Robust and Sparse Generalized Linear Models for High-Dimensional Data via Maximum Mean Discrepancy

Xiaoning Kang[1] Lulu Kang[2]

[1]Institute of Supply Chain Analytics and International Business College
Dongbei University of Finance and Economics, Dalian, China

[2]Department of Mathematics and Statistics
University of Massachusetts Amherst, MA, USA

mODa14
June 19, 2026

Introduction & Motivation

- **High-Dimensional Settings ($p \gg n$):** Modern datasets (genomics, finance) require models that are both sparse and resilient.
- **Vulnerability of Standard GLMs:** Standard Maximum Likelihood Estimation (MLE) combined with ℓ_1 (Lasso) penalties is highly sensitive to:
  - Outliers in responses
  - High-leverage points (outliers in covariate space)
  - Heavy-tailed error distributions
- **Robustness Deficit:** Classical shrinkage regularizers stabilize estimation under high-dimensionality but inherit the lack of robustness from their underlying loss functions.

Existing Robust Regression Frameworks

Traditional Robust Estimators

- **M-estimators (Huber Loss Huber [1964]):** Robust against response outliers, but vulnerable to high-leverage points in high dimensions.
- **Density Power Divergence (DPD) & L_2E Basu et al. [1998], Chi and Scott [2014]:** Effective for specific distribution families, but face non-convex optimization hurdles.
- **Robust Lasso variants Chang et al. [2018], Khan et al. [2007], Loh [2017]:** Extend M-estimators to high dimensions but inherit coordinate-wise limitations.

The Core Issue

Most classical robust M-estimators rely on coordinate-wise bounded influence functions, which often fail to handle leverage points effectively when the number of predictors is large.

Maximum Mean Discrepancy: A Universal Framework

Key Idea from Alquier and Gerber [2024]

MMD provides a **joint-distribution matching** criterion that naturally handles outliers in both response and covariate space simultaneously.

Why MMD?

Comparing $P(X, Y)$ to $P_\theta(X, Y)$ in an RKHS

$\Rightarrow$ Bounded influence functions without requiring model correctness

- **Advantage over M-estimators:** No need to detect leverage points explicitly
- **Advantage over DPD:** Works for any distribution family with a characteristic kernel

Maximum Mean Discrepancy (MMD) Preliminaries

MMD is a distance between probability distributions defined via kernel embedding into a Reproducing Kernel Hilbert Space (RKHS) $\mathcal{H}_K$.

Definition (Squared MMD Distance)

For two distributions $P_1, P_2 \in \mathcal{P}(\mathcal{Z})$ and a symmetric, positive definite kernel K:

$$D(K, \mathcal{Z}, P_1, P_2)^2 = \mathbb{E}_{\boldsymbol{z},\boldsymbol{z}' \sim P_1}[K(\boldsymbol{z}, \boldsymbol{z}')] - 2\mathbb{E}_{\boldsymbol{z} \sim P_1, \boldsymbol{t} \sim P_2}[K(\boldsymbol{z}, \boldsymbol{t})] + \mathbb{E}_{\boldsymbol{t},\boldsymbol{t}' \sim P_2}[K(\boldsymbol{t}, \boldsymbol{t}')]$$

- If K is a **characteristic kernel** (e.g., Gaussian), $D(K, \mathcal{Z}, P_1, P_2) = 0 \iff P_1 = P_2$.
- Offers a joint-distribution matching framework ($P(X, Y)$ vs. $P_{\boldsymbol{\theta}}(X, Y)$).

MMD for GLMs: Empirical vs. Model Distribution

Let data $D_n = \{\boldsymbol{x}_i, y_i\}_{i=1}^n$ follow empirical distribution $\hat{P}^n$. Assume the kernel factorizes: $K(\boldsymbol{z}_1, \boldsymbol{z}_2) = K_x(\boldsymbol{x}_1, \boldsymbol{x}_2) K_y(y_1, y_2)$.

Model Distribution $P_{\boldsymbol{\theta}}^n$

$$P_{\boldsymbol{\theta}}^n(A \times B) = \frac{1}{n} \sum_{i=1}^n \delta_{\boldsymbol{x}_i}(A) P_{g(\boldsymbol{\theta}, \boldsymbol{x}_i)}(B)$$

Discrete uniform on $\{\boldsymbol{x}_i\}$ with $Y|\boldsymbol{x}_i \sim P_{g(\boldsymbol{\theta}, \boldsymbol{x}_i)}$

The squared MMD distance decomposes as:

$$D^2 = \underbrace{\frac{1}{n^2} \sum_{i,j} K_x(\boldsymbol{x}_i, \boldsymbol{x}_j) \mathbb{E}[K_y(Y, Y')]}_{①} \underbrace{- \frac{2}{n^2} \sum_{i,j} K_x(\boldsymbol{x}_i, \boldsymbol{x}_j) \mathbb{E}[K_y(Y, y_i)]}_{②} + \underbrace{\frac{1}{n^2} \sum_{i,j} K(\boldsymbol{z}_i, \boldsymbol{z}_j)}_{③ \text{ (constant)}}$$

Unpenalized MMD for GLM

Let data $D_n = \{\boldsymbol{x}_i, y_i\}_{i=1}^n$ follow empirical distribution $\hat{P}^n$. Assume the kernel factorizes: $K(\boldsymbol{z}_1, \boldsymbol{z}_2) = K_x(\boldsymbol{x}_1, \boldsymbol{x}_2)K_y(y_1, y_2)$.

- **Full $O(n^2)$ Estimator:**

$$\hat{\boldsymbol{\theta}}_n \in \arg\min_{\boldsymbol{\theta}} \sum_{i,j=1}^{n} K_x(\boldsymbol{x}_i, \boldsymbol{x}_j) l(\boldsymbol{\theta}, \boldsymbol{x}_i, \boldsymbol{x}_j, y_i)$$

  where $l(\boldsymbol{\theta}, \boldsymbol{x}_i, \boldsymbol{x}_j, y_i) = \mathbb{E}_{Y,Y'}[K_y(Y, Y')] - 2\mathbb{E}_Y[K_y(Y, y_i)]$.

- **Approximate $O(n)$ Estimator:** Under small bandwidth assumptions on K_x, cross-terms ($i \neq j$) vanish:

$$\tilde{\boldsymbol{\theta}}_n \in \arg\min_{\boldsymbol{\theta}} \sum_{i=1}^{n} \tilde{l}(\boldsymbol{\theta}, \boldsymbol{x}_i, y_i)$$

Penalized MMD for High Dimensions

In high-dimensional settings ($p \gg n$), unpenalized MMD estimators may suffer from over-fitting or lack of identifiability.

Proposed Penalized Estimators

Full $O(n^2)$ version:

$$\hat{\boldsymbol{\theta}}_n \in \arg\min_{\boldsymbol{\theta}} \sum_{i,j=1}^{n} K_x(\boldsymbol{x}_i, \boldsymbol{x}_j) l(\boldsymbol{\theta}, \boldsymbol{x}_i, \boldsymbol{x}_j, y_i) + \lambda \|\boldsymbol{\theta}\|_1$$

Computationally efficient $O(n)$ version:

$$\tilde{\boldsymbol{\theta}}_n \in \arg\min_{\boldsymbol{\theta}} \sum_{i=1}^{n} \tilde{l}(\boldsymbol{\theta}, \boldsymbol{x}_i, y_i) + \lambda \|\boldsymbol{\theta}\|_1$$

$\lambda \geq 0$ controls sparsity. Selected via 5-fold cross-validation.

Remark: Why MMD for Robust Variable Selection?

> *"Unlike penalized M-estimators which focus on $P(Y|X)$, the MMD objective is a **joint-distribution matching criterion** that minimizes a distance between $P(X, Y)$ and $P_\theta(X, Y)$ in an RKHS. This allows the estimator to handle outliers in both the response and the covariate space (leverage points) simultaneously."*

Key insight: MMD loss provides a bounded gradient even under heavy-tailed contamination, preventing the KKT conditions of the Lasso from being dominated by outliers.

Case 1: Gaussian Linear Regression

Let $Y(\boldsymbol{x}) \sim \mathcal{N}(\boldsymbol{x}^\top \boldsymbol{\theta}, \sigma^2)$. We select a Gaussian kernel for K_y with bandwidth h_y.

- **Simplified pairwise loss term $l(\boldsymbol{\theta}, \boldsymbol{x}_i, \boldsymbol{x}_j, y_i)$:**

$$l(\boldsymbol{\theta}, \boldsymbol{x}_i, \boldsymbol{x}_j, y_i) = \frac{h_y}{\sqrt{2\sigma^2 + h_y^2}} \exp\left(-\frac{1}{2}\frac{\boldsymbol{\theta}^\top (\boldsymbol{x}_j - \boldsymbol{x}_i)(\boldsymbol{x}_j - \boldsymbol{x}_i)^\top \boldsymbol{\theta}}{2\sigma^2 + h_y^2}\right) - \frac{2h_y}{\sqrt{\sigma^2 + h_y^2}} \exp\left(-\frac{1}{2}\frac{(y_i - \boldsymbol{x}_j^\top \boldsymbol{\theta})^2}{\sigma^2 + h_y^2}\right)$$

- **Simplified $O(n)$ loss term $\tilde{l}(\boldsymbol{\theta}, \boldsymbol{x}_i, y_i)$:**

$$\tilde{l}(\boldsymbol{\theta}, \boldsymbol{x}_i, y_i) = \frac{h_y}{\sqrt{2\sigma^2 + h_y^2}} - \frac{2h_y}{\sqrt{\sigma^2 + h_y^2}} \exp\left(-\frac{1}{2}\frac{(y_i - \boldsymbol{x}_i^\top \boldsymbol{\theta})^2}{\sigma^2 + h_y^2}\right)$$

Case 2: Binary Logistic Regression

Let $Y(\boldsymbol{x}) \sim \text{Bernoulli}(\pi(\boldsymbol{x}))$, with $\pi(\boldsymbol{x}) = \frac{e^{\boldsymbol{x}^\top \boldsymbol{\theta}}}{1+e^{\boldsymbol{x}^\top \boldsymbol{\theta}}}$. We utilize a discrete **geometric kernel** $K_y(y_1, y_2) = 0.5 h_y (1 - h_y)^{|y_1 - y_2|}$.

- **Simplified pairwise loss term:**

$$\begin{aligned} l(\boldsymbol{\theta}, \boldsymbol{x}_i, \boldsymbol{x}_j, y_i) = 0.5 h_y [1 - h_y(\pi_i + \pi_j) + 2 h_y \pi_i \pi_j] \\ - h_y [(1 - h_y)^{1-y_i} \pi_j + (1 - h_y)^{y_i} (1 - \pi_j)] \end{aligned}$$

- **Simplified $O(n)$ approximate loss term:**

$$\tilde{l}(\boldsymbol{\theta}, \boldsymbol{x}_i, y_i) = h_y^2 \pi_i^{2(1-y_i)} (1 - \pi_i)^{2y_i} - 0.5 h_y$$

Optimization: ADMM + AdaGrad

The objective combines non-convex MMD loss with non-smooth ℓ_1 penalty. Use ADMM to decouple them Boyd et al. [2011].

Augmented Lagrangian ($O(n)$ case)

$$\mathcal{L}(\boldsymbol{\theta}, \boldsymbol{\eta}, \boldsymbol{\gamma}) = \underbrace{\frac{1}{n}\sum_{i=1}^{n} \tilde{l}(\boldsymbol{\theta}, \boldsymbol{x}_i, y_i)}_{\text{MMD loss}} + \lambda\|\boldsymbol{\eta}\|_1 + \boldsymbol{\gamma}^\top(\boldsymbol{\theta} - \boldsymbol{\eta}) + \frac{\rho}{2}\|\boldsymbol{\theta} - \boldsymbol{\eta}\|_2^2$$

- $\boldsymbol{\eta}$: Auxiliary variable for sparsity
- $\boldsymbol{\gamma}$: Lagrange multipliers
- $\rho > 0$: Penalty parameter (set to 1)

ADMM Update Steps

For iteration $k+1$:

1. **η-step (Soft Thresholding):**

$$\boldsymbol{\eta}^{(k+1)} = S\left(\boldsymbol{\theta}^{(k)} + \frac{\boldsymbol{\gamma}^{(k)}}{\rho}, \frac{\lambda}{\rho}\right), \quad S(a,b) = \text{sign}(a)\max(|a| - b, 0)$$

2. **θ-step (AdaGrad Duchi et al. [2011]):**

$$\nabla_{\boldsymbol{\theta}}\mathcal{L} = \frac{1}{n}\sum_{i=1}^{n}\frac{\partial \tilde{l}_i}{\partial \boldsymbol{\theta}} + \boldsymbol{\gamma}^{(k)} + \rho(\boldsymbol{\theta} - \boldsymbol{\eta}^{(k+1)})$$

   Learning rate $= 0.1$, adaptively scaled per parameter.

3. **γ-step (Dual Ascent):**

$$\boldsymbol{\gamma}^{(k+1)} = \boldsymbol{\gamma}^{(k)} + \rho(\boldsymbol{\theta}^{(k+1)} - \boldsymbol{\eta}^{(k+1)})$$

Local Convexity of Gaussian MMD Loss

The Hessian matrix of $\tilde{l}_i$:

$$\nabla^2_{\boldsymbol{\theta}} \tilde{l}_i = \frac{2h_y}{(\sigma^2 + h_y^2)^{3/2}} \exp\left(-\frac{(y_i - \boldsymbol{x}_i^\top \boldsymbol{\theta})^2}{2(\sigma^2 + h_y^2)}\right) \left[1 - \frac{(y_i - \boldsymbol{x}_i^\top \boldsymbol{\theta})^2}{\sigma^2 + h_y^2}\right] \boldsymbol{x}_i \boldsymbol{x}_i^\top$$

Local Convexity Condition

$\nabla^2_{\boldsymbol{\theta}} \tilde{l}_i$ is PSD $\iff$

$$(y_i - \boldsymbol{x}_i^\top \boldsymbol{\theta})^2 \leq \sigma^2 + h_y^2$$

Since $\mathrm{E}[(y_i - \boldsymbol{x}_i^\top \boldsymbol{\theta})^2] = \sigma^2$, this holds near the true parameter.

Local Convexity of Logistic MMD Loss

For logistic regression, $\tilde{l}(\boldsymbol{\theta}, \boldsymbol{x}_i, y_i) \propto \pi_i^{2(1-y_i)}(1-\pi_i)^{2y_i}$.

Local Convexity Bounds

The loss is locally convex if for all $i = 1, \ldots, n$:

$$1 - \left(\frac{2}{3}\right)^{y_i} \leq \pi(\boldsymbol{x}_i, \boldsymbol{\theta}) \leq \left(\frac{2}{3}\right)^{1-y_i}$$

- $y_i = 0 \Rightarrow \pi_i \in [0, 2/3]$
- $y_i = 1 \Rightarrow \pi_i \in [1/3, 1]$

Lasso initialization typically places $\boldsymbol{\theta}$ in this convex region.

Simulation Study Design

Gaussian Linear Regression: $n = 100$, $p = 200$,
$\beta = (4, 4, 3, 3, -3, -3, -4, -4, \mathbf{0}_{p-8})^\top$

Factor	**Settings**
Covariance $\mathbf{\Sigma}_X$	Identity $\boldsymbol{I}_p$, AR(0.7)
Error distribution ϵ_i	$\mathcal{N}(0,1)$, Laplace$(0,1)$, t_5
Contamination level τ	0%, 5%, 10%
Outlier types $\mathcal{T}$	$\mathcal{X}_1$, $\mathcal{X}_2$ (leverage), $\mathcal{Y}$ (response)
Benchmarks	Lasso, Huber-Lasso, Loh's M-estimator Loh [2017]

Binary Logistic Regression: $n = 100$, $p = 100, 200$, $\beta = (1_{10}, \mathbf{0}_{p-10})^\top$

- Contamination: $\mathcal{X}_1$, $\mathcal{X}_2$, $\mathcal{XY}_1$, $\mathcal{XY}_2$ (label flipping + leverage)
- Benchmarks: Logistic Lasso, enetLTS **?**

Kernel Bandwidth Selection

Median Heuristic ?

For continuous responses:

- h_y = median of pairwise distances $\{y_i\}_{i=1}^n$
- h_x = median of pairwise distances $\{\boldsymbol{x}_i\}_{i=1}^n$

Adaptive per replicate.

Binary Response

Median heuristic not applicable. Fix $h_y = \sqrt{2}/2 \approx 0.707$ Alquier and Gerber [2024], **?**.

- Matching labels: $K_y(0,0) = K_y(1,1) = 0.5h_y \approx 0.3535$
- Mismatching labels: $K_y(0,1) = K_y(1,0) = 0.5h_y(1-h_y) \approx 0.1035$

Gaussian Results: MSE (Estimation Accuracy)

Identity Σ_X, Laplace Errors, $\tau = 0.1$, $\mathcal{X}_2$ Contamination

Method	Lasso	Huber	Loh	$\hat{\theta}_n$	$\tilde{\theta}_n$
MSE	0.0393	0.0205	**0.0224**	0.0386	0.0354
(std)	(0.027)	(0.014)	(0.015)	(0.028)	(0.022)

- Huber and Loh outperform Lasso under leverage point contamination
- MMD estimators show competitive performance

AR(0.7) Σ_X, t_5 Errors, $\tau = 0.1$, $\mathcal{X}_2$ Contamination

Method	Lasso	Huber	Loh	$\hat{\theta}_n$	$\tilde{\theta}_n$
MSE	0.0729	0.0428	**0.0280**	0.0708	0.0690
(std)	(0.048)	(0.033)	(0.016)	(0.047)	(0.043)

Loh's estimator performs best; MMD methods are robust but slightly higher MSE.

Gaussian Results: FSL (Variable Selection)

Key strength of penalized MMD: Significantly lower False Selection Loss

Identity Σ_X, $N(0,1)$ Errors, $\tau = 0.05$, $\mathcal{Y}$ Contamination

Method	Lasso	Huber	Loh	$\hat{\theta}_n$	$\tilde{\theta}_n$
FSL	18.0	19.0	14.0	11.3	**4.40**
(std)	(6.6)	(5.8)	(5.3)	(6.8)	(3.5)

AR(0.7) Σ_X, Laplace Errors, $\tau = 0.1$, $\mathcal{Y}$ Contamination

Method	Lasso	Huber	Loh	$\hat{\theta}_n$	$\tilde{\theta}_n$
FSL	14.7	20.7	17.7	**9.30**	8.80
(std)	(8.3)	(11)	(13)	(6.5)	(8.0)

Conclusion: MMD loss is less "distracted" by outliers during feature selection.

Gaussian Results: PE (Prediction Error)

Identity Σ_X, Laplace Errors, $\tau = 0.1$, $\mathcal{Y}$ Contamination

Method	Lasso	Huber	Loh	$\hat{\theta}_n$	$\tilde{\theta}_n$
PE	15.1	**5.02**	15.0	15.2	15.5
(std)	(8.8)	(1.6)	(8.6)	(8.3)	(11)

Huber excels for response outliers, as expected.

AR(0.7) Σ_X, $N(0,1)$ Errors, $\tau = 0.1$, $\mathcal{X}_1$ Contamination

Method	Lasso	Huber	Loh	$\hat{\theta}_n$	$\tilde{\theta}_n$
PE	11.3	11.4	11.4	**11.1**	11.7
(std)	(1.5)	(1.7)	(2.1)	(1.2)	(1.4)

MMD $\hat{\theta}_n$ achieves lowest PE under leverage point contamination.

Logistic Results: MSE (Identity Σ_X, $p = 200$)

Contamination	Lasso	LTS	LTS.full	$\hat{\theta}_n$	$\tilde{\theta}_n$
None ($\tau = 0$)	**6.77**	10.9	8.87	7.17	7.28
$\mathcal{X}_2$ ($\tau = 0.05$)	9.45	14.5	12.6	9.48	**7.77**
$\mathcal{XY}_1$ ($\tau = 0.05$)	9.44	14.9	12.8	9.44	**7.65**
$\mathcal{X}_2$ ($\tau = 0.1$)	9.39	13.7	11.6	9.39	**7.63**
$\mathcal{XY}_1$ ($\tau = 0.1$)	9.59	16.5	13.3	9.57	**8.31**

Key finding: $\tilde{\theta}_n$ (O(n) approximation) consistently achieves lowest MSE under contamination affecting active predictors. enetLTS (LTS, LTS.full) struggles in high dimensions.

Logistic Results: Misclassification Error (AR(0.7), $p = 200$)

Contamination	Lasso	LTS	LTS.full	$\hat{\theta}_n$	$\tilde{\theta}_n$
$\mathcal{X}_2$ $(\tau = 0.05)$	44.5	44.8	44.4	46.0	**16.9**
$\mathcal{XY}_1$ $(\tau = 0.05)$	44.5	45.5	42.4	45.2	**15.0**
$\mathcal{X}_2$ $(\tau = 0.1)$	40.7	45.9	47.4	43.0	**15.3**
$\mathcal{XY}_1$ $(\tau = 0.1)$	44.4	47.3	47.8	46.2	**15.9**

Dramatic improvement: $\tilde{\theta}_n$ reduces misclassification error from $\sim$45% to $\sim$15-17% when active predictors are corrupted!
The $O(n)$ approximate estimator often outperforms the full $O(n^2)$ version.

Logistic Results: FSL (AR(0.7), $p = 200$)

Contamination	Lasso	LTS	LTS.full	$\hat{\theta}_n$	$\tilde{\theta}_n$
None ($\tau = 0$)	14.8	15.7	20.4	**7.93**	10.6
$\mathcal{X}_2$ ($\tau = 0.05$)	12.0	24.8	61.5	**10.1**	7.07
$\mathcal{XY}_1$ ($\tau = 0.1$)	11.3	28.9	51.6	**10.4**	7.17

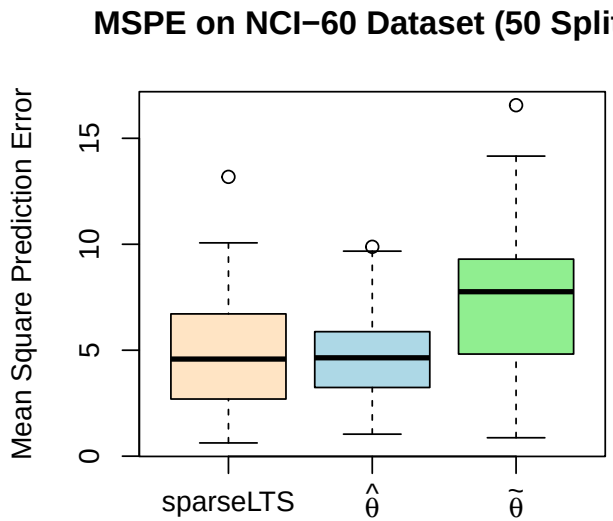
- $\hat{\theta}_n$ achieves lowest FSL in most settings
- Demonstrates ability to distinguish true active predictors from noise even when those predictors are the source of contamination
- enetLTS shows instability (FSL up to 73.6!)

Real Data: NCI-60 Cancer Cell Line ?

- $n = 59$ observations, $p = 100$ genes (after screening with Huber correlation)
- Response: Protein expression of the 92nd protein
- Compared against `sparseLTS` **??**

Result (50 random splits, training $n = 45$, test $n = 14$):

- $\hat{\theta}_n$ exhibits **lower median MSPE** than sparseLTS
- **Narrower interquartile range** (higher stability)
- $\tilde{\theta}_n$ performs worse than both



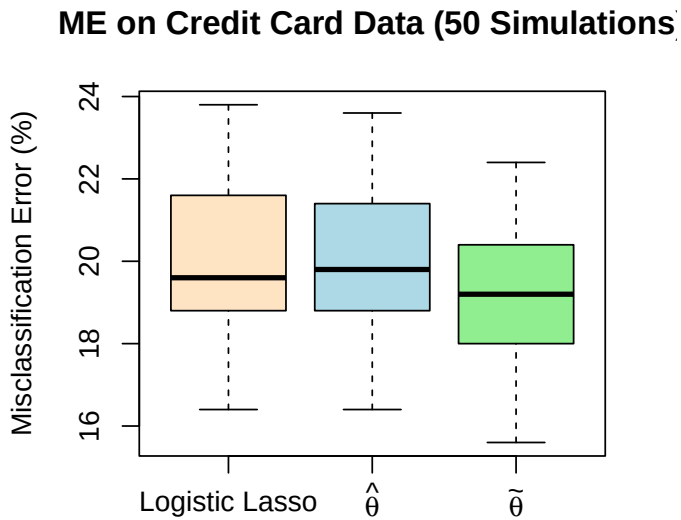
Real Data: Credit Card Default ?

- 30,000 observations, 23 predictors
- Training: $n = 2,000$ (stratified), Test: $m = 500$
- Compared against standard logistic Lasso

Result (50 random splits):

- $\tilde{\theta}_n$ provides **more robust classification boundary** than Lasso
- **Lower average prediction error**
- **Significantly reduced variability** across splits

MMD's joint-distribution perspective advantageous for noisy financial data with class imbalance.



Computational Complexity

Estimator	Complexity per iteration	Use case
$\hat{\theta}_n$ (full)	$O(n^2 p)$	Moderate n (e.g., $n = 100$)
$\tilde{\theta}_n$ (approx)	$O(np)$	Large n (e.g., $n = 2000$)

$O(n)$ Approximation Justification

When $K_x(\boldsymbol{x}_i, \boldsymbol{x}_j) = \exp(-\|\boldsymbol{x}_i - \boldsymbol{x}_j\|^2/h)$ with small h:

- $K_x(\boldsymbol{x}_i, \boldsymbol{x}_j) \approx 0$ for $i \neq j$ in high dimensions
- Cross-terms become negligible
- Empirical evidence: $\tilde{\theta}_n$ performs comparably or better than $\hat{\theta}_n$ in simulations

Summary of Contributions

1. **Penalized MMD framework for GLMs**
   - Simultaneous robust estimation and variable selection in high dimensions
   - ℓ_1 penalty for sparsity
2. **Computationally efficient $O(n)$ approximation**
   - Based on fast-decaying kernel assumption
   - Enables scaling to larger datasets
3. **ADMM + AdaGrad optimization solver**
   - Decouples non-convex MMD loss from non-smooth ℓ_1 penalty
   - Handles local non-convexity via adaptive gradient scaling

Key Findings

- **Universal robustness:** MMD methods handle leverage points, response outliers, and heavy-tailed errors simultaneously
- **Superior variable selection:** Consistently lowest FSL across contaminated scenarios
- **Logistic regression breakthrough:** $\tilde{\theta}_n$ reduces ME from $\sim$45% to $\sim$15-17% when active predictors are corrupted
- **Computational viability:** $O(n)$ approximation performs comparably to full $O(n^2)$ version
- **Real-world validation:** Strong performance on NCI-60 (genomics) and Credit Default (finance) datasets

Summary & Future Directions

- **Contributions:**
  - Formulated a penalized MMD framework for GLMs enabling concurrent parameter estimation and variable selection in high dimensions.
  - Developed computationally efficient $O(n)$ approximations.
  - Proposed an ADMM + AdaGrad solver resilient to non-convex landscapes.
- **Strengths:** Strong resilience to high-leverage points, heavy-tailed noise, and response outliers.
- **Future Work:**
  - Establish theoretical oracle properties of the penalized estimators.
  - Explore automated, data-driven bandwidth selection methods.
  - Extend the framework to group-sparse penalties.

References I

Alquier, P., and Gerber, M. [2024], "Universal robust regression via maximum mean discrepancy," *Biometrika*, 111(1), 71–92.

Basu, A., Harris, I. R., Hjort, N. L., and Jones, M. [1998], "Robust and efficient estimation by minimising a density power divergence," *Biometrika*, 85(3), 549–559.

Boyd, S., Parikh, N., Chu, E., Peleato, B., Eckstein, J. et al. [2011], "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Foundations and Trends® in Machine learning*, 3(1), 1–122.

Chang, L., Roberts, S., and Welsh, A. [2018], "Robust lasso regression using Tukey's biweight criterion," *Technometrics*, 60(1), 36–47.

Chi, E. C., and Scott, D. W. [2014], "Robust parametric classification and variable selection by a minimum distance criterion," *Journal of Computational and Graphical Statistics*, 23(1), 111–128.

Duchi, J., Hazan, E., and Singer, Y. [2011], "Adaptive subgradient methods for online learning and stochastic optimization.," *Journal of machine learning research*, 12(7).

Huber, P. J. [1964], "Robust Estimation of a Location Parameter," *The Annals of Mathematical Statistics*, 35(1), 73–101.

Khan, J. A., Aelst, S. V., and Zamar, R. H. [2007], "Robust Linear Model Selection Based on Least Angle Regression," *Journal of the American Statistical Association*, 102(480), 1289–1299.

Loh, P.-L. [2017], "Statistical consistency and asymptotic normality for high-dimensional robust M-estimators," *The Annals of Statistics*, 45(2), 866–896.