Taming Heavy-tailed Features by Shrinkage

Ziwei Zhu, Wenjing Zhou

University of Michigan, Department of Statistics

April 2021

Motivation

- Heavy-tailed data abound in modern data analytics.

1

2

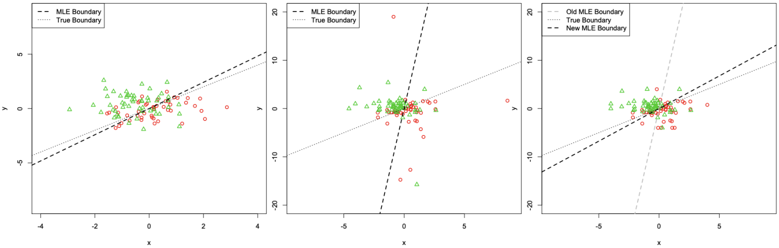
3

Figure 1: Real-life scenarios with heavy-tailed data

1 http://www.algorithmica-technologies.com/en/case_studies/financial-data-analysis-for-contract-planning
2 https://www.chla.org/press-release/children-s-hospital-los-angeles-publishes-largest-genomic-study-covid-19-children-date
3 https://bernardmarr.com/default.asp?contentID=2126

Motivation

- Heavy-tailed features can aggravate the corruption on the response and jeopardize standard statistical approaches.



(a) Gaussian features (b) Student's t_2 features (c) Shrunk Student's t_2 features

Figure 2: Logistic regression with 10% mislabeled data on different features

Generalized linear model (GLM)

- Suppose we have n observations $\{(y_i, \mathrm{x}_i)\}_{i=1}^n$, where y_i is the response and x_i is the feature vector valued in $\mathbb{R}^d$. Under the GLM with the canonical link, the probability density function of the response y_i is defined as

$$f_n(\mathrm{y}; \mathrm{X}, \boldsymbol{\beta}^*) = \prod_{i=1}^n f(y_i; \eta_i^*) = \prod_{i=1}^n \left\{ c(y_i) \exp\left(\frac{y_i \eta_i^* - b(\eta_i^*)}{\phi} \right) \right\}, \tag{1}$$

  where $\mathrm{y} = (y_1, \cdots, y_n)^\top$, $\mathrm{X} = (\mathrm{x}_1, \cdots, \mathrm{x}_n)^\top$, $\boldsymbol{\beta}^* \in \mathbb{R}^d$ is the regression coefficient vector, $\eta_i^* := \mathrm{x}_i^\top \boldsymbol{\beta}^*$, $b(\cdot)$ is a known function that is twice differentiable with a positive second derivative and $\phi > 0$ is the dispersion parameter.
- The response y is assumed to be generated from a particular distribution in an exponential family.

Corrupted generalized linear model (CGLM)

- For the ith observation, only corrupted response $z_i = y_i + \epsilon_i$ can be observed, where ϵ_i is random noise.
- The response is not limited within the exponential family.
- More real-world problems with complex structures:
  - the linear regression model with heavy-tailed noise
  - the logistic regression with mislabeled samples.
- The flexibility of the original GLM is significantly improved.

Shrinkage

- Low-dimensional regime:

  ℓ_4-norm shrinkage on features, clipping on response

$$\widetilde{\mathrm{x}}_i := \frac{\min(\|\mathrm{x}_i\|_4, \tau_1)}{\|\mathrm{x}_i\|_4}\mathrm{x}_i$$

$$\widetilde{z}_i := \min(|z_i|, \tau_2) z_i / |z_i|$$

- High-dimensional regime:

  elementwise shrinkage on features, clipping on response

$$\widetilde{x}_{ij} := \min(|x_{ij}|, \tau_1) x_{ij} / |x_{ij}|.$$

$$\widetilde{z}_i := \min(|z_i|, \tau_2) z_i / |z_i|$$

- τ_1 and τ_2 are predetermined thresholds

ℓ_4-norm ball

- The norm determines the strength of the constraint.
- ℓ_4-norm shrinkage balances the bias and the variance.

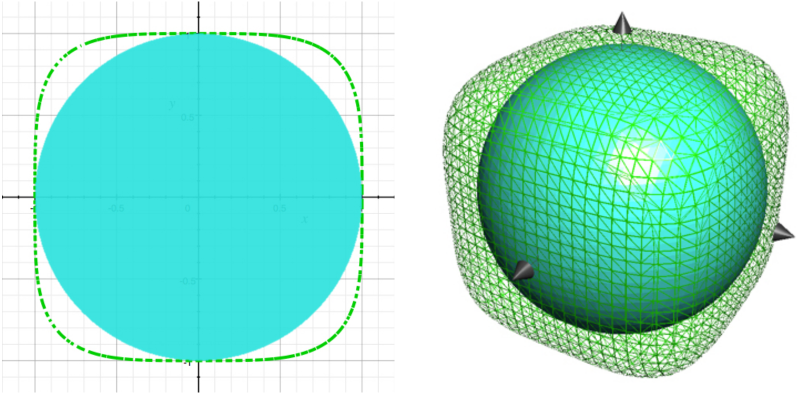


Figure 3: Unit Euclidean ball and ℓ_4-norm ball in 2D and 3D

Simulation: logistic regression with mislabeled data

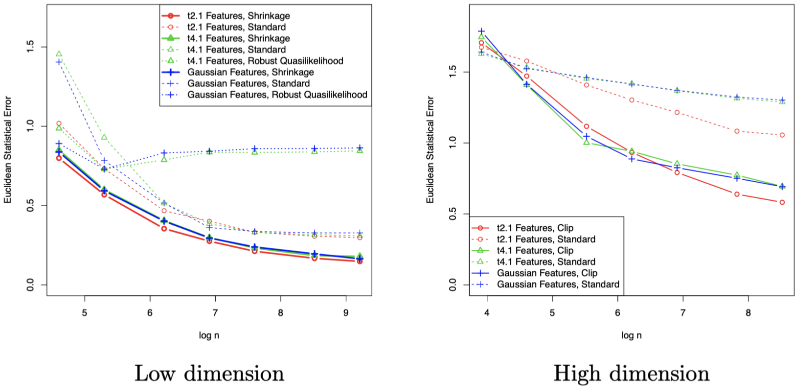


Figure 4: Logistic regression with 10% mislabeled data

Simulation: CNN on MNIST dataset

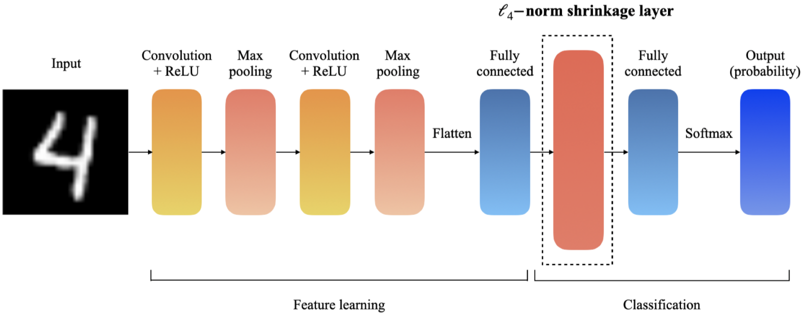


Figure 5: Architecture of the shrinkage convolutional neural network

Simulation: CNN on MNIST dataset

Table 1: Average testing misclassification rate (with standard error in the parentheses) on noisy MNIST images under mislabeling probability 40%

Noisy Pixel Ratio	Original CNN	Shrinkage CNN
0	3.64% (0.20%)	2.93% (0.09%)
0.1	6.88% (0.22%)	4.18% (0.17%)
0.2	6.90% (0.21%)	4.37% (0.16%)
0.4	10.69% (0.29%)	6.65% (0.24%)
0.6	18.82% (0.88%)	12.80% (0.65%)

Taming Heavy-tailed Features by Shrinkage