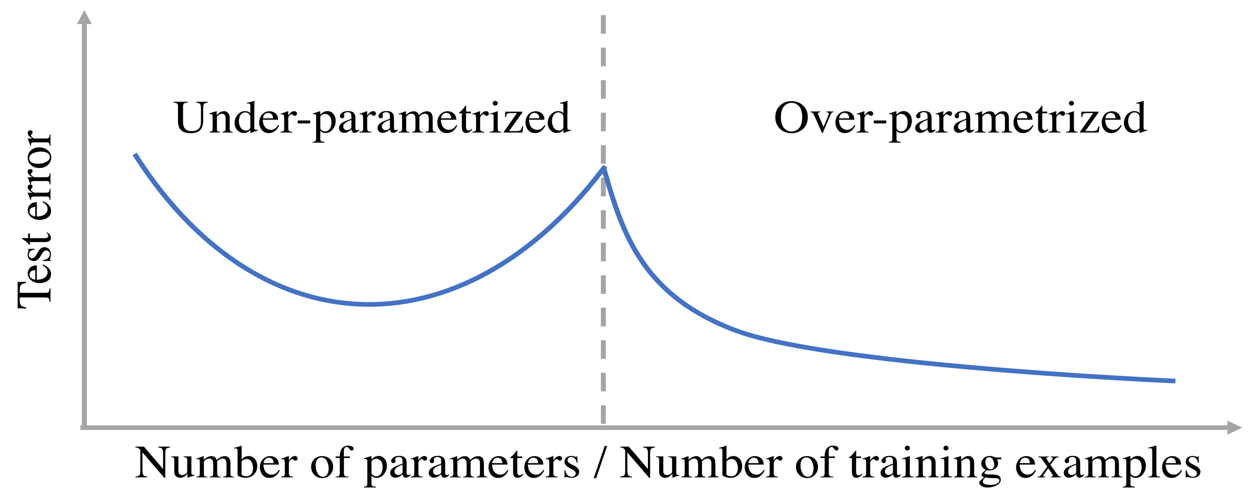
Rethinking complexity and the bias-variance tradeoff: *Using Minimum Description Length (MDL)*

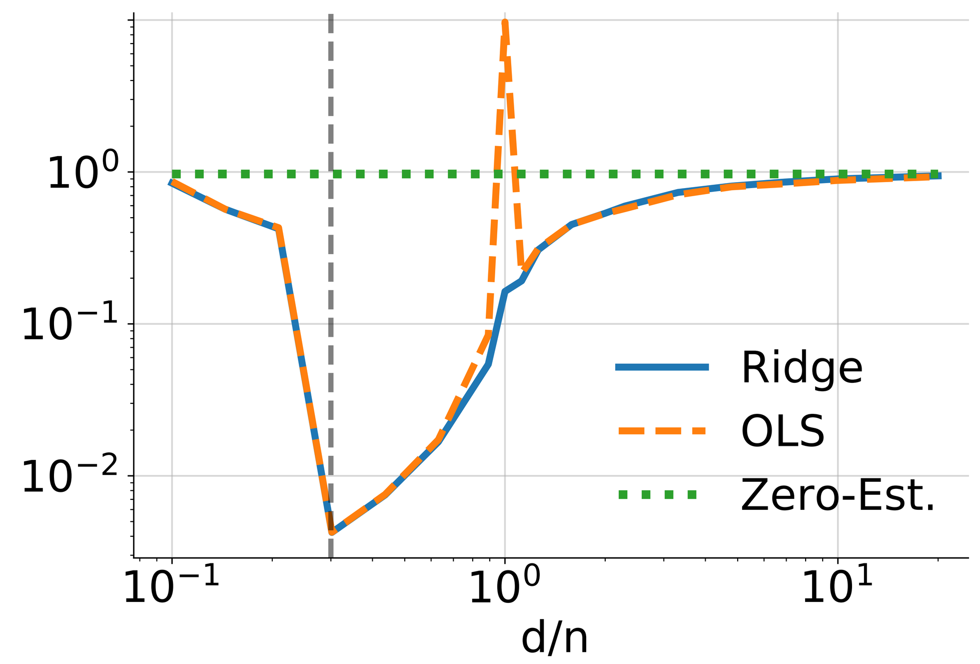
Berkeley | EECS

Raaz Dwivedi, Chandan Singh,
Bin Yu, Martin Wainwright

INTRO: DOUBLE-DESCENT



- Bad estimators don't exhibit U-shaped bias-variance tradeoff, even in low-dimensions; why should we expect that from OLS?

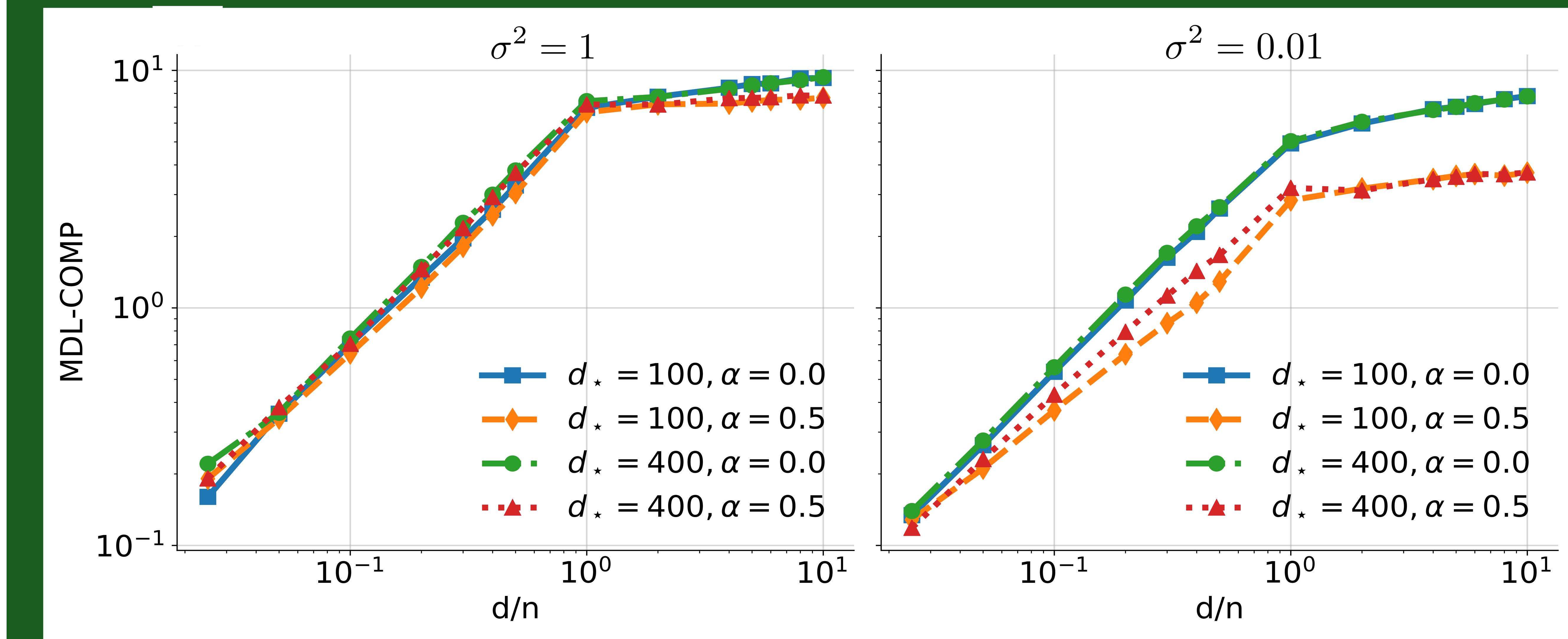


- Why is complexity = # parameters in overparameterized models?

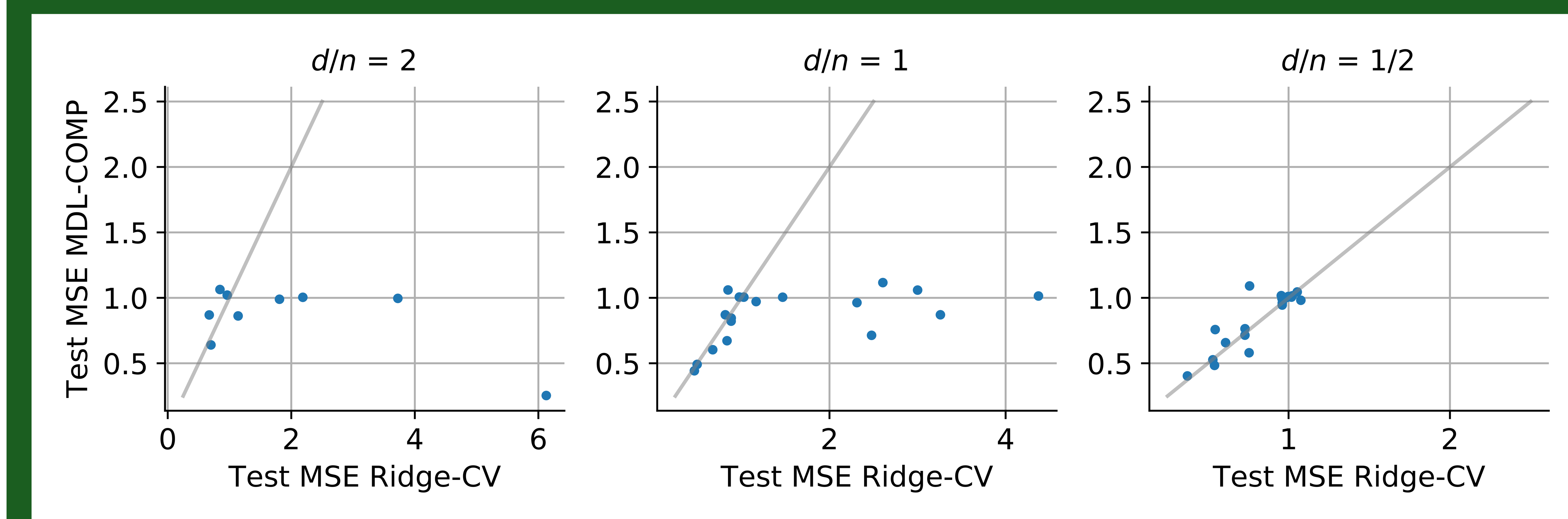
REVISIT COMPLEXITY

1. Build on the normalized maximum likelihood (NML) principle in MDL: luckiness NML codes induced by ridge estimators to account for degeneracy in high-dimensions
2. Define MDL-COMP as optimal excess codelength over the LNML codes
3. Find a measure that depends on covariate design, and signal strength and is not mere parameter count

MDL-Complexity can grow logarithmic with parameters for overparameterized models



& inform hyper-parameter tuning in limited-data settings, using just a single fold computation



Bottom plot: 19 Real datasets from Penn ML Benchmark (CV = cross-validation)
Top plot: Gaussian synthetic data

More experiments + theory @https://arxiv.org/abs/2006.10189 raaz.rsk@berkeley.edu

RESULTS

- MDL-COMP has a non-linear scaling
- Informs in-sample generalization for linear and kernel methods
- Provides competitive performance to cross-validation (CV) in out-of-sample

MDL-COMP Expressions

- *Linear models*: $y_i = x_i^\top \theta^\star + \mathcal{N}(0, \sigma^2)$

$$\text{MDL-COMP} = \frac{1}{n}\sum_{i=1}^{d} \log\left(\rho_i + \frac{\sigma^2}{w_i^2}\right)$$

$w = \mathbf{U}^\top \theta^\star$, $\mathbf{U}, \{\rho_i\}$ ~eigenvectors and eigenvalues of $\mathbf{X}^\top\mathbf{X}$

- *Gaussian design*: $d_\star$ = #*true feats*, d = #*fitted feats*, n = #*samples*; with $d_\star = n$

$$\approx \begin{cases} \frac{d}{n}\log\left(1 + d_\star/\|\theta^\star\|^2\right) & \text{when } d < n \\ \log\left[d(1/\|\theta^\star\|^2 + 1/n)\right] & \text{when } d > n. \end{cases}$$

- For the plots on the top: $\theta^\star \sim U(\mathbb{S}^{d_\star - 1})$

$$x_i \sim \mathcal{N}(0, \mathrm{diag}(1, 2^{-\alpha}, \ldots, d^{-\alpha}))$$

- *Kernel methods*: $y_i = f^\star(x_i) + \mathcal{N}(0, \sigma^2)$

$$\text{MDL-COMP} = \inf_\lambda \left(\frac{\lambda\|f^\star\|_{\mathbb{H}}^2}{2n\sigma^2} + \frac{1}{2n}\sum_{i=1}^{n}\log\left(1 + \frac{\rho_i}{\lambda}\right)\right)$$

$\{\rho_i\}$~ eigenvalues of $\mathbf{K} = (\mathrm{kernel}(x_i, x_j))_{i,j=1}^n$

MDL-COMP based hyper-parameter tuning

- *Kernel ridge regression*:

$$\min_\lambda \frac{\|\mathbf{K}\widehat{\theta} - \mathbf{y}\|^2 + \lambda\widehat{\theta}^\top\mathbf{K}\widehat{\theta}}{\sigma^2} + \sum_{i=1}^{d}\log\left(1 + \frac{\rho_i}{\lambda}\right)$$

$\widehat{\theta} = (\mathbf{K} + \lambda\mathbf{I})^{-1}\mathbf{y}, \{\rho_i\} \sim$ eigenvalues of $\mathbf{K}$

- *For linear model*: $\mathbf{K} = \mathbf{X}\mathbf{X}^\top$