Lecture 15: Robust Statistics

Lecturer: Sam Hopkins **Date:** April 2, 2026
Scribes: Raymond Bahng, Maureen Zhang

1. Learning from Corrupted Data

A central question in statistics is how to learn about an unknown distribution D from samples. The classical setup assumes we receive n independent, identically distributed samples $X_1^*, \ldots, X_n^* \sim D$ and aim to estimate some property of D, such as its mean. *Robust statistics* asks: what can we still learn when an adversary is allowed to corrupt some of those samples?

This question has been studied since the 1960s, but for decades the field lacked efficient algorithms for high-dimensional problems. The last decade has seen a breakthrough, as we now have efficient, provably correct algorithms for tasks such as mean estimation, learning a Gaussian in total variation, and clustering, all under adversarial contamination.

1.1. Strong Contamination Model

Definition 1.1 (Strong Contamination)**.** Let D be an unknown distribution and $\epsilon \in [0, 1/2)$. The *strong contamination* (also known as Huber contamination) process proceeds as follows:

1. Draw $X_1^*, \ldots, X_n^* \sim_{\text{iid}} D$.
2. Hand the entire list to a computationally unbounded, adaptive adversary who knows the samples and the algorithm we intend to run.
3. The adversary may replace any ϵn of the points with arbitrary values of its choosing.
4. The resulting, possibly corrupted, list $X_1, \ldots, X_n$ is given to the learning algorithm.

We say an algorithm *ϵ-robustly* solves a learning task if it succeeds on every input produced by this process for the given ϵ.

Remark 1.1 (Adaptive vs. oblivious adversary). In the strong contamination model, the adversary sees all samples before deciding which to corrupt. A weaker, "oblivious" model lets the adversary corrupt each sample independently after seeing only that sample. These two models are not equivalent in general; however, for many tasks they are equivalent up to polynomial losses in sample complexity, via a randomized subsampling argument (see Section 2).

Remark 1.2 (Security framing). From the perspective of security, the strong contamination model is a natural model of a *data-poisoning* attack: an adversary who can inspect the training dataset and inject malicious points.

1.2. Model Misspecification

For statisticians, the contamination model has a compelling alternative interpretation that does not involve adversaries at all.

Definition 1.2 (Model Misspecification)**.** Suppose the statistician assumes the data-generating distribution D belongs to some class $\mathcal{D}$. In reality, D lies outside $\mathcal{D}$ but is close to it:

$$\mathrm{TV}(D,\,\mathcal{D}) \;:=\; \min_{D_0 \in \mathcal{D}} \mathrm{TV}(D, D_0) \;\le\; \epsilon.$$

The learning algorithm receives i.i.d. samples from D but is designed under the assumption that $D \in \mathcal{D}$.

Here TV denotes total variation distance, defined below.

Definition 1.3 (Total Variation Distance)**.** For distributions D_0, D_1 on a measurable space Ω,

$$\mathrm{TV}(D_0, D_1) \;=\; \sup_{S \subseteq \Omega} \left| D_0(S) - D_1(S) \right| \;=\; \min_{\text{couplings } D \text{ of } D_0, D_1} \Pr_{(X,Y)\sim D}[X \neq Y]\,,$$

where the minimum is over all joint distributions (X, Y) with marginals D_0 and D_1, respectively. Such a joint distribution is called a *coupling*.

Definition 1.4 (Coupling)**.** A *coupling* of distributions D_0 and D_1 on a measurable space Ω is a joint distribution D on $\Omega \times \Omega$ such that if $(X, Y) \sim D$, then $X \sim D_0$ and $Y \sim D_1$.

Example 1.3 (Illustrative coupling example)**.** Let $D_0 = \mathrm{Bernoulli}(1/2)$ and $D_1 = \mathrm{Bernoulli}(1/2+\epsilon)$ for some small $\epsilon > 0$. The most natural coupling draws $X \sim D_0$ and sets $Y = X$ whenever possible, that is, X and Y agree unless forced to differ. Since D_1 puts slightly more weight on 1 than D_0 does, the two variables must disagree precisely on this excess, which happens with probability ϵ. So under this coupling, $\Pr[X \neq Y] = \epsilon$. Since $\mathrm{TV}(D_0, D_1) = \epsilon$, this coupling is optimal as it minimizes the probability of disagreement. Such an optimal coupling always exists and is called the *maximal coupling*.

The following claim shows that an algorithm robust to strong contamination is also robust to model misspecification.

Claim 1.4. *Suppose algorithm $\mathcal{A}$ is ϵ-robust under the strong contamination model for distributions in $\mathcal{D}$. Then $\mathcal{A}$ also succeeds on i.i.d. samples from any D with $\mathrm{TV}(D, \mathcal{D}) \le \epsilon$.*

Proof sketch. Let $D_0 \in \mathcal{D}$ be a distribution achieving $\mathrm{TV}(D_0, D) \le \epsilon$, and let (X, Y) be the maximal coupling of D_0 and D. The algorithm receives i.i.d. samples $Y_1, \ldots, Y_n \sim D$. We construct an adversary for the strong contamination model as follows: draw $X_i^* \sim D_0$ and couple each X_i^* with Y_i using the marginal coupling (X_i^*, Y_i). With probability $1 - \mathrm{TV}(D_0, D) \ge 1 - \epsilon$, we have $X_i^* = Y_i$ and the adversary leaves the sample unchanged. With the remaining probability at most ϵ, the adversary replaces X_i^* with Y_i. The number of indices where $X_i^* \neq Y_i$ is $\mathrm{Binomial}(n, \epsilon)$. By a Chernoff bound, it exceeds $2\epsilon n$ only with probability $e^{-O(\epsilon n)}$, so with high probability the adversary modifies at most $2\epsilon n$ samples, giving a valid strongly contaminated sample from D_0. Because the constant factor of 2 is absorbed into the $O(\cdot)$ in the error bound, $\mathcal{A}$ succeeds with high probability. □

Model misspecification was the original statistical motivation for studying this framework, so it is very helpful that designing algorithms for the strong contamination model automatically yields guarantees for the misspecification as well.

2. Comparing Contamination Models

Several contamination models appear in the literature, so it is useful to understand how they relate.

1. **Strong (adaptive) contamination.** The adversary sees all n samples jointly before deciding which ϵn to replace. This is the model of Definition 1.1.
2. **Oblivious contamination.** The adversary decides, independently for each sample, whether to corrupt it (probability ϵ) or leave it unchanged. This is a weaker model.
3. **Additive mixture model.** The observed samples are drawn i.i.d. from the mixture $(1-\epsilon)D_0 + \epsilon D_1$ for an arbitrary distribution D_1. This is even weaker as the adversary cannot coordinate its corruptions across samples.

The relationships among these models are subtle, and the full picture was not covered in lecture. Broadly, strong contamination is the hardest model for the learner and the additive mixture the easiest, though the precise implications between them depend on details not covered in lecture.

One informal technique for relating models is *random subsampling*: if one takes a random subsample of strongly contaminated data, the adversary's coordinated choices become harder to exploit. Concretely, suppose we subsample at rate p. An adaptive adversary who corrupts a δ/p fraction of points before subsampling corrupts only a δ fraction of the subsampled points on average, and the choice of which points to corrupt was made before the subsampling randomness was revealed. This suggests that algorithms designed for weaker adversaries may be applicable in stronger settings, at the cost of needing somewhat more samples, though making this precise requires care.

3. Fundamental Limits on Robust Learning Feasibility

Before designing algorithms, we examine for which values of ϵ can we hope to learn anything?

3.1. The $\epsilon < 1/2$ Case

Suppose the adversary corrupts a 51% majority of the samples. Can we still estimate the mean of D?

Claim 3.1. *When $\epsilon \geq 1/2$, no algorithm can robustly estimate the mean of an arbitrary distribution.*

Proof. Consider two distinct distributions D_a and D_b with well-separated means. An adversary who controls half the samples can replace half the D_a-samples with draws from D_b and vice versa. The algorithm then receives a mixture of D_a and D_b samples in either case, and cannot determine which distribution is the true distribution. □

3.2. List-Decodable Learning

Although $\epsilon \geq 1/2$ precludes unique identification of the mean, it does not preclude partial recovery.

Definition 3.1 (List-Decodable Mean Estimation)**.** An algorithm (k, ϵ)-*list-decodably* estimates the mean of D if, given contaminated samples with ϵn corruptions, it outputs a list of k candidate means such that at least one candidate is close to the true mean μ.

This notion is directly motivated by list-decoding in coding theory. When $\epsilon < 1$, a list of size $O(1/(1-\epsilon))$ suffices. List-decoding is also intimately connected to clustering. If the clean data comes from a mixture of k distributions with equal weights, list-decodable learning with $\epsilon = 1 - 1/k$ recovers a list of size $O(k)$ containing approximations to the means of all k mixture components [1]. Accurately assigning points to their respective mixture components additionally requires sufficient mean separation.

Remark 3.2. Even a single clean sample, which alone is too noisy to estimate the mean, can be used to select the correct element from the list produced by a list-decodable algorithm. This has applications to settings with multiple, partially trustworthy data sources.

For the remainder of these notes we focus on the regime $\epsilon < 1/2$, treating ϵ as a small constant (e.g., $\epsilon = 0.01$).

4. Tradeoffs Between Distributional Assumptions and Robust Learnability

A central theme in robust statistics is the following:

Assumptions vs. Learnability

There is a fundamental tradeoff between the strength of assumptions placed on the clean distribution D and the quality of the guarantees achievable by robust learning algorithms.

We illustrate this tradeoff through two extreme scenarios for robust mean estimation.

Scenario 1: No assumptions on D. Suppose D is an arbitrary distribution on $\mathbb{R}$. Even with $\epsilon = 1\%$, we cannot robustly estimate the mean. To see why, consider two alternative explanations for the observed data:

- **Possibility A:** The true distribution is a Gaussian centered far to the left, and the adversary has corrupted 1% of the samples by replacing them with points far to the right.
- **Possibility B:** The true distribution is a Gaussian centered far to the right, and the adversary has placed the 1% corruptions to the left.

If the class $\mathcal{D}$ contains both distributions as possible true distributions, then the resulting sample is statistically indistinguishable under the two scenarios, so no algorithm can determine the true mean.

Scenario 2: Dirac delta distributions (zero variance). Now suppose $\mathcal{D}$ consists entirely of point masses (Dirac deltas), so the clean data is a single repeated value. Robust learning is trivial: with ϵ corruptions, at least $(1-\epsilon)n$ samples equal the true mean. We output the majority value. This extreme assumption makes robustness easy but is unrealistic.

In practice, we seek assumptions that are realistic (e.g., sub-Gaussian tails or bounded covariance) and that still permit efficient, optimal robust estimation. The next section quantifies this tradeoff for two natural assumptions classes.

5. Robust Mean Estimation: Two Key Theorems

We now state the two foundational results in algorithmic robust statistics, which characterize the optimal error achievable under different assumptions on the moments of D.

Notation. For symmetric matrices $A, B \in \mathbb{R}^{d\times d}$, we write $A \preceq B$ to mean that $B - A$ is positive semidefinite, or that $v^\top (B-A)v \geq 0$ for all $v \in \mathbb{R}^d$. In particular, $\Sigma \preceq \sigma^2 I$ means that the covariance matrix Σ has all eigenvalues at most σ^2, or equivalently that $\mathrm{Var}[\langle v, X\rangle] \leq \sigma^2$ for every unit vector $v \in \mathbb{R}^d$.

Theorem 5.1 (Robust Mean Estimation for Gaussians)**.** *Let $\mathcal{D}$ be the class of all Gaussian distributions $N(\mu, \Sigma)$ on $\mathbb{R}^d$ with $\Sigma \preceq I$. There exists an efficient algorithm that, given sufficiently many strongly ϵ-contaminated samples from some $N(\mu, \Sigma) \in \mathcal{D}$, outputs $\hat{\mu}$ satisfying*

$$\|\mu - \hat{\mu}\| \leq O(\epsilon).$$

More generally, if $\Sigma \preceq \sigma^2 I$, the error is $O(\epsilon\sigma)$.

Theorem 5.2 (Robust Mean Estimation with Bounded Second Moments)**.** *Let $\mathcal{D}$ be the class of all distributions on $\mathbb{R}^d$ with covariance $\Sigma \preceq \sigma^2 I$. There exists an efficient algorithm that, given sufficiently many strongly ϵ-contaminated samples, outputs $\hat{\mu}$ satisfying*

$$\|\mu - \hat{\mu}\| \leq O(\sqrt{\epsilon}) \cdot \sigma.$$

Both bounds are tight, such that as $\epsilon \to 0$, the error vanishes. As $\sigma \to 0$, the error vanishes. The $O(\epsilon)$ guarantee for Gaussians is stronger than the $O(\sqrt{\epsilon})$ bound for bounded-covariance distributions, reflecting the stronger assumption.

The following discussion provides a high-level intuition for the optimality of these rates. The lecture sketched these lower bounds through examples rather than full formal derivations.

5.1. Why These Rates Are Optimal

We can verify the rates cannot be improved by examining distributions that are close in total variation but have means far apart.

Lower bound for Theorem 5.1 ($O(\epsilon)$). Consider the two Gaussians $N(0, I)$ and $N(c\,e_1, I)$ where $e_1 = (1, 0, \ldots, 0) \in \mathbb{R}^d$ and $c > 0$ is small. Their total variation distance satisfies $\mathrm{TV}(N(0, I),\ N(c\,e_1, I)) = O(c)$. An adversary can "simulate" either distribution from the other by corrupting an $O(c)$ fraction of samples. Choosing $c = \Theta(\epsilon)$ makes the two scenarios indistinguishable under ϵ-corruption, so no algorithm can achieve error $o(\epsilon)$.

Lower bound for Theorem 5.2 ($O(\sqrt{\epsilon})$). Consider the distribution $D_0 = (1 - \epsilon)\,\delta_0 + \epsilon\,\delta_{1/\sqrt{\epsilon}}$ (a point mass at 0 and a point mass at $1/\sqrt{\epsilon}$, with weights $1 - \epsilon$ and ϵ). This distribution has mean $\sqrt{\epsilon}$ and variance at most 1. Its total variation distance from the point mass δ_0 is exactly ϵ. An adversary can make the two scenarios of true mean 0 vs. true mean $\sqrt{\epsilon}$ indistinguishable by corrupting ϵ fraction of samples, showing that error $o(\sqrt{\epsilon})$ is impossible under only bounded second moment assumptions.

6. The Fundamental Lemma of Robust Mean Estimation

The theoretical backbone of both Theorem 5.1 and Theorem 5.2 is the following lemma, which bounds how much two random variables' means can differ given that they are close in TV and have bounded variance.

Lemma 6.1 (Fundamental Lemma of RME)**.** *Let Z and Y be real-valued random variables with*

(a) $\mathrm{TV}(Z, Y) \le \epsilon$, *and*

(b) $\mathrm{Var}[Z] \le \sigma^2$ *and* $\mathrm{Var}[Y] \le \sigma^2$.

Then $|\mathbb{E}[Z] - \mathbb{E}[Y]| \le O(\sqrt{\epsilon}) \cdot \sigma$.

Proof sketch. Let (Z, Y) be a coupling achieving $\Pr[Z \neq Y] \le \epsilon$. Then

$$|\mathbb{E}[Z] - \mathbb{E}[Y]| = |\mathbb{E}[Z - Y]| = |\mathbb{E}[(Z - Y) \cdot \mathbf{1}[Z \neq Y]]|.$$

Applying Cauchy–Schwarz,

$$|\mathbb{E}[(Z - Y) \cdot \mathbf{1}[Z \neq Y]]| \ \le\ \sqrt{\mathbb{E}[(Z - Y)^2]} \cdot \sqrt{\Pr[Z \neq Y]}.$$

Since the disagreement event has probability at most ϵ and both random variables have bounded variance, this yields a bound of order $O(\sqrt{\epsilon}) \cdot \sigma$. □

Remark 6.2. The improved $O(\epsilon)$ guarantee for Gaussians (Theorem 5.1) comes from the fact that Gaussians have much more tightly controlled tails than a general bounded-covariance distribution. This extra structure means that two Gaussians whose means differ by more than $O(\epsilon)$ must differ in total variation by more than ϵ, so the adversary has less room to hide the corruption.

6.1. Applying the Fundamental Lemma

We now show how Lemma 6.1 implies the $O(\sqrt{\epsilon})$ guarantee of Theorem 5.2. Suppose we start with clean samples $X_1^*, \ldots, X_n^* \sim D$ which the adversary corrupts to $X_1, \ldots, X_n$. An algorithm attempts to *reconstruct* uncorrupted-looking samples $Y_1, \ldots, Y_n$ and outputs their empirical mean.

Let $\mu(\cdot)$ and $\Sigma(\cdot)$ denote the empirical mean and covariance of a list of vectors. Define:

- $\mathcal{Z}$ = the uniform distribution over $\{X_1^*, \ldots, X_n^*\}$ (clean),
- $\mathcal{Y}$ = the uniform distribution over $\{Y_1, \ldots, Y_n\}$ (reconstructed).

We need $\mathcal{Y}$ to satisfy two properties:

(a) $Y_i = X_i$ for at least $(1-\epsilon)n$ indices i, and

(b) $\Sigma(\mathcal{Y}) \preceq I$.

Note that property (a) implies $\mathrm{TV}(\mathcal{Z}, \mathcal{Y}) \le 2\epsilon$, as the adversary changed at most ϵn points from the clean samples to produce the corrupted samples, and we change at most ϵn more to produce $\mathcal{Y}$, so the triangle inequality for TV distance gives the factor of 2. Property (b) mirrors the bounded covariance assumption we placed on the clean distribution D, so the true clean samples $X_1^*, \ldots, X_n^*$ would satisfy it too.[1] Given both properties, we can invoke Lemma 6.1 with $\sigma = 1$ and contamination fraction 2ϵ, which gives $\|\mu(\mathcal{Z}) - \mu(\mathcal{Y})\| \le O(\sqrt{\epsilon})$.

7. Algorithms for Robust Mean Estimation

7.1. An Inefficient Algorithm

We first describe an *information-theoretically* optimal algorithm, ignoring computational efficiency, that achieves the bound of Theorem 5.2.

Algorithm 1: Inefficient Robust Mean Estimator

Input: Corrupted samples $X_1, \ldots, X_n \in \mathbb{R}^d$ and corruption fraction ϵ
Output: Robust mean estimate $\hat{\mu}$

Find $Y_1, \ldots, Y_n \in \mathbb{R}^d$ satisfying:;
  (a) $X_i = Y_i$ for at least $(1-\epsilon)n$ indices i;
  (b) Empirical covariance $\frac{1}{n}\sum_{i=1}^n (Y_i - \mu(Y))(Y_i - \mu(Y))^\top \preceq I$;
return $\mu(Y) = \frac{1}{n}\sum_{i=1}^n Y_i$

Correctness. A valid choice of $Y_1, \ldots, Y_n$ is $Y_i = X_i^*$ (the true clean samples), since the true distribution satisfies bounded covariance by assumption.[1] By the analysis of Section 6.1, any $Y_1, \ldots, Y_n$ satisfying (a) and (b) has $\mu(Y)$ within $O(\sqrt{\epsilon})$ of the true mean.

[1] Strictly speaking, the true distribution having covariance $\Sigma \preceq I$ does not guarantee that the empirical covariance of n finite samples satisfies the same bound. By concentration inequalities (e.g., Chebyshev's inequality applied to the second moment), the empirical covariance exceeds I only with small probability for large n. We neglect this failure probability throughout. The algorithm therefore succeeds with high probability rather than with probability 1.

Inefficiency. The search over all subsets of size $(1-\epsilon)n$ is exponential in n. We now describe an efficient algorithm.

7.2. The Filter Algorithm

The efficient approach, sometimes called the *spectral filter* or simply the *filter* algorithm, replaces the combinatorial search with an iterative, PCA-based procedure. Rather than moving samples, we *remove* samples. A sample can always be moved to the empirical mean without increasing the empirical covariance, so removal is the key operation.

Algorithm 2: Filter Algorithm for Robust Mean Estimation

Input: Contaminated samples $X_1, \ldots, X_n \in \mathbb{R}^d$ and corruption fraction ϵ
Output: Robust mean estimate $\hat{\mu}$

repeat
  Compute empirical mean $\mu = \mu(X)$ and covariance $\Sigma = \Sigma(X)$;
  if $\Sigma \preceq I$ *(all eigenvalues* ≤ 1*)* **then**
    halt; **return** μ;
  else
    Let v be the direction of highest variance in Σ;
    Remove all X_i for which $|\langle X_i - \mu, v\rangle|$ is large
  end
until *halted*;

At each iteration, the threshold for removal is chosen so that $O(\epsilon n)$ points are removed in total across all iterations.

Intuition. The key insight is that the adversary can corrupt the empirical mean only by injecting samples in a concentrated direction. Specifically, if an adversary injects ϵn samples all displaced by a vector v of norm $\|v\|$ from the true mean, then

$$\|\hat{\mu}_{\text{corrupted}} - \mu\| = \|(1-\epsilon)\mu + \epsilon(\mu + v) - \mu\| = \epsilon\|v\|.$$

In d dimensions, points drawn from $N(\mu, I)$ satisfy $\|X_i^* - \mu\| \approx \sqrt{d}$, so naively $\|v\| \leq \sqrt{d}$, yielding error $\epsilon\sqrt{d}$, which grows with the dimension. The filter avoids this by detecting the high-variance direction v along which corrupted samples concentrate, then removing the outliers in that direction.

Additional intuition on why the filter works:

1. *Good samples have bounded variance in every direction.* Under the clean distribution, the empirical covariance should have all eigenvalues at most 1. So if we find an eigenvalue exceeding 1, we know corrupted points must be responsible for the excess variance in that direction.[1]

2. *Corrupted samples are detectable.* Corrupted samples must contribute disproportionately along the high-variance direction v. Otherwise, the adversary could not have inflated that eigenvalue.

3. *Progress per iteration.* Each iteration removes a batch of samples, at least as many bad as good (up to constant factors), and the number of good samples that can be removed is at most ϵn, so the algorithm terminates in $O(1/\epsilon)$ rounds.

The filter algorithm runs in $\mathrm{poly}(n, d)$ time and achieves the $O(\sqrt{\epsilon})$ error bound of Theorem 5.2.

Remark 7.1 (Generality of the algorithmic template). The filter is a special case of a general algorithmic template:

1. Start with the corrupted samples $X_1, \ldots, X_n$.
2. Check a list of "good-sample conditions" (e.g., covariance bounded, higher moments bounded, etc.).
3. If all conditions hold, halt and output the empirical mean.
4. Otherwise, find the violated condition; use it to identify a "certificate of corruption" pointing to specific samples responsible for the violation, and remove those samples.
5. Iterate.

The lecture emphasized that this template extends beyond mean estimation and has analogues in linear regression, clustering, and mixture learning. The iterative procedure can often be replaced by a single convex program (e.g., via sum-of-squares optimization) whose dual solution simultaneously certifies goodness and identifies bad samples.

8. Connections to Modern Machine Learning and Data Poisoning

8.1. Why Naive Outlier Removal Fails in High Dimensions

A first instinct for robust estimation might be to identify and remove samples that appear to be outliers, such as those with unusually large distance to the empirical mean. In d dimensions, samples from $N(\mu, I)$ have norm $\|X_i - \mu\| \approx \sqrt{d}$. A naive algorithm might remove all samples outside a ball of radius $\sqrt{d}$; however, an adversary can defeat this by placing all corrupted samples *inside* the ball, near the boundary, shifted in a single direction. Such samples pass the norm test yet collectively shift the empirical mean by $\epsilon\sqrt{d}$: a dimension-dependent error. The filter algorithm detects this attack precisely because it looks for high-variance directions rather than high-norm individual samples.

8.2. Data Poisoning Defenses in Neural Networks

The filter idea has been applied to defend against data-poisoning attacks in neural networks. The key observation is that in a well-trained model, the internal representations (embeddings) of inputs in a late layer reflect their semantic content. If poisoned examples have a different semantic meaning from the clean data, their embeddings will cluster in a direction distinct from the clean embeddings, which is precisely the situation the filter can detect.

This defense, known as *spectral signatures* [2], identifies poisoned training examples by computing the top singular vectors of the matrix of per-example embeddings and flagging samples with anomalous projections onto those vectors.

Remark 8.1. The spectral-signature defense fails against more sophisticated attacks that corrupt the *embedding* itself rather than only the final classification layer. If an adversary can inject samples that cause the model to learn a distorted embedding, then clean and poisoned samples are no longer geometrically distinguishable, and the filter provides no guarantee. Designing robust defenses against such attacks is an open problem.

8.3. Open Questions

1. **Robustly learning generative models.** Can we learn, say, a diffusion model robustly (i.e., with provable guarantees under strong contamination)? Learning a Gaussian robustly is well understood, but diffusion models correspond to much richer distribution classes.
2. **Robust score matching.** Score-matching underlies denoising diffusion probabilistic models. If we could perform *robust score matching* under adversarial contamination, we might obtain robust generative models. Whether robust score matching is possible, and whether it implies robust generation, are open questions.
3. **Understanding the metric of corruption.** Current defenses depend on having a well-understood geometric structure (e.g., Euclidean distance in embedding space) to separate corrupted from clean data. A general theory of what the right metric for detecting corruptions is across different data modalities is lacking.

9. Summary

The main takeaways from the lecture are:

- Strong contamination models adversarial corruption by allowing an adversary to replace an ϵ-fraction of the sample set.
- The same model can be interpreted as robustness to misspecification, via total variation distance and couplings.
- If $\epsilon \geq 1/2$, unique recovery is impossible in general, motivating list-decodable learning.
- Robust mean estimation exhibits a clear tradeoff between assumptions and guarantees:
  - Gaussian with bounded covariance gives error $O(\epsilon\sigma)$.
  - Arbitrary bounded-covariance distributions give error $O(\sqrt{\epsilon}\,\sigma)$.
- High-dimensional robust estimation is hard because naive pointwise outlier removal suffers dimension-dependent error.
- A central algorithmic paradigm is to reconstruct or filter the data until the remaining sample set satisfies the structural properties expected of clean data.

References

[1] Moses Charikar, Jacob Steinhardt, and Gregory Valiant. Learning from untrusted data. In *Proceedings of the 49th Annual ACM STOC*, pages 47–60, 2017.

[2] Brandon Tran, Jerry Li, and Aleksander Madry. Spectral signatures in backdoor attacks. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 31, 2018.