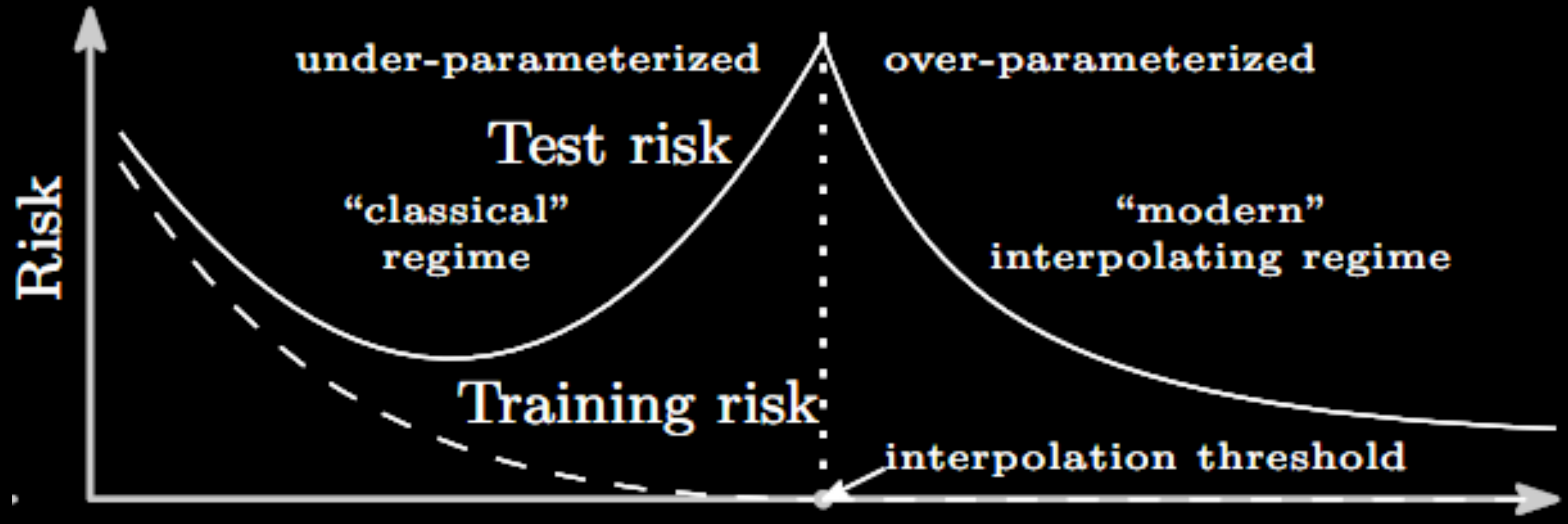
Revisiting minimum description length complexity for overparameterized models

raaz dwivedi, chandan singh, bin yu & martin wainwright

Non-U shaped “tradeoff” curves in modern ML settings

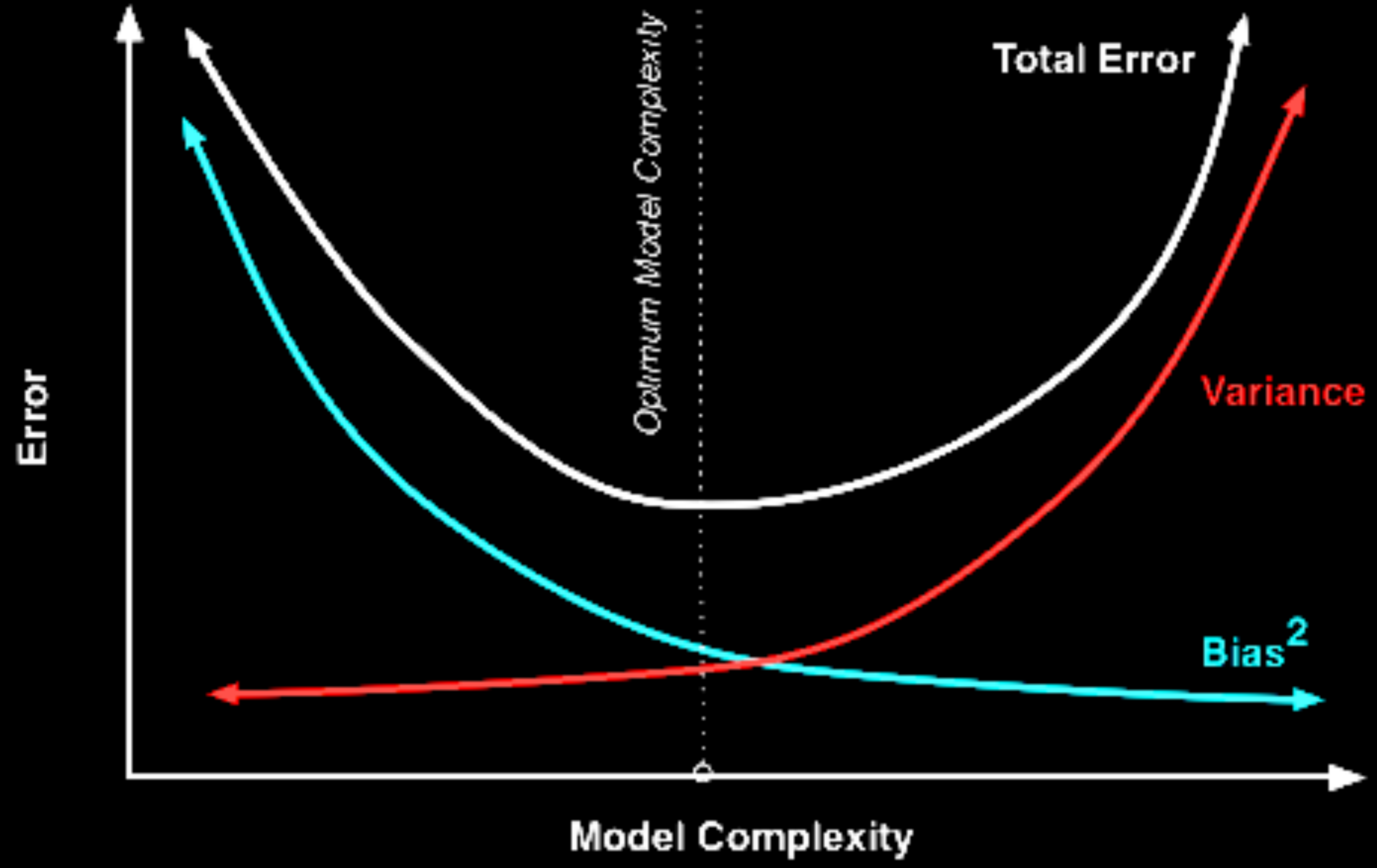


“Complexity” of $\mathscr{H}$

Belkin-Hsu-Ma-Mandal 18, Muthukumar-Vodrahalli-Sahai 19, Hastie-Montanari-Rosset-Tibshirani 19, …

Bias-variance tradeoff

- Occam's razor: Pick the simplest model that provides a good fit to the training data
- U-shaped curves: Established for low-dimensional settings with "good" estimators

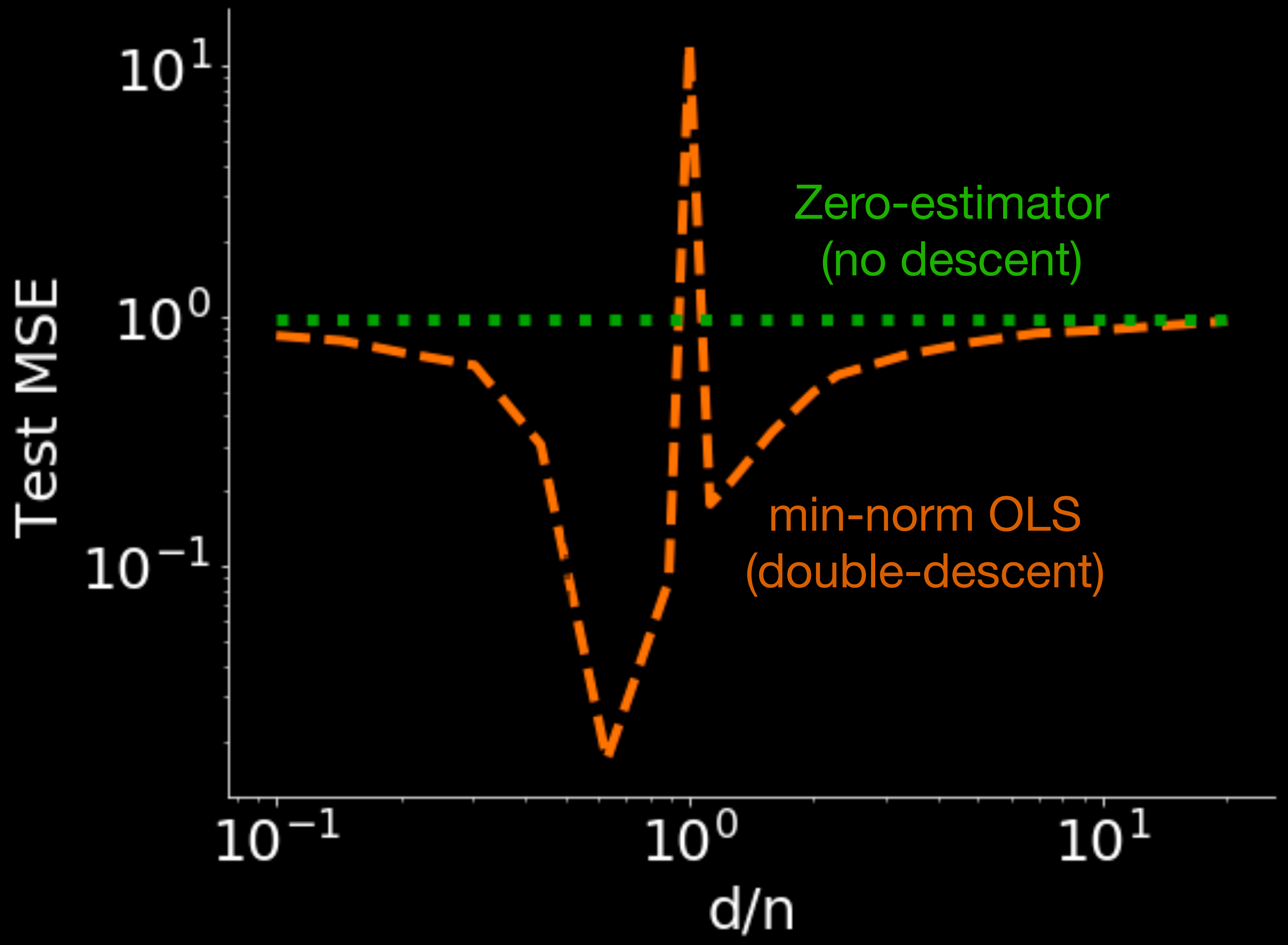


Bias-variance tradeoff: Few things to note..

- We should expect a tradeoff *given*
  - some fixed data
  - as the “complexity” of the fitted estimator varies

Bias-variance tradeoff: Few things to note..

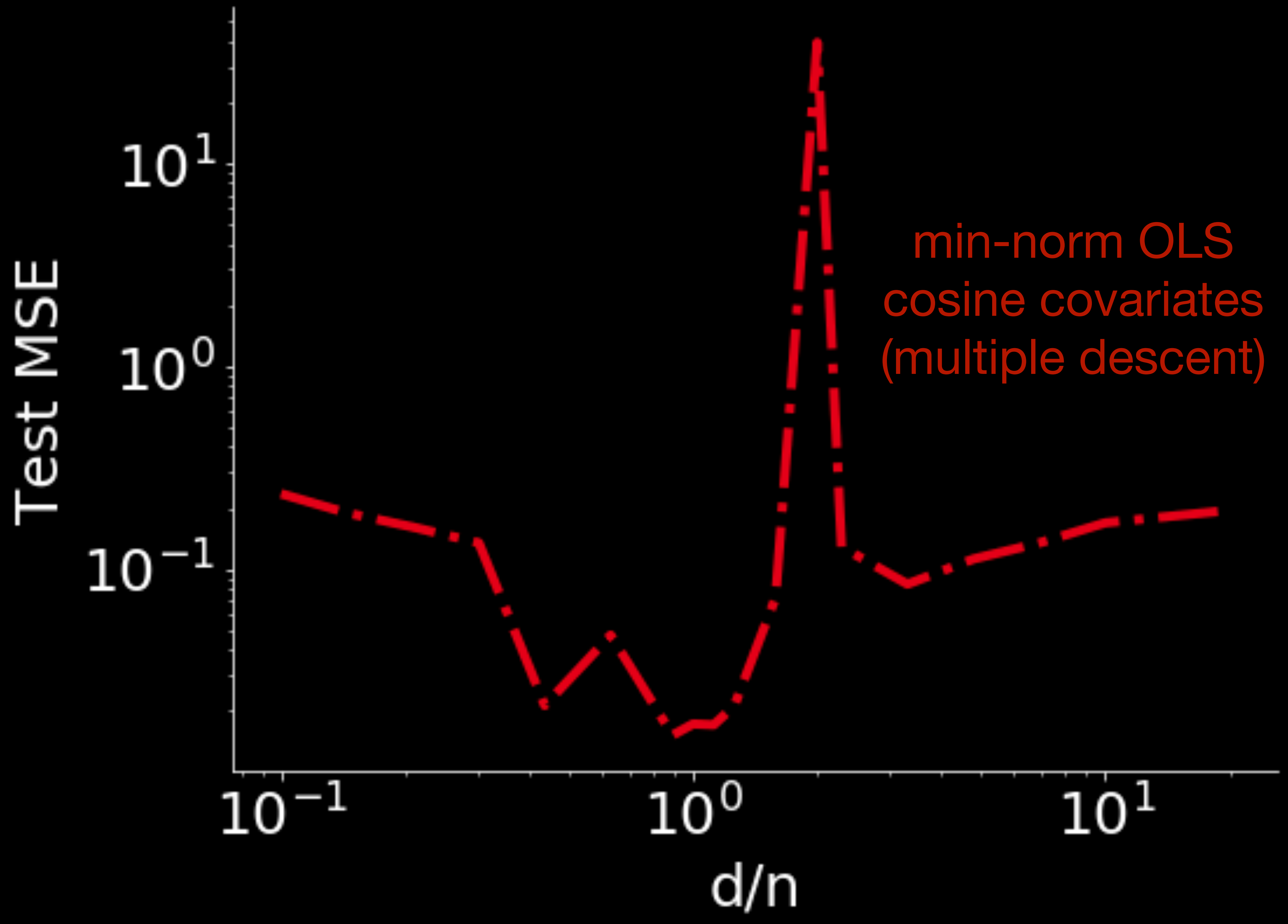
- We should expect a tradeoff *given*
  - some fixed data
  - as the "complexity" of the fitted estimator varies
- Need not observe a tradeoff for
  - poor choice of estimators



d = number of features
n = number of samples

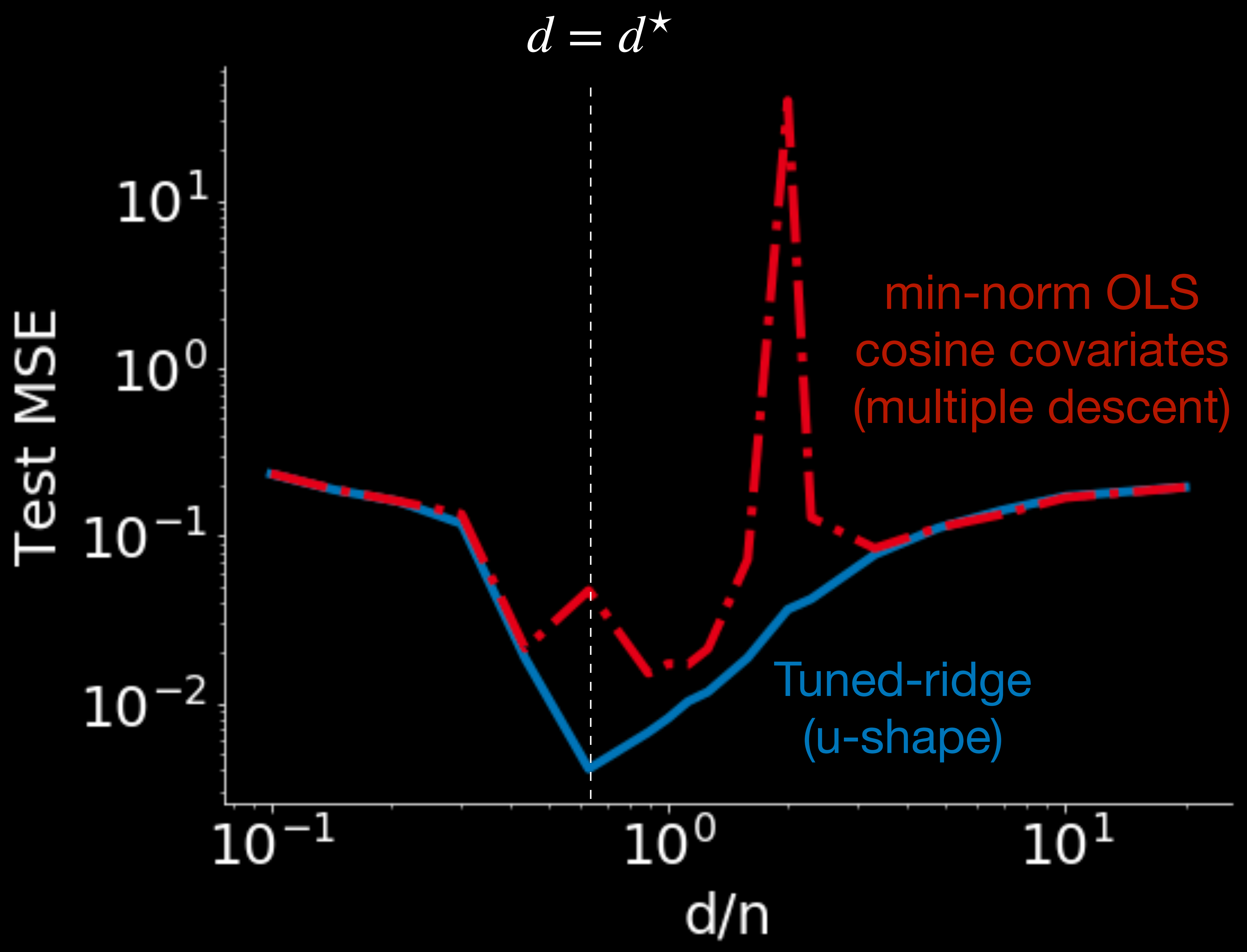
Bias-variance tradeoff: Few things to note..

- We should expect a tradeoff *given*
  - some fixed data
  - as the "complexity" of the fitted estimator varies
- Need not observe a tradeoff for
  - poor choice of estimators



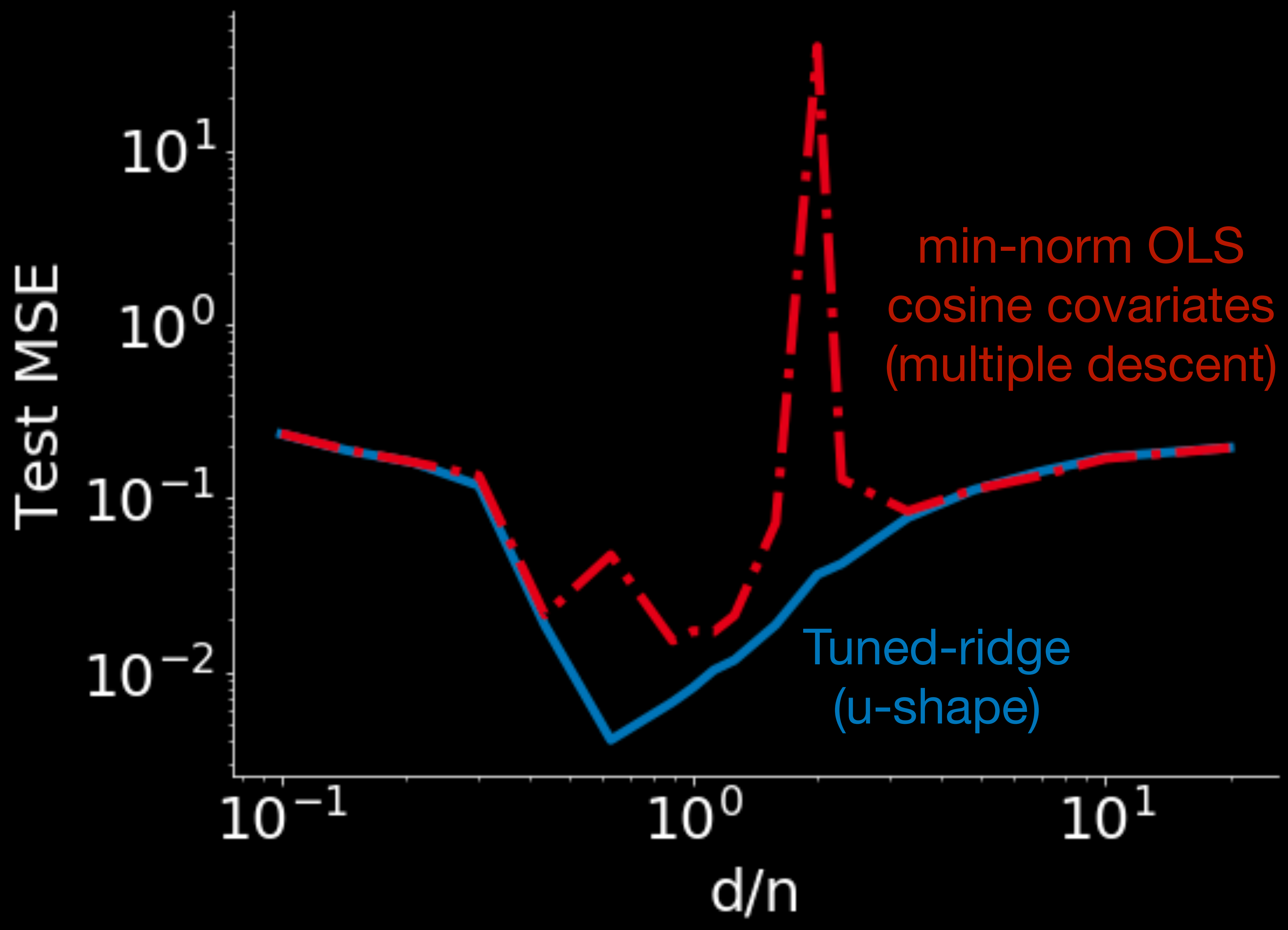
Bias-variance tradeoff: Few things to note..

- We should expect a tradeoff *given*
  - some fixed data
  - as the "complexity" of the fitted estimator varies
- Need not observe a tradeoff for
  - poor choice of estimators



Bias-variance tradeoff: Few things to note..

- We should expect a tradeoff *given*
  - some fixed data
  - as the “complexity” of the fitted estimator varies
- Need not observe a tradeoff for
  - poor choice of estimators
  - poor choice of complexity



Is parameter counting a valid complexity measure?

Complexity: A tricky concept

- A fundamental notion: Kolmogorov’s algorithmic complexity
- Complexity in Statistics and ML
    - Test error ~ Train error + Complexity / n^a
    - useful for model selection
    - x-axis on bias-variance tradeoff—often vaguely defined; parameter count often used

Parameter counting as complexity: Origins
(for linear models)

- Akaike Information Criterion (AIC): $d/2$
- Bayesian information criterion (BIC): $\frac{d}{2} \log n$
- Rademacher complexity: $\mathbb{E}\left[\sup_{\theta \in \Theta} \sum \epsilon_i x_i^\top \theta \right] \sim d$
- Degrees of freedom: trace$(X^\top X) \sim d$
- Vapnik-Chervonenkis dimension: d
- Minimum Description Length complexity: $\frac{d}{2} \log n$ (asymptotically)

OLS achieves the minimax error in low-dimensions of order $\frac{d}{n}$

but in high-dimensions these complexity measure neither work nor theoretically well-justified

this talk:
a data-dependent complexity using minimum description length that is not just parameter count

Minimum Description Length (MDL)

- Another formalism of Occam's razor

  "Choose the model that gives the shortest description of data"

- Developed by Rissanen in the 70s with roots in Kolmogorov's algorithmic complexity, making it computable using Shannon's information theory

- Different forms over the years: Two-stage MDL, mixture MDL, normalized maximum likelihood

Rissanen 76, 80, Barron-Rissanen-Yu 96, Hansen-Yu 02, Grunwald 07

Underlying principle: Probability models as codes

- Model $\longleftrightarrow$ Code
  Good fit $\longleftrightarrow$ Shorter codelength (description)
- Given any distribution Q on the space $\mathscr{Y}$, we can associate a code such that to encode any observation y, we need $\log(1/Q(y))$ bits
  - This interpretation does not need a generative model

Optimal code: With known true model $P^\star$

- When $y \sim P^\star$, the expected code-length when using code Q is given by

$$\mathbb{E}_{y \sim P^\star} \log \left(\frac{1}{Q(y)} \right)$$

Optimal code: With known true model $P^\star$ is $P^\star$

- When $y \sim P^\star$, the expected code-length when using code Q is given by

$$\mathbb{E}_{y \sim P^\star} \log \left(\frac{1}{Q(y)} \right) = \underbrace{\boxed{\mathbb{KL}(P^\star \| Q)}}_{\text{Redundancy}} + H(P^\star)$$

- Minimized when $Q = P^\star$, since redundancy is non-negative

Optimal code: With known true model $P^\star$ is $P^\star$

- When $y \sim P^\star$, the expected code-length when using code Q is given by

$$\mathbb{E}_{y\sim P^\star} \log\left(\frac{1}{Q(y)}\right) = \underbrace{\mathbb{KL}(P^\star \| Q)}_{\text{Redundancy}} + H(P^\star)$$

- Minimized when $Q = P^\star$, since redundancy is non-negative
- $P^\star$ also minimizes the worst-case regret

$$p^\star = \arg\min_q \max_y \left[\log\left(\frac{1}{q(y)}\right) - \log\left(\frac{1}{p^\star(y)}\right)\right] \text{ such that } \int q(z)dz \leq 1$$

Optimal code when $P^\star$ is *unknown*

- Given a class of models $\{p_\theta, \theta \in \Theta\}$, not necessarily containing $p^\star$, consider the generalization of the min-max regret problem:

$$\min_q \max_y \left[\log\left(\frac{1}{q(y)}\right) - \min_\theta \log\left(\frac{1}{p_\theta(y)}\right)\right] \text{ such that } \int q(z)dz \leq 1$$

- Shtarkov (1981) showed that

$$q_{NML}(y) \propto \max_\theta p_\theta(y) \qquad \text{i.e.,} \qquad q_{NML}(y) = \frac{\max_\theta p_\theta(y)}{\int \max_{\theta'} p_{\theta'}(z)dz}$$

solves the optimization problem above where NML stands for ``normalized maximum likelihood"; the normalization makes this a universal (valid for any y) code

NML Complexity

- $\log \int \max_{\theta} p_{\theta}(z) dz$ is both the worst-case and the average regret of
    - Referred to as the NML or Shtarkov complexity for the class $\{p_\theta, \theta \in \Theta\}$
- For d-dimensional parametric-class $\{p_\theta, \theta \in \Theta\}$, Rissanen showed that the Shtarkov complexity simplifies to $\dfrac{d}{2} \log n$ (under regularity conditions)

When $\int \max_{\theta \in \Theta} p_\theta(z) dz$ is infinite, the NML distribution is ill-defined

Issues with NML: Linear model

- Consider linear regression with n samples and d feature:

$$p_\theta(y) = \mathcal{N}(X\theta, \sigma^2 I_n)$$

(we assume X and σ^2 fixed and known)

- Then Q_{NML} is given by

$$q_{NML}(y) \propto \max_\theta p_\theta(y) = p_{\hat{\theta}}(y) = \frac{1}{(2\pi\sigma^2)^{n/2}} \exp\left(-\frac{1}{2\sigma^2}\|X\hat{\theta}_{OLS} - y\|^2\right)$$

the normalization constant $\int \max_\theta p_\theta(z)dz$ is infinite if $\mathcal{Y}$ is unbounded

Fixes for NML

- Truncate the output space $\mathscr{Y}$: [Barron-Rissanen-Yu 96]
- **This talk**: Use regularization and define a modified NML complexity

Ridge luckiness normalized maximum likelihood

- Instead of $\max_{\theta} p_{\theta}(y)$, we use $\max_{\theta} p_{\theta}(y) w_{\theta}$ for some ``luckiness factor" w_{θ}
- Use w_{θ} induced by ridge regularization—-since tuned ridge estimators provide good performance for all range of d:

$$q_{\Lambda}(y) \propto \max_{\theta} \exp\left(-\frac{1}{2\sigma^2}\|X\theta - y\|^2 - \frac{1}{2\sigma^2}\theta^{\top}\Lambda\theta\right)$$

Ridge luckiness normalized maximum likelihood

- Instead of $\max_\theta p_\theta(y)$, we use $\max_\theta p_\theta(y) w_\theta$ for some ``luckiness factor” w_θ
- Use w_θ induced by ridge regularization—-since tuned ridge estimators provide good performance for all range of d:

$$q_\Lambda(y) \propto \exp\left(-\frac{1}{2\sigma^2}\|X\widehat{\theta}_\Lambda - y\|^2 - \frac{1}{2\sigma^2}\widehat{\theta}_\Lambda^\top \Lambda \widehat{\theta}_\Lambda\right)$$

where

$$\widehat{\theta}_\Lambda = \min_\theta \|X\theta - y\|^2 + \theta^\top \Lambda \theta = (X^\top X + \Lambda)^{-1} X^\top y$$

- To derive complexity: Optimize over Λ

LNML codes induced by ridge estimators

- Optimize over the following class

$$\mathcal{Q}_{\text{ridge}} = \{Q_\Lambda, \Lambda = UDU^\top, D \succeq 0\}$$

  where U denotes the eigenvectors of the matrix $X^\top X$

- Need to account for encoding Λ (not present in usual NML): For $\Lambda = U\text{diag}(\lambda_1, \ldots, \lambda_d)U^\top$

$$\mathcal{L}(\Lambda) = \sum \log(\lambda_i/\Delta)$$

  for small enough (discretization) Δ

MDL-COMP: Optimal LNML code in the ridge class

- MDL-COMP captures the best possible redundancy (excess codelength) of Q_{ridge} compared to $P^\star$:

$$\mathcal{R}_{opt} = \frac{1}{n} \min_{Q \in \mathbb{Q}_{ridge}} \mathbb{KL}(P^\star \| Q)$$

$$MDL-COMP = \mathcal{R}_{opt} + \frac{1}{n}\mathcal{L}(\Lambda_{opt})$$

Main result: Analytical MDL-COMP for linear models

- Let ρ_i denote the eigenvalues of $X^\top X$ and let $w_i = U^\top \theta^\star$. When $y \sim \mathcal{N}(X\theta^\star, \sigma^2 I_n)$, then

$$\mathscr{R}_{opt} = \frac{1}{n} \sum_{i=1}^{\min\{n,d\}} \log\left(1 + \frac{\rho_i w_i^2}{\sigma^2}\right)$$

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

Not just parameter count but data dependent—a function of the covariate design, and the interaction between signal and covariates

Unpacking the result for Gaussian X

- When $X \in \mathbb{R}^{n \times d}$ has i.i.d. $\mathcal{N}(0,1/n)$ entries, then

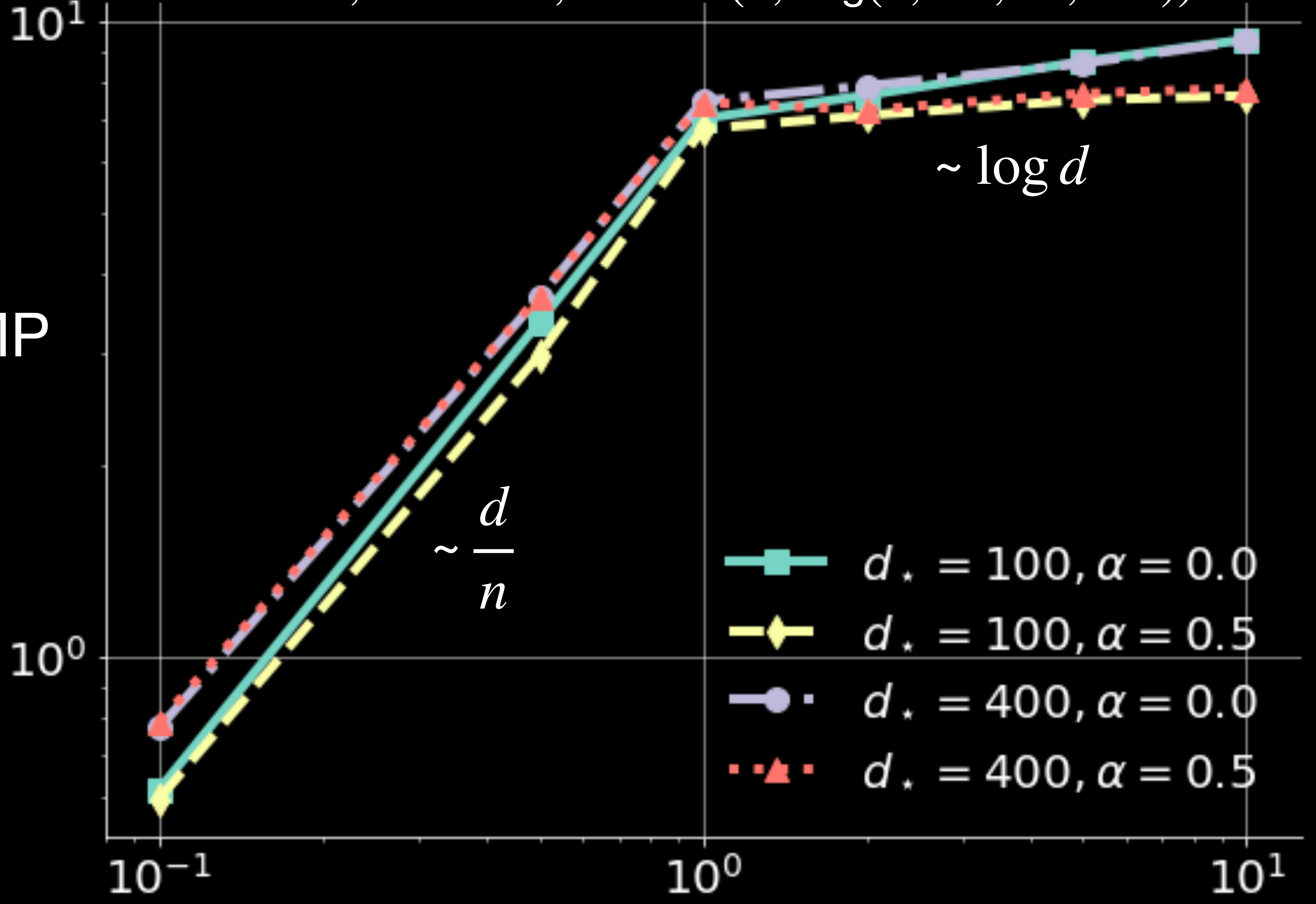
$$\text{MDL-COMP} \approx \begin{cases} \frac{d}{n}\log\left(1+\frac{d_\star}{r^2}\right)+\frac{d}{n}\log\left(\frac{1}{\Delta}\right), & \text{if } d \in [1,d_\star] \\ \frac{d}{n}\log\left(1+\frac{d}{r^2}\right)+\frac{d}{n}\log\left(\frac{1}{\Delta}\right), & \text{if } d \in [d_\star, n] \\ \log\left(\frac{d}{n}+\frac{d}{r^2}\right)+\log\left(\frac{1}{\Delta}\right), & \text{if } d \in [n,\infty) \end{cases}$$

[here $d_\star$ denotes the true dimensionality of $\theta^\star$, and we assume $\mathbb{E}[y\,|\,X] = \tilde{X}\theta^\star$ where $\tilde{X}$ denotes the first $d_\star$ columns of X; and $r^2 = \|\theta^\star\|^2$]

Numerical computation

$n = 200,\ \sigma^2 = 0.1,\ X \sim \mathcal{N}(0, \mathrm{diag}(1, 2^{-\alpha}, \ldots, d^{-\alpha}))$

MDL-COMP



d/n

Consequences for double descent

- Since MDL-COMP (for Gaussian covariates) is monotone in d, the double-descent curve for the OLS or ridge remains qualitatively the same
- The double descent likely due to the estimator choice

Other optimality properties from MDL-COMP

- Λ_{opt} provides optimal regularization for the in-sample risk (a proxy for test error)

$$\Lambda_{opt} = \arg\min_{\Lambda} \mathbb{E}\left(\sum_{i=1}^{n} (x_i^\top \hat{\theta}_\Lambda - x_i^\top \theta^\star)^2 \right)$$

- Q_{opt} corresponds to the min-max optimal code over a family of distributions, i.e.,

$$Q_{opt} = \arg \min_{Q \in \mathcal{Q}_{ridge}} \max_{P \in \mathcal{P}} \mathbb{E}_{y \sim P} \log \left(\frac{1}{q(y)} \right)$$

where $\mathcal{P} = \{P \,|\, E_P(y\,|\,X) = X\theta^\star, \mathrm{Var}(y\,|\,X) \leq \sigma^2 I_n\}$

Extension to kernel methods

- To be added

Can MDL-COMP be useful for practice?

Let's make it computable from data

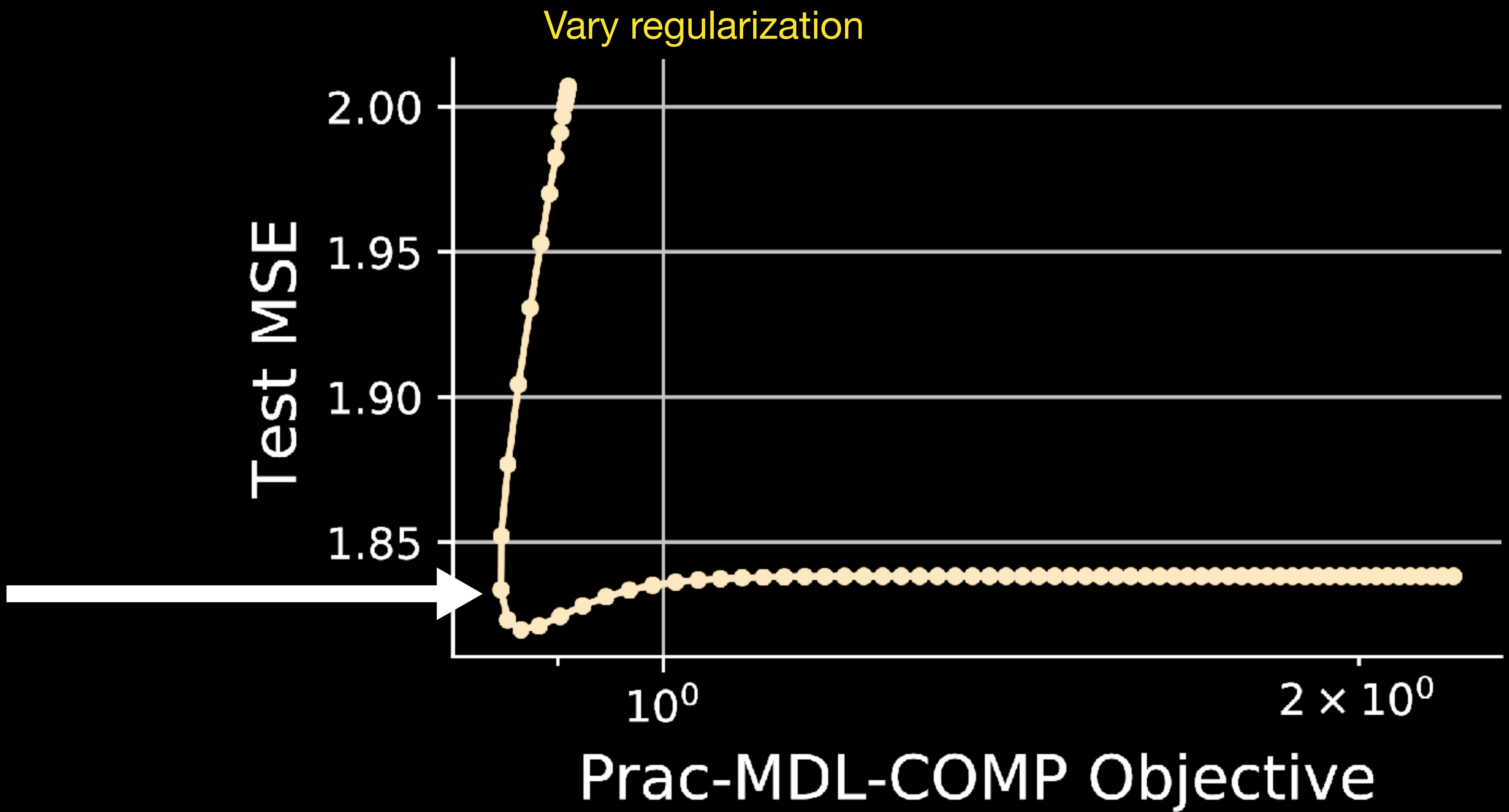
$$\text{Prac-MDL-COMP} = \min_{\lambda} \log\left(\frac{1}{q_\lambda(y)}\right)$$

$$= \min_{\lambda} \left[\frac{\|X\hat{\theta}_\lambda - y\|^2}{2\sigma^2} + \frac{\lambda\|\hat{\theta}_\lambda\|^2}{2\sigma^2} + \sum_{i=1}^{\min\{n,d\}} \log\left(1 + \frac{\rho_i}{\lambda}\right) \right]$$

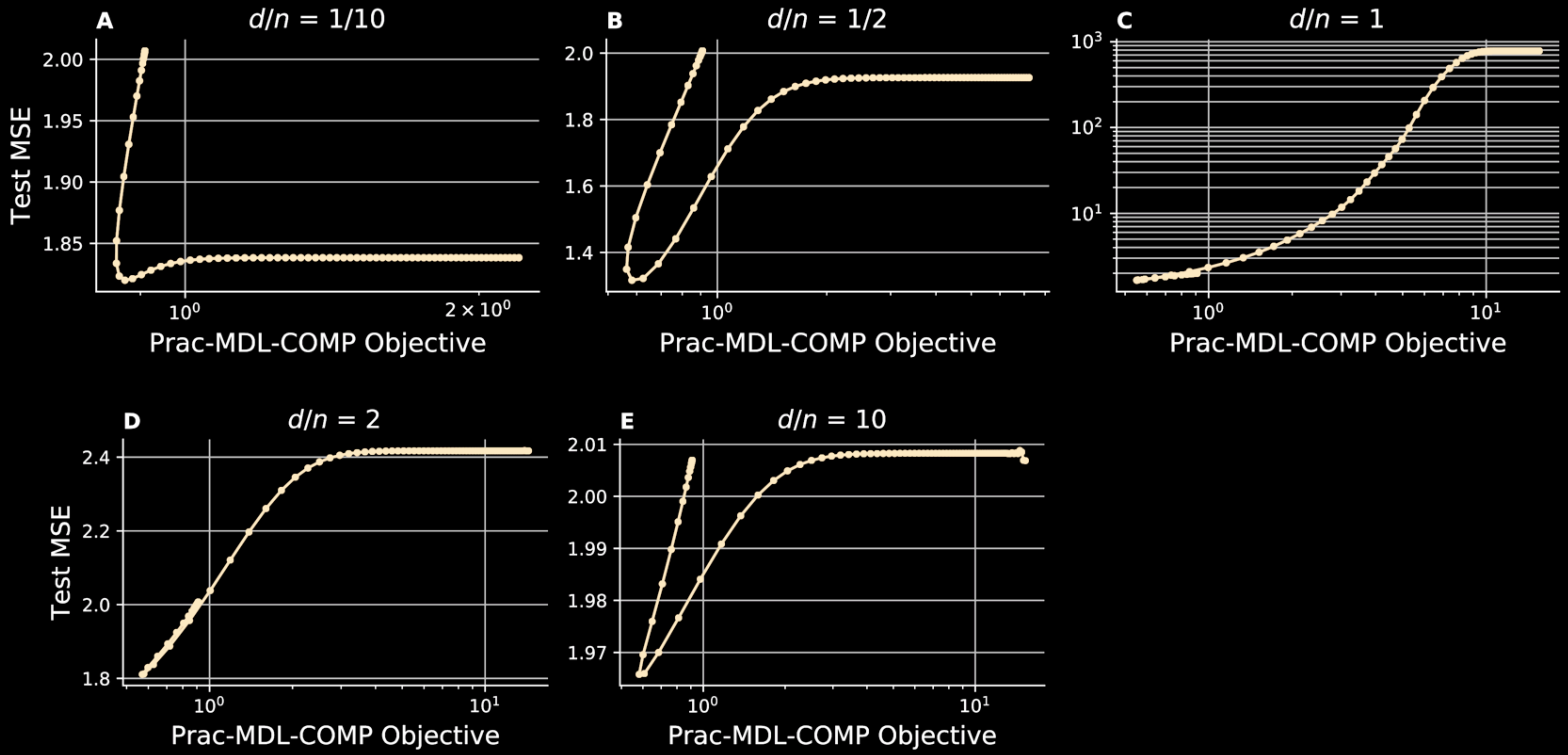
where

$\hat{\theta}_\lambda = (X^\top X + \lambda I)^{-1} X^\top y$ and ρ_i denote the eigenvalues of $X^\top X$

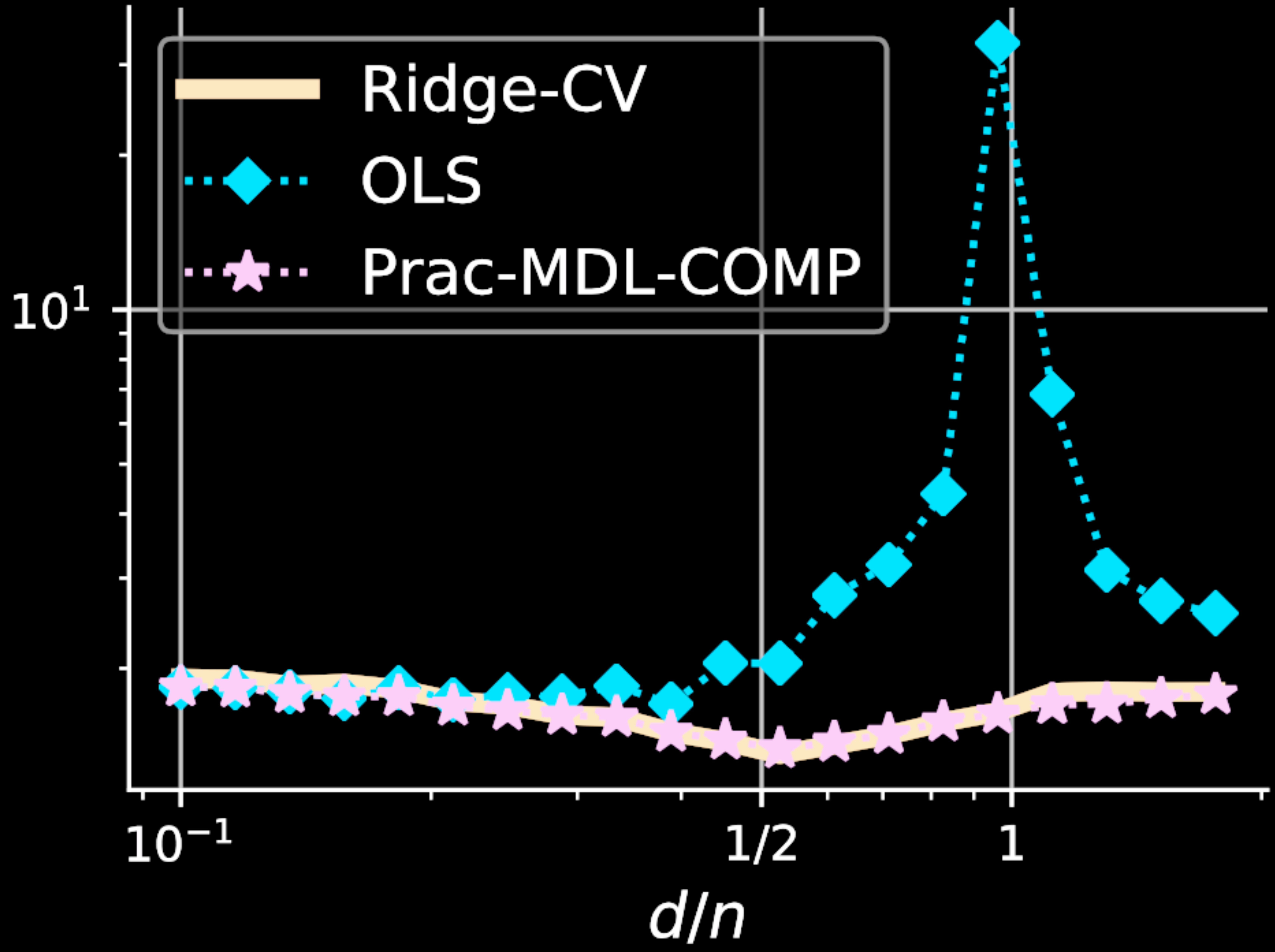
Model selection with Prac-MDL-COMP



Model selection with Prac-MDL-COMP



Look Ma, no peak

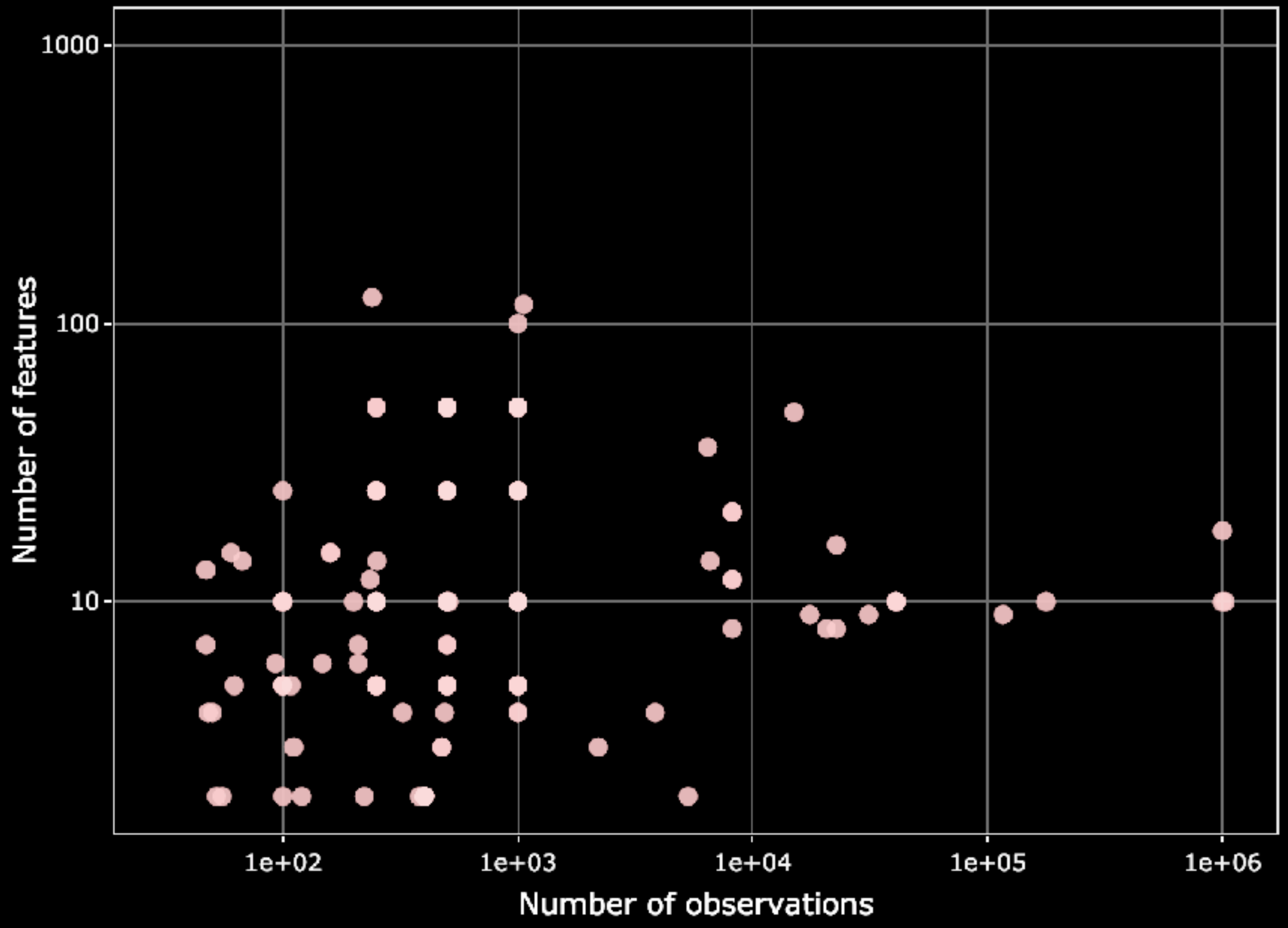


Using Prac-MDL-COMP for hyperparameter tuning

$$\min_{\lambda} \left[\frac{\|X\widehat{\theta}_\lambda - y\|^2}{2\sigma^2} + \frac{\lambda\|\widehat{\theta}_\lambda\|^2}{2\sigma^2} + \sum_{i=1}^{\min\{n,d\}} \log\left(1 + \frac{\rho_i}{\lambda}\right)\right]$$

K-fold computational savings compared to K-fold cross validation

Experiments on PMLB datasets

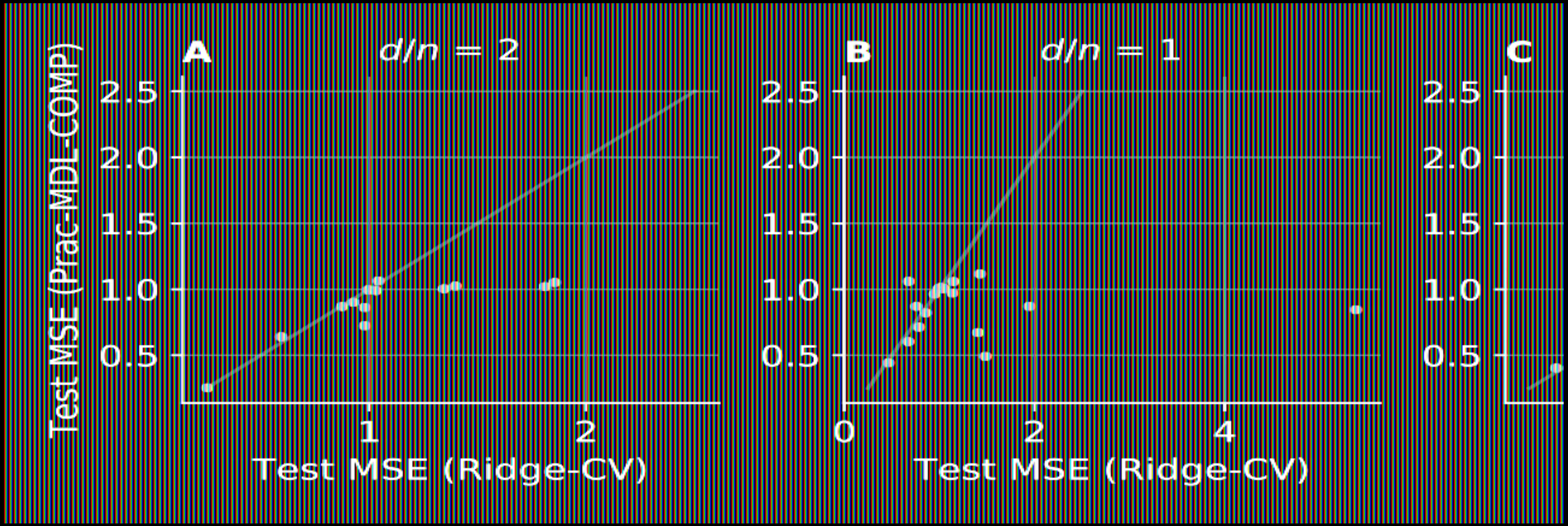


Diverse set of tabular datasets

- Predicting breast cancer from image features
- Predicting automobile prices
- Election results from previous elections

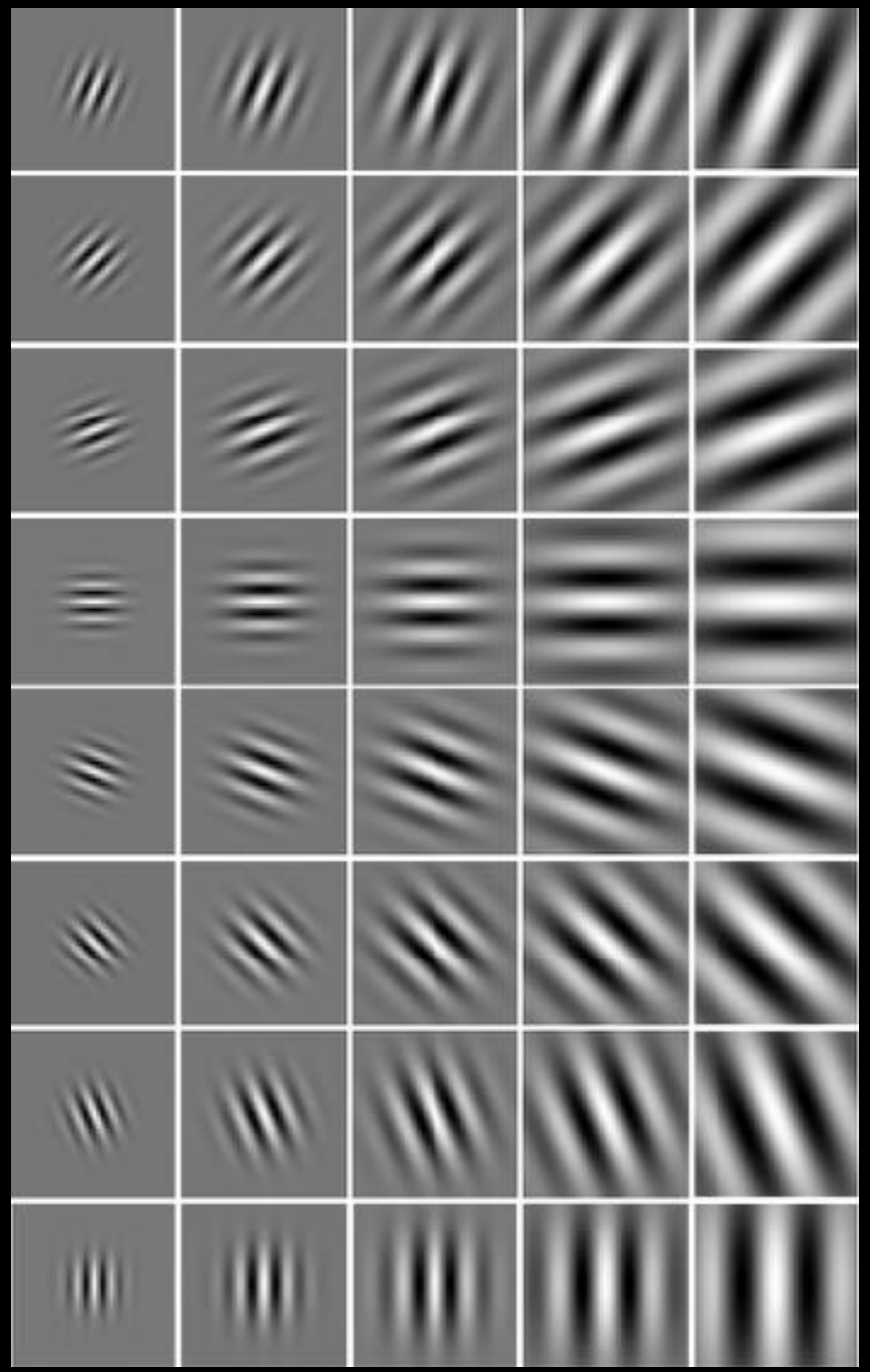
PMLB: a large benchmark suite for machine learning evaluation and comparison
Olson-Cava-Orzechowski-Urbanowicz-Moore 17

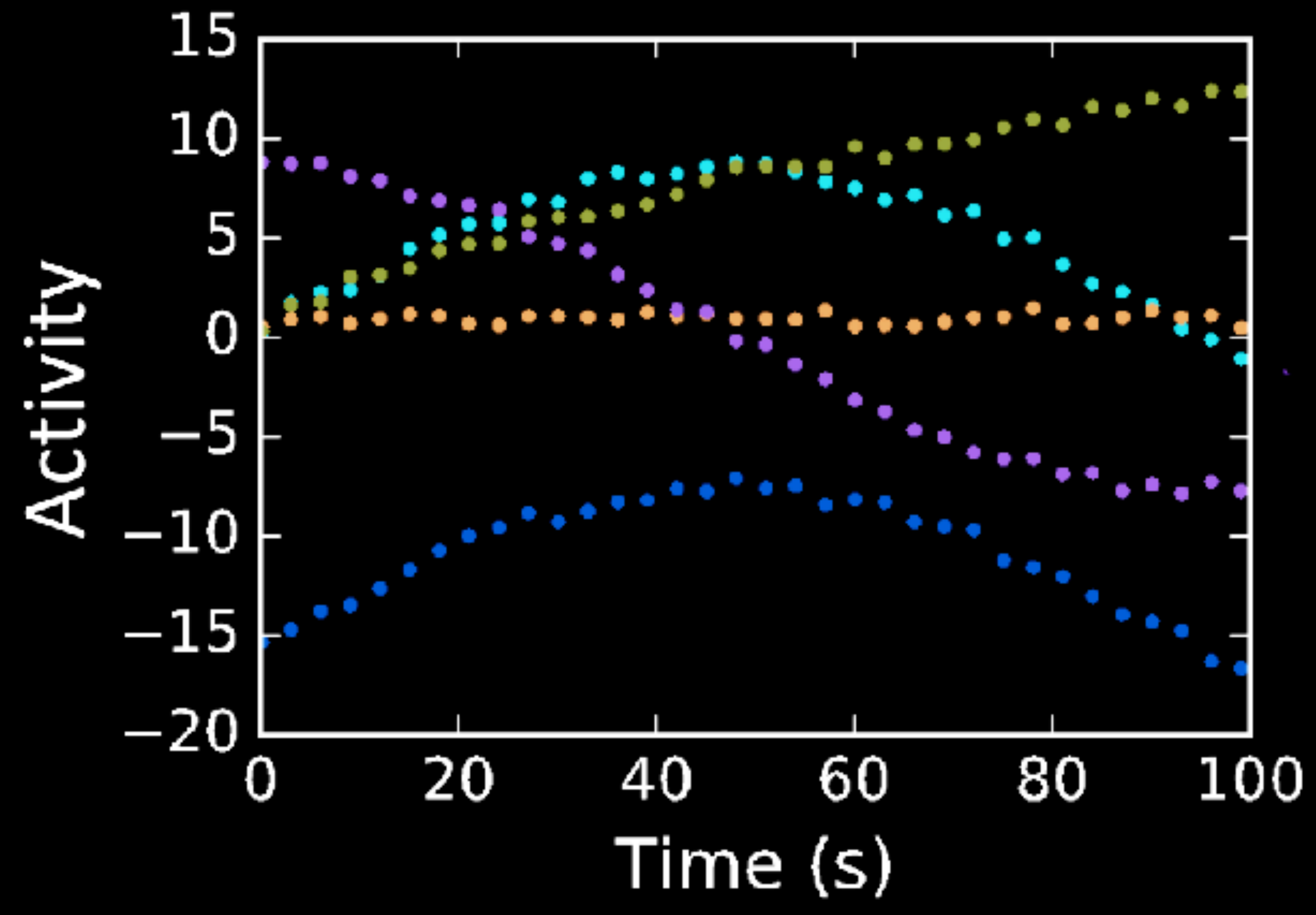
Experiments on PMLB datasets



MDL-COMP better for hyper-parameter tuning in low-data settings

fMRI experimental setup



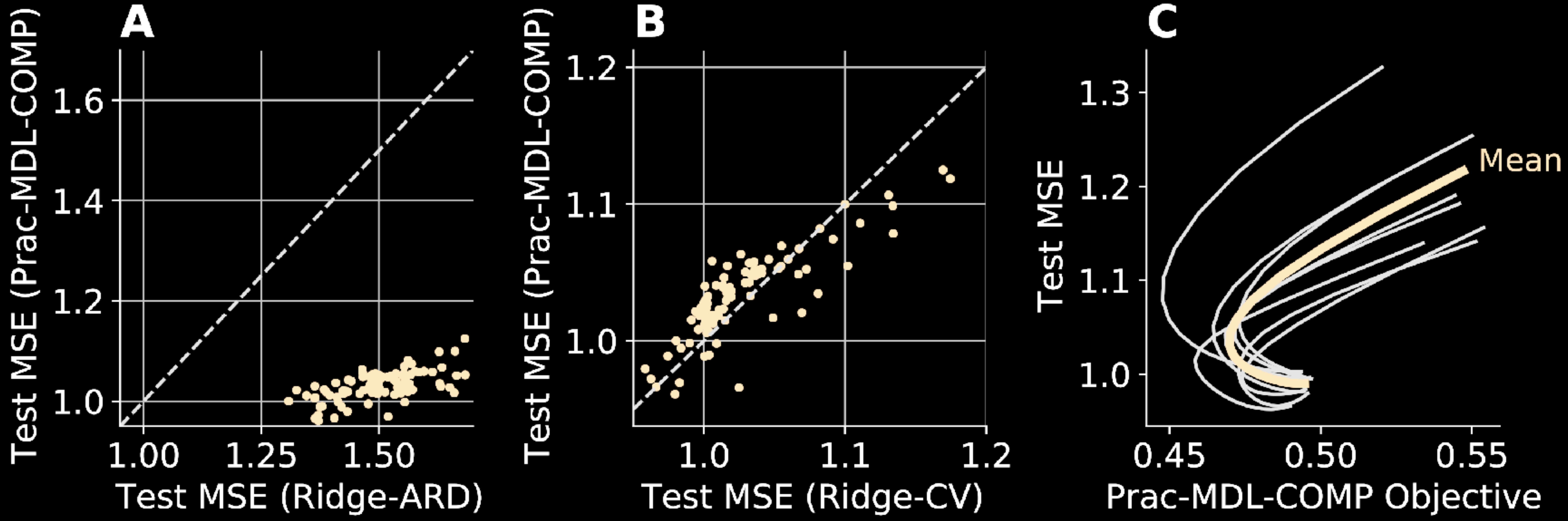


Video clip → Extract gabor features → Predictive model → fMRI voxel responses

Nishimoto-Vu-Naselaris-Benjamini-Yu-Gallant 11

d = 1280
n_train = 7200
n_test = 540

Experiments on fMRI data from 100 voxels



MDL-COMP better than Bayesian-ARD regression, and pretty comparable to CV tuning

Neural tangent kernels (NTK)

NTK approximates neural net with infinite width

Jacot et al. 2018

- Varies with number of layers and nonlinearity

- $K(x, x') = \mathbb{E}_{\theta \sim W}\left[\left\langle \frac{\partial f(\theta, x)}{\partial \theta}, \frac{\partial f(\theta, x')}{\partial \theta} \right\rangle\right]$

- Analytical expressions for simple architectures (e.g., cosine kernel for 2 layer Relu networks)

- Software libraries for computing the kernel for deeper networks

Kernel version of the computation

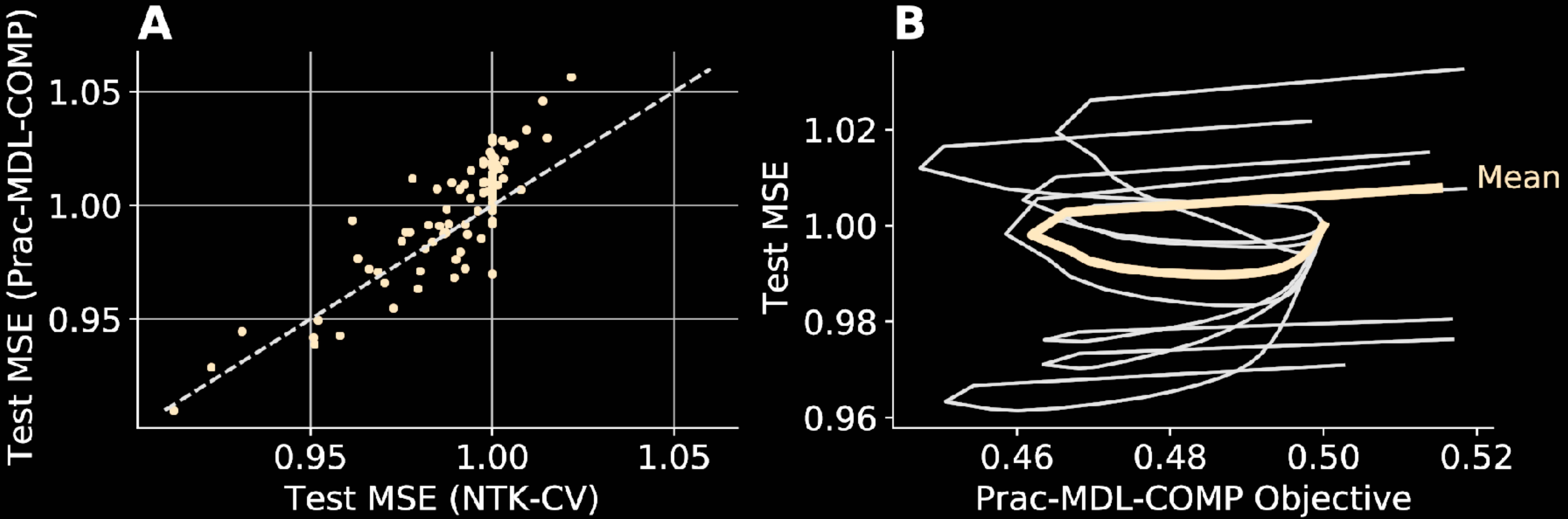
$$\text{Prac-MDL-COMP} = \min_{\lambda} \log\left(\frac{1}{q_\lambda(y)}\right)$$

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

where

$\widehat{\theta}_\lambda = (K+\lambda I)^{-1}y$ and ρ_i denote the eigenvalues of the kernel matrix K

Experiments on NTK with fMRI data voxels



Once again, MDL-COMP pretty comparable to CV tuning

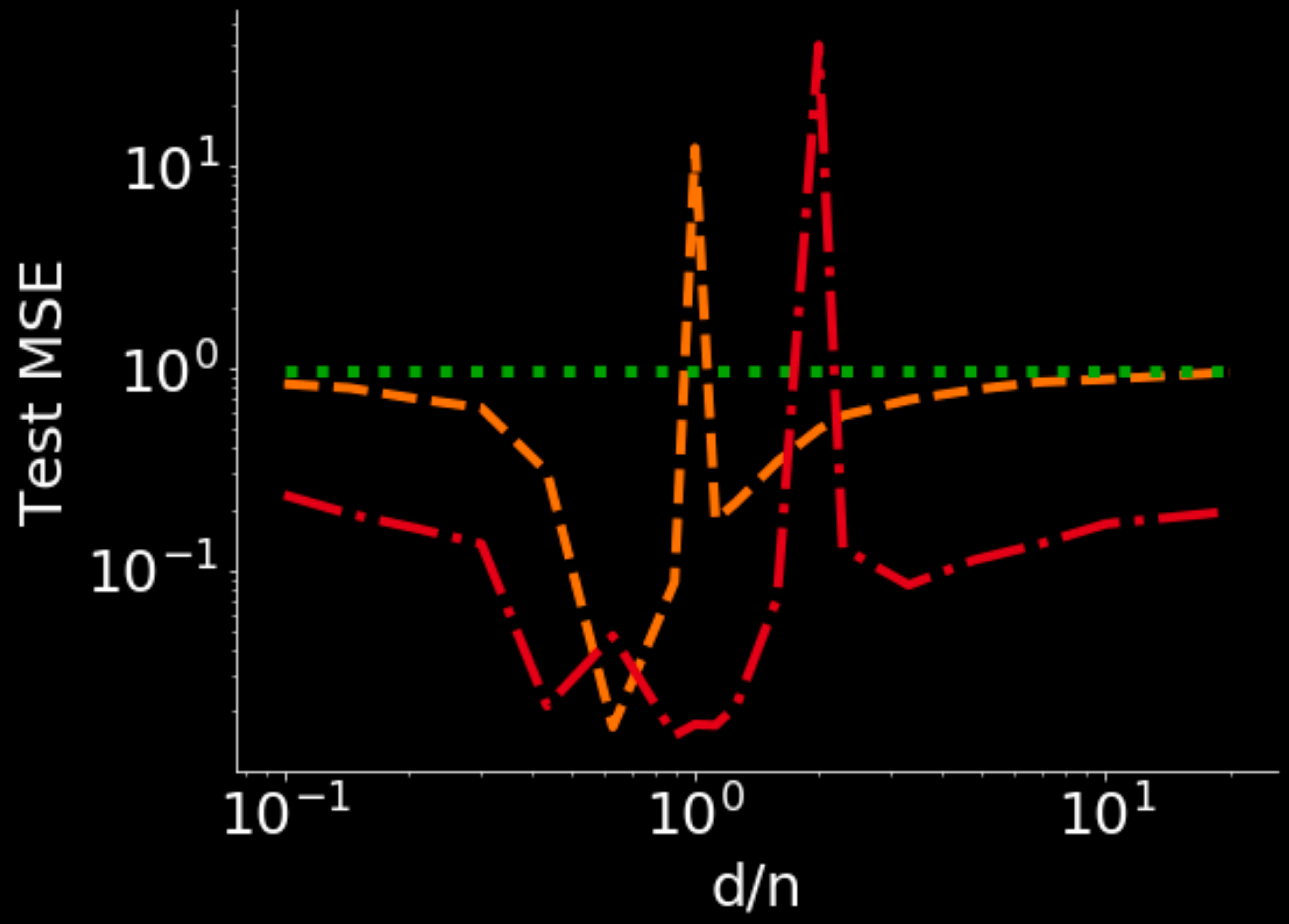
Summary

- MDL-COMP—a modified NML complexity measure using ``optimal” ridge estimators
    - not just parameter count—$\log d$ scaling in overparameterized regime for Gaussian covariates
    - Provides competitive-to-cross validation but computationally more efficient ridge hyper-parameter tuning
- Going forward
    - Establish relationship between MDL-COMP and out-of-sample generalization?
    - Closer to real deep networks: MDL-COMP analytical computations hard for complex models—-Approximations?

Additional slides

Bias-variance tradeoff: Few things to note..

- We should expect a tradeoff *given*
  - some fixed data
  - as the "complexity" of the fitted estimator changes
- Do not expect a tradeoff for
  - poor choice of estimators
  - poor choice of complexity



MDL-COMP for kernel methods

Universal codes induced by kernel ridge regression

- Define the code Q_λ:

$$q_\lambda(y) \propto \exp\left(-\frac{1}{2\sigma^2}\|K\hat{\theta}_\lambda - y\|^2 - \frac{\lambda}{2\sigma^2}\hat{\theta}_\lambda^\top K \hat{\theta}_\lambda\right)$$

where

$$\hat{\theta}_\lambda = \min_\theta \|K\theta - y\|^2 + \lambda\theta^\top K\theta = (K + \lambda I)^{-1} y$$

- This choice comes from kernel ridge regression:

$$\min_{f \in \mathscr{H}} \sum_{i=1}^{n} (f(x_i) - y_i)^2 + \lambda \|f\|_{\mathscr{H}}^2$$

Kernel ridge regression

- One can show that for the optimization proboem

$$\min_{f \in \mathscr{H}} \sum_{i=1}^{n} (f(x_i) - y_i)^2 + \lambda \|f\|_{\mathscr{H}}^2,$$

it suffices to consider the functions of the form

$$f = \sum_{i=1}^{n} \theta_i K(x_i, \cdot\,),$$

and this leads to the kernel ridge regression problem in the previous slide

MDL-COMP for kernel regression

- Let ρ_i denote the eigenvalues of the kernel matrix $(K(x_i, x_j))_{i,j=1}^n$ and suppose $y \sim \mathcal{N}(f^\star(X), \sigma^2 I_n)$ for some $f^\star$ in RKHS of K, then

$$\mathcal{R}_{opt} \leq \frac{1}{2n}\left[\min_{\lambda} \frac{\lambda \|f^\star\|_{\mathcal{H}}^2}{\sigma^2} + \sum_{i=1}^{n} \log\left(1 + \frac{\rho_i}{\lambda}\right)\right]$$

(no easy closed-form)

- Since there is only a single hyper-parameter, we can directly take

$$MDL - COMP = \mathcal{R}_{opt}$$

Unpacking MDL-COMP for Sobolev kernels

- For Sobolev kernel of smoothness α, the eigenvalues decay like $\rho_i \sim i^{-2\alpha}$, and one can derive

$$\mathscr{R}_{opt} \leq C \left(\frac{\|f^\star\|_{\mathscr{H}}^2}{\sigma^2} \right)^{\frac{1}{2\alpha+1}} \cdot n^{-\frac{2\alpha}{2\alpha+1}}$$

Proofs

Proof sketch for linear models

$$\mathcal{D}_{\mathrm{KL}}(\mathbb{P}_{\theta_\star} \,\|\, \mathbb{Q}_{\boldsymbol{\Lambda}}) = \mathbb{E}_{\mathbf{y}}\left[\log \frac{p(\mathbf{y};\mathbf{X},\theta_\star)}{q_{\boldsymbol{\Lambda}}(\mathbf{y})}\right]$$

$$= \mathbb{E}_{\mathbf{y}}\left[\log\left(\frac{\frac{1}{(2\pi\sigma^2)^{n/2}}\exp\left(-\frac{1}{2\sigma^2}\|\mathbf{y}-\mathbf{X}\theta_\star\|^2\right)}{\frac{1}{\mathrm{C}_{\boldsymbol{\Lambda}}(2\pi\sigma^2)^{n/2}}\exp\left(-\frac{1}{2\sigma^2}\|\mathbf{y}-\mathbf{X}\widehat{\theta}\|^2-\frac{1}{2\sigma^2}\widehat{\theta}^\top\boldsymbol{\Lambda}\widehat{\theta}\right)}\right)\right]$$

$$= -\underbrace{\mathbb{E}_{\mathbf{y}}\left[\frac{1}{2\sigma^2}\|\mathbf{y}-\mathbf{X}\theta_\star\|^2\right]}_{=:T_1} + \underbrace{\mathbb{E}\left[\frac{1}{2\sigma^2}\|\mathbf{y}-\mathbf{X}\widehat{\theta}\|^2+\frac{1}{2\sigma^2}\widehat{\theta}^\top\boldsymbol{\Lambda}\widehat{\theta}\right]}_{=:T_2} + \underbrace{\log \mathrm{C}_{\boldsymbol{\Lambda}}}_{=:T_3}. \tag{31}$$

$$T_2 = \frac{(n - \min\{n,d\})}{2} + \frac{1}{2} \sum_{i=1}^{\min\{n,d\}} \frac{(\rho_i w_i^2/\sigma^2 + 1)\lambda_i}{\lambda_i + \rho_i}, \quad \text{and} \tag{33a}$$

$$T_3 = \frac{1}{2} \sum_{i=1}^{\min\{n,d\}} \log\left(\frac{\rho_i + \lambda_i}{\lambda_i}\right) \tag{33b}$$

$$\frac{1}{n}\mathcal{D}_{\mathrm{KL}}(\mathbb{P}_{\theta_\star}\,\|\,\mathbb{Q}_{\mathbf{\Lambda}}) = T_1 + T_2 + T_3$$

$$= -\frac{\min\{n,d\}}{2n} + \frac{1}{2n}\sum_{i=1}^{\min\{n,d\}} \underbrace{\left(\frac{(\rho_i w_i^2/\sigma^2+1)\lambda_i}{\lambda_i+\rho_i} + \log\left(\frac{\rho_i+\lambda_i}{\lambda_i}\right)\right)}_{=:f_i(\lambda_i)}. \tag{34}$$

Finally to compute the $\mathcal{R}_{\mathrm{opt}}$ (32), we need to minimize the KL-divergence (34) where we note the objective depends merely on $\lambda_1,\ldots,\lambda_{\min\{n,d\}}$. We note that the objective (RHS of equation (34)) is separable in each term λ_i. We have

$$f_i'(\lambda_i) = 0 \iff -\frac{(\rho_i w_i^2/\sigma^2+1)}{(1+\rho_i/\lambda_i)^2} + \frac{1}{1+\rho_i/\lambda_i} = 0 \iff \lambda_i^{\mathrm{opt}} = \frac{\sigma^2}{w_i^2}. \tag{35}$$

Proof sketch for the result with Gaussian X

- When $X \in \mathbb{R}^{n \times d}$ has i.i.d. $\mathcal{N}(0,1/n)$ entries, then for $X^\top X = U\text{diag}(\rho_1, \ldots, \rho_d)U^\top$
  - The matrix U has uniform distribution over the set of $d \times d$ orthonormal matrices and hence for any fixed $\theta^\star$, the coordinates of $w = U^\top \theta^\star$ are identically distributed, and we can use the approximation $w_i^2 \approx \dfrac{\|\theta^\star\|^2}{d}$

Proof sketch for the result with Gaussian X

- When $X \in \mathbb{R}^{n \times d}$ has i.i.d. $\mathcal{N}(0,1/n)$ entries, then for $X^\top X = U\text{diag}(\rho_1, \ldots, \rho_d)U^\top$
  - The eigenvalues ρ_i follow Marcenko-Pastur Law with the following approximation
  - $d \ll n, \quad X^\top X \approx I_d, \quad \rho_i \approx 1$
  - $d > n, \quad X^\top X \approx \begin{bmatrix} \frac{d}{n} I_n & 0 \\ 0 & 0 \end{bmatrix}, \quad \rho_i \begin{cases} \approx \frac{d}{n}, & i \leq n \\ = 0, & i > n \end{cases}$

Marcenko-Pastur 67, Silverstein 95, Tulino-Verdu 04

Two-stage MDL

Two-stage MDL

- Consider a parametric class of codes $\{p_\theta, \theta \in \Theta\}$, and then use the valid codelength for any fixed p_θ

$$\log\left(\frac{1}{p_\theta(y)}\right)$$

- Minimizing this codelength is same as MLE over the given parametric class
- But the choice of $\widehat{\theta}$ varies with y, so need to account for the codelength needed to transmit the value of $\widehat{\theta}$

Two-stage MDL

- Thus the overall codelength is

$$\log\left(\frac{1}{p_{\hat{\theta}}(y)}\right) + \frac{d}{2}\log n$$

Codelength for data

Codelength for d-dimensional parameter upto $1/\sqrt{n}$ resolution

- For a fixed parametric class, same as MLE (since the second term is constant)
- For a family of parametric classes, same as BIC procedure (model selection)

MDL-COMP vs Cross-validation

- For $n \times d$ covariates, for each value of λ, the computational costs are
  - **K-fold cross-validation**: K x OLS solver = $K \times (nd^2 + \min(n^3, d^3))$
  - **Prac-MDL-COMP**: 1 x SVD solver = $nd^2 + n^2 d$

Prac-MDL-COMP provides a proxy for complexity and saves K-fold computation!

Issues with NML

Issues with NML: Linear model

- Then Q_{NML} is given by

$$q_{NML}(y) \propto \max_{\theta} p_{\theta}(y) = p_{\widehat{\theta}}(y) = \frac{1}{(2\pi\sigma^2)^{n/2}} \exp\left(-\frac{1}{2\sigma^2}\|X\widehat{\theta} - y\|^2\right)$$

$$\widehat{\theta} = \arg\max p_{\theta}(y) = \arg\min_{\theta} \|X\theta - y\|^2 = \widehat{\theta}_{OLS}$$

(We can use min-norm OLS when $d > n$)

Issues with NML: Linear model

- If $\mathscr{Y}$ is not compact (even when d<n)

$$\int \frac{1}{(2\pi\sigma^2)^{n/2}} \exp\left(-\frac{1}{2\sigma^2}\|X\hat{\theta} - y\|^2\right) dy = \infty$$

- Easiest to see when $d > n$ so that $X\hat{\theta} = y$, and we have

$$\int \frac{1}{(2\pi\sigma^2)^{n/2}} \exp\left(-\frac{1}{2\sigma^2}\|X\hat{\theta} - y\|^2\right) dy = \int_{\mathbb{R}^n} \frac{1}{(2\pi\sigma^2)^{n/2}} dy = \infty$$

Grunwald 07