从手动 Ridge 到 Hansen L-curve：IPS 项目正则化方法的演进

技术笔记 | 2026-02-15

项目：IPS Unlabeled Learning

Viska Wei

核心结论

- **旧方法**：手动 Ridge 正则化，λ 固定为 10^{-4}（MLE）或 $10^{-3} \cdot \text{tr}(\mathbf{A})/K$（Selftest 自适应缩放）。无法适应不同模型、维度、观测频率下的条件数变化。
- **新方法**：Hansen L-curve 自动选择——对 **A** 做 SVD，在 $(\log\|\text{残差}\|, \log\|\text{解}\|)$ 曲线上找最大曲率点。
- **关键改进**：Model B Selftest Φ 误差从 70% $\to$ 9%，Model Morse d=2 从 8.5% $\to$ 2.2%。
- **我们没有使用** Generalized Tikhonov $(\mathbf{A} + \lambda\mathbf{\Gamma}^\top\mathbf{\Gamma})$，即没有引入 Bayesian 先验协方差。当前仍是 Standard Tikhonov $(\mathbf{A} + \lambda\mathbf{I})$。

Contents

1 背景：为什么需要正则化？ 3
- 1.1 IPS 学习问题的本质 . . . 3
- 1.2 正规方程 . . . 3
- 1.3 为什么不能直接求逆？ . . . 3

2 旧方法：手动 Ridge 正则化 4
- 2.1 Standard Tikhonov（我们使用的形式） . . . 4
- 2.2 我们**没有**使用 Generalized Tikhonov . . . 4
- 2.3 旧代码中的三种 λ 策略 . . . 4
  - 2.3.1 MLE & Sinkhorn：固定 λ / n . . . 5
  - 2.3.2 Self-test：自适应缩放 $\lambda \cdot \text{tr}(\mathbf{A})/K$ . . . 5
- 2.4 旧方法的灾难性后果：Model B 的案例 . . . 5

3 新方法：Hansen L-curve 自动选择 6
- 3.1 L-curve 的直觉 . . . 6
- 3.2 SVD 视角下的解析推导 . . . 6
- 3.3 曲率公式 . . . 7
- 3.4 实现细节 . . . 7
  - 3.4.1 λ 的搜索范围 . . . 7
  - 3.4.2 三层决策逻辑 . . . 7
- 3.5 矩阵归一化 . . . 8

4 结果对比 **8**
4.1 B.1 表格：dt_obs = 0.001, d = 2 8
4.2 MLE / Sinkhorn 完全不受影响 9

5 为什么 MLE 不受正则化影响？ **9**
5.1 MLE 的误差分解 9
5.2 Selftest 的误差分解 10

6 为什么 LJ 的 Selftest 反而变差了？ **10**
6.1 离散化偏差主导 10
6.2 L-curve 的盲区 10

7 L-curve vs GCV **10**
7.1 GCV 的思路 11
7.2 比较 11

8 未来可能的改进 **11**
8.1 Block-specific 正则化（Generalized Tikhonov） 11
8.2 针对 LJ 类强力场模型的 debiased 方法 11
8.3 GCV 作为对比基准 11

9 代码变更总结 **12**
9.1 变更的文件 12
9.2 算法伪代码 12

10 关键结论 **12**

1 背景：为什么需要正则化？

1.1 IPS 学习问题的本质

我们的目标是从粒子系统的观测数据中恢复两个势函数：

- 约束势 $V(x)$：每个粒子受到的外场力
- 相互作用势 $\Phi(x_i - x_j)$：粒子之间的相互作用力

用基函数展开后，问题变成估计系数 $\boldsymbol{\theta} = (\alpha, \beta)$，其中 α 对应 V 的基函数系数，β 对应 Φ 的基函数系数。

1.2 正规方程

无论是 MLE（速度回归）还是 Self-test（能量平衡），最终都归结为求解一个线性系统：

$$\mathbf{A}\boldsymbol{\theta} = \mathbf{b} \tag{1}$$

其中：

- $\mathbf{A} \in \mathbb{R}^{K \times K}$ 是半正定矩阵（$K = K_V + K_\Phi$，通常 $K = 3 \sim 5$）
- $\mathbf{b} \in \mathbb{R}^K$ 是右端向量
- 真解满足 $\mathbf{A}\boldsymbol{\theta}^* = \mathbf{b}$（连续时间极限下）

1.3 为什么不能直接求逆？

类比：天平称重

想象用一架精密天平称量两种外观相似的粉末（A 粉和 B 粉）的重量。如果两种粉末的密度非常接近（比如 $\rho_A = 7.8, \rho_B = 7.9$），那么即使你精确称出了总重量，也很难区分这两种粉末各有多少——因为把一克 A 换成一克 B，总重量只变化 0.1 克。测量中的微小噪声就会导致「A 多了 50 克，B 少了 50 克」这样荒谬的估计。
这就是**条件数**（condition number）大的含义：系统的两个方向（A 和 B 的量）几乎「共线」，微小的观测误差被放大成巨大的估计误差。

在我们的问题中，$\mathbf{A}$ 的条件数可能很大，原因有三：

(1) Φ-Φ block 比 V-V block 更病态。 V 的梯度来自 N 个独立的粒子位置（N 个样本），而 Φ 的梯度来自 $N(N-1)/2$ 个共享粒子的成对差向量，且在耗散项中带 $1/N^2$ 因子。设计矩阵的 Φ 列之间相关性高，导致 $\mathbf{A}_{\Phi\Phi}$ 的特征值小于 $\mathbf{A}_{VV}$。

(2) 基函数的形状相似性（deconvolution 困难）。 例如 Lennard-Jones 势的两个基函数 r^{-12} 和 r^{-6}，其梯度在粒子常见的距离区间内高度相关（相关系数 ≈ 0.94）。这是经典的反卷积（deconvolution）逆问题的病态性。

(3) 不同 Δt 和 d 下，A 的数量级差异巨大。 Self-test 的 $\mathbf{A}$ 中包含 Δt 因子（耗散项 $\propto \Delta t$），所以 $\Delta t = 0.001$ 时 $\mathbf{A}$ 的元素比 $\Delta t = 0.1$ 时小 100 倍。固定的 λ 在一种情况下可能恰好，在另一种情况下却完全不合适。

2 旧方法：手动 Ridge 正则化

2.1 Standard Tikhonov（我们使用的形式）

我们一直使用的是最基本的 Tikhonov 正则化：

$$\boxed{(\mathbf{A} + \lambda \mathbf{I})\,\hat{\boldsymbol{\theta}} = \mathbf{b}} \tag{2}$$

等价于最小化：

$$\min_{\boldsymbol{\theta}} \underbrace{\frac{1}{2}\boldsymbol{\theta}^\top \mathbf{A}\boldsymbol{\theta} - \mathbf{b}^\top \boldsymbol{\theta}}_{\text{数据拟合（原始损失）}} + \underbrace{\frac{\lambda}{2}\|\boldsymbol{\theta}\|^2}_{\text{正则化惩罚}}$$

类比：弹簧回到原点

正则化项 $\frac{\lambda}{2}\|\boldsymbol{\theta}\|^2$ 就像在每个参数上装了一根弹簧，把它拉向零点。λ 越大，弹簧越硬，参数越接近零。

- $\lambda = 0$：完全无弹簧——参数自由飞，如果数据有噪声，参数可能飞到天上去
- λ 太大：弹簧太硬——所有参数被压到零附近，忽略了数据信号（bias 太大）
- λ 恰好：在「跟随噪声」和「忽略信号」之间找到平衡

2.2 我们没有使用 Generalized Tikhonov

Generalized Tikhonov 的形式为：

$$(\mathbf{A} + \lambda\, \boldsymbol{\Gamma}^\top \boldsymbol{\Gamma})\,\hat{\boldsymbol{\theta}} = \mathbf{b} \tag{3}$$

其中 $\boldsymbol{\Gamma}$ 是一个**先验矩阵**（regularization matrix），不一定是单位矩阵。对应的最小化问题是：

$$\min_{\boldsymbol{\theta}} \frac{1}{2}\boldsymbol{\theta}^\top \mathbf{A}\boldsymbol{\theta} - \mathbf{b}^\top \boldsymbol{\theta} + \frac{\lambda}{2}\|\boldsymbol{\Gamma}\boldsymbol{\theta}\|^2$$

Bayesian 解读。 如果把 $\boldsymbol{\theta}$ 看作随机变量，$\boldsymbol{\Gamma}$ 对应**先验协方差的逆**。Standard Tikhonov（$\boldsymbol{\Gamma} = \mathbf{I}$）等价于先验 $\boldsymbol{\theta} \sim \mathcal{N}(0, \lambda^{-1}\mathbf{I})$——即假设所有参数同等重要，且先验均值为零。

为什么 Generalized Tikhonov 可能更好？

- **Block-specific 正则化**：对 V 和 Φ 的系数使用不同的 λ。因为 $\mathbf{A}_{VV}$ 条件良好而 $\mathbf{A}_{\Phi\Phi}$ 病态，对 α（V 系数）不需要正则化，对 β（Φ 系数）需要较强的正则化。用 $\boldsymbol{\Gamma} = \mathrm{diag}(\lambda_V \mathbf{I}_{K_V}, \lambda_\Phi \mathbf{I}_{K_\Phi})$ 可以实现这一点。
- **非对角先验**：如果已知某些基函数组合是「物理上不合理的」，可以通过 $\boldsymbol{\Gamma}$ 惩罚这些方向。
- **Smoothness 惩罚**：对导数的惩罚 $\|\boldsymbol{\theta}'\|^2$（二阶 Tikhonov）用差分矩阵作为 $\boldsymbol{\Gamma}$。

我们没有使用任何这些。当前的实现严格使用 $\boldsymbol{\Gamma} = \mathbf{I}$（标准 Tikhonov），对所有参数施加相同强度的 L_2 惩罚。这意味着 V 和 Φ 的系数被同一个 λ 正则化，尽管它们的条件数可能相差甚远。

2.3 旧代码中的三种 λ 策略

旧代码对三种 solver 使用了不同的手动 λ 策略：

2.3.1 MLE & Sinkhorn：固定 λ / n

Listing 1: MLE 的旧正则化逻辑（`lib/solvers.py`）

```
# MLE: lambda_eff = lambda / n_samples
n_samples = M * (L - 1)  # e.g. 2000 * 99 = 198,000
reg_effective = 1e-4 / n_samples  # ~ 5e-10
ATA_reg = ATA + reg_effective * np.eye(K_total)
```

注意 $\lambda_{\text{eff}} = 10^{-4}/198000 \approx 5 \times 10^{-10}$——**几乎为零**！这是因为 MLE 在归一化 **A** 之后，10^{-4} 也除以了样本数，变得微不足道。

MLE 弹簧类比

弹簧常数 $= 5 \times 10^{-10}$，就像用一根头发丝去拉一块铁球。铁球完全不动——参数完全由数据决定，正则化形同虚设。

2.3.2 Self-test：自适应缩放 $\lambda \cdot \mathbf{tr}(\mathbf{A})/K$

Listing 2: Selftest 的旧正则化逻辑

```
# Self-test: lambda_eff = lambda * trace(A) / K
reg_scale = np.trace(A) / K_total  # average eigenvalue proxy
reg_effective = 1e-3 * reg_scale  # scales with matrix
A_reg = A + reg_effective * np.eye(K_total)
```

这里用了一个聪明的技巧：$\text{tr}(\mathbf{A})/K$ 近似于 **A** 的平均特征值，所以 λ_{eff} 会随 **A** 的大小自动缩放。这部分解决了不同 Δt 下 **A** 数量级不同的问题。

但仍有致命缺陷：

- 固定的 $\lambda = 10^{-3}$ 是拍脑袋选的——对 model_a 可能合适，对 model_b 可能太大
- 对所有维度 d 使用同一个 λ——但最优 λ 随 d 变化
- 无法区分 V 和 Φ 子空间的不同条件数

2.4 旧方法的灾难性后果：Model B 的案例

Model B 的 Φ 使用 InverseInteraction 势（$\gamma = 0.5$），只有 **1 个** Φ 基函数（$K_\Phi = 1$）。Selftest 的 **A** 矩阵是 3×3（$K_V = 2, K_\Phi = 1$）。

旧的自适应缩放对这种低秩问题产生了过度正则化：

旧结果（手动 λ）	d=2	d=10	d=20
Oracle Selftest ∇V%	2.69	2.02	2.58
Oracle Selftest $\nabla\Phi$%	**70.24**	**96.90**	**98.60**

Φ 误差接近 100% 意味着 β **系数被正则化压到了零**——solver 基本上预测“没有相互作用”。

3 新方法：Hansen L-curve 自动选择

3.1 L-curve 的直觉

对于正则化问题 $(\mathbf{A}+\lambda\mathbf{I})\hat{\theta}=\mathbf{b}$，解为：

$$\hat{\boldsymbol{\theta}}(\lambda)=(\mathbf{A}+\lambda\mathbf{I})^{-1}\mathbf{b}$$

随着 λ 从 0 变到 ∞：

- **残差范数** $\rho(\lambda)=\|\mathbf{A}\hat{\theta}-\mathbf{b}\|$ 从 0 增大到 $\|\mathbf{b}\|$（数据拟合越来越差）
- **解范数** $\eta(\lambda)=\|\hat{\boldsymbol{\theta}}\|$ 从 $\|\mathbf{A}^{-1}\mathbf{b}\|$（可能很大）减小到 0（解被压缩到零）

把 $(\log\rho,\log\eta)$ 画在二维平面上，曲线呈现出一个明显的 **"L" 形**拐点。

类比：调节音响的音量旋钮

想象你在嘈杂的房间里听音乐。音量旋钮就是 λ 的反面（$1/\lambda$）：

- **旋钮拧到最大**（$\lambda\to 0$）：音乐响亮清晰，但也把所有噪音一起放大了——你听到的「信号 + 噪音」一团糟
- **旋钮拧到最小**（$\lambda\to\infty$）：完全安静，噪音没了，但音乐也没了
- **L-curve 拐点**：恰好在「音乐信号已经足够清晰，再调大只会放大噪音」的转折处

L-curve 方法就是自动找到这个「拐点」——继续增大音量只带来噪声、不增加信号的临界位置。

3.2 SVD 视角下的解析推导

对 **A** 做奇异值分解（**A** 是对称正定的，所以 SVD = 特征分解）：

$$\mathbf{A}=\mathbf{U}\,\mathrm{diag}(\sigma_1,\ldots,\sigma_K)\,\mathbf{U}^\top$$

定义 Fourier 系数 $\beta_i=\mathbf{u}_i^\top\mathbf{b}$（**b** 在特征方向上的投影），则正则化解为：

$$\hat{\boldsymbol{\theta}}(\lambda)=\sum_{i=1}^{K}\underbrace{\frac{\sigma_i}{\sigma_i+\lambda}}_{\text{filter factor } f_i}\cdot\frac{\beta_i}{\sigma_i}\cdot\mathbf{u}_i$$

- 当 $\sigma_i\gg\lambda$：$f_i\approx 1$，该方向不受影响
- 当 $\sigma_i\ll\lambda$：$f_i\approx\sigma_i/\lambda\approx 0$，该方向被压制

残差和解范数的解析表达式：

$$\rho^2(\lambda)=\sum_{i=1}^{K}(1-f_i)^2\beta_i^2 \tag{4}$$

$$\eta^2(\lambda)=\sum_{i=1}^{K}\frac{f_i^2\beta_i^2}{\sigma_i^2} \tag{5}$$

3.3 曲率公式

L-curve 拐点就是 $(\xi, \eta_l) = (\log \rho, \log \eta)$ 曲线上曲率 κ 最大的点。Hansen (1992) 给出了解析曲率公式：

$$\kappa(\lambda) = \frac{\xi'(\lambda)\,\eta_l''(\lambda) - \eta_l'(\lambda)\,\xi''(\lambda)}{(\xi'^2 + \eta_l'^2)^{3/2}} \tag{6}$$

其中导数都可以用 σ_i 和 β_i 解析计算（不需要数值微分）。

3.4 实现细节

3.4.1 λ 的搜索范围

Listing 3: L-curve λ 搜索范围

```
# Search range: from below sigma_min^2 to above sigma_max^2
lam_lo = s.min()**2 * 1e-4
lam_hi = s.max()**2 * 10
lambdas = np.logspace(np.log10(max(lam_lo, 1e-16)),
                      np.log10(lam_hi), 200)
```

范围是 $[\sigma^2_{\text{min}} \times 10^{-4},\ \sigma^2_{\text{max}} \times 10]$，其中 σ_i 是 **A** 的奇异值。这确保了：

- 下界远低于最小特征值的平方——覆盖了「几乎不正则化」的区域
- 上界远高于最大特征值的平方——覆盖了「完全压制」的区域
- 对数均匀采样 200 个候选点——足够密以找到拐点

注意

你提到的 $[\lambda_{\text{min}}, \lambda_{\text{max}}]$ 范围的经典选法通常是基于 **A** 的特征值：$\lambda_{\text{min}} = \sigma_K^2$（最小特征值的平方），$\lambda_{\text{max}} = \sigma_1^2$（最大特征值的平方）。我们的实现范围更宽（各方向扩展了 10^{-4} 和 10 倍），以确保不遗漏拐点。这是 Hansen (2001) 推荐的做法。

3.4.2 三层决策逻辑

找到最大曲率后，我们根据曲率的大小分三种情况：

Listing 4: 三层决策逻辑

```
max_kappa = np.max(kappa)
if max_kappa > 0:
    # Case 1: clear L-shaped corner
    lam_opt = lambdas[np.argmax(kappa)]
elif abs(max_kappa) > 0.01:
    # Case 2: no positive corner, but significant
    #   curvature structure exists
    lam_opt = lambdas[np.argmax(kappa)]
else:
    # Case 3: flat L-curve (|kappa| < 0.01)
    #   -> well-conditioned, minimal reg
    lam_opt = s[-1]**2 * 1e-6
```

Case 1 ($\kappa_{\text{max}} > 0$)： 标准 L-curve 拐点。在曲率为正的最大处选择 λ。

Case 2 ($\kappa_{\max} \leq 0$ 但 $|\kappa_{\max}| > 0.01$)： 没有正曲率拐点，但曲线仍有显著的弯曲结构。这通常出现在 selftest 类的能量平衡系统中。选最不负（least negative）的曲率点作为 bias-variance 平衡点。

Case 3 ($|\kappa_{\max}| < 0.01$)： L-curve 几乎是平的——说明系统条件良好，不需要正则化。使用最小的 $\lambda = \sigma_{\min}^2 \times 10^{-6}$ 仅保证数值稳定性。

关键发现

$|\kappa_{\max}|$ 的值完美区分了两类问题：

方法	$\lvert\kappa_{\max}\rvert$	解释
MLE / Sinkhorn	$\sim 7 \times 10^{-4}$	平坦 L-curve → 条件良好 → Case 3
Selftest	$\sim 0.5 - 4$	显著弯曲 → 需要正则化 → Case 1 或 2

阈值 0.01 干净地将两类情况分开。

3.5 矩阵归一化

在调用 L-curve 之前，我们对 **A** 和 **b** 做了归一化（除以样本数）：

```
n_samples = M * (L - 1)
ATA /= n_samples   # O(1) entries
ATb /= n_samples
```

这非常重要！如果不归一化：

- $\mathbf{A}_{\text{unnorm}} = n \cdot \mathbf{A}_{\text{norm}}$，特征值是 n 倍大
- λ 的搜索范围会被推到 n^2 量级
- L-curve 的形状可能被拉伸变形

归一化后 **A** 的特征值是 $O(1)$，L-curve 在合理的 λ 范围内有良好的数值行为。

4 结果对比

4.1 B.1 表格：dt_obs = 0.001, d = 2

Table 1: Oracle Selftest 结果对比：旧手动 λ vs 新 L-curve（$\nabla V\%$ / $\nabla\Phi\%$）

	旧方法（手动 λ）		**新方法（L-curve）**	
Model	$\nabla V\%$	$\nabla\Phi\%$	$\nabla V\%$	$\nabla\Phi\%$
A (d=2)	6.56	6.59	0.74	1.73
B (d=2)	2.69	70.24	0.39	**9.45**
C / LJ (d=2)	9.42	10.43	11.49	16.62
D / Morse (d=2)	8.58	8.50	**0.74**	**2.24**
B (d=10)	2.02	96.90	0.19	**19.90**
B (d=20)	2.58	98.60	0.43	**26.47**

赢家。

- **Model B**：Φ 误差从 $70 \to 9\%$（d=2）、$97 \to 20\%$（d=10）、$99 \to 26\%$（d=20）。改善 **4–7 倍**。
- **Model Morse d=2**：V 从 $8.58 \to 0.74\%$，Φ 从 $8.50 \to 2.24\%$。成为 Morse 模型所有方法中的**最佳结果**（击败 NN！）。

输家。

- **Model LJ d=2**：V 从 $9.42 \to 11.49\%$，Φ 从 $10.43 \to 16.62\%$。变差了约 60%。原因见第 6 节。

4.2 MLE / Sinkhorn 完全不受影响

Model	MLE (旧)	MLE (新/L-curve)
A (d=2)	6.08 / 12.52	6.08 / 12.52
B (d=2)	0.08 / 3.88	0.08 / 3.88
LJ (d=2)	25.63 / 39.90	25.63 / 39.90

完全一致。原因：

1. 旧 MLE 的 $\lambda_{\text{eff}} = 10^{-4}/n \approx 5 \times 10^{-10}$——已经约等于零
2. L-curve 检测到 flat L-curve（$|\kappa| \sim 10^{-4}$），选择 $\lambda = \sigma_{\min}^2 \times 10^{-6}$——也约等于零
3. 两种方法都得到的是「无正则化」的最小二乘解

5 为什么 MLE 不受正则化影响？

你正确地指出了：Φ 的估计对 MLE 和 Selftest 来说都是一个反卷积逆问题，**A** 矩阵都是病态的。那为什么 L-curve 只改善了 Selftest 而没有改善 MLE？

答案不在于条件数，而在于**误差的瓶颈在哪里**。

5.1 MLE 的误差分解

MLE 做的是速度回归：$v_{\text{obs}} = \boldsymbol{\theta} + \epsilon$，其中 是设计矩阵，$\epsilon$ 是噪声。正规方程的解为：

$$\hat{\boldsymbol{\theta}} = (\mathbf{A} + \lambda \mathbf{I})^{-1}\mathbf{b} = (\mathbf{A} + \lambda \mathbf{I})^{-1}\mathbf{A}\boldsymbol{\theta}^* + (\mathbf{A} + \lambda \mathbf{I})^{-1\top}\epsilon$$

- **正则化偏差**：$\|(\mathbf{A} + \lambda \mathbf{I})^{-1}\mathbf{A}\boldsymbol{\theta}^* - \boldsymbol{\theta}^*\| \sim O(\lambda)$
- **噪声放大**：$\|(\mathbf{A} + \lambda \mathbf{I})^{-1\top}\epsilon\| \sim O(\sigma/\sqrt{n \cdot \Delta t})$
- **离散化偏差**：$O(\Delta t)$（有限差分速度估计的系统误差）

关键：$n = M \times (L-1) = 198{,}000$，所以噪声项在 $\sqrt{n}$ 平均后已经非常小。占主导的是 $O(\Delta t)$ 离散化偏差，这与 λ 完全无关。因此，无论 λ 是 10^{-10} 还是 10^{-6}，MLE 结果不变。

5.2 Selftest 的误差分解

Selftest 最小化能量平衡损失，正规方程为 $\mathbf{A}\boldsymbol{\theta} = \mathbf{b}$，但 $\mathbf{b}$ 包含能量差 ΔE——这些随机量的方差更大，且离散化偏差为 $O(\Delta t^2 \times |\text{force}|^2)$。

关键区别：Selftest 的旧自适应缩放 $\lambda_{\text{eff}} = 10^{-3} \times \text{tr}(\mathbf{A})/K$ 产生的 λ 是 $O(10^{-3})$——比 MLE 的 $O(10^{-10})$ 大了**七个数量级**。这个大 λ 有时会严重过度正则化（如 Model B），L-curve 能找到更合适的值。

一句话总结

MLE 的旧 λ 已经是零，L-curve 选的也是零——没有改进空间。
Selftest 的旧 λ 可能太大（10^{-3}），L-curve 能找到更小的最优值。

6 为什么 LJ 的 Selftest 反而变差了？

6.1 离散化偏差主导

Lennard-Jones 势的力 $\nabla\Phi_{\text{LJ}} \sim 24r^{-13} - 12r^{-7}$ 在 $r \lesssim 1$ 时**极其巨大**。Selftest 的 $\mathbf{b}$ 向量中包含了来自 Euler-Maruyama 离散化的系统偏差：

$$\mathbf{b}_{\text{obs}} = \mathbf{A}\boldsymbol{\theta}^* + \underbrace{\mathbf{b}_{\text{bias}}}_{\text{discretization}} + \underbrace{\mathbf{b}_{\text{noise}}}_{\text{statistical}}$$

对于 LJ：$\|\mathbf{b}_{\text{bias}}\| \propto \Delta t^2 \times |\text{force}|^2$

Model	典型 $\lvert\nabla\Phi\rvert$	$\Delta t^2 \times \lvert\nabla\Phi\rvert^2$	相对 Model A
A	2–5	4×10^{-6} – 2.5×10^{-5}	$1\times$
LJ	**12–160**	$\mathbf{10^{-4}}$ – $\mathbf{2.6 \times 10^{-2}}$	**100–1000**$\times$

6.2 L-curve 的盲区

L-curve 优化的是**统计噪声 vs 正则化偏差**的权衡。它完全不知道存在离散化偏差。

类比：带偏差的指南针

你用一个指南针导航，但指南针有磁偏角（系统偏差）。L-curve 相当于一个智能校准器，它能自动补偿指针的随机抖动（噪声），但对固定的磁偏角（系统偏差）一无所知。
对于 Model A（磁偏角很小），消除抖动后方向就很准。
对于 LJ（磁偏角很大），消除抖动后方向仍然是偏的。更糟的是，旧方法的过度正则化恰好部分抵消了磁偏角（两个 bias 方向相反，偶然互相补偿），而 L-curve 去掉了这个「幸运的过度正则化」，反而暴露了底层的系统偏差。

这在逆问题文献中被称为 **Bakushinskii veto**：当模型误差（model misspecification）主导统计噪声时，基于噪声的参数选择规则会失败。

7 L-curve vs GCV

广义交叉验证（Generalized Cross-Validation, GCV）是另一种常见的 λ 选择方法。

7.1 GCV 的思路

GCV 最小化：

$$\text{GCV}(\lambda) = \frac{\|\mathbf{A}\hat{\boldsymbol{\theta}}(\lambda) - \mathbf{b}\|^2}{\left(\text{tr}(\mathbf{I} - \mathbf{A}(\mathbf{A} + \lambda\mathbf{I})^{-1})\right)^2}$$

直觉：分子是残差（拟合好坏），分母惩罚了模型的「有效自由度」。GCV 选择使泛化误差最小的 λ。

7.2 比较

	L-curve	**GCV**
原理	几何：log-log 曲线拐点	统计：交叉验证泛化误差
鲁棒性	对噪声水平不敏感；但对 λ 网格敏感	更 robust；不需要用户调参
精确性	选好 λ 范围和 regularization norm 后更精确	稳定但可能不够精确
弱点	可能找不到拐点（flat L-curve）	对相关噪声敏感；可能 under-regularize
我们的场景	适合 selftest（有明显拐点，$\lvert\kappa\rvert \sim 0.5$–$4$）	可能更适合 MLE（但 MLE 不需要正则化）

你说得对：总体上 GCV 比 L-curve 更鲁棒，但做好了的 L-curve 可以更精确。“做好” 的关键在于 (1) λ 搜索范围的选择，(2) regularization matrix $\boldsymbol{\Gamma}$ 的选择——这些我们都还有改进空间。

但 λ 选择不是本文的主要贡献，目前 L-curve 已经足够好用。

8 未来可能的改进

8.1 Block-specific 正则化（Generalized Tikhonov）

当前对 V 和 Φ 系数使用统一的 λ，但两者条件数差异巨大。可以使用：

$$\boldsymbol{\Gamma} = \begin{pmatrix} \sqrt{\lambda_V}\,\mathbf{I}_{K_V} & 0 \\ 0 & \sqrt{\lambda_\Phi}\,\mathbf{I}_{K_\Phi} \end{pmatrix}$$

分别对 V 和 Φ 子空间做 L-curve，选择两个独立的 λ_V 和 λ_Φ。

8.2 针对 LJ 类强力场模型的 debiased 方法

L-curve 对 LJ 失效的根本原因是离散化偏差。可能的解决方案：

- 使用更高阶的离散化格式（如 Milstein 或 Runge-Kutta）来减小偏差
- 使用 Richardson 外推：在两个不同的 Δt 下求解，外推到 $\Delta t \to 0$
- 在 L-curve 的 discrepancy principle 中加入已知的离散化偏差估计

8.3 GCV 作为对比基准

实现 GCV 并与 L-curve 对比，特别是在 selftest 场景下。如果 GCV 对 LJ 更鲁棒（可能因为它的 leave-one-out 结构间接考虑了某些系统偏差），这将是一个有价值的发现。

9 代码变更总结

9.1 变更的文件

文件	变更内容
`lib/solvers.py`	新增 `_hansen_lcurve_lambda()`；三个 solver 的 `reg` 默认值改为 `` `auto' ``
`scripts/run_baselines.py`	注意：`run_one()` 从未将 `reg` 参数传给 solver (dead parameter bug) ——solver 一直使用默认值
`LED_ips_nn.tex`	更新正则化描述、Algorithm 1、所有结果表格
`ref_weak_IPS_learning.bib`	新增 Hansen 1992 引用
`results.md`	更新 B.1 表格所有 Oracle Selftest 值

9.2 算法伪代码

Algorithm 1 Hansen L-curve λ 选择

Require: 正规方程矩阵 $\mathbf{A} \in \mathbb{R}^{K\times K}$，右端向量 $\mathbf{b} \in \mathbb{R}^K$

计算 SVD：$\mathbf{A} = \mathbf{U}\,\mathrm{diag}(\sigma_1, \ldots, \sigma_K)\,\mathbf{U}^\top$
计算 Fourier 系数：$\beta_i = \mathbf{u}_i^\top \mathbf{b}$
生成候选网格：$\lambda_j \in [\sigma_K^2 \times 10^{-4},\ \sigma_1^2 \times 10]$，对数均匀 200 个点
for $j = 1, \ldots, 200$ **do**
    计算 filter factors：$f_i = \sigma_i^2/(\sigma_i^2 + \lambda_j)$
    计算 ρ^2, η^2 及其对 λ 的一阶和二阶导数 (解析公式)
    计算曲率：$\kappa_j = (\xi'\eta_l'' - \eta_l'\xi'')/(\xi'^2 + \eta_l'^2)^{3/2}$
end for
$\kappa_{\max} \leftarrow \max_j \kappa_j$
if $\kappa_{\max} > 0$ **then**
    $\lambda^* \leftarrow \lambda_{\arg\max \kappa}$ // 明确的 L-curve 拐点
else if $|\kappa_{\max}| > 0.01$ **then**
    $\lambda^* \leftarrow \lambda_{\arg\max \kappa}$ // 有弯曲结构但无正曲率
else
    $\lambda^* \leftarrow \sigma_K^2 \times 10^{-6}$ // 平坦 L-curve, 最小正则化
end if
return λ^*

10 关键结论

1. **Standard Tikhonov** $(\mathbf{A} + \lambda\mathbf{I})$：我们使用的是最基本的形式，正则化矩阵 $\mathbf{\Gamma} = \mathbf{I}$。没有引入 Bayesian 先验协方差 (Generalized Tikhonov)。
2. **旧方法的缺陷**：MLE 的手动 λ 太小 (形同虚设)，Selftest 的自适应 λ 虽然部分解决了量级问题，但固定的 $\lambda = 10^{-3}$ 对某些模型 (如 Model B) 严重过度正则化。
3. **L-curve 的改进**：对 Selftest 效果显著 (Model B Φ: 70% $\rightarrow$ 9%，Model Morse: 8.5% $\rightarrow$ 2.2%)，但对 MLE 无效 (旧 λ 已经等于零)，对 LJ 反而变差 (离散化偏差主导)。
4. **L-curve 不是 GCV**：L-curve 是几何方法 (找 log-log 曲线拐点)，GCV 是统计方法 (最小化交叉验证误差)。总体上 GCV 更鲁棒，但 L-curve 做好了可以更精确。两者都不能处理离散化偏差。

5. **可以进一步改进**：block-specific λ（为 V 和 Φ 分别选择），但这不是本文主要贡献，当前的 L-curve 已经大幅改善了最关键的瓶颈。