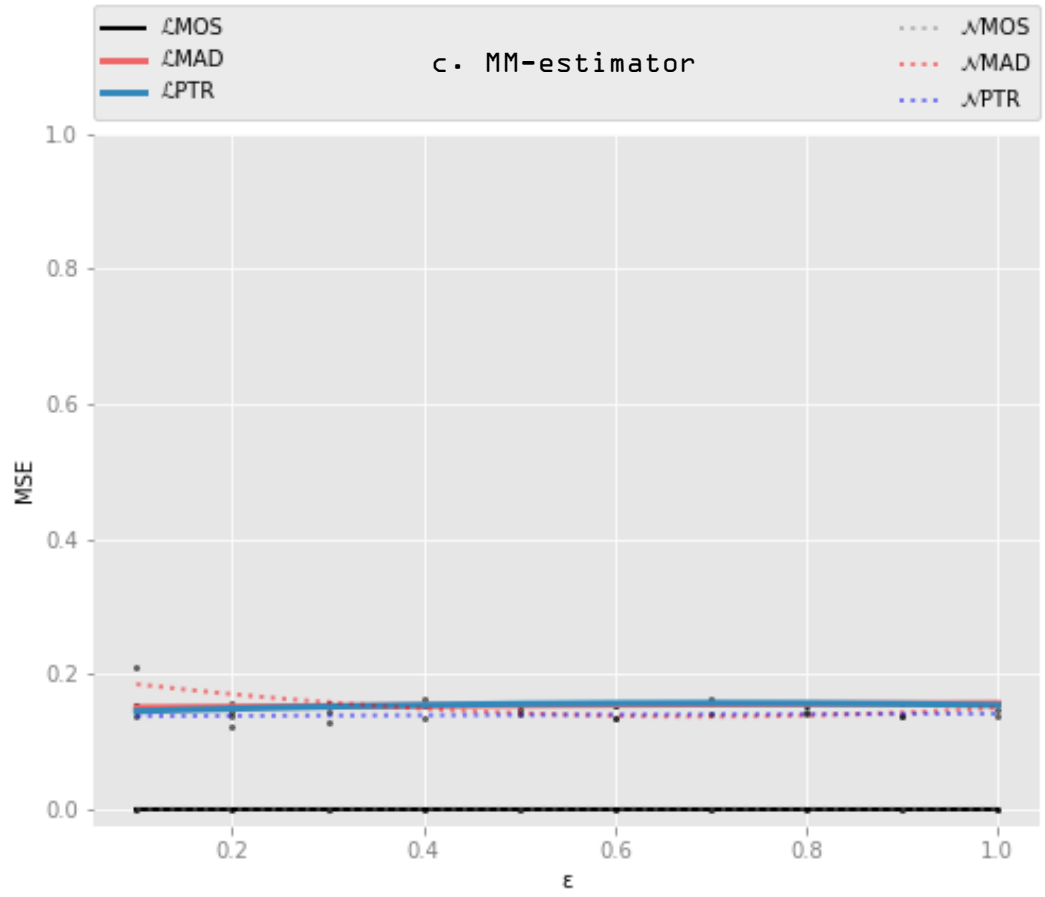
$$\mathcal{N}(0,1) \times \sqrt{2}/(\varepsilon \cdot 10^{1)}$$

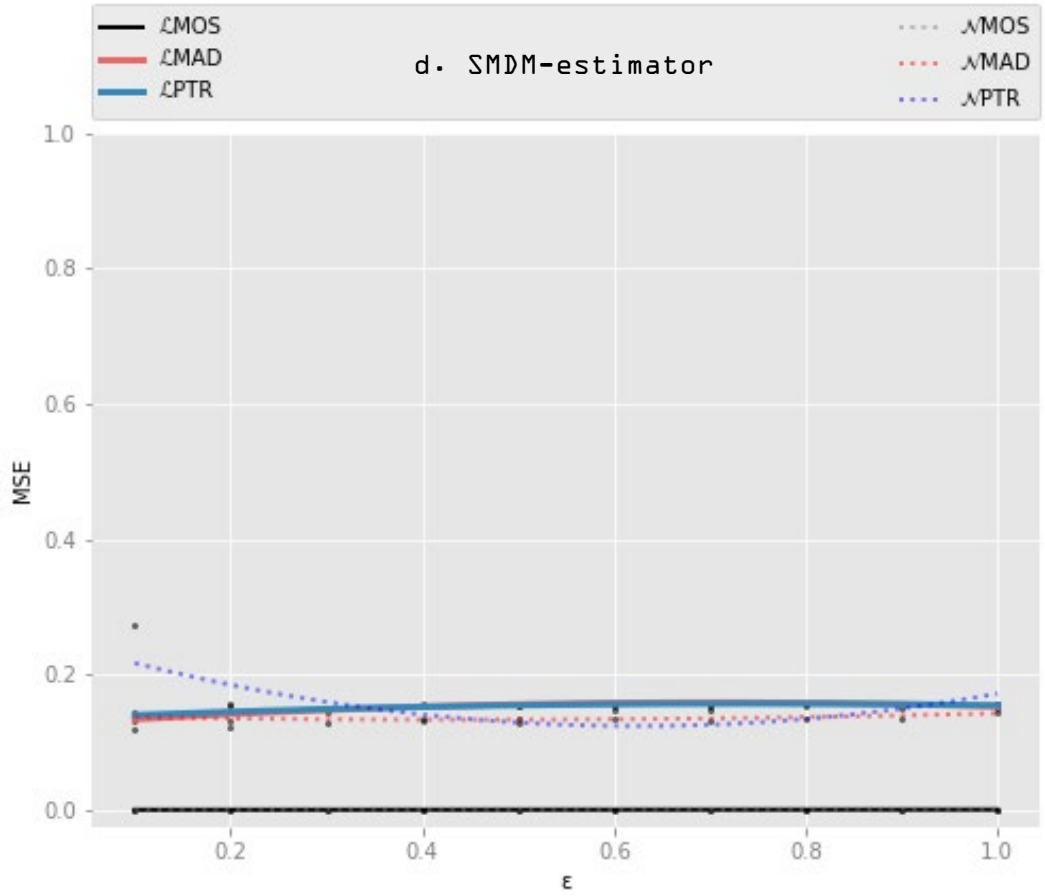


B.2: SYNTHETIC UNIVARIATE PROBLEM

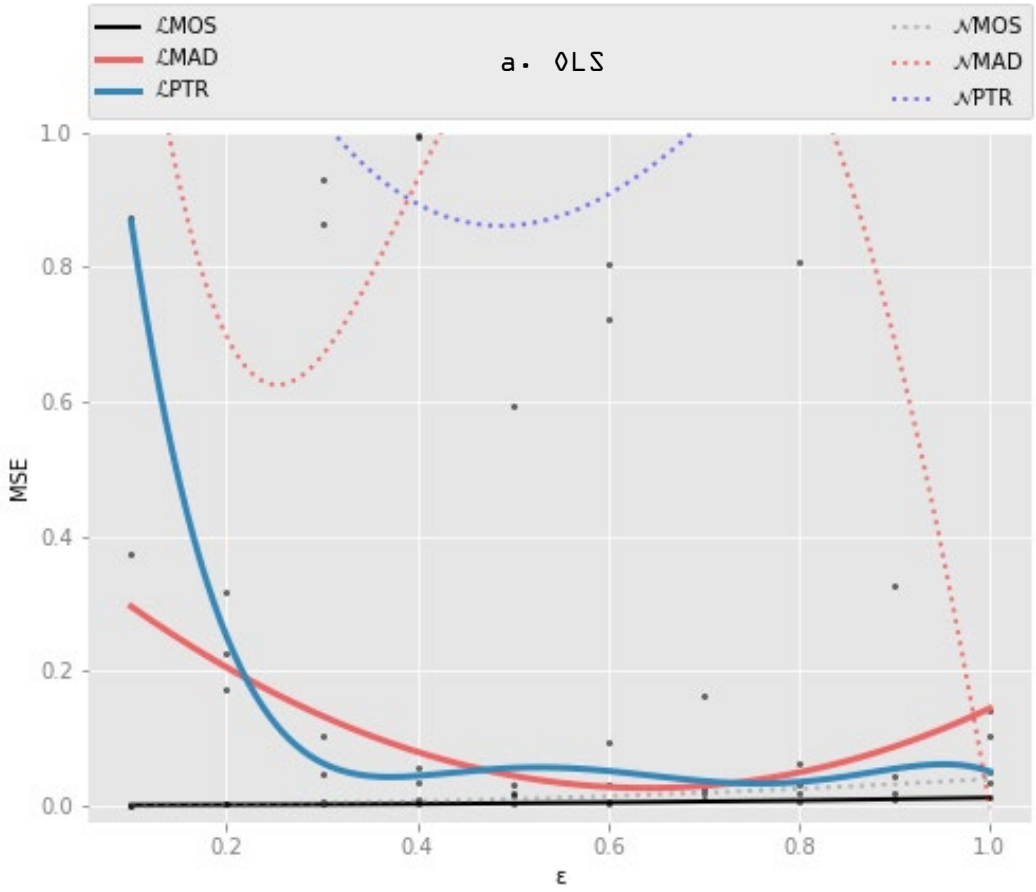


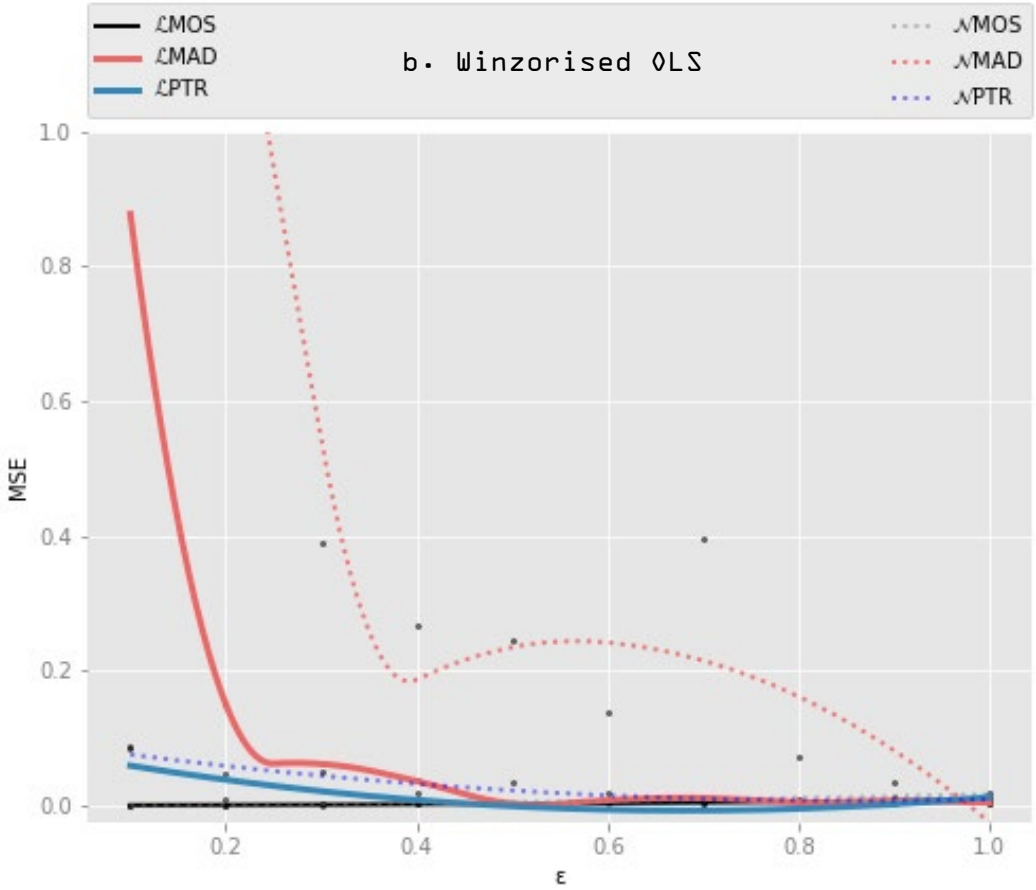


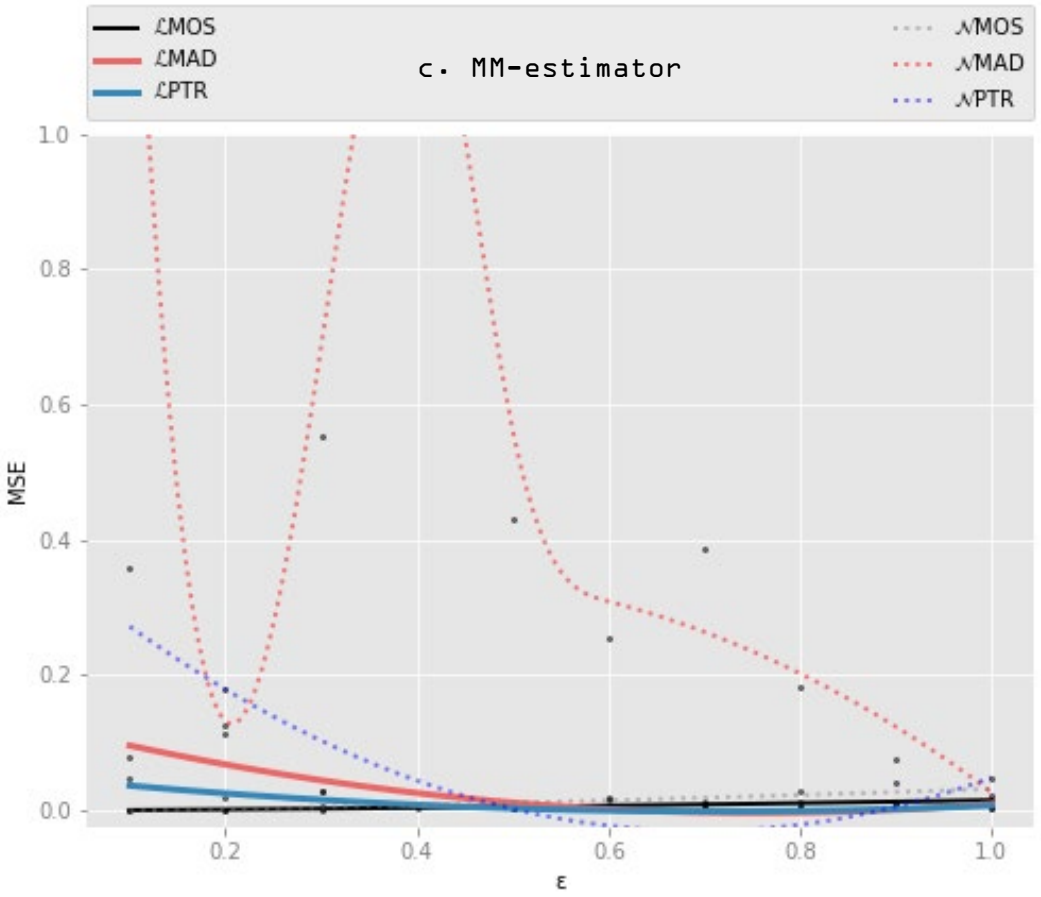

ℒMOS
ℒMAD
ℒPTR
c. MM-estimator
𝒩MOS
𝒩MAD
𝒩PTR
MSE
ε

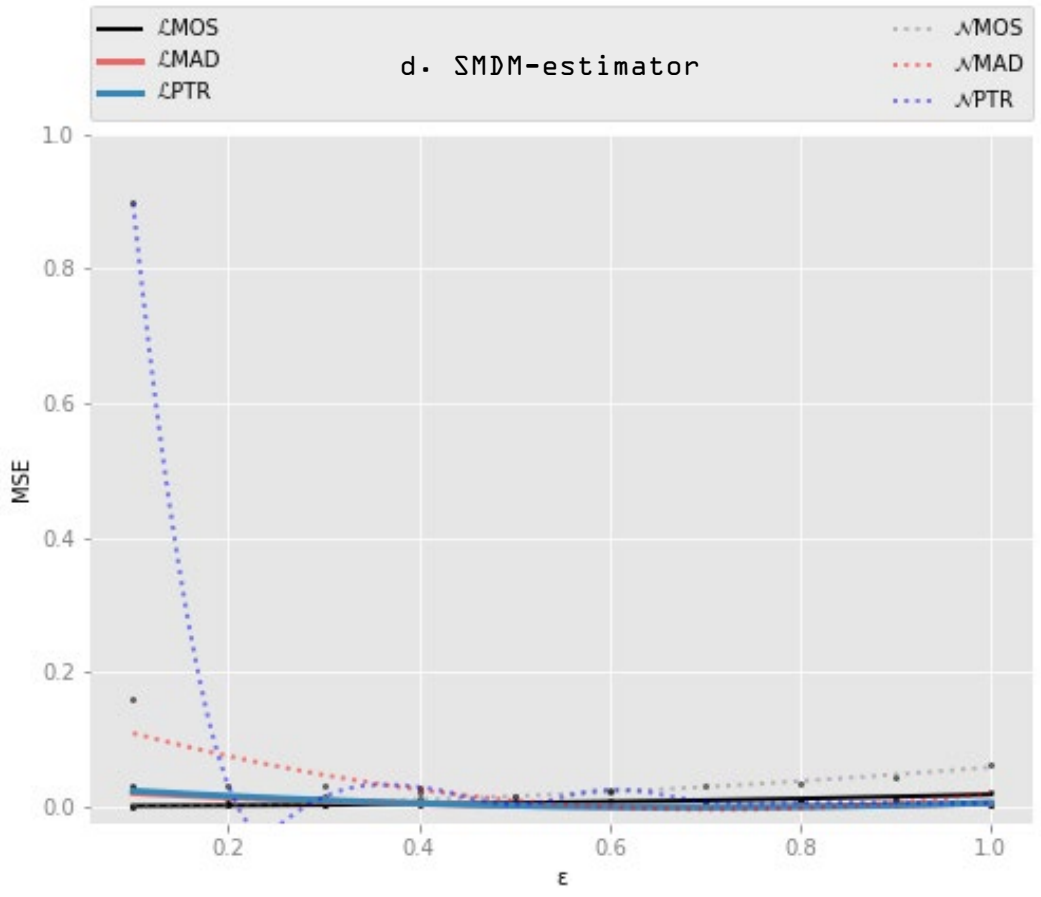

ℒMOS
ℒMAD
ℒPTR
d. SMDM-estimator
𝒩MOS
𝒩MAD
𝒩PTR
MSE
ε

B.3: EMPIRICAL UNIVARIATE PROBLEM

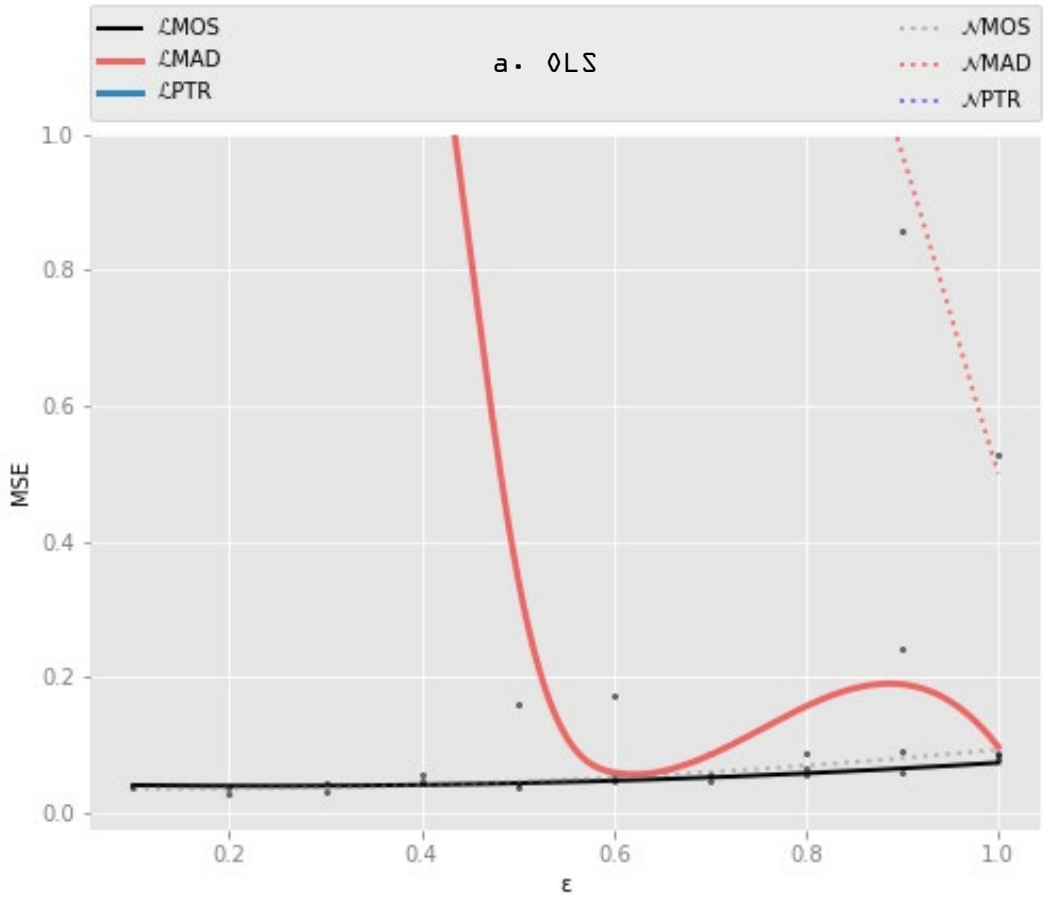


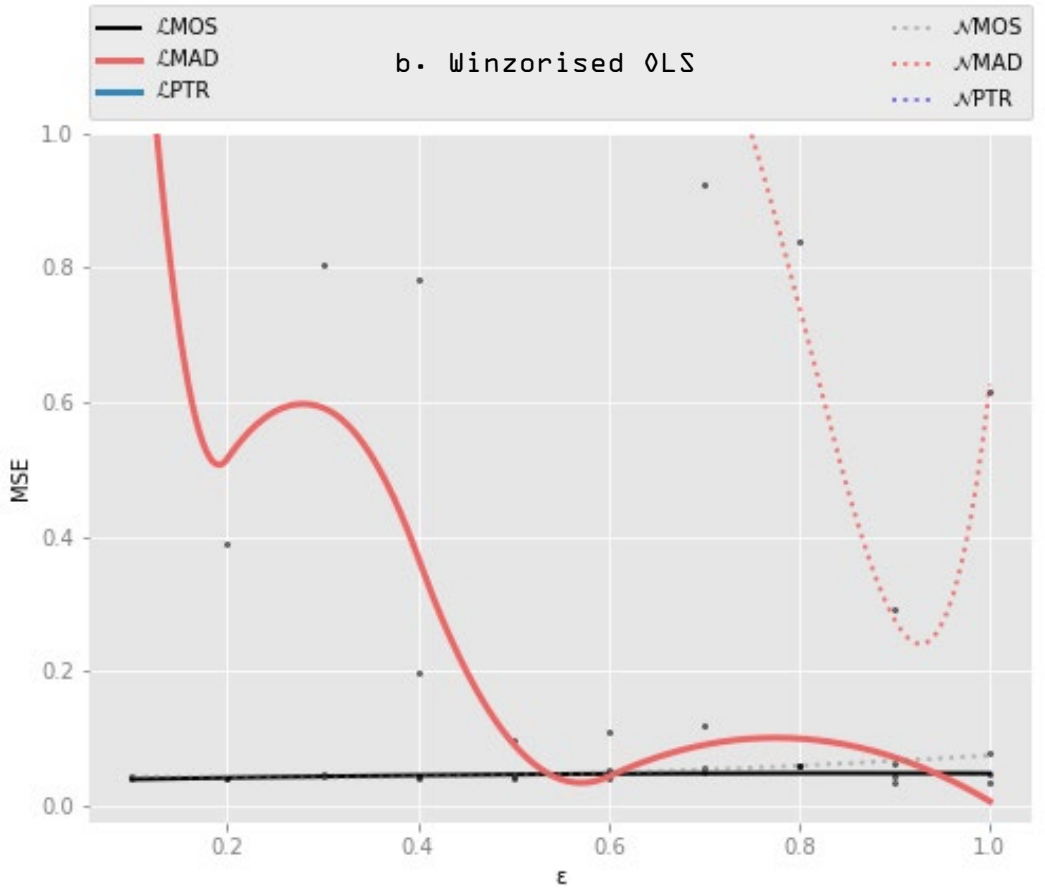


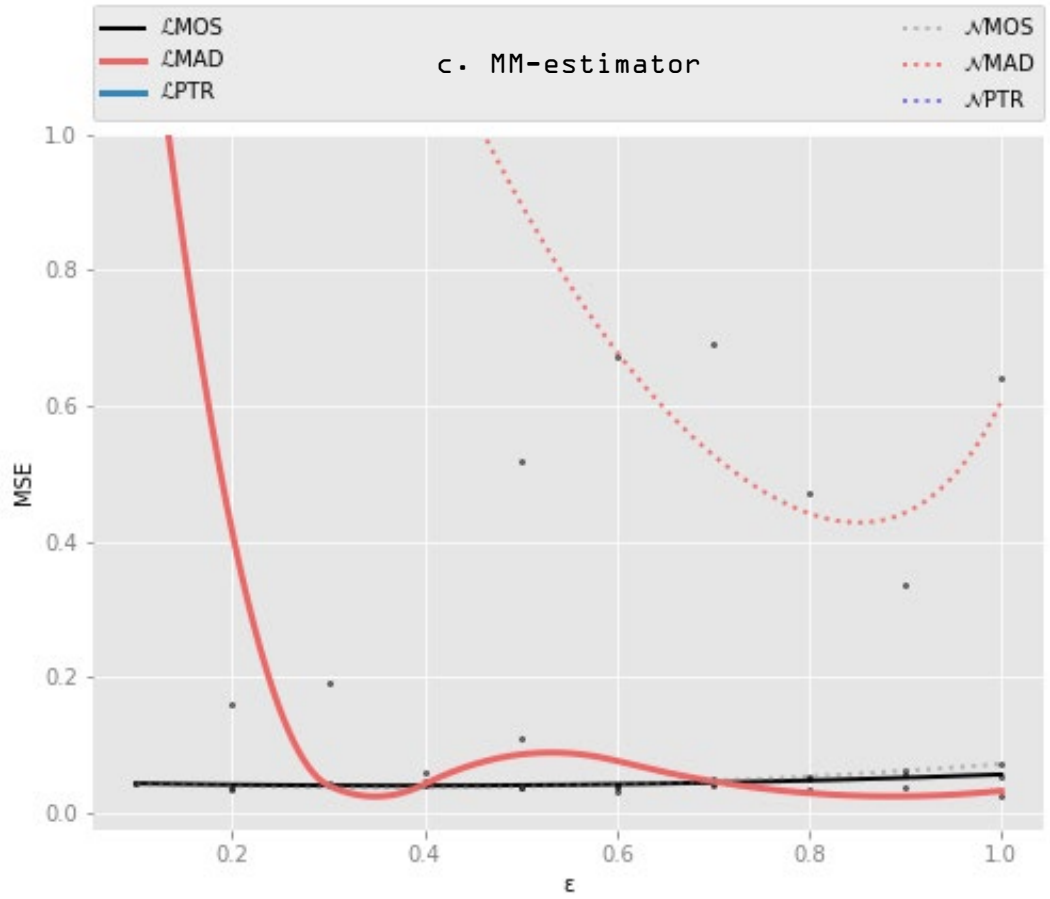

ℒMOS
ℒMAD
ℒPTR
c. MM-estimator
𝒩MOS
𝒩MAD
𝒩PTR
MSE
ε
0.0
0.2
0.4
0.6
0.8
1.0

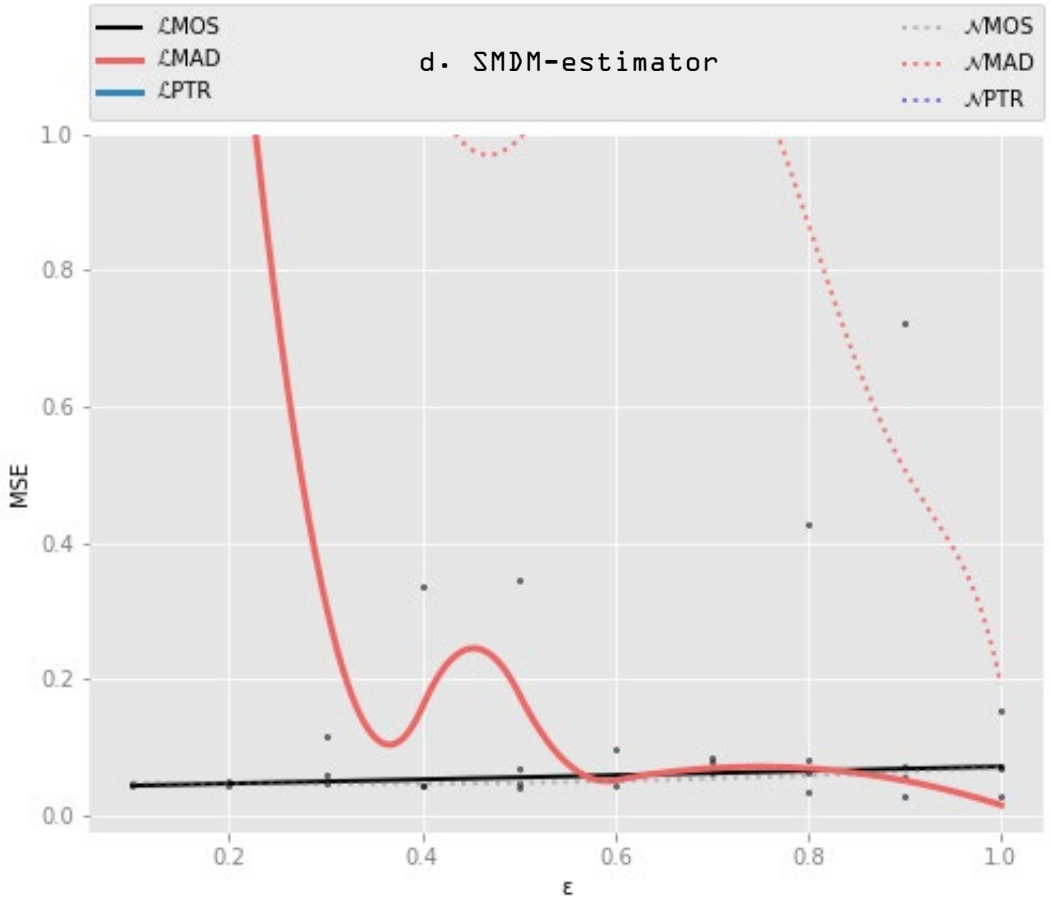

ℒMOS
ℒMAD
ℒPTR
d. SMDM-estimator
𝒩MOS
𝒩MAD
𝒩PTR
MSE
ε
0.0
0.2
0.4
0.6
0.8
1.0

B.4: EMPIRICAL BIIVARIATE PROBLEM






ℒMOS
ℒMAD
ℒPTR
c. MM-estimator
𝒩MOS
𝒩MAD
𝒩PTR
MSE
ε
0.0
0.2
0.4
0.6
0.8
1.0


ℒMOS
ℒMAD
ℒPTR
d. SMDM-estimator
𝒩MOS
𝒩MAD
𝒩PTR
MSE
ε
0.0
0.2
0.4
0.6
0.8
1.0

C. TABLE

We count all $\theta \neq \perp$

C.1. SYNTHETIC UNIVARIATE

$p = 111$	OLS	W-OLS	MM	SMDM
PTR - $\mathcal{L}$	56	72	86	87
PTR - $\mathcal{N}$	61	64	79	80
MAD - $\mathcal{L}$	111	111	111	111
MAD - $\mathcal{N}$	111	111	111	111

C.2. EMPIRICAL UNIVARIATE

$p = 100$	OLS	W-OLS	MM	SMDM
PTR - $\mathcal{L}$	11	57	48	36
PTR - $\mathcal{N}$	10	52	51	33
MAD - $\mathcal{L}$	91	99	93	96
MAD - $\mathcal{N}$	93	100	94	95

C.3. EMPIRICAL BIVARIATE

$p = 5$	OLS	W-OLS	MM	SMDM
PTR - $\mathcal{L}$	0	0	0	0
PTR - $\mathcal{N}$	0	0	0	0
MAD - $\mathcal{L}$	3	4	5	5
MAD - $\mathcal{N}$	4	5	5	5

THIS PAGE IS INTENTIONALLY LEFT BLANK