Lecture 9: LAD, Linear Programming & Algorithmic Solution

Econometrics 2 — *From the LAD Estimator to the LP Canonical Form*

Instructor: Prof. S. Arvanitis | **Notes transcribed by:** Ch. Dourgoutis | **Digitisation & added notes:** T. Kourtalis

Semester: Spring 2026 | **Official slides (eClass):** LAD Slides (26 pp.)

▷ **Amber = Handwritten Notes (professor's words)**

◇ **Teal = Student's Notes (added explanations & figures)**

Recall from Earlier Lectures

These notes supplement the official slides. The amber boxes reproduce the handwritten lecture material. The teal boxes add geometric intuition, worked examples, and connections to earlier lectures. The slides (linked above) should be read alongside this document for the formal statements, code, and Monte Carlo figures.

1 Why the Conditional Median? (Professor's Comments)

▷ **Handwritten Notes** (what the professor said)

The question is: why should we care about finding $\text{Med}(Y_{(i)} \mid X_n)$? Two reasons:

1. It may interest us **independently**, for example in questions concerning income distributions.
2. **Statistical difficulties** related to the properties of the distribution of $\varepsilon_{(i)} \mid X_n$. For example, if the conditional distribution behaves "as towards extreme values", in the sense that as $x \to +\infty$:

$$P\big(|\varepsilon_{(i)}| > x \mid X_n\big) \sim \frac{c}{x^{\alpha}}, \qquad c > 0, \quad \alpha \in (0, \bar{\alpha})$$

   - When $\alpha \le 2$: $\text{Var}(\varepsilon_{(i)} \mid X_n)$ does not exist.
   - When $\alpha \le 1$: neither does $\mathbb{E}(\varepsilon_{(i)} \mid X_n)$.

   Consequences for OLS:

   (a) $\alpha \le 1 \Rightarrow$ OLSE is **not consistent**.

(b) $1 < \alpha \le 2 \Rightarrow$ OLSE has **slower rate than** $\sqrt{n}$ and is **not asymptotically normal** (relevant: CLT and Limit Theorems for α-stable distributions).

Note: Under some assumptions, even when $\alpha < 1$, the LADE can still be consistent with rate $\sqrt{n}$ and asymptotically normal.

◇ Student's Notes

Connection to earlier lectures:
In Lecture 5 we introduced the three dimensions of estimation: methodological, technical, and numerical. The heavy-tail scenario above shows *why* LAD exists as a methodological alternative to OLS: when the error distribution has heavy tails, the mean may not exist and OLS loses its asymptotic properties, but the median *always* exists (Lecture 8, Definition of median).

The tail-index taxonomy at a glance:

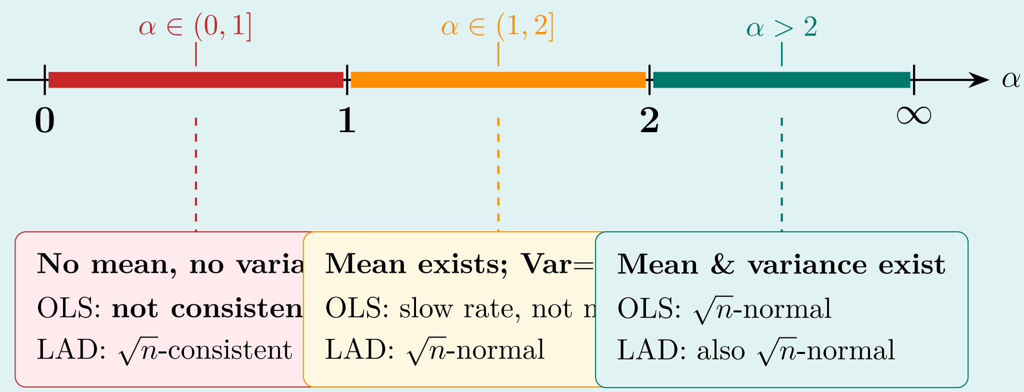

Figure 1: The tail index α determines which moments exist and which asymptotic properties survive. For $\alpha \le 2$, LAD is the safer choice because the median always exists even when the mean or variance do not.

Why the median survives when the mean does not:
The mean is defined as $\mathbb{E}[|\varepsilon|] < \infty$. For a power-law tail $P(|\varepsilon| > x) \sim c/x^{\alpha}$:

$$\mathbb{E}[|\varepsilon|] = \int_0^\infty P(|\varepsilon| > x)\, dx \;\sim\; c \int_0^\infty x^{-\alpha}\, dx \quad \text{diverges for } \alpha \le 1.$$

The median, by contrast, only needs $F(\zeta) = 1/2$ to be solvable—a condition satisfied for *any* distribution, regardless of tail behaviour. This is the fundamental robustness advantage of LAD stated in the professor's second reason above.

2 Property of Absolute Value

▷ **Handwritten Notes** (what the professor said)

We recall the decomposition of the absolute value, needed for the LP reformulation:

$$|x| = \begin{cases} -x, & x < 0 \\ x, & x \geq 0 \end{cases} = -x \cdot \mathbf{1}_{x<0} + x \cdot \mathbf{1}_{x\geq 0}$$

with indicator functions $\mathbf{1}_{x<0} = \begin{cases} 1 & x < 0 \\ 0 & x \geq 0 \end{cases}$, $\mathbf{1}_{x\geq 0} = \begin{cases} 0 & x < 0 \\ 1 & x \geq 0 \end{cases}$.
Any real number x decomposes as $x = x^+ - x^-$ where:

$$x^+ := x \cdot \mathbf{1}_{x\geq 0} \geq 0, \qquad x^- := -x \cdot \mathbf{1}_{x<0} \geq 0, \qquad |x| = x^+ + x^-.$$

◇ **Student's Notes**

Why this decomposition is the key to LP:
The LAD objective $\sum_i |e_i|$ is *not* differentiable at $e_i = 0$ (Lecture 8 warning; Lecture 13 smoothness issue). Writing $|e_i| = e_i^+ + e_i^-$ converts it into a *sum of linear terms* subject to sign constraints—exactly the form that LP solvers handle.

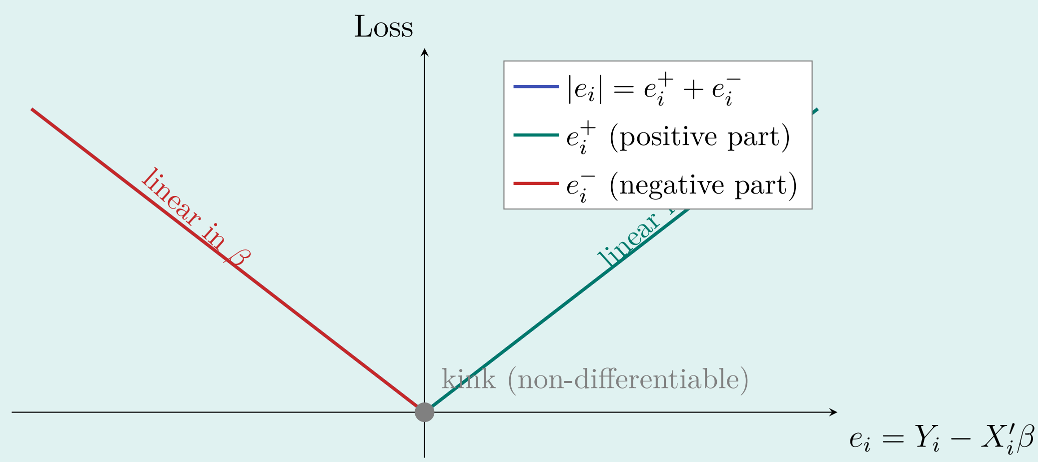

Figure 2: The absolute value (indigo) splits into two linear pieces: e_i^+ (teal, when residual is positive) and e_i^- (rose, when residual is negative). Each piece is linear in β, making the total objective linear once we introduce the auxiliary variables $u_i^\pm$.

3 From LAD to Linear Programming

▷ **Handwritten Notes** (what the professor said)

We showed that:

$$|Y_{(i)} - X_{(i)}\beta| = (Y_{(i)} - X_{(i)}\beta) \cdot \mathbf{1}_{Y_{(i)} \geq X_{(i)}\beta} + (X_{(i)}\beta - Y_{(i)}) \cdot \mathbf{1}_{Y_{(i)} < X_{(i)}\beta}$$

Introducing auxiliary variables:

$$u^+_{(i)} := (Y_{(i)} - X_{(i)}\beta)\, \mathbf{1}_{Y_{(i)} \geq X_{(i)}\beta} \geq 0$$
$$u^-_{(i)} := (X_{(i)}\beta - Y_{(i)})\, \mathbf{1}_{Y_{(i)} < X_{(i)}\beta} \geq 0$$

so $Y_{(i)} - X_{(i)}\beta = u^+_{(i)} - u^-_{(i)}$ and $|Y_{(i)} - X_{(i)}\beta| = u^+_{(i)} + u^-_{(i)}$.
Also writing $\beta = \beta^+ - \beta^-$ with $\beta^+, \beta^- \geq 0$:

$$\hat{\beta}_{LAD} \in \arg\min_{\beta \in \Theta} \sum_{i=1}^{n} |Y_{(i)} - X_{(i)}\beta| \;=\; \arg \min_{\substack{\beta^+,\beta^- \geq 0 \\ u^+,u^- \geq 0}} \sum_{i=1}^{n} (u^+_{(i)} + u^-_{(i)})$$

subject to:

$$Y_{(i)} - X_{(i)}\beta^+ + X_{(i)}\beta^- = u^+_{(i)} - u^-_{(i)}, \qquad u^+_{(i)}, u^-_{(i)} \geq 0.$$

This is a **Linear Programming problem** (Γ Π).

■ **Derivation**

Step-by-step verification that the substitution is correct

Starting from any β, define $e_i := Y_i - X_i'\beta$.
Case 1: $e_i \geq 0$. Then $u^+_i = e_i \geq 0$, $u^-_i = 0$, so $u^+_i + u^-_i = e_i = |e_i|$. ✓
Case 2: $e_i < 0$. Then $u^+_i = 0$, $u^-_i = -e_i > 0$, so $u^+_i + u^-_i = -e_i = |e_i|$. ✓
In both cases, minimising $\sum(u^+_i + u^-_i)$ subject to the equality constraint $e_i = u^+_i - u^-_i$ is equivalent to minimising $\sum |e_i|$. The decomposition $\beta = \beta^+ - \beta^-$ ensures all decision variables are non-negative, as required by LP. □

4 Canonical Form of the LP

▷ **Handwritten Notes** (what the professor said)

A linear program in canonical form is:

$$\min_{x \in \mathbb{R}^m} c'x \qquad \text{subject to} \quad Ax = b,\; x \geq 0$$

In the LAD case with $\Theta = \mathbb{R}^p$, set $m = 2p + 2n$ and:

$$x = \begin{pmatrix} \beta^+ \\ \beta^- \\ u^+ \\ u^- \end{pmatrix}, \quad c = \begin{pmatrix} 0_{p\times 1} \\ 0_{p\times 1} \\ \mathbf{1}_{n\times 1} \\ \mathbf{1}_{n\times 1} \end{pmatrix}, \quad A = \begin{pmatrix} X_n & -X_n & I_{n\times n} & -I_{n\times n} \end{pmatrix}, \quad b = Y_n.$$

◇ **Student's Notes**

Why this is an LP and not something harder:
The LAD problem has three features that make it an LP:

1. The objective $\sum(u_i^+ + u_i^-)$ is **linear** in (u^+, u^-).
2. The constraints $Y_i - X_i'\beta^+ + X_i'\beta^- = u_i^+ - u_i^-$ are **linear** in all decision variables.
3. All constraints are **non-negativity** constraints ($x \geq 0$).

Concrete dimensions ($p = 2$, $n = 100$):

Object	Size	What it contains
x (decision vector)	204×1	$(\beta_1^+, \beta_2^+, \beta_1^-, \beta_2^-, u_1^+, \ldots, u_{100}^+, u_1^-, \ldots, u_{100}^-)$
c (cost vector)	204×1	$(0, 0, 0, 0, \underbrace{1, \ldots, 1}_{100}, \underbrace{1, \ldots, 1}_{100})$
A (constraint matrix)	100×204	$(X_n \vert -X_n \vert I_{100} \vert -I_{100})$
b (RHS)	100×1	Y_n

A LAD problem with $n = 100$, $p = 2$ becomes an LP with **204 variables** and **100 equality constraints**. As shown in the slides [Slides p. 19], this is computationally trivial on modern hardware (seconds).

5 LP Geometry: Polyhedra, Vertices, and Faces

◇ **Student's Notes**

The geometric picture behind LP [Slides p. 6]–[Slides p. 7]:
The feasible set $\{x \geq 0 : Ax = b\}$ is a **polyhedron**—the intersection of finitely many half-spaces. The objective $c'x$ is linear, so its level sets are parallel hyperplanes. As we decrease the constant k in $c'x = k$, the hyperplane sweeps toward the polyhedron.

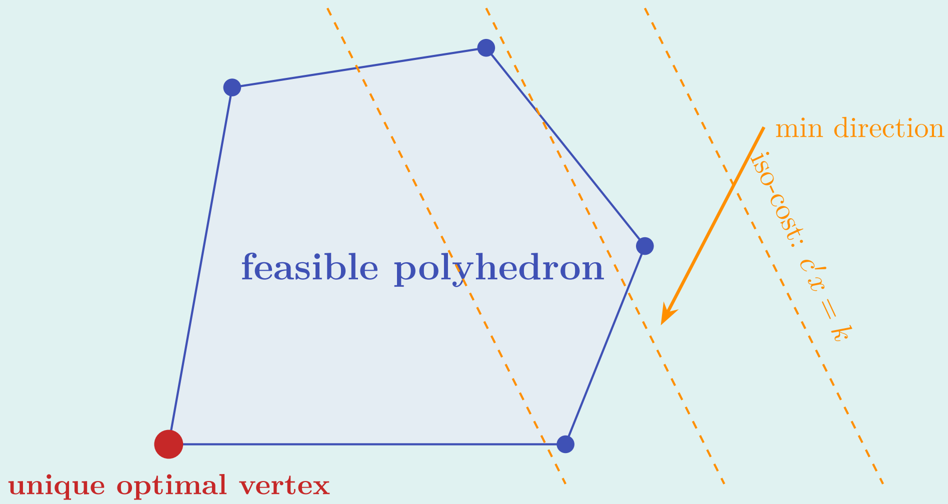

Figure 3: The LP feasible set is a polyhedron (indigo). The objective hyperplanes (amber dashed) sweep until the last one touches the polyhedron at the optimal vertex (rose). When the solution is unique*, it always occurs at a vertex.*

What happens when the solution is not unique:
If the cost vector c happens to be parallel to an *edge* or *face* of the polyhedron, every point on that face achieves the same minimum. The optimal set is an entire **face** of the polyhedron, not a single point.

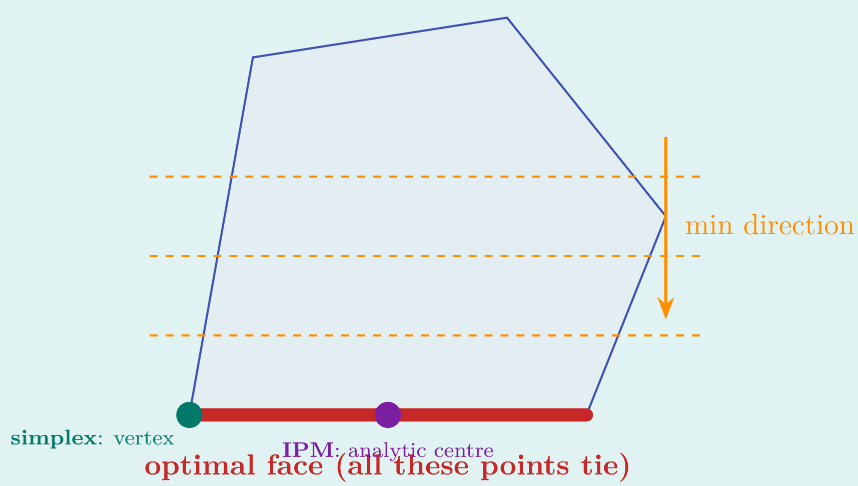

Figure 4: When the optimal set is a face (rose), simplex and interior-point choose different *points on it. Simplex returns the vertex (teal); interior-point returns the analytic centre (purple). This is the source of algorithmic non-coincidence in the degenerate Monte Carlo design.*

6 The Two Algorithms: How They Work

◇ **Student's Notes**

Simplex [Slides p. 8]: moves along the boundary, vertex to vertex. Each "pivot" swaps one basic variable, improving the objective. When the optimum is unique it finds it; when there is an optimal face it typically returns an extreme point (vertex) of that face.

Interior-point (IPM) [Slides p. 9]: moves through the *interior* of the feasible set following a "central path". Formally, it minimises the barrier-penalised objective:

$$\min \; c'x - \mu \sum_j \log x_j \quad \text{s.t.} \quad Ax = b, \quad \mu \downarrow 0.$$

As $\mu \to 0$, the barrier penalty disappears and the solution converges to the LP optimum. When the optimal set is a face, IPM tends toward the **analytic centre** of that face (the point that maximises $\sum \log x_j$ on the optimal face).

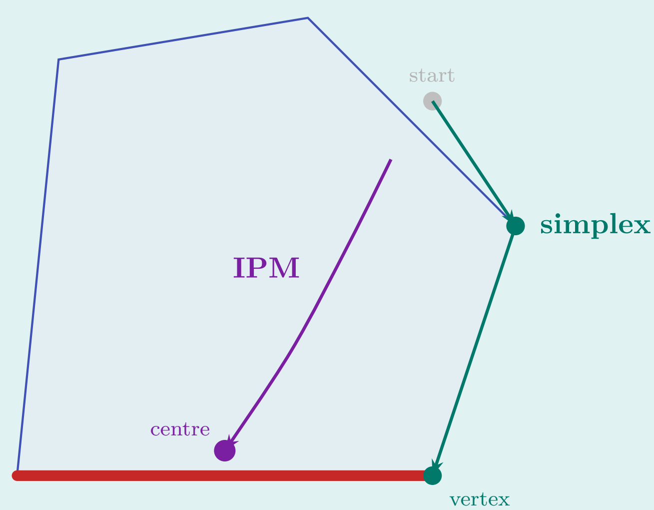

Figure 5: Simplex (teal) walks along the boundary, vertex to vertex. IPM (purple) cuts through the interior. When the optimal set is a face (rose), they land at different points on it. This matches the diagram in the official slides [Slides p. 10].

Comparison table:

Feature	Simplex (`highs-ds`)	Interior-Point (`highs-ipm`)
Path	Boundary (edges)	Interior
Step type	Basis pivot	Newton step on barrier
Worst-case complexity	Exponential	Polynomial
Unique optimum	Returns it	Returns it
Non-unique optimum	Returns a **vertex**	Returns the **centre**

Key Result

The Central Observation [Slides p. 11], [Slides p. 26]

When the $\arg\min$ *of the sample objective is a* ***set*** *(not a singleton), the numerical algorithm acts as a* ***selection rule*** *within that set.*

This means:

$$\underbrace{\text{objective function}}_{\text{methodology}} + \underbrace{\text{geometry of }\Theta}_{\text{identification}} + \underbrace{\text{algorithm}}_{\text{numerical}} = \text{computed estimator}$$

This is exactly the interplay of the three dimensions from **Lecture 2, §1**: methodological, technical, and numerical. In the *generic design*, the algorithm is irrelevant (unique minimum). In the *degenerate design*, the algorithm *is* part of the estimator's definition.

7 Monte Carlo Setup

▷ **Handwritten Notes** (what the professor said)

We consider the model:

$$Y_{(i)} = X_{(i)}\beta + \varepsilon_{(i)} = (1, z_{(i)})(\beta_0, \beta_1)' + \varepsilon_{(i)}$$

Single-valued (generic) design:

$$z_{(i)} \sim N(0,1), \quad \varepsilon_{(i)} \sim \text{Laplace}(0,\sigma), \quad \text{i.i.d.}$$

Multi-valued (degenerate) design:

$$z_{(i)} \in \{-1,0,1\}, \quad \#\{i : z_{(i)} = k\} \approx \tfrac{n}{3}, \quad \varepsilon_{(i)} \in \{-\sigma,+\sigma\}, \quad P(\varepsilon_{(i)} = \sigma) = P(\varepsilon_{(i)} = -\sigma) = \tfrac{1}{2}$$

Note: In the degenerate design the median of $\varepsilon_{(i)}$ *is not unique—see slides for a full discussion of consequences for identification and why the two solvers may return different solutions.*

◇ Student's Notes

Why these two designs test completely different things:

Property	Generic design	Degenerate design
Lecture 8 A3: $\text{Med}(\varepsilon \mid X) = 0$	✓ (Laplace: unique median 0)	✓ (but *any* $\zeta \in [-\sigma, \sigma]$ qualifies)
Lecture 8 A4: $f(0) > 0$	✓ ($f_{\text{Laplace}}(0) = \frac{1}{2\sigma}$)	× (discrete, no density)
Lecture 13: unique min of $M(\beta)$	✓ (M strictly convex)	× (M has flat regions)
Lecture 10: well-separated minimum	✓ ($\delta > 0$ exists)	× (no $\delta > 0$)
Expected: algorithmic coincidence	✓ (unique arg min)	× (flat optimal face)
Expected: consistency	✓ ($\hat{\beta} \xrightarrow{p} \beta_0$)	× (identification fails)

The setup parameters [Slides p. 18]: $\beta_0 = (1, 2)'$, $n \in \{30, 100, 300, 1000\}$, $R = 200$ repetitions, $\sigma = 1$, two solvers: `highs-ds` (dual simplex) and `highs-ipm` (interior-point).

Two metrics reported:

- $\|\hat{\beta}^{DS} - \hat{\beta}^{IPM}\|_2$: **algorithmic discrepancy** — measures whether the two solvers agree.
- $\|\hat{\beta} - \beta_0\|_2$: **estimation error** — measures how close we are to the truth.

8 Monte Carlo Results and Interpretation

8.1 Generic Design: Everything Works

◇ Student's Notes

Results [Slides p. 20]:

n	Alg. discrepancy	Estimation error	Theory predicts
30	0	0.277	$c/\sqrt{30} \approx 0.22$
100	0	0.141	$c/\sqrt{100} = 0.12$
300	0	0.082	$c/\sqrt{300} \approx 0.07$
1000	0	0.038	$c/\sqrt{1000} \approx 0.04$

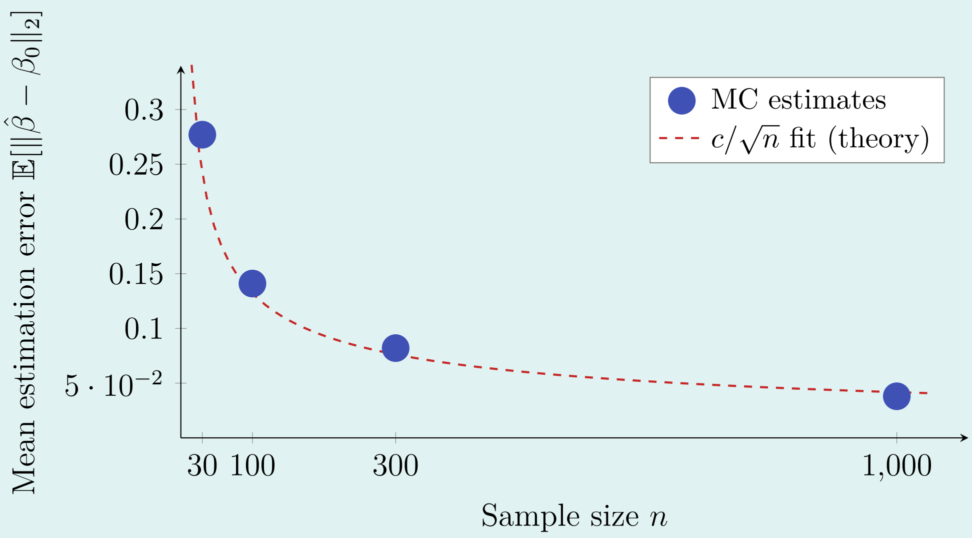

Figure 6: Generic design estimation error vs. n. The Monte Carlo points (indigo) align closely with the $c/\sqrt{n}$ theoretical rate (rose), confirming $\sqrt{n}$-consistency from Lecture 13. Algorithmic discrepancy is identically zero for all n (not plotted).

Interpretation:

1. **Zero algorithmic discrepancy:** The sample LAD criterion has a unique minimum (assumptions A3–A4 hold), so both solvers find the same unique vertex. The LP arg min is a singleton.
2. **Estimation error shrinks as $1/\sqrt{n}$:** This is empirical confirmation of the $\sqrt{n}$-consistency theorem (Lecture 13). The assumptions A1–A5 all hold, and the limit function is strictly convex.

8.2 Degenerate Design: Identification Fails

◇ **Student's Notes**

Results [Slides p. 21]:

n	Alg. discrepancy (mean)	Estimation error (mean)
30	0.180	large, stays large
100	0.116	large, stays large
300	0.042	large, stays large
1000	0.014	large, stays large

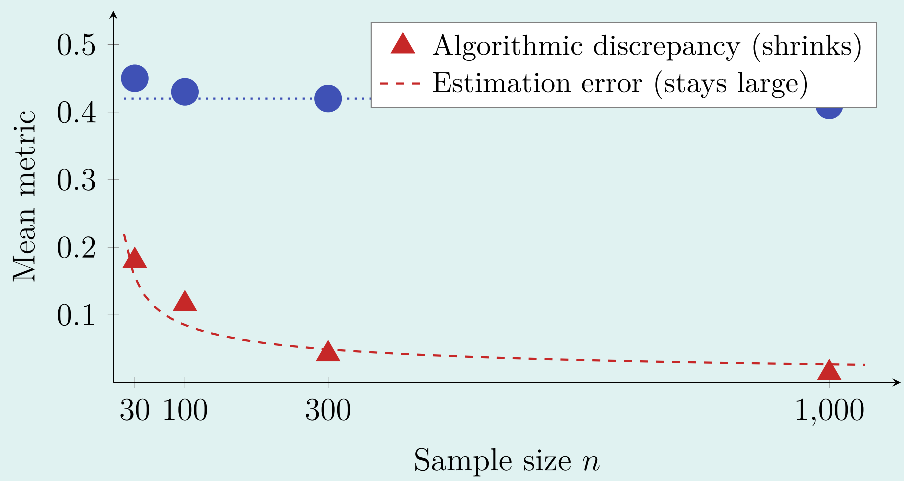

Figure 7: Degenerate design. The algorithmic discrepancy (rose triangles, dashed fit) shrinks toward zero as n grows — the two solvers eventually agree on some *point. But the estimation error (indigo dots) remains high and flat — there is no consistent estimator because identification fails.*

Interpretation:

1. **Algorithmic discrepancy shrinks but does not vanish fast:** As n grows, the optimal face of the LP narrows, so simplex and IPM pick points that are closer together. But they are not identical for any finite n.
2. **Estimation error does NOT shrink:** Because A4 fails (discrete ε has no density), the population limit function $M(\beta)$ has *flat regions* — the asymptotic identification condition from Lecture 10 fails. No algorithm can cure a population identification failure.

! Watch Out

Asymptotic identification failure is not computational.
It is tempting to think: "If I use a better solver, I will get a better estimate in the degenerate design." This is wrong.

The issue is at the *population level*: the limit function $M(\beta)$ is flat, so there is no unique β_0 to converge to. No finite-sample computation can fix this — it is a model specification problem.

Correct diagnosis using earlier lectures:
Lecture 8, A4 requires $f(0) > 0$ (density positive at zero). For $\varepsilon_{(i)} \in \{-\sigma, +\sigma\}$, there is no density at all. Therefore:

- The median of $\varepsilon_{(i)}$ is the *entire interval* $[-\sigma, \sigma]$ (Lecture 8, discrete example).
- $M(\beta)$ has a flat region (Lecture 10, non-well-separated minimum).
- The two-step consistency recipe from Lectures 10–11 fails at Step (b).

9 Connecting the Threads: What We Learn

★ Intuition

Lecture 9 sits at the intersection of all three dimensions from Lecture 2, §1:

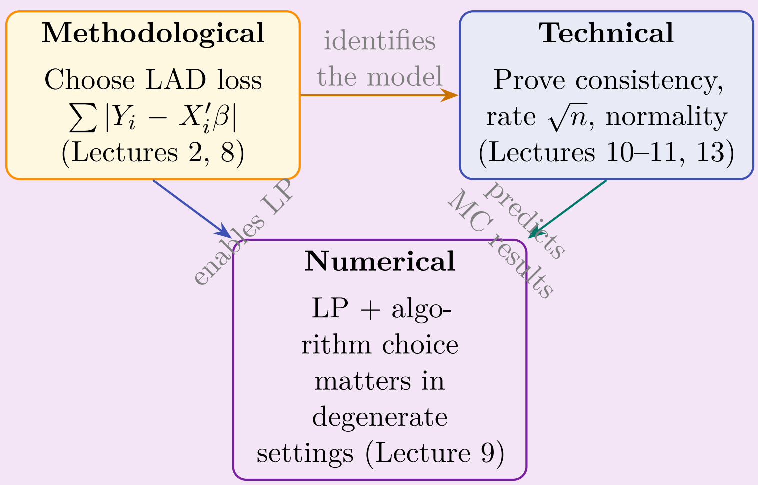

Figure 8: Lecture 9 bridges all three dimensions of econometric practice from Lecture 2.

The concluding lesson of the slides [Slides p. 25]–[Slides p. 26]:

Design	Algorithmic coincidence	Estimation
Generic (unique min)	✓ Solvers agree	✓ Error → 0
Degenerate (flat regions)	× Solvers differ	× Error stays large

General moral: numerical optimisation is not merely an implementation detail. In non-unique settings, it is literally *part of the definition* of the computed estimator:

$$\text{objective} + \text{geometry of } \Theta + \text{algorithm} = \text{estimator}.$$

Cross-Reference Map

◇ Student's Notes

Topic	Official slides	Earlier lectures
Why conditional median	[Slides p. 2]	Lect. 8 (median definition)
Heavy-tail taxonomy	(comments in lecture)	Lect. 5 (rates of conv.), Lect. 8
$\lvert x\rvert = x^+ + x^-$ decomposition	[Slides p. 3]	Lect. 8 (LAD definition)
LAD → LP transformation	[Slides p. 3]–[Slides p. 4]	Lect. 5 (numerical difficulties)
LP canonical form	[Slides p. 5]	—
Polyhedron geometry	[Slides p. 6]–[Slides p. 7]	—
Simplex algorithm	[Slides p. 8]	Lect. 2 (numerical dimension)
Interior-point algorithm	[Slides p. 9]	Lect. 2 (numerical dimension)
Algorithm selection rule	[Slides p. 11], [Slides p. 26]	Lect. 2 §1, Lect. 4
MC: generic design	[Slides p. 13], [Slides p. 20]	Lect. 8 A1–A5, Lect. 13
MC: degenerate design	[Slides p. 14]–[Slides p. 15], [Slides p. 21]	Lect. 10 (well-separation)
Identification failure	[Slides p. 25]–[Slides p. 26]	Lect. 10–11, Lect. 13