

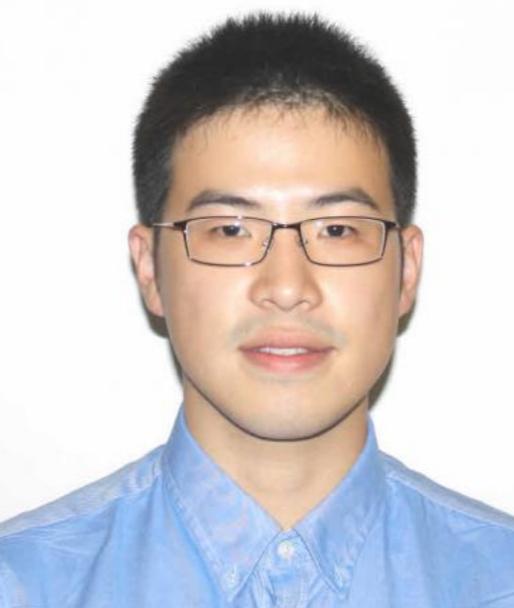
Robust Estimation and Generative Adversarial Networks

Yuan YAO
HKUST





Chao Gao (Chicago)

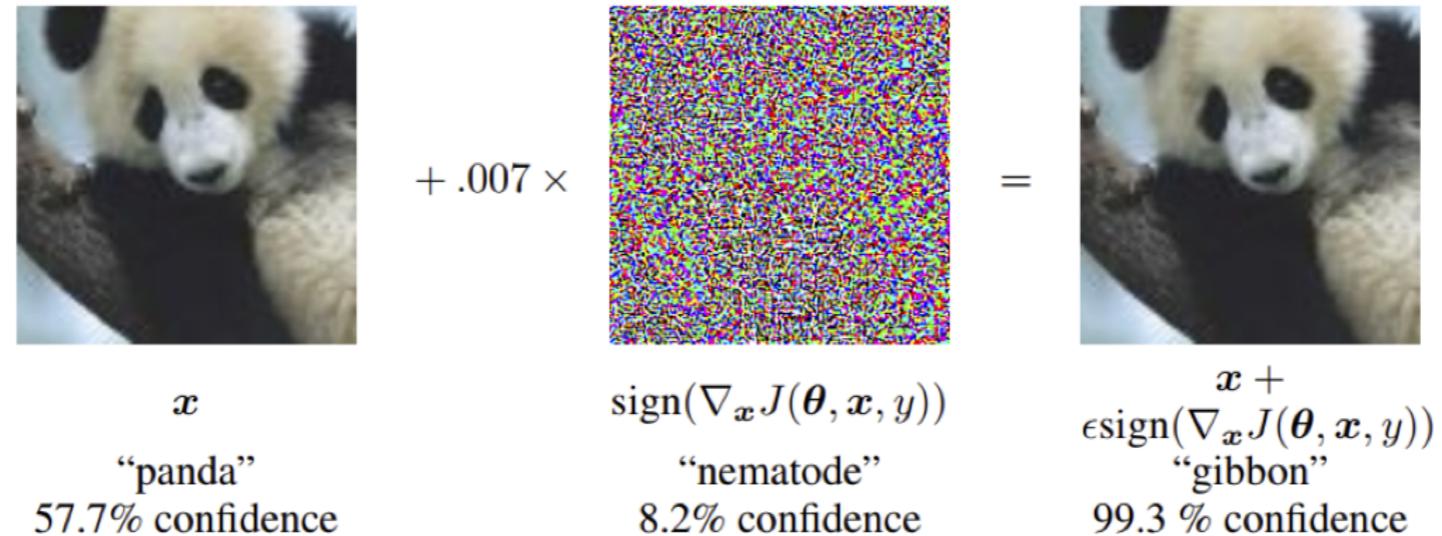


Jiyu Liu (Yale)



Weizhi Zhu (HKUST)

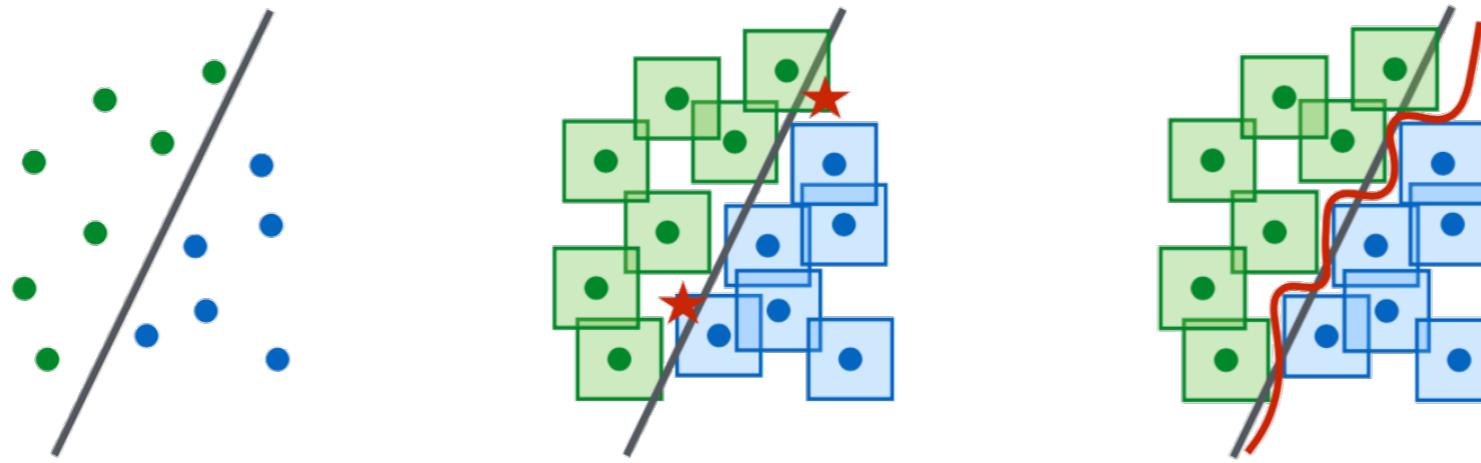
Deep Learning is Notoriously Not Robust!



[Goodfellow et al., 2014]

- Imperceptible adversarial examples are ubiquitous to fail neural networks
- How can one achieve **robustness**?

Robust Optimization



- Traditional training:

$$\min_{\theta} J_n(\theta, \mathbf{z} = (x_i, y_i)_{i=1}^n)$$

- e.g. square or cross-entropy loss as negative log-likelihood of logit models

- Robust optimization:

$$\min_{\theta} \max_{\|\epsilon_i\| \leq \delta} J_n(\theta, \mathbf{z} = (x_i + \epsilon_i, y_i)_{i=1}^n)$$

- robust to any distributions, yet perhaps too conservative

Distributional Robust Optimization

- Distributional Robust Optimization:

$$\min_{\theta} \max_{\epsilon} \mathbb{E}_{z \sim P_\epsilon \in \mathcal{D}} [J_n(\theta, z)]$$

- \mathcal{D} is a set of ambiguous distributions, e.g. Wasserstein ambiguity set

$$\mathcal{D} = \{P_\epsilon : W_2(P_\epsilon, \text{uniform distribution}) \leq \epsilon\}$$

where DRO may be reduced to **regularized maximum likelihood** estimates
([Shafieezadeh-Abadeh, Esfahani, Kuhn, NIPS'2015](#))

- intermediate approach with statistically contaminated distributions
- *sometimes, contamination might be unstructured...*

$$\mathcal{D} = \{P_\epsilon : TV(P_\epsilon, \text{uniform distribution}) \leq \epsilon\}?$$

Huber's Model

$$X_1, \dots, X_n \sim (1 - \epsilon)P_\theta + \epsilon Q$$

[Huber 1964]

Huber's Model

$$X_1, \dots, X_n \sim (1 - \epsilon)P_\theta + \epsilon Q$$

parameter of interest

[Huber 1964]

Huber's Model

$$X_1, \dots, X_n \sim (1 - \epsilon)P_\theta + \epsilon Q$$

contamination proportion

parameter of interest

[Huber 1964]

Huber's Model

$$X_1, \dots, X_n \sim (1 - \epsilon)P_\theta + \epsilon Q$$

contamination proportion

parameter of interest

arbitrary contamination

[Huber 1964]

An Example

$$X_1, \dots, X_n \sim (1 - \epsilon)N(\theta, I_p) + \epsilon Q.$$

An Example

$$X_1, \dots, X_n \sim (1 - \epsilon)N(\theta, I_p) + \epsilon Q.$$

how to estimate ?

Robust Maximum-Likelihood

Does not work!

$$X_1, \dots, X_n \sim (1 - \epsilon)N(\theta, I_p) + \epsilon Q.$$

Robust Maximum-Likelihood

Does not work!

$$X_1, \dots, X_n \sim (1 - \epsilon)N(\theta, I_p) + \epsilon Q.$$

$$\begin{aligned}\ell(\theta, Q) &= \text{negative log-likelihood} = \sum_{i=1}^n (\theta - X_i)^2 \\ &\sim (1 - \epsilon)\mathbb{E}_{N(\theta)}(\theta - X)^2 + \epsilon\mathbb{E}_Q(\theta - X)^2\end{aligned}$$

the sample mean

$$\hat{\theta}_{mean} = \frac{1}{n} \sum_{i=1}^n X_i = \arg \min_{\theta} \ell(\theta, Q)$$

$$\min_{\theta} \max_Q \ell(\theta, Q) \geq \max_Q \min_{\theta} \ell(\theta, Q) = \max_Q \ell(\hat{\theta}_{mean}, Q) = \infty$$

Medians

1. Coordinatewise median

$\hat{\theta} = (\hat{\theta}_j)$, where $\hat{\theta}_j = \text{Median}(\{X_{ij}\}_{i=1}^n)$;

Medians

1. Coordinatewise median

$\hat{\theta} = (\hat{\theta}_j)$, where $\hat{\theta}_j = \text{Median}(\{X_{ij}\}_{i=1}^n)$;

2. Tukey's median

$$\hat{\theta} = \arg \max_{\eta \in \mathbb{R}^p} \min_{\|u\|=1} \frac{1}{n} \sum_{i=1}^n \mathbb{I}\{u^T X_i > u^T \eta\}.$$

Comparisons

	Coordinatewise Median	Tukey's Median
breakdown point	$1/2$	$1/3$
statistical precision (no contamination)	$\frac{p}{n}$	$\frac{p}{n}$
statistical precision (with contamination)	$\frac{p}{n} + p\epsilon^2$	$\frac{p}{n} + \epsilon^2$: minimax [Chen-Gao-Ren'15]
computational complexity	Polynomial	NP-hard [Amenta et al. '00]

Note: R-package for Tukey median can not deal with more than **10** dimensions!

[<https://github.com/ChenMengjie/DepthDescent>]

Multivariate Location Depth

$$\text{Definition: } \left\{ \frac{1}{n} \sum_{i=1}^n \mathbb{I}\{u^T X_i > u^T \eta\} \wedge \frac{1}{n} \sum_{i=1}^n \mathbb{I}\{u^T X_i \leq u^T \eta\} \right\}$$

Multivariate Location Depth

$$\min_{\|u\|=1} \left\{ \frac{1}{n} \sum_{i=1}^n \mathbb{I}\{u^T X_i > u^T \eta\} \wedge \frac{1}{n} \sum_{i=1}^n \mathbb{I}\{u^T X_i \leq u^T \eta\} \right\}$$

Multivariate Location Depth

$$\hat{\theta} = \arg \max_{\eta \in \mathbb{R}^p} \min_{\|u\|=1} \left\{ \frac{1}{n} \sum_{i=1}^n \mathbb{I}\{u^T X_i > u^T \eta\} \wedge \frac{1}{n} \sum_{i=1}^n \mathbb{I}\{u^T X_i \leq u^T \eta\} \right\}$$

Multivariate Location Depth

$$\hat{\theta} = \arg \max_{\eta \in \mathbb{R}^p} \min_{\|u\|=1} \left\{ \frac{1}{n} \sum_{i=1}^n \mathbb{I}\{u^T X_i > u^T \eta\} \wedge \frac{1}{n} \sum_{i=1}^n \mathbb{I}\{u^T X_i \leq u^T \eta\} \right\}$$

$$= \arg \max_{\eta \in \mathbb{R}^p} \min_{\|u\|=1} \frac{1}{n} \sum_{i=1}^n \mathbb{I}\{u^T X_i > u^T \eta\}.$$

[Tukey, 1975]

Regression Depth

model

$$y|X \sim N(X^T\beta, \sigma^2)$$

Regression Depth

model

$$y|X \sim N(X^T \beta, \sigma^2)$$

embedding

$$Xy|X \sim N(XX^T \beta, \sigma^2 XX^T)$$

Regression Depth

model

$$y|X \sim N(X^T \beta, \sigma^2)$$

embedding

$$Xy|X \sim N(XX^T \beta, \sigma^2 XX^T)$$

projection

$$u^T X y | X \sim N(u^T X X^T \beta, \sigma^2 u^T X X^T u)$$

Regression Depth

model

$$y|X \sim N(X^T \beta, \sigma^2)$$

embedding

$$Xy|X \sim N(XX^T \beta, \sigma^2 XX^T)$$

projection

$$u^T X y | X \sim N(u^T X X^T \beta, \sigma^2 u^T X X^T u)$$

$$\left\{ \frac{1}{n} \sum_{i=1}^n \mathbb{I}\{u^T X_i (y_i - X_i^T \eta) > 0\} \wedge \frac{1}{n} \sum_{i=1}^n \mathbb{I}\{u^T X_i (y_i - X_i^T \eta) \leq 0\} \right\}$$

Regression Depth

model

$$y|X \sim N(X^T \beta, \sigma^2)$$

embedding

$$Xy|X \sim N(XX^T \beta, \sigma^2 XX^T)$$

projection

$$u^T X y | X \sim N(u^T X X^T \beta, \sigma^2 u^T X X^T u)$$

$$\min_{u \in \mathbb{R}^p} \left\{ \frac{1}{n} \sum_{i=1}^n \mathbb{I}\{u^T X_i (y_i - X_i^T \eta) > 0\} \wedge \frac{1}{n} \sum_{i=1}^n \mathbb{I}\{u^T X_i (y_i - X_i^T \eta) \leq 0\} \right\}$$

Regression Depth

model

$$y|X \sim N(X^T \beta, \sigma^2)$$

embedding

$$Xy|X \sim N(XX^T \beta, \sigma^2 XX^T)$$

projection

$$u^T X y | X \sim N(u^T X X^T \beta, \sigma^2 u^T X X^T u)$$

$$\hat{\beta} = \operatorname{argmax}_{\eta \in \mathbb{R}^p} \min_{u \in \mathbb{R}^p} \left\{ \frac{1}{n} \sum_{i=1}^n \mathbb{I}\{u^T X_i (y_i - X_i^T \eta) > 0\} \wedge \frac{1}{n} \sum_{i=1}^n \mathbb{I}\{u^T X_i (y_i - X_i^T \eta) \leq 0\} \right\}$$

Regression Depth

model

$$y|X \sim N(X^T \beta, \sigma^2)$$

embedding

$$Xy|X \sim N(XX^T \beta, \sigma^2 XX^T)$$

projection

$$u^T X y | X \sim N(u^T X X^T \beta, \sigma^2 u^T X X^T u)$$

$$\hat{\beta} = \operatorname{argmax}_{\eta \in \mathbb{R}^p} \min_{u \in \mathbb{R}^p} \left\{ \frac{1}{n} \sum_{i=1}^n \mathbb{I}\{u^T X_i (y_i - X_i^T \eta) > 0\} \wedge \frac{1}{n} \sum_{i=1}^n \mathbb{I}\{u^T X_i (y_i - X_i^T \eta) \leq 0\} \right\}$$

[Rousseeuw & Hubert, 1999]

Tukey's depth is not a special case of regression depth.

Multi-task Regression Depth

$$(X, Y) \in \mathbb{R}^p \times \mathbb{R}^m \sim \mathbb{P}$$

Multi-task Regression Depth

$$(X, Y) \in \mathbb{R}^p \times \mathbb{R}^m \sim \mathbb{P}$$

$$B \in \mathbb{R}^{p \times m}$$

Multi-task Regression Depth

$$(X, Y) \in \mathbb{R}^p \times \mathbb{R}^m \sim \mathbb{P}$$

$$B \in \mathbb{R}^{p \times m}$$

population version:

$$\mathcal{D}_{\mathcal{U}}(B, \mathbb{P}) = \inf_{U \in \mathcal{U}} \mathbb{P} \left\{ \langle U^T X, Y - B^T X \rangle \geq 0 \right\}$$

Multi-task Regression Depth

$$(X, Y) \in \mathbb{R}^p \times \mathbb{R}^m \sim \mathbb{P}$$

$$B \in \mathbb{R}^{p \times m}$$

population version:

$$\mathcal{D}_{\mathcal{U}}(B, \mathbb{P}) = \inf_{U \in \mathcal{U}} \mathbb{P} \left\{ \langle U^T X, Y - B^T X \rangle \geq 0 \right\}$$

empirical version:

$$\mathcal{D}_{\mathcal{U}}(B, \{(X_i, Y_i)\}_{i=1}^n) = \inf_{U \in \mathcal{U}} \frac{1}{n} \sum_{i=1}^n \mathbb{I} \left\{ \langle U^T X_i, Y_i - B^T X_i \rangle \geq 0 \right\}$$

Multi-task Regression Depth

$$(X, Y) \in \mathbb{R}^p \times \mathbb{R}^m \sim \mathbb{P}$$

$$B \in \mathbb{R}^{p \times m}$$

population version:

$$\mathcal{D}_{\mathcal{U}}(B, \mathbb{P}) = \inf_{U \in \mathcal{U}} \mathbb{P} \left\{ \langle U^T X, Y - B^T X \rangle \geq 0 \right\}$$

empirical version:

$$\mathcal{D}_{\mathcal{U}}(B, \{(X_i, Y_i)\}_{i=1}^n) = \inf_{U \in \mathcal{U}} \frac{1}{n} \sum_{i=1}^n \mathbb{I} \left\{ \langle U^T X_i, Y_i - B^T X_i \rangle \geq 0 \right\}$$

Multi-task Regression Depth

$$\mathcal{D}_{\mathcal{U}}(B, \mathbb{P}) = \inf_{U \in \mathcal{U}} \mathbb{P} \left\{ \langle U^T X, Y - B^T X \rangle \geq 0 \right\}$$

Multi-task Regression Depth

$$\mathcal{D}_{\mathcal{U}}(B, \mathbb{P}) = \inf_{U \in \mathcal{U}} \mathbb{P} \left\{ \langle U^T X, Y - B^T X \rangle \geq 0 \right\}$$

$$p = 1, X = 1 \in \mathbb{R},$$

$$\mathcal{D}_{\mathcal{U}}(b, \mathbb{P}) = \inf_{u \in \mathcal{U}} \mathbb{P} \left\{ u^T (Y - b) \geq 0 \right\}$$

Multi-task Regression Depth

$$\mathcal{D}_{\mathcal{U}}(B, \mathbb{P}) = \inf_{U \in \mathcal{U}} \mathbb{P} \left\{ \langle U^T X, Y - B^T X \rangle \geq 0 \right\}$$

$$p = 1, X = 1 \in \mathbb{R},$$

$$\mathcal{D}_{\mathcal{U}}(b, \mathbb{P}) = \inf_{u \in \mathcal{U}} \mathbb{P} \left\{ u^T (Y - b) \geq 0 \right\}$$

$$m = 1,$$

$$\mathcal{D}_{\mathcal{U}}(\beta, \mathbb{P}) = \inf_{U \in \mathcal{U}} \mathbb{P} \left\{ u^T X (y - \beta^T X) \geq 0 \right\}$$

Multi-task Regression Depth

Proposition. For any $\delta > 0$,

$$\sup_{B \in \mathbb{R}^{p \times m}} |\mathcal{D}(B, \mathbb{P}_n) - \mathcal{D}(B, \mathbb{P})| \leq C \sqrt{\frac{pm}{n}} + \sqrt{\frac{\log(1/\delta)}{2n}},$$

with probability at least $1 - 2\delta$.

Multi-task Regression Depth

Proposition. For any $\delta > 0$,

$$\sup_{B \in \mathbb{R}^{p \times m}} |\mathcal{D}(B, \mathbb{P}_n) - \mathcal{D}(B, \mathbb{P})| \leq C \sqrt{\frac{pm}{n}} + \sqrt{\frac{\log(1/\delta)}{2n}},$$

with probability at least $1 - 2\delta$.

Proposition.

$$\sup_{B,Q} |\mathcal{D}(B, (1 - \epsilon P_{B^*}) + \epsilon Q) - \mathcal{D}(B, P_{B^*})| \leq \epsilon$$

Multi-task Regression Depth

$$(X, Y) \sim P_B$$

Multi-task Regression Depth

$$(X, Y) \sim P_B : X \sim N(0, \Sigma), \quad Y|X \sim N(B^T X, \sigma^2 I_m)$$

Multi-task Regression Depth

$$(X, Y) \sim P_B : X \sim N(0, \Sigma), \quad Y|X \sim N(B^T X, \sigma^2 I_m)$$

$$(X_1, Y_1), ..., (X_n, Y_n) \sim (1 - \epsilon)P_B + \epsilon Q$$

Multi-task Regression Depth

$$(X, Y) \sim P_B : X \sim N(0, \Sigma), \quad Y|X \sim N(B^T X, \sigma^2 I_m)$$

$$(X_1, Y_1), \dots, (X_n, Y_n) \sim (1 - \epsilon)P_B + \epsilon Q$$

Theorem [G17]. For some $C > 0$,

$$\text{Tr}((\widehat{B} - B)^T \Sigma (\widehat{B} - B)) \leq C \sigma^2 \left(\frac{pm}{n} \vee \epsilon^2 \right),$$

$$\|\widehat{B} - B\|_{\text{F}}^2 \leq C \frac{\sigma^2}{\kappa^2} \left(\frac{pm}{n} \vee \epsilon^2 \right),$$

with high probability uniformly over B, Q .

Covariance Matrix

$$X_1, \dots, X_n \sim (1 - \epsilon)N(0, \Sigma) + \epsilon Q.$$

Covariance Matrix

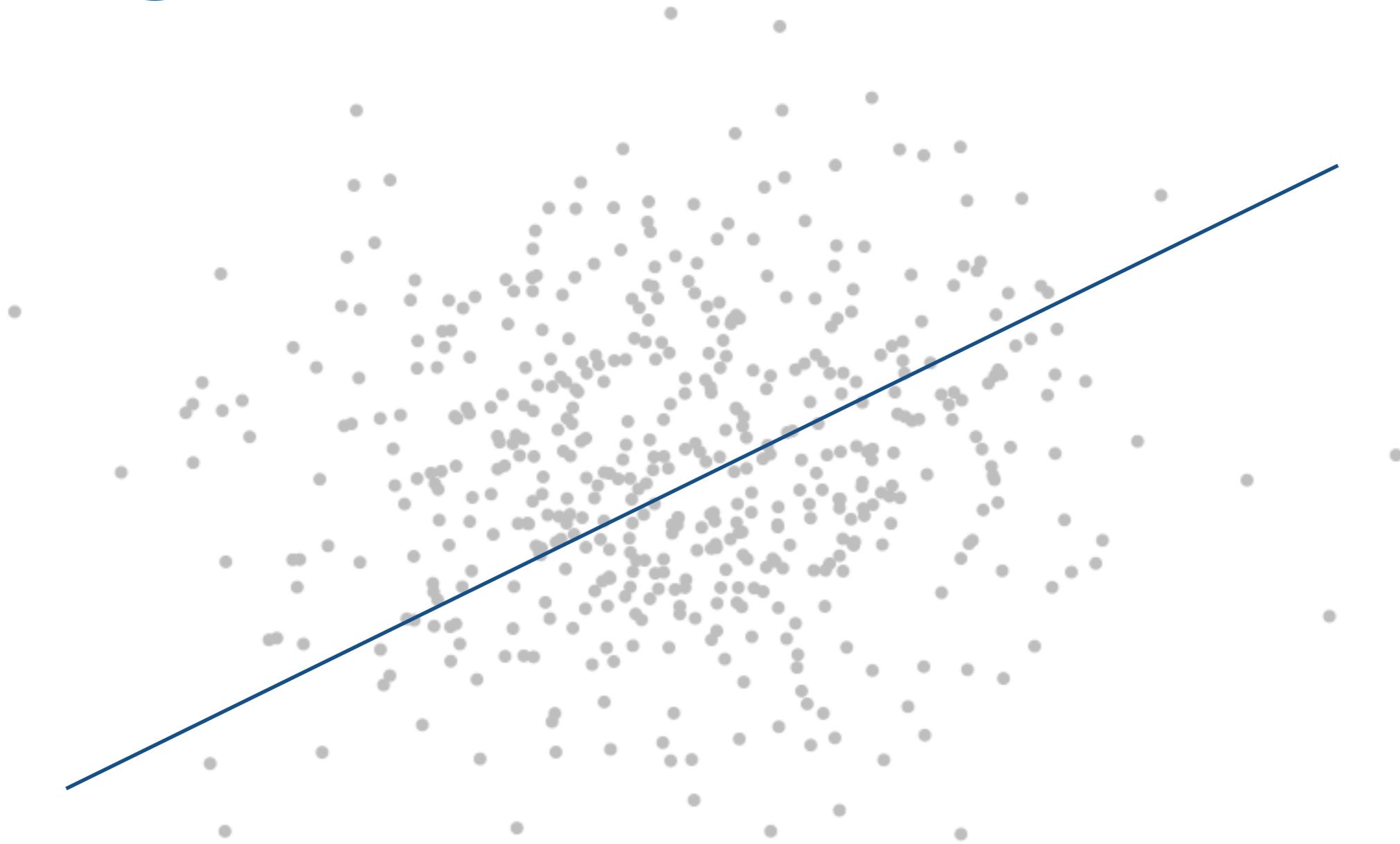
$$X_1, \dots, X_n \sim (1 - \epsilon)N(0, \Sigma) + \epsilon Q.$$

how to estimate ?

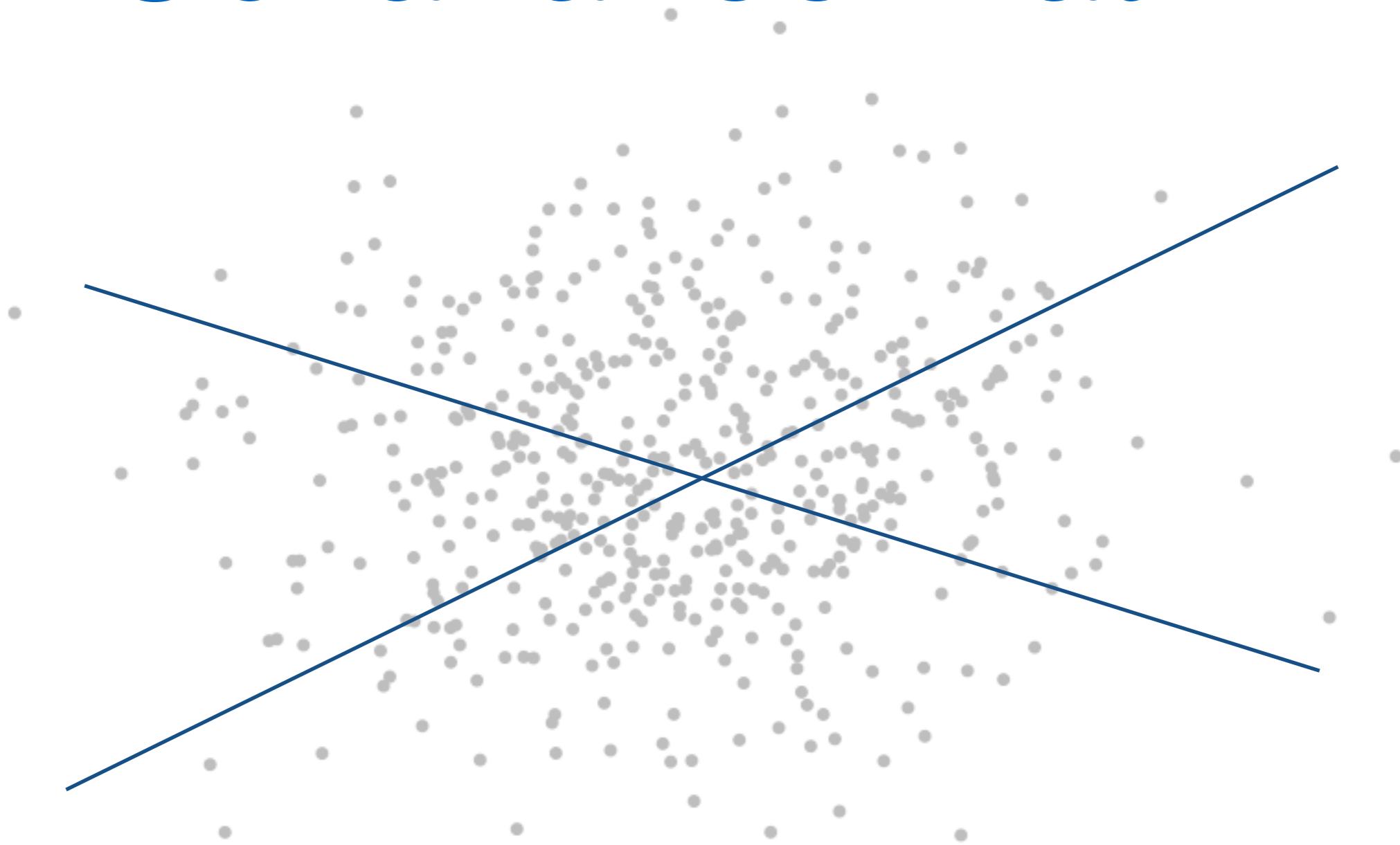
Covariance Matrix



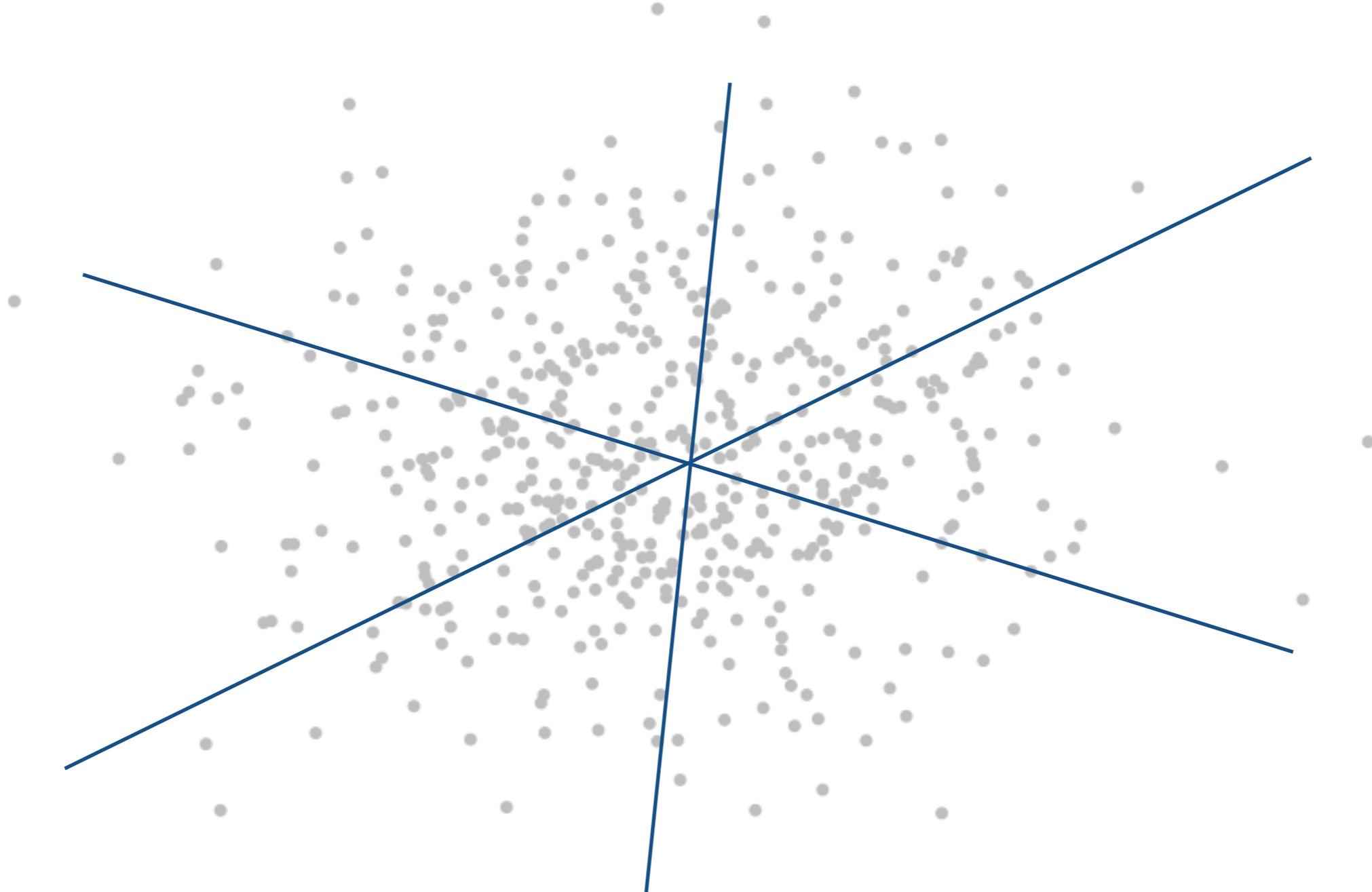
Covariance Matrix



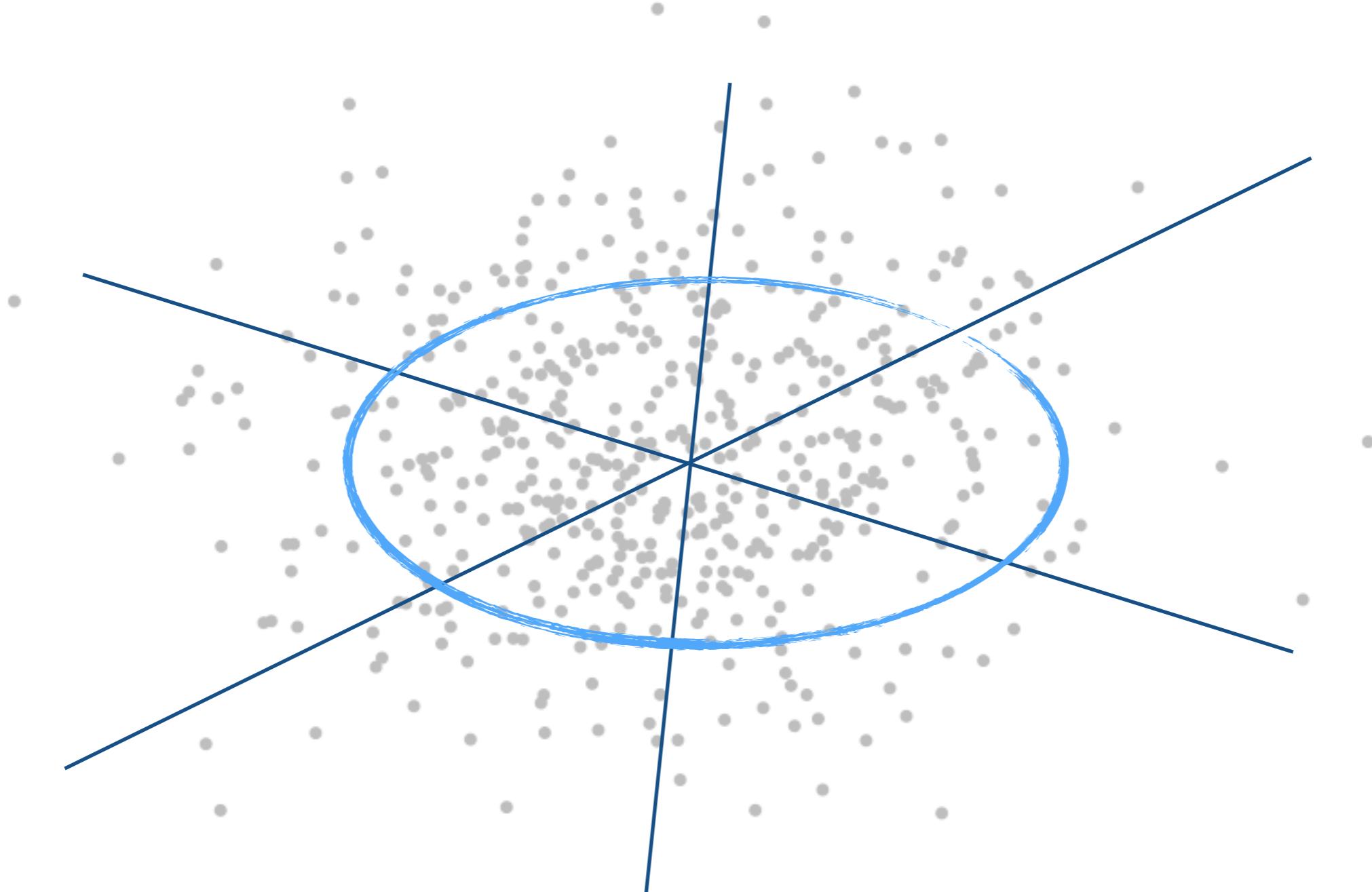
Covariance Matrix



Covariance Matrix



Covariance Matrix



Covariance Matrix

$$\mathcal{D}(\Gamma, \{X_i\}_{i=1}^n) = \min_{\|u\|=1} \min \left\{ \frac{1}{n} \sum_{i=1}^n \mathbb{I}\{|u^T X_i|^2 \geq u^T \Gamma u\}, \frac{1}{n} \sum_{i=1}^n \mathbb{I}\{|u^T X_i|^2 < u^T \Gamma u\} \right\}$$

Covariance Matrix

$$\mathcal{D}(\Gamma, \{X_i\}_{i=1}^n) = \min_{\|u\|=1} \min \left\{ \frac{1}{n} \sum_{i=1}^n \mathbb{I}\{|u^T X_i|^2 \geq u^T \Gamma u\}, \frac{1}{n} \sum_{i=1}^n \mathbb{I}\{|u^T X_i|^2 < u^T \Gamma u\} \right\}$$

$$\hat{\Gamma} = \arg \max_{\Gamma \succeq 0} \mathcal{D}(\Gamma, \{X_i\}_{i=1}^n) \quad \quad \hat{\Sigma} = \hat{\Gamma}/\beta$$

Covariance Matrix

$$\mathcal{D}(\Gamma, \{X_i\}_{i=1}^n) = \min_{\|u\|=1} \min \left\{ \frac{1}{n} \sum_{i=1}^n \mathbb{I}\{|u^T X_i|^2 \geq u^T \Gamma u\}, \frac{1}{n} \sum_{i=1}^n \mathbb{I}\{|u^T X_i|^2 < u^T \Gamma u\} \right\}$$

$$\hat{\Gamma} = \arg \max_{\Gamma \succeq 0} \mathcal{D}(\Gamma, \{X_i\}_{i=1}^n) \quad \hat{\Sigma} = \hat{\Gamma}/\beta$$

Theorem [CGR15]. For some $C > 0$,

$$\|\hat{\Sigma} - \Sigma\|_{\text{op}}^2 \leq C \left(\frac{p}{n} \vee \epsilon^2 \right)$$

with high probability uniformly over Σ, Q .

Summary

mean	$\ \cdot\ ^2$	$\frac{p}{n} \vee \epsilon^2$
reduced rank regression	$\ \cdot\ _F^2$	$\frac{\sigma^2}{\kappa^2} \frac{r(p+m)}{n} \vee \frac{\sigma^2}{\kappa^2} \epsilon^2$
Gaussian graphical model	$\ \cdot\ _{\ell_1}^2$	$\frac{s^2 \log(ep/s)}{n} \vee s\epsilon^2$
covariance matrix	$\ \cdot\ _{\text{op}}^2$	$\frac{p}{n} \vee \epsilon^2$
sparse PCA	$\ \cdot\ _F^2$	$\frac{s \log(ep/s)}{n\lambda^2} \vee \frac{\epsilon^2}{\lambda^2}$

Summary

mean	$\ \cdot\ ^2$	$\frac{p}{n} \sqrt{\epsilon^2}$
reduced rank regression	$\ \cdot\ _F^2$	$\frac{\sigma^2}{\kappa^2} \frac{r(p+m)}{n} \sqrt{\frac{\sigma^2}{\kappa^2} \epsilon^2}$
Gaussian graphical model	$\ \cdot\ _{\ell_1}^2$	$\frac{s^2 \log(ep/s)}{n} \sqrt{s\epsilon^2}$
covariance matrix	$\ \cdot\ _{\text{op}}^2$	$\frac{p}{n} \sqrt{\epsilon^2}$
sparse PCA	$\ \cdot\ _F^2$	$\frac{s \log(ep/s)}{n\lambda^2} \sqrt{\frac{\epsilon^2}{\lambda^2}}$

Computation

Computational Challenges

$$X_1, \dots, X_n \sim (1 - \epsilon)N(\theta, I_p) + \epsilon Q.$$

Computational Challenges

$$X_1, \dots, X_n \sim (1 - \epsilon)N(\theta, I_p) + \epsilon Q.$$

Lai, Rao, Vempala
Diakonikolas, Kamath, Kane, Li, Moitra, Stewart
Balakrishnan, Du, Singh

Computational Challenges

$$X_1, \dots, X_n \sim (1 - \epsilon)N(\theta, I_p) + \epsilon Q.$$

Lai, Rao, Vempala
Diakonikolas, Kamath, Kane, Li, Moitra, Stewart
Balakrishnan, Du, Singh

- Polynomial algorithms are proposed [[Diakonikolas et al.'16](#), [Lai et al. 16](#)] of minimax optimal statistical precision
 - needs information on second or higher order of moments
 - some priori knowledge about ϵ

Advantages of Tukey Median

-

Advantages of Tukey Median

- A well-defined objective function

Advantages of Tukey Median

- **A well-defined objective function**
- **Adaptive to ϵ and Σ**

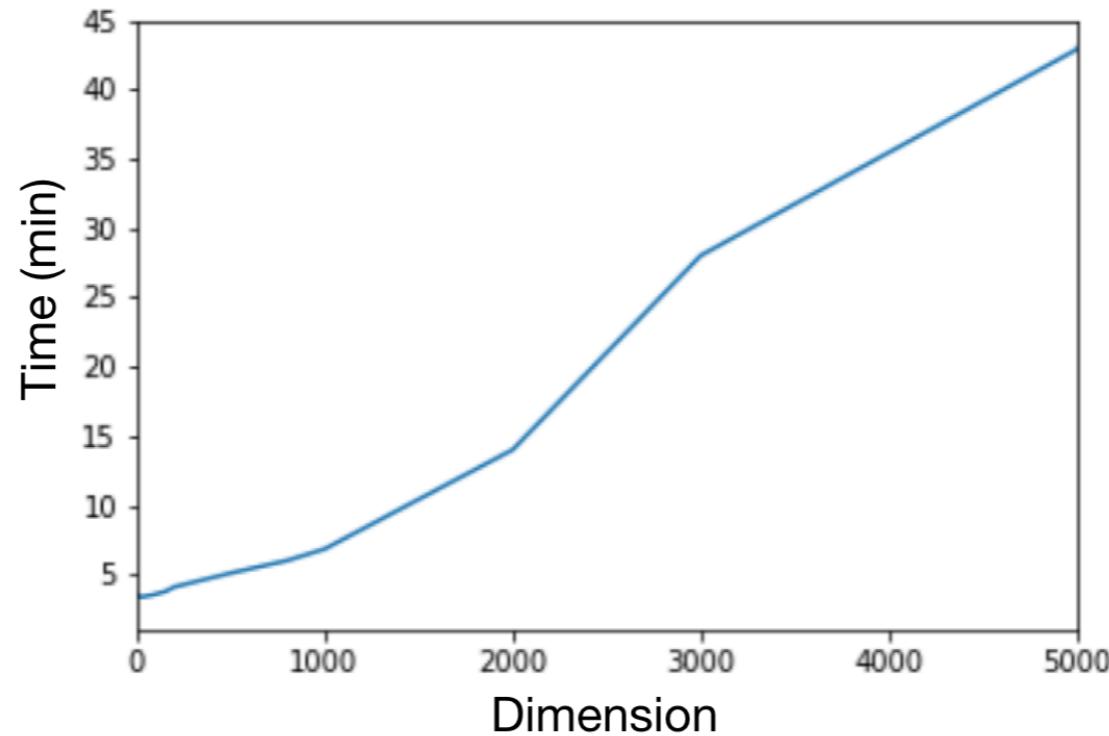
Advantages of Tukey Median

- A well-defined objective function
- Adaptive to ϵ and Σ
- Optimal for any elliptical distribution

A practically good algorithm?

Generative Adversarial Networks

[Goodfellow et al. 2014]



Note: R-package for Tukey median can not deal with more than 10 dimensions [<https://github.com/ChenMengjie/DepthDescent>]

Robust Learning of Cauchy Distributions

Table 4: Comparison of various methods of robust location estimation under Cauchy distributions. Samples are drawn from $(1 - \epsilon)\text{Cauchy}(0_p, I_p) + \epsilon Q$ with $\epsilon = 0.2, p = 50$ and various choices of Q . Sample size: 50,000. Discriminator net structure: 50-50-25-1. Generator $g_\omega(\xi)$ structure: 48-48-32-24-12-1 with absolute value activation function in the output layer.

Contamination Q	JS-GAN (G_1)	JS-GAN (G_2)	Dimension Halving	Iterative Filtering
Cauchy($1.5 * 1_p, I_p$)	0.0664 (0.0065)	0.0743 (0.0103)	0.3529 (0.0543)	0.1244 (0.0114)
Cauchy($5.0 * 1_p, I_p$)	0.0480 (0.0058)	0.0540 (0.0064)	0.4855 (0.0616)	0.1687 (0.0310)
Cauchy($1.5 * 1_p, 5 * I_p$)	0.0754 (0.0135)	0.0742 (0.0111)	0.3726 (0.0530)	0.1220 (0.0112)
Normal($1.5 * 1_p, 5 * I_p$)	0.0702 (0.0064)	0.0713 (0.0088)	0.3915 (0.0232)	0.1048 (0.0288))

- *Dimension Halving:* [Lai et al.'16]
<https://github.com/kal2000/AgnosticMeanAndCovarianceCode>.
- *Iterative Filtering:* [Diakonikolas et al.'17]
<https://github.com/hoonose/robust-filter>.

f-GAN

Given a strictly convex function f that satisfies $f(1) = 0$, the f -divergence between two probability distributions P and Q is defined by

$$D_f(P\|Q) = \int f\left(\frac{p}{q}\right) dQ. \quad (8)$$

Let f^* be the convex conjugate of f . A variational lower bound of (8) is

$$D_f(P\|Q) \geq \sup_{T \in \mathcal{T}} [\mathbb{E}_P T(X) - \mathbb{E}_Q f^*(T(X))]. \quad (9)$$

where equality holds whenever the class \mathcal{T} contains the function $f'(p/q)$.

[Nowozin-Cseke-Tomioka'16] f-GAN minimizes the variational lower bound (9)

$$\hat{P} = \arg \min_{Q \in \mathcal{Q}} \sup_{T \in \mathcal{T}} \left[\frac{1}{n} \sum_{i=1}^n T(X_i) - \mathbb{E}_Q f^*(T(X)) \right]. \quad (10)$$

with i.i.d. observations $X_1, \dots, X_n \sim P$.

From f-GAN to Tukey's Median: f-learning

Consider the special case

$$\mathcal{T} = \left\{ f' \left(\frac{\tilde{q}}{q} \right) : \tilde{q} \in \tilde{\mathcal{Q}} \right\}. \quad (11)$$

which is tight if $P \in \tilde{\mathcal{Q}}$. The sample version leads to the following f -learning

$$\hat{P} = \arg \min_{Q \in \mathcal{Q}} \sup_{\tilde{Q} \in \tilde{\mathcal{Q}}} \left[\frac{1}{n} \sum_{i=1}^n f' \left(\frac{\tilde{q}(X_i)}{q(X_i)} \right) - \mathbb{E}_Q f^* \left(f' \left(\frac{\tilde{q}(X)}{q(X)} \right) \right) \right]. \quad (12)$$

- If $f(x) = x \log x$, $\mathcal{Q} = \tilde{\mathcal{Q}}$, (12) \Rightarrow Maximum Likelihood Estimate
- If $f(x) = (x - 1) +$, then $D_f(P||Q) = \frac{1}{2} \int |p - q|$ is the TV-distance,
 $f^*(t) = t \mathbb{I}\{0 \leq t \leq 1\}$, f -GAN \Rightarrow TV-GAN
 - $\mathcal{Q} = \{N(\eta, I_p) : \eta \in \mathbb{R}^p\}$ and $\tilde{\mathcal{Q}} = \{N(\tilde{\eta}, I_p) : \|\tilde{\eta} - \eta\| \leq r\}$, (12) $\stackrel{r \rightarrow 0}{\Rightarrow}$ Tukey's Median

f-Learning

f-Learning

Jensen-Shannon

$$f(x) = x \log x - (x + 1) \log(x + 1)$$

GAN

[Goodfellow et al.]

f-Learning

Jensen-Shannon	$f(x) = x \log x - (x + 1) \log(x + 1)$	GAN
Kullback-Leibler	$f(x) = x \log x$	MLE

[Goodfellow et al.]

f-Learning

Jensen-Shannon	$f(x) = x \log x - (x + 1) \log(x + 1)$	GAN
Kullback-Leibler	$f(x) = x \log x$	MLE
Hellinger Squared	$f(x) = 2 - 2\sqrt{x}$	rho

[Goodfellow et al., Baraud and Birge]

f-Learning

Jensen-Shannon	$f(x) = x \log x - (x + 1) \log(x + 1)$	GAN
Kullback-Leibler	$f(x) = x \log x$	MLE
Hellinger Squared	$f(x) = 2 - 2\sqrt{x}$	rho
Total Variation	$f(x) = (x - 1)_+$	depth

[Goodfellow et al., Baraud and Birge]

TV-Learning

$$\min_{Q \in \mathcal{Q}} \max_{\tilde{Q} \in \tilde{\mathcal{Q}}} \left\{ \frac{1}{n} \sum_{i=1}^n \mathbb{I} \left\{ \frac{\tilde{q}(X_i)}{q(X_i)} \geq 1 \right\} - Q \left(\frac{\tilde{q}}{q} \geq 1 \right) \right\}$$

TV-Learning

$$\min_{Q \in \mathcal{Q}} \max_{\tilde{Q} \in \tilde{\mathcal{Q}}} \left\{ \frac{1}{n} \sum_{i=1}^n \mathbb{I} \left\{ \frac{\tilde{q}(X_i)}{q(X_i)} \geq 1 \right\} - Q \left(\frac{\tilde{q}}{q} \geq 1 \right) \right\}$$

$$\mathcal{Q} = \left\{ N(\theta, I_p) : \theta \in \mathbb{R}^p \right\} \quad \tilde{\mathcal{Q}} = \left\{ N(\tilde{\theta}, I_p) : \tilde{\theta} \in \mathcal{N}_r(\theta) \right\}$$

TV-Learning

$$\min_{Q \in \mathcal{Q}} \max_{\tilde{Q} \in \tilde{\mathcal{Q}}} \left\{ \frac{1}{n} \sum_{i=1}^n \mathbb{I} \left\{ \frac{\tilde{q}(X_i)}{q(X_i)} \geq 1 \right\} - Q \left(\frac{\tilde{q}}{q} \geq 1 \right) \right\}$$

$$\mathcal{Q} = \left\{ N(\theta, I_p) : \theta \in \mathbb{R}^p \right\} \quad \tilde{\mathcal{Q}} = \left\{ N(\tilde{\theta}, I_p) : \tilde{\theta} \in \mathcal{N}_r(\theta) \right\}$$



$$r \rightarrow 0$$

TV-Learning

$$\min_{Q \in \mathcal{Q}} \max_{\tilde{Q} \in \tilde{\mathcal{Q}}} \left\{ \frac{1}{n} \sum_{i=1}^n \mathbb{I} \left\{ \frac{\tilde{q}(X_i)}{q(X_i)} \geq 1 \right\} - Q \left(\frac{\tilde{q}}{q} \geq 1 \right) \right\}$$

$$\mathcal{Q} = \left\{ N(\theta, I_p) : \theta \in \mathbb{R}^p \right\} \quad \tilde{\mathcal{Q}} = \left\{ N(\tilde{\theta}, I_p) : \tilde{\theta} \in \mathcal{N}_r(\theta) \right\}$$



$r \rightarrow 0$

Tukey depth

$$\max_{\theta \in \mathbb{R}} \min_{\|u\|=1} \frac{1}{n} \sum_{i=1}^n \mathbb{I} \left\{ u^T X_i \geq u^T \theta \right\}$$

TV-Learning

$$\min_{Q \in \mathcal{Q}} \max_{\tilde{Q} \in \tilde{\mathcal{Q}}} \left\{ \frac{1}{n} \sum_{i=1}^n \mathbb{I} \left\{ \frac{\tilde{q}(X_i)}{q(X_i)} \geq 1 \right\} - Q \left(\frac{\tilde{q}}{q} \geq 1 \right) \right\}$$

TV-Learning

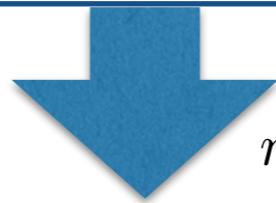
$$\min_{Q \in \mathcal{Q}} \max_{\tilde{Q} \in \tilde{\mathcal{Q}}} \left\{ \frac{1}{n} \sum_{i=1}^n \mathbb{I} \left\{ \frac{\tilde{q}(X_i)}{q(X_i)} \geq 1 \right\} - Q \left(\frac{\tilde{q}}{q} \geq 1 \right) \right\}$$

$$\mathcal{Q} = \left\{ N(0, \Sigma) : \Sigma \in \mathbb{R}^{p \times p} \right\} \quad \tilde{\mathcal{Q}} = \left\{ N(0, \tilde{\Sigma}) : \tilde{\Sigma} = \Sigma + r u u^T, \|u\| = 1 \right\}$$

TV-Learning

$$\min_{Q \in \mathcal{Q}} \max_{\tilde{Q} \in \tilde{\mathcal{Q}}} \left\{ \frac{1}{n} \sum_{i=1}^n \mathbb{I} \left\{ \frac{\tilde{q}(X_i)}{q(X_i)} \geq 1 \right\} - Q \left(\frac{\tilde{q}}{q} \geq 1 \right) \right\}$$

$$\mathcal{Q} = \left\{ N(0, \Sigma) : \Sigma \in \mathbb{R}^{p \times p} \right\} \quad \tilde{\mathcal{Q}} = \left\{ N(0, \tilde{\Sigma}) : \tilde{\Sigma} = \Sigma + r u u^T, \|u\| = 1 \right\}$$



$$r \rightarrow 0$$

TV-Learning

$$\min_{Q \in \mathcal{Q}} \max_{\tilde{Q} \in \tilde{\mathcal{Q}}} \left\{ \frac{1}{n} \sum_{i=1}^n \mathbb{I} \left\{ \frac{\tilde{q}(X_i)}{q(X_i)} \geq 1 \right\} - Q \left(\frac{\tilde{q}}{q} \geq 1 \right) \right\}$$

$$\mathcal{Q} = \left\{ N(0, \Sigma) : \Sigma \in \mathbb{R}^{p \times p} \right\} \quad \tilde{\mathcal{Q}} = \left\{ N(0, \tilde{\Sigma}) : \tilde{\Sigma} = \Sigma + r u u^T, \|u\| = 1 \right\}$$



$r \rightarrow 0$

**(related to)
matrix depth**

$$\max_{\Sigma} \min_{\|u\|=1} \left[\left(\frac{1}{n} \sum_{i=1}^n \mathbb{I}\{|u^T X_i|^2 \leq u^T \Sigma u\} - \mathbb{P}(\chi_1^2 \leq 1) \right) \wedge \left(\frac{1}{n} \sum_{i=1}^n \mathbb{I}\{|u^T X_i|^2 > u^T \Sigma u\} - \mathbb{P}(\chi_1^2 > 1) \right) \right]$$

robust
statistics
community

deep
learning
community

robust
statistics
community

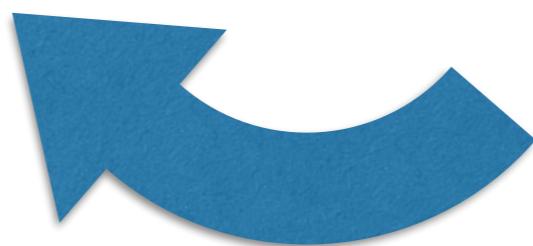
f-Learning
f-GAN

deep
learning
community

robust
statistics
community

f-Learning
f-GAN

deep
learning
community



practically good algorithms

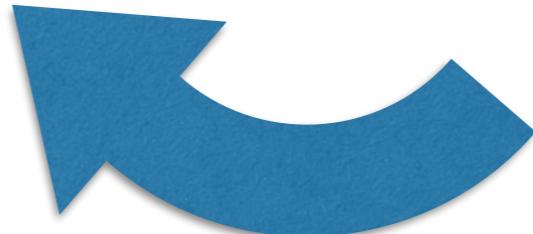
theoretical foundation



robust
statistics
community

f-Learning
f-GAN

deep
learning
community



practically good algorithms

TV-GAN

$$\widehat{\theta} = \operatorname{argmin}_{\eta} \sup_{w,b} \left[\frac{1}{n} \sum_{i=1}^n \frac{1}{1 + e^{-w^T X_i - b}} - E_{\eta} \frac{1}{1 + e^{-w^T X - b}} \right]$$

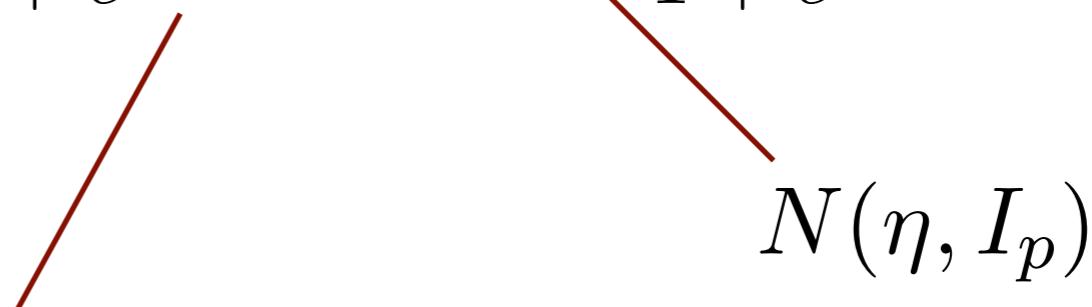
TV-GAN

$$\widehat{\theta} = \operatorname{argmin}_{\eta} \sup_{w,b} \left[\frac{1}{n} \sum_{i=1}^n \frac{1}{1 + e^{-w^T X_i - b}} - E_{\eta} \frac{1}{1 + e^{-w^T X - b}} \right]$$

$N(\eta, I_p)$

TV-GAN

$$\hat{\theta} = \operatorname{argmin}_{\eta} \sup_{w,b} \left[\frac{1}{n} \sum_{i=1}^n \frac{1}{1 + e^{-w^T X_i - b}} - E_{\eta} \frac{1}{1 + e^{-w^T X - b}} \right]$$


$$N(\eta, I_p)$$

logistic regression classifier

TV-GAN

$$\hat{\theta} = \operatorname{argmin}_{\eta} \sup_{w,b} \left[\frac{1}{n} \sum_{i=1}^n \frac{1}{1 + e^{-w^T X_i - b}} - E_{\eta} \frac{1}{1 + e^{-w^T X - b}} \right]$$


$$N(\eta, I_p)$$

logistic regression classifier

Theorem [GLYZ18]. For some $C > 0$,

$$\|\hat{\theta} - \theta\|^2 \leq C \left(\frac{p}{n} \vee \epsilon^2 \right)$$

with high probability uniformly over $\theta \in \mathbb{R}^p, Q$.

TV-GAN

rugged landscape!

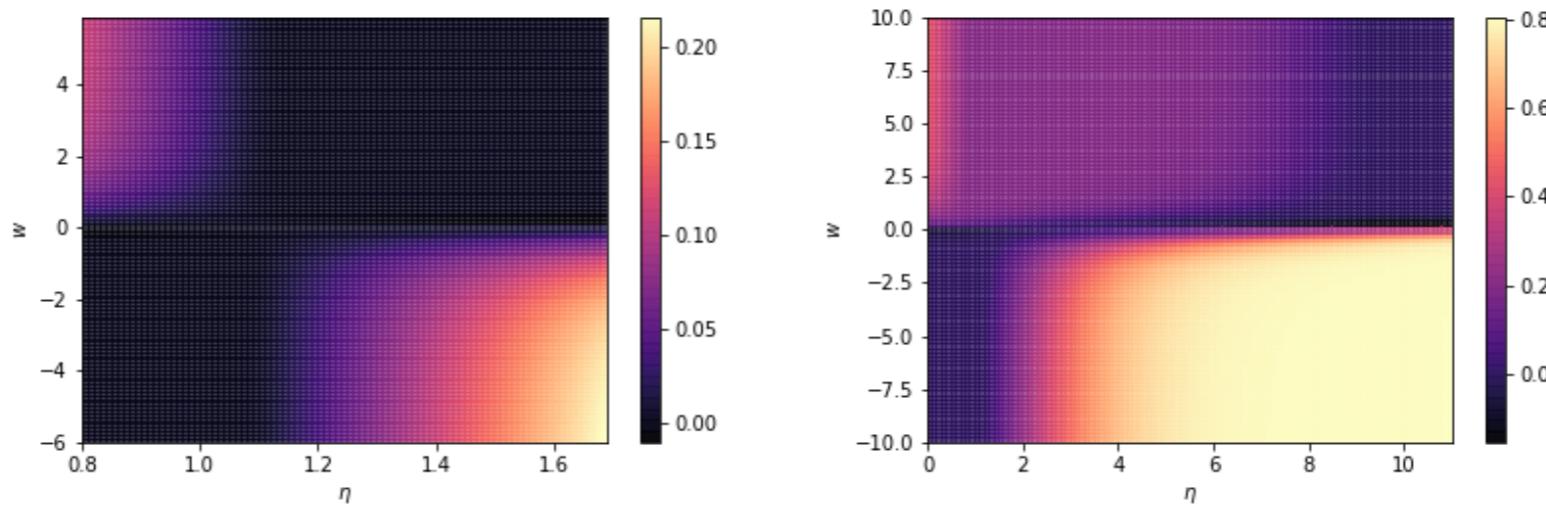


Figure: Heatmaps of the landscape of $F(\eta, w) = \sup_b [E_P \text{sigmoid}(wX + b) - E_{N(\eta, 1)} \text{sigmoid}(wX + b)]$, where b is maximized out for visualization. Left: samples are drawn from $P = (1 - \epsilon)N(1, 1) + \epsilon N(1.5, 1)$ with $\epsilon = 0.2$. Right: samples are drawn from $P = (1 - \epsilon)N(1, 1) + \epsilon N(10, 1)$ with $\epsilon = 0.2$. Left: the landscape is good in the sense that no matter whether we start from the left-top area or the right-bottom area of the heatmap, gradient ascent on η does not consistently increase or decrease the value of η . This is because the signal becomes weak when it is close to the saddle point around $\eta = 1$. Right: it is clear that $\tilde{F}(w) = F(\eta, w)$ has two local maxima for a given η , achieved at $w = +\infty$ and $w = -\infty$. In fact, the global maximum for $\tilde{F}(w)$ has a phase transition from $w = +\infty$ to $w = -\infty$ as η grows. For example, the maximum is achieved at $w = +\infty$ when $\eta = 1$ (blue solid) and is achieved at $w = -\infty$ when $\eta = 5$ (red solid). Unfortunately, even if we initialize with $\eta_0 = 1$ and $w_0 > 0$, gradient ascents on η will only increase the value of η (green dash), and thus as long as the discriminator cannot reach the global maximizer, w will be stuck in the positive half space $\{w : w > 0\}$ and further increase the value of η .

The Original JS-GAN

[Goodfellow et al. 2014] For $f(x) = x \log x - (x + 1) \log \frac{x+1}{2}$,

$$\hat{\theta} = \arg \min_{\eta \in \mathbb{R}^p} \max_{D \in \mathcal{D}} \left[\frac{1}{n} \sum_{i=1}^n \log D(X_i) + \mathbb{E}_{\mathcal{N}(\eta, I_p)} \log(1 - D(X)) \right] + \log 4. \quad (15)$$

What are \mathcal{D} , the class of discriminators?

- Single layer (no hidden layer):

$$\mathcal{D} = \left\{ D(x) = \text{sigmoid}(w^T x + b) : w \in \mathbb{R}^p, b \in \mathbb{R} \right\}$$

- One-hidden or Multiple layer:

$$\mathcal{D} = \left\{ D(x) = \text{sigmoid}(w^T g(X)) \right\}$$

JS-GAN

$$\widehat{\theta} = \operatorname{argmin}_{\eta \in \mathbb{R}^p} \max_{T \in \mathcal{T}} \left[\frac{1}{n} \sum_{i=1}^n \log T(X_i) + E_\eta \log(1 - T(X)) \right] + \log 4$$



JS-GAN

$$\widehat{\theta} = \operatorname{argmin}_{\eta \in \mathbb{R}^p} \max_{T \in \mathcal{T}} \left[\frac{1}{n} \sum_{i=1}^n \log T(X_i) + E_\eta \log(1 - T(X)) \right] + \log 4$$

numerical experiment

$$X_1, \dots, X_n \sim (1 - \epsilon)N(\theta, I_p) + \epsilon N(\tilde{\theta}, I_p)$$

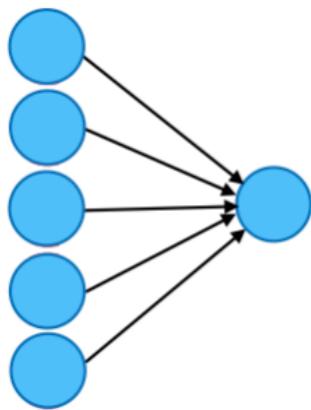


JS-GAN

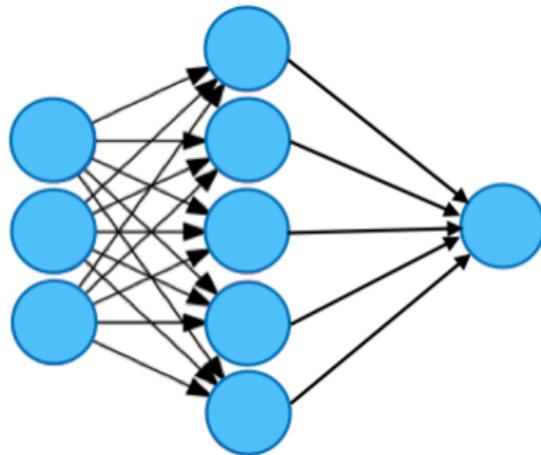
$$\hat{\theta} = \operatorname{argmin}_{\eta \in \mathbb{R}^p} \max_{T \in \mathcal{T}} \left[\frac{1}{n} \sum_{i=1}^n \log T(X_i) + E_\eta \log(1 - T(X)) \right] + \log 4$$

numerical experiment

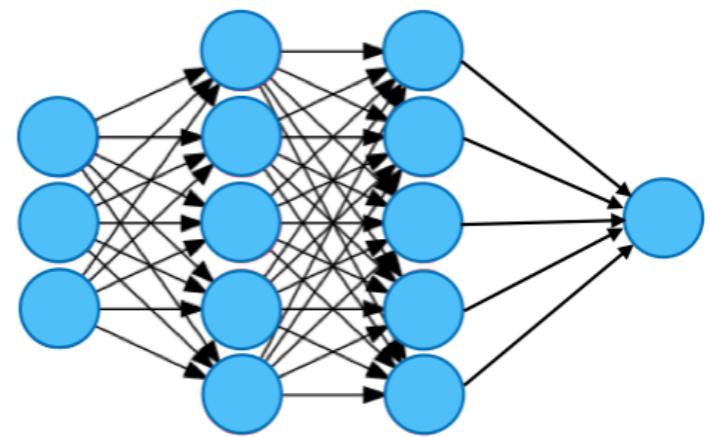
$$X_1, \dots, X_n \sim (1 - \epsilon)N(\theta, I_p) + \epsilon N(\tilde{\theta}, I_p)$$



✗



✓



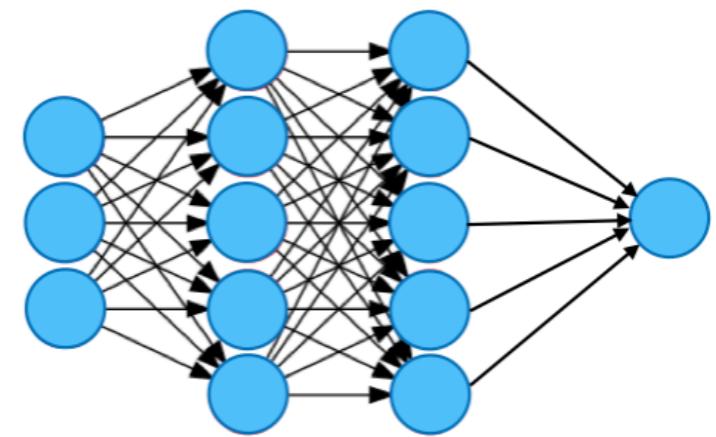
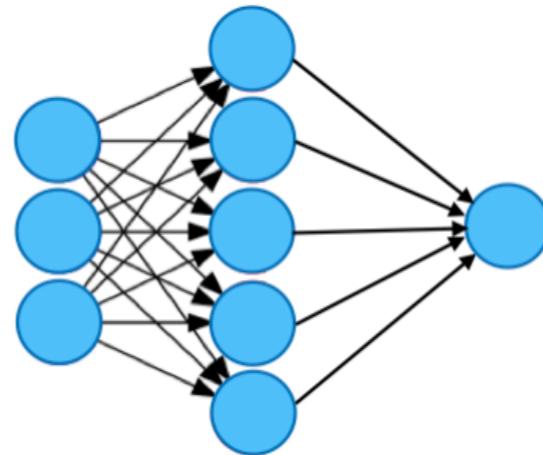
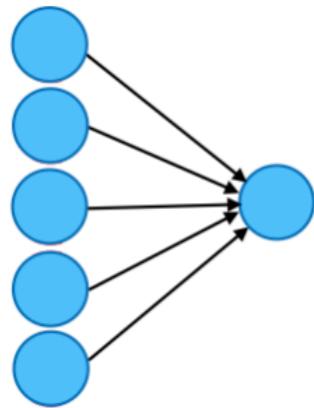
✓

JS-GAN

$$\hat{\theta} = \operatorname{argmin}_{\eta \in \mathbb{R}^p} \max_{T \in \mathcal{T}} \left[\frac{1}{n} \sum_{i=1}^n \log T(X_i) + E_\eta \log(1 - T(X)) \right] + \log 4$$

numerical experiment

$$X_1, \dots, X_n \sim (1 - \epsilon)N(\theta, I_p) + \epsilon N(\tilde{\theta}, I_p)$$



$$\hat{\theta} \approx (1 - \epsilon)\theta + \epsilon\tilde{\theta}$$

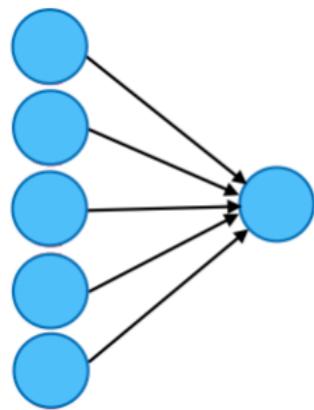


JS-GAN

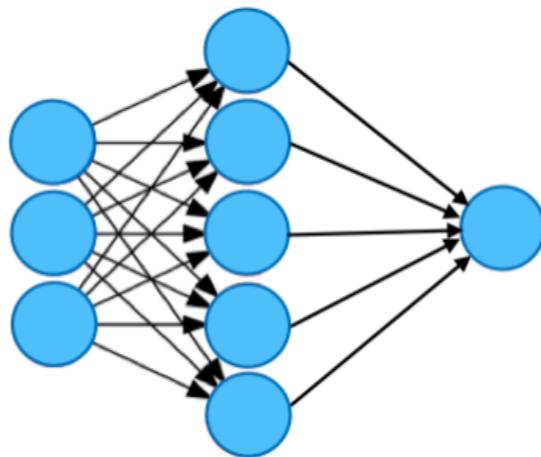
$$\widehat{\theta} = \operatorname{argmin}_{\eta \in \mathbb{R}^p} \max_{T \in \mathcal{T}} \left[\frac{1}{n} \sum_{i=1}^n \log T(X_i) + E_\eta \log(1 - T(X)) \right] + \log 4$$

numerical experiment

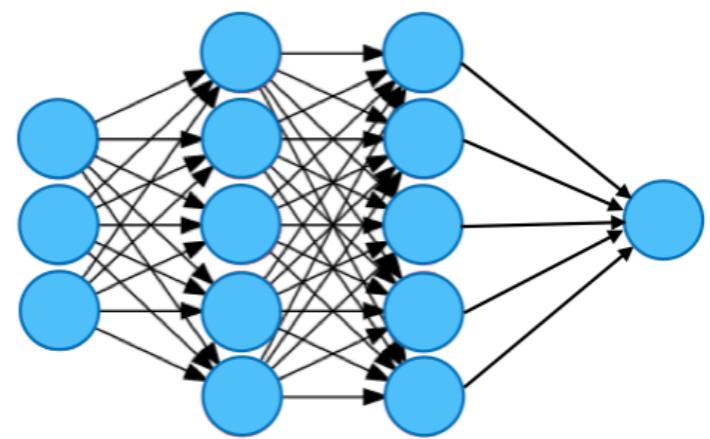
$$X_1, \dots, X_n \sim (1 - \epsilon)N(\theta, I_p) + \epsilon N(\tilde{\theta}, I_p)$$



$$\widehat{\theta} \approx (1 - \epsilon)\theta + \epsilon\widetilde{\theta}$$



$$\hat{\theta} \approx \theta$$



$$\hat{\theta} \approx \theta$$

JS-GAN

A classifier with hidden layers leads to robustness. Why?

JS-GAN

A classifier with hidden layers leads to robustness. Why?

$$\text{JS}_g(\mathbb{P}, \mathbb{Q}) = \max_{w \in \mathbb{R}^d} \left[\mathbb{P} \log \frac{1}{1 + e^{-w^T g(X)}} + \mathbb{Q} \log \frac{1}{1 + e^{w^T g(X)}} \right] + \log 4.$$

JS-GAN

A classifier with hidden layers leads to robustness. Why?

$$\text{JS}_g(\mathbb{P}, \mathbb{Q}) = \max_{w \in \mathbb{R}^d} \left[\mathbb{P} \log \frac{1}{1 + e^{-w^T g(X)}} + \mathbb{Q} \log \frac{1}{1 + e^{w^T g(X)}} \right] + \log 4.$$

Proposition.

$$\text{JS}_g(\mathbb{P}, \mathbb{Q}) = 0 \iff \mathbb{P}g(X) = \mathbb{Q}g(X)$$

JS-GAN

$$\widehat{\theta} = \operatorname{argmin}_{\eta \in \mathbb{R}^p} \max_{T \in \mathcal{T}} \left[\frac{1}{n} \sum_{i=1}^n \log T(X_i) + E_\eta \log(1 - T(X)) \right] + \log 4$$

Theorem [GLYZ18]. For a neural network class \mathcal{T} with at least one hidden layer and appropriate regularization, we have

$$\|\widehat{\theta} - \theta\|^2 \lesssim \begin{cases} \frac{p}{n} + \epsilon^2 & \text{(indicator/sigmoid/ramp)} \\ \frac{p \log p}{n} + \epsilon^2 & \text{(ReLUs+sigmoid features)} \end{cases}$$

with high probability uniformly over $\theta \in \mathbb{R}^p, Q$.

JS-GAN: Adaptation to Unknown Covariance

**unknown
covariance?**

$$X_1, \dots, X_n \sim (1 - \epsilon)N(\theta, \Sigma) + \epsilon Q$$

JS-GAN: Adaptation to Unknown Covariance

unknown covariance?

$$X_1, \dots, X_n \sim (1 - \epsilon)N(\theta, \Sigma) + \epsilon Q$$

$$(\hat{\theta}, \hat{\Sigma}) = \operatorname{argmin}_{\eta, \Gamma} \max_{T \in \mathcal{T}} \left[\frac{1}{n} \sum_{i=1}^n \log T(X_i) + \mathbb{E}_{X \sim N(\eta, \Gamma)} \log(1 - T(X)) \right]$$

JS-GAN: Adaptation to Unknown Covariance

unknown covariance?

$$X_1, \dots, X_n \sim (1 - \epsilon)N(\theta, \Sigma) + \epsilon Q$$

$$(\hat{\theta}, \hat{\Sigma}) = \operatorname{argmin}_{\eta, \Gamma} \max_{T \in \mathcal{T}} \left[\frac{1}{n} \sum_{i=1}^n \log T(X_i) + \mathbb{E}_{X \sim N(\eta, \Gamma)} \log(1 - T(X)) \right]$$

no need to change the discriminator class

Generalization

Strong Contamination model:

$X_1, \dots, X_n \stackrel{iid}{\sim} P$ for some P satisfying $\text{TV}(P, E(\theta, \Sigma, H)) \leq \epsilon$

$$(\hat{\theta}, \hat{\Sigma}, \hat{H}) = \underset{\eta \in \mathbb{R}^p, \Gamma \in \mathcal{E}_p(M), H \in \mathcal{H}(M')}{\operatorname{argmin}} \max_{T \in \mathcal{T}} \left[\frac{1}{n} \sum_{i=1}^n S(T(X_i), 1) + \mathbb{E}_{X \sim E(\eta, \Gamma, G)} S(T(X), 0) \right]$$

Theorem [GYZ19]. For a neural network class \mathcal{T} with at least one hidden layer and appropriate regularization, we have

$$\|\hat{\theta} - \theta\|^2 \leq C \left(\frac{p}{n} \vee \epsilon^2 \right),$$

$$\|\hat{\Sigma} - \Sigma\|_{\text{op}}^2 \leq C \left(\frac{p}{n} \vee \epsilon^2 \right),$$

Robust Learning of Gaussian Distributions

Q	n	p	ϵ	TV-GAN	JS-GAN	Dimension Halving	Iterative Filtering
$N(0.5 * 1_p, I_p)$	50,000	100	.2	0.0953 (0.0064)	0.1144 (0.0154)	0.3247 (0.0058)	0.1472 (0.0071)
$N(0.5 * 1_p, I_p)$	5,000	100	.2	0.1941 (0.0173)	0.2182 (0.0527)	0.3568 (0.0197)	0.2285 (0.0103)
$N(0.5 * 1_p, I_p)$	50,000	200	.2	0.1108 (0.0093)	0.1573 (0.0815)	0.3251 (0.0078)	0.1525 (0.0045)
$N(0.5 * 1_p, I_p)$	50,000	100	.05	0.0913 (0.0527)	0.1390 (0.0050)	0.0814 (0.0056)	0.0530 (0.0052)
$N(5 * 1_p, I_p)$	50,000	100	.2	2.7721 (0.1285)	0.0534 (0.0041)	0.3229 (0.0087)	0.1471 (0.0059)
$N(0.5 * 1_p, \Sigma)$	50,000	100	.2	0.1189 (0.0195)	0.1148 (0.0234)	0.3241 (0.0088)	0.1426 (0.0113)
Cauchy($0.5 * 1_p$)	50,000	100	.2	0.0738 (0.0053)	0.0525 (0.0029)	0.1045 (0.0071)	0.0633 (0.0042)

Table: Comparison of various robust mean estimation methods. The smallest error of each case is highlighted in bold.

- *Dimension Halving*: [Lai et al.'16]
<https://github.com/kal2000/AgnosticMeanAndCovarianceCode>.
- *Iterative Filtering*: [Diakonikolas et al.'17]
<https://github.com/hoonose/robust-filter>.

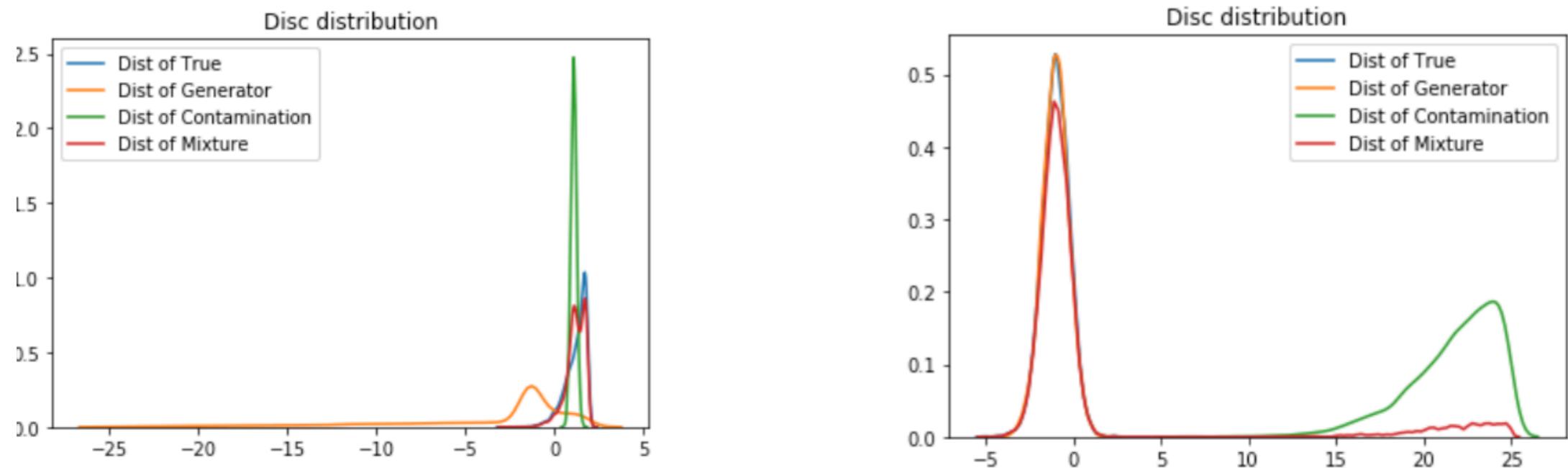
Robust Learning of Cauchy Distributions

Table 4: Comparison of various methods of robust location estimation under Cauchy distributions. Samples are drawn from $(1 - \epsilon)\text{Cauchy}(0_p, I_p) + \epsilon Q$ with $\epsilon = 0.2, p = 50$ and various choices of Q . Sample size: 50,000. Discriminator net structure: 50-50-25-1. Generator $g_\omega(\xi)$ structure: 48-48-32-24-12-1 with absolute value activation function in the output layer.

Contamination Q	JS-GAN (G_1)	JS-GAN (G_2)	Dimension Halving	Iterative Filtering
Cauchy($1.5 * 1_p, I_p$)	0.0664 (0.0065)	0.0743 (0.0103)	0.3529 (0.0543)	0.1244 (0.0114)
Cauchy($5.0 * 1_p, I_p$)	0.0480 (0.0058)	0.0540 (0.0064)	0.4855 (0.0616)	0.1687 (0.0310)
Cauchy($1.5 * 1_p, 5 * I_p$)	0.0754 (0.0135)	0.0742 (0.0111)	0.3726 (0.0530)	0.1220 (0.0112)
Normal($1.5 * 1_p, 5 * I_p$)	0.0702 (0.0064)	0.0713 (0.0088)	0.3915 (0.0232)	0.1048 (0.0288))

- *Dimension Halving:* [Lai et al.'16]
<https://github.com/kal2000/AgnosticMeanAndCovarianceCode>.
- *Iterative Filtering:* [Diakonikolas et al.'17]
<https://github.com/hoonose/robust-filter>.

Discriminator identifies outliers



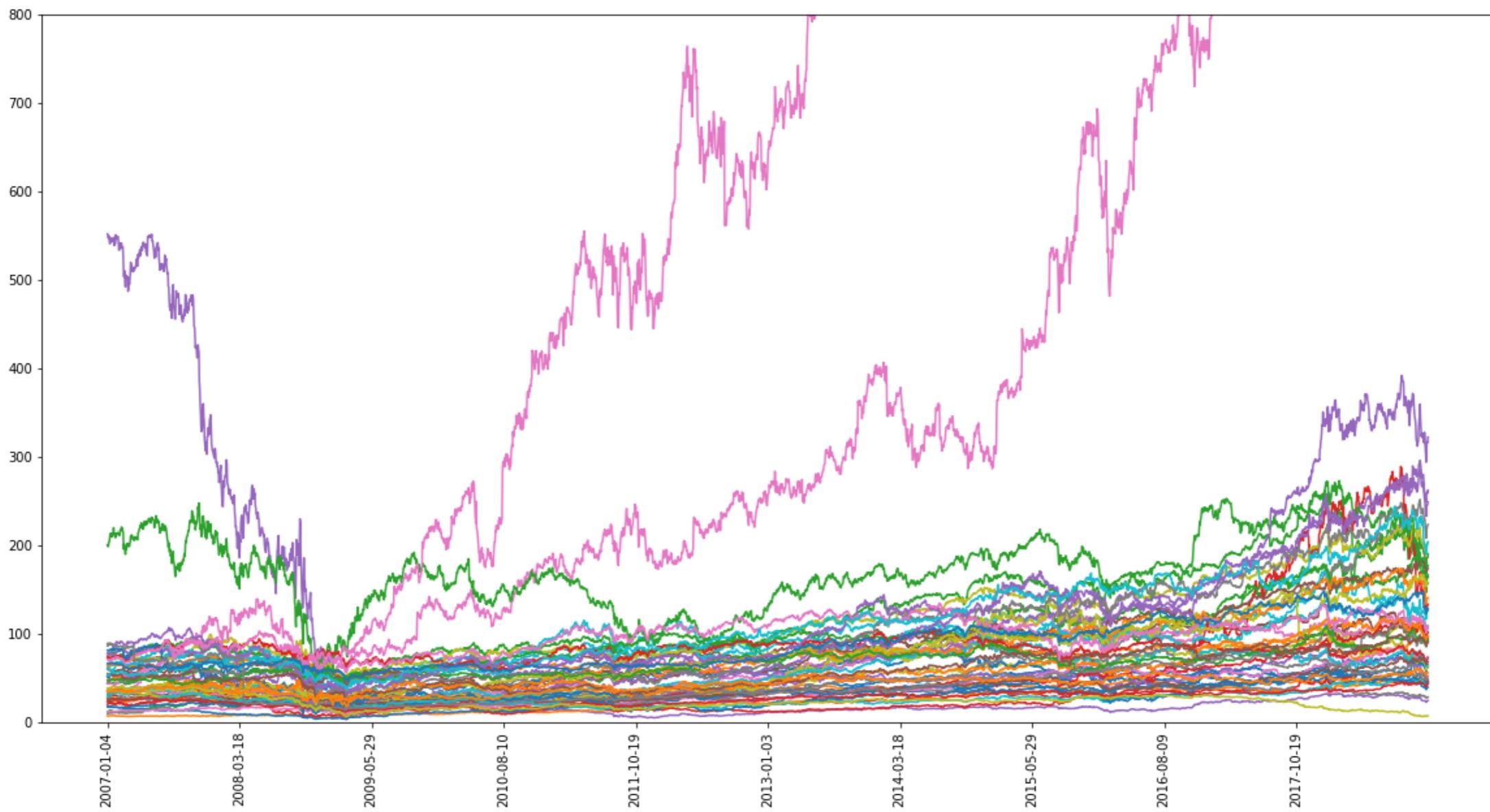
$$(1 - \epsilon)N(0_p, I_p) + \epsilon Q$$

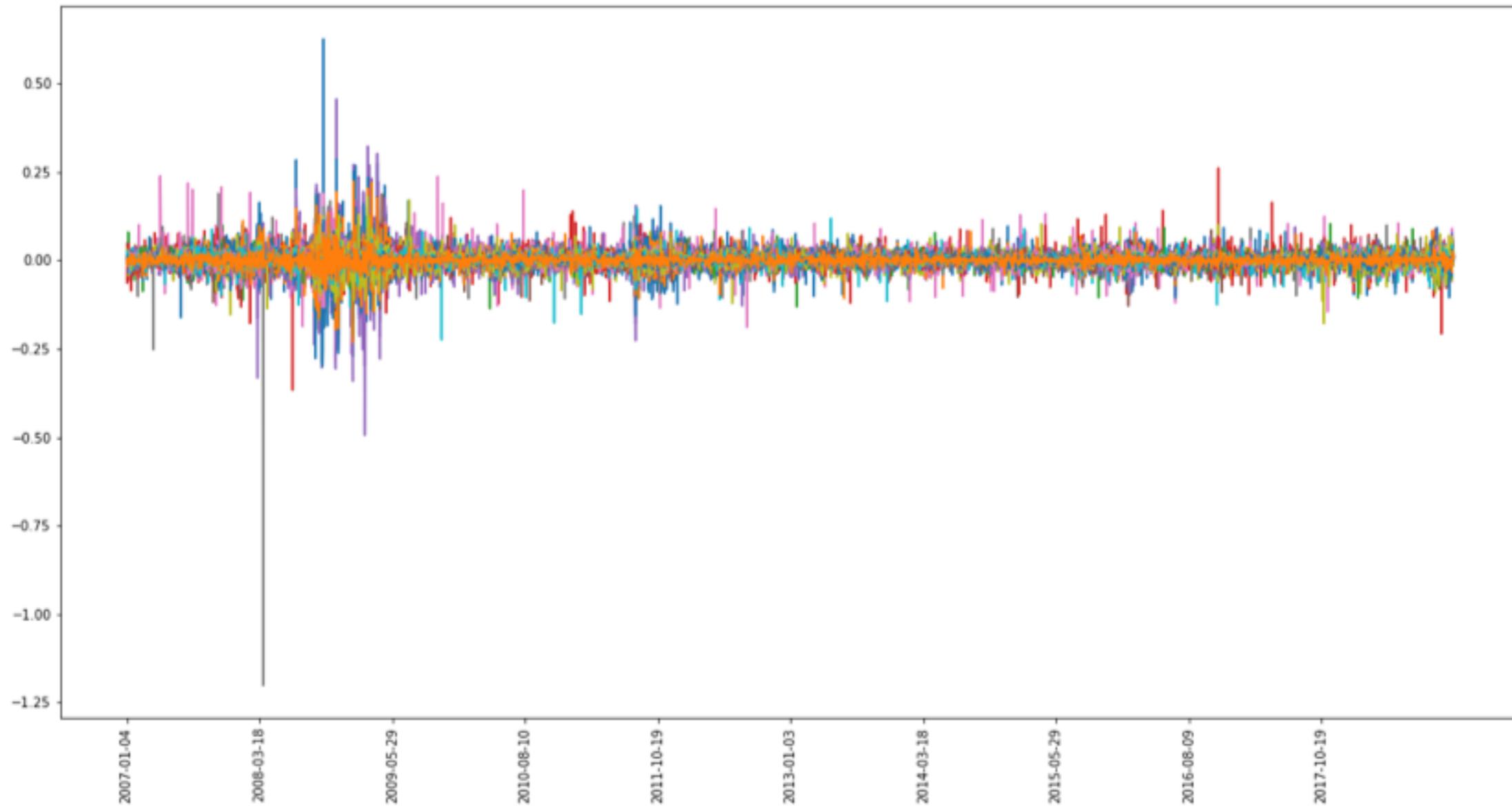
$$N(5 * 1_p, I_p)$$

- Discriminator helps identify outliers or contaminated samples
- Generator fits uncontaminated portion of true samples

Application: Price of 50 stocks from 2007/01 to 2018/12

Corps are selected by ranking in market capitalization





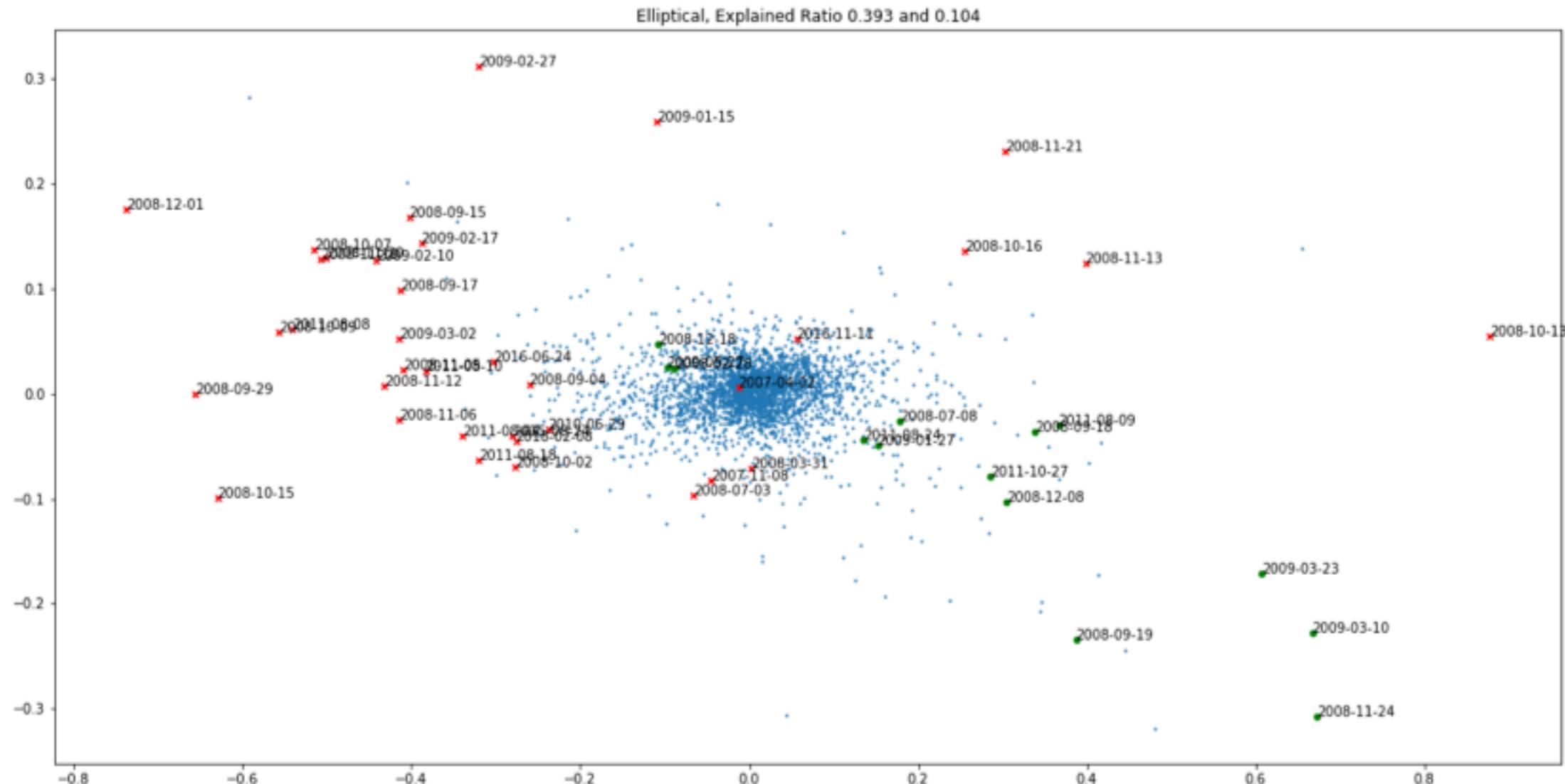
Log-return. $y[i] = \log(\text{price}_{\{i+1\}}/\text{price}_{\{i\}})$

Fit data by Elliptica-GAN.

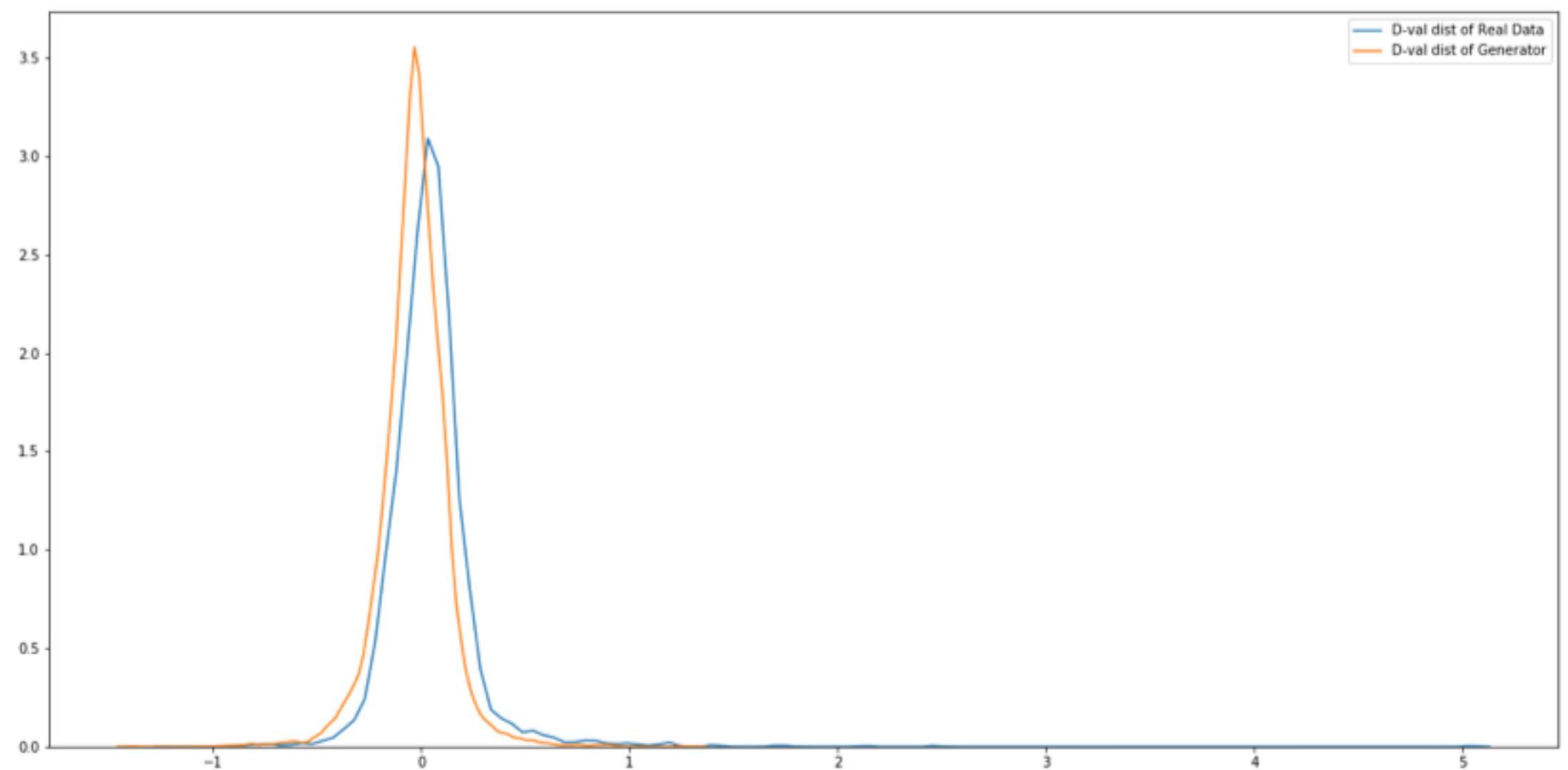
Apply SVD on scatter.

Dimension reduction on R^2 .

outlier **x and **o** are selected from Discriminator value distribution.**

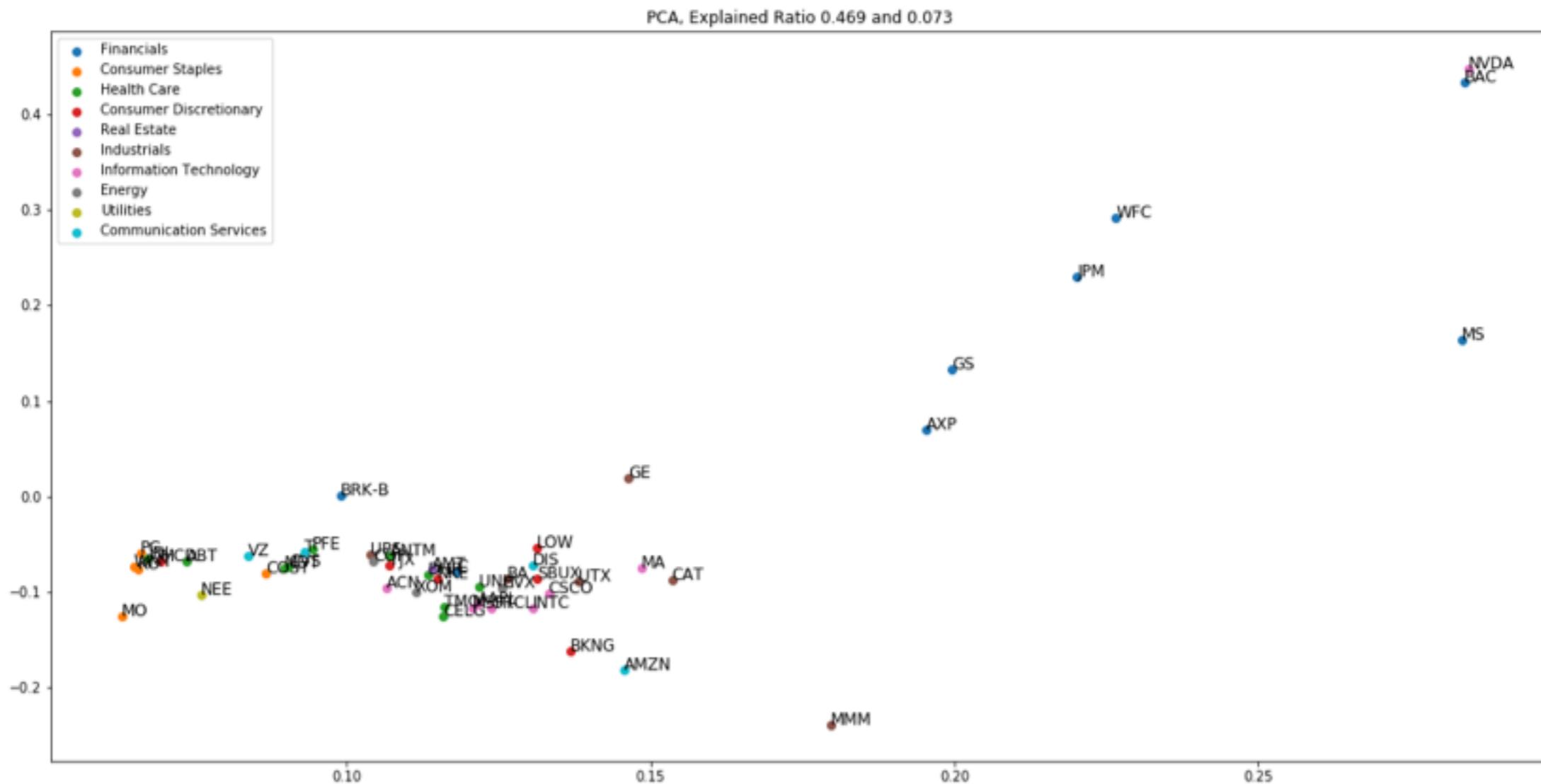


Discriminator value distribution from (Elliptical) Generator and real samples. Outliers are chosen from samples larger/ lower than a chosen percentile of Generator distribution



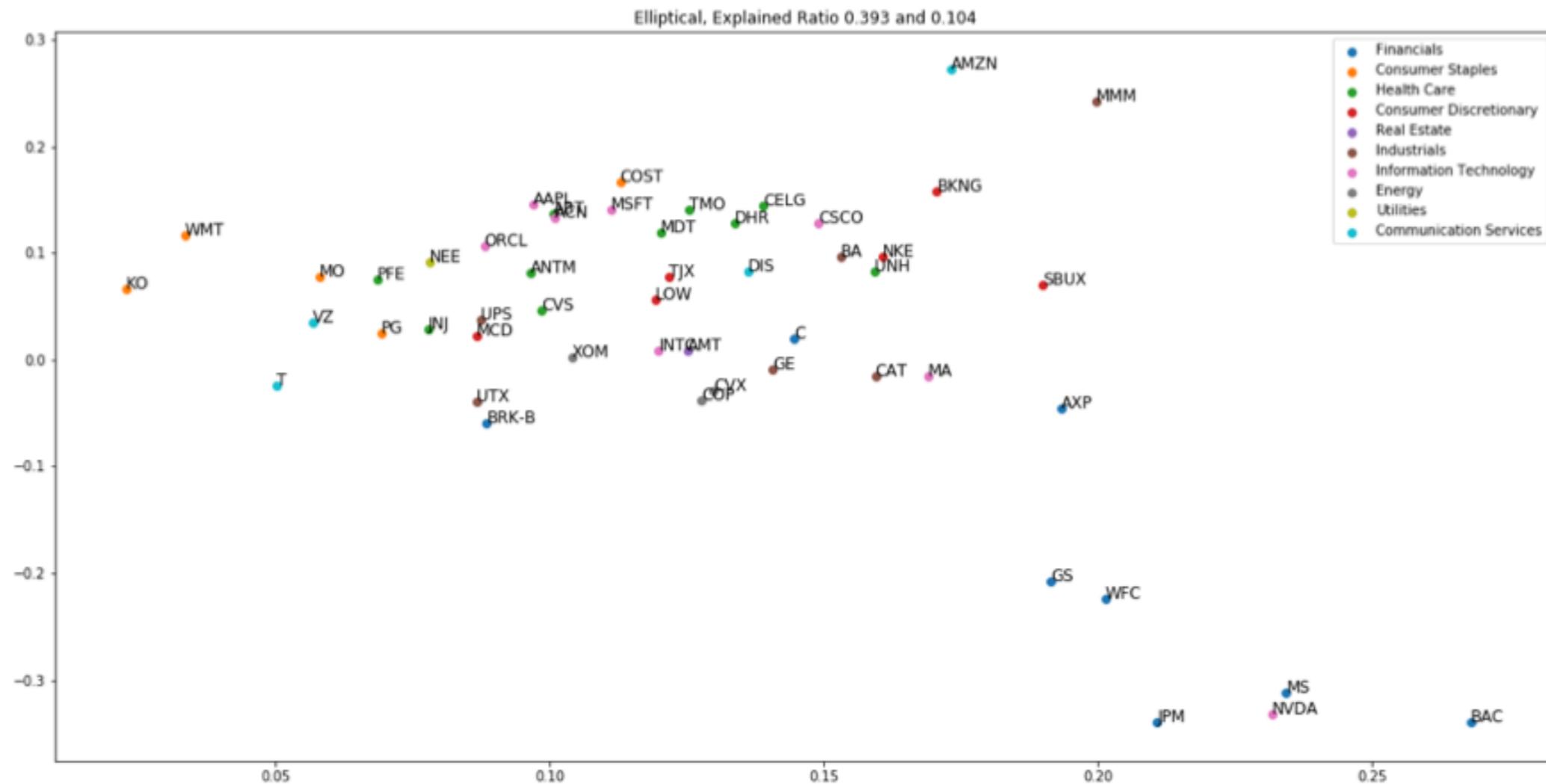
Loading of PCA.

First two direction are dominated by few corps → not robust



Loading of Elliptical Scatter:

Comparing with PCA, it's more robust in the sense that it does not totally dominate by Financial company (JPM, GS)



Reference

- Gao, Liu, Yao, Zhu, Robust Estimation and Generative Adversarial Networks, *ICLR 2019*,
<https://arxiv.org/abs/1810.02030>
- Gao, Yao, Zhu, Generative Adversarial Networks for Robust Scatter Estimation: A Proper Scoring Rule Perspective, <https://arxiv.org/abs/1903.01944>

Thank You

