Model Regularization

Overfitting, Bias-variance decomposition, L1 and L2 regularization, probabilistic interpretation

Machine Learning and Data Mining, 2023

Presented by: Abdalaziz Al-Maeeni

Prepared by: Artem Maevskiy

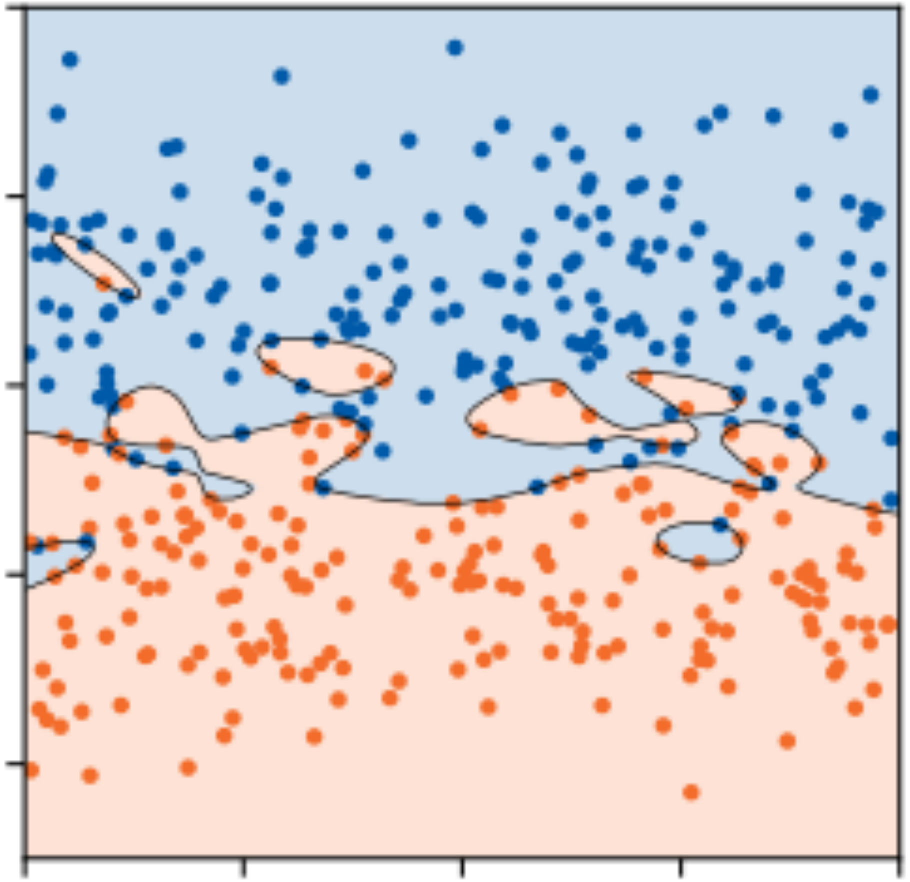
National Research University Higher School of Economics

LAMBDA • HSE

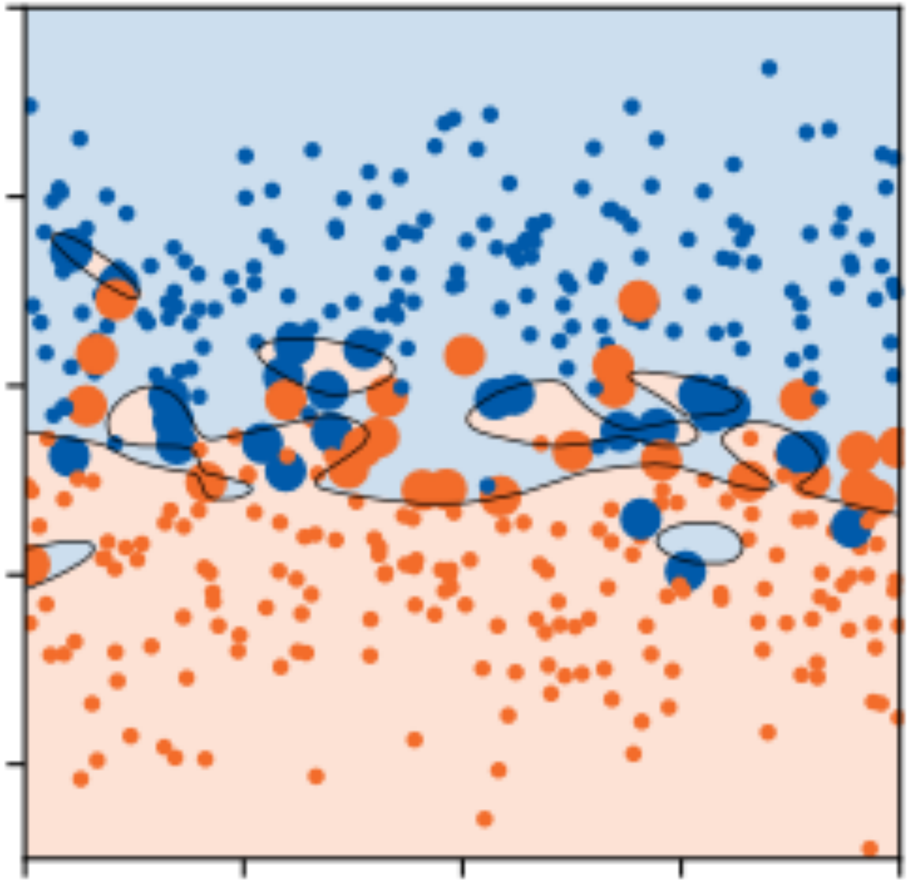
September 30, 2023

The problem of overfitting

Overfitting in classification



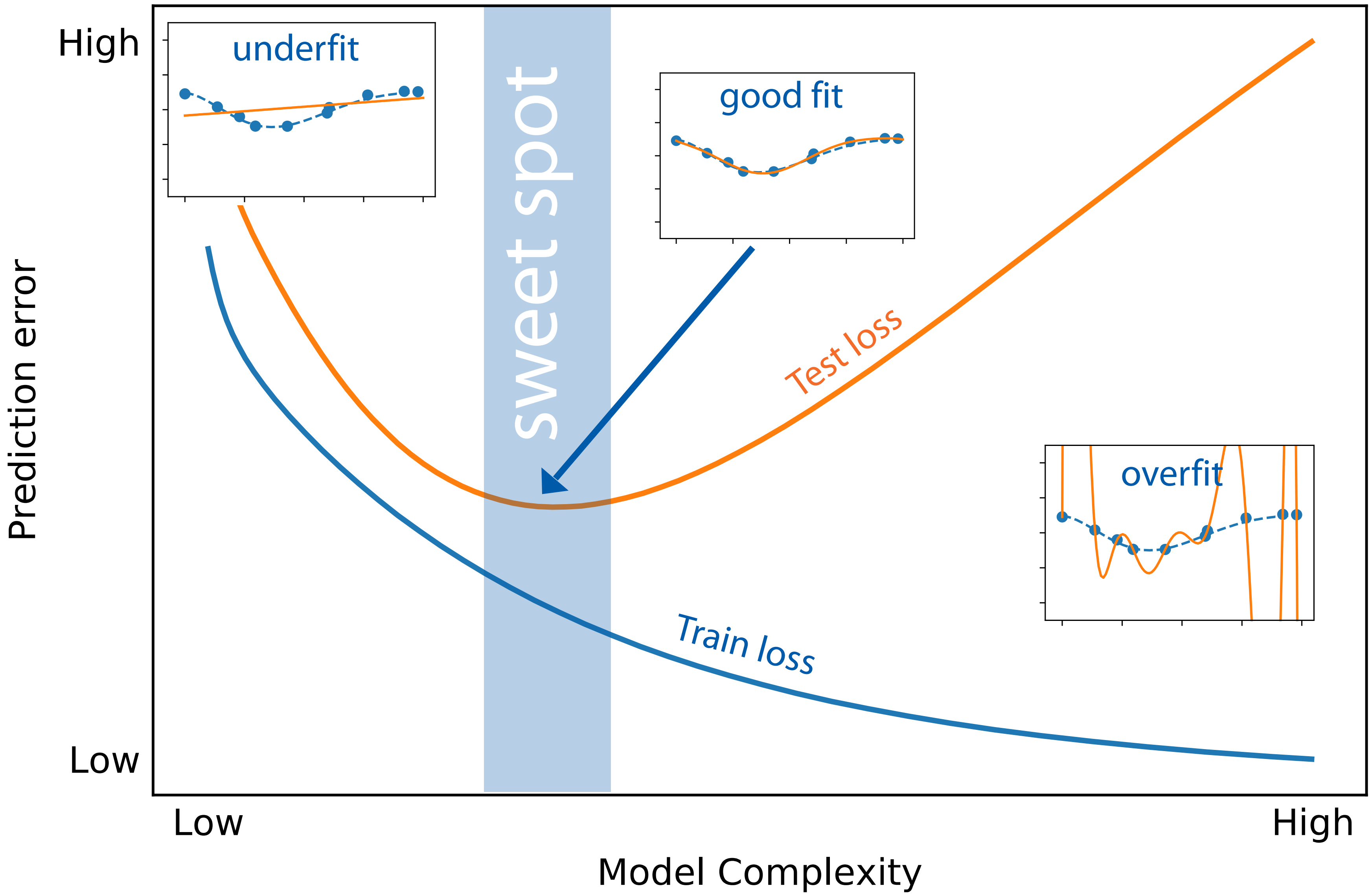
Training set



Test set

Large points =
classification error

How to check whether a model is good?



Check the loss on the test data – i.e. data that the learning algorithm hasn't seen

The goal is to find the right level of limitations – not too strict, not too loose

Prediction error decomposition

Prediction error decomposition

Assume there's the following (unknown) relation between the features and targets:

$$y = f(x) + \varepsilon$$

where ε is some random noize:

$$\mathbb{E}[\varepsilon] = 0$$

$$\mathbb{D}[\varepsilon] = \sigma_\varepsilon^2$$

Prediction error decomposition

Assume there's the following (unknown) relation between the features and targets:

$$y = f(x) + \varepsilon$$

where ε is some random noize:

$$\mathbb{E}[\varepsilon] = 0$$

$$\mathbb{D}[\varepsilon] = \sigma_\varepsilon^2$$

Let's denote our training set as τ.

We want to study the expected squared error for the model $\hat{f}_\tau$ trained on it:

$$\mathrm{exp.sq.err}(x) = \mathop{\mathbb{E}}_{\tau, y\ x}\left[\left(\hat{f}_\tau(x) - y\right)^2\right]$$

Prediction error decomposition

$$\text{exp.sq.err}(x) = \mathop{\mathbb{E}}_{\tau, y \mid x}\left[\left(\hat{f}_\tau(x) - y\right)^2\right]$$

$$= \mathop{\mathbb{E}}_{\tau, y \mid x}\left[\left(\quad \hat{f}_\tau(x) \qquad\qquad\qquad\qquad\qquad\qquad - y \quad \right)^2\right]$$

Prediction error decomposition

$$\mathrm{exp.sq.err}(x) = \mathop{\mathbb{E}}_{\tau,y\,x}\left[\left(\hat{f}_\tau(x) - y\right)^2\right]$$

$$= \mathop{\mathbb{E}}_{\tau,y\,x}\left[\left(\hat{f}_\tau(x) - \mathop{\mathbb{E}}_{\tau'}\left[\hat{f}_{\tau'}(x)\right] + \mathop{\mathbb{E}}_{\tau'}\left[\hat{f}_{\tau'}(x)\right] - y\right)^2\right]$$

Prediction of the
“expected model”

Prediction error decomposition

$$\text{exp.sq.err}(x) = \mathop{\mathbb{E}}_{\tau, y \mid x}\left[\left(\hat{f}_\tau(x) - y\right)^2\right]$$

$$= \mathop{\mathbb{E}}_{\tau, y \mid x}\left[\left(\hat{f}_\tau(x) - \mathop{\mathbb{E}}_{\tau'}\left[\hat{f}_{\tau'}(x)\right] + \mathop{\mathbb{E}}_{\tau'}\left[\hat{f}_{\tau'}(x)\right] - f(x) + f(x) - y\right)^2\right]$$

Ground truth
(without the noise)

Prediction error decomposition

$$\text{exp.sq.err}(x) = \mathop{\mathbb{E}}_{\tau,y\,x}\left[\left(\hat{f}_\tau(x) - y\right)^2\right]$$

$$= \mathop{\mathbb{E}}_{\tau,y\,x}\left[\left(\left(\hat{f}_\tau(x) - \mathop{\mathbb{E}}_{\tau'}\left[\hat{f}_{\tau'}(x)\right]\right) + \left(\mathop{\mathbb{E}}_{\tau'}\left[\hat{f}_{\tau'}(x)\right] - f(x)\right) + \left(f(x) - y\right)\right)^2\right]$$

(grouping the terms, then expanding the square)

Prediction error decomposition

$$\text{exp.sq.err}(x) = \underset{\tau,y\ x}{\mathbb{E}}\left[\left(\hat{f}_\tau(x) - y\right)^2\right]$$

$$= \underset{\tau,y\ x}{\mathbb{E}}\left[\left(\left(\hat{f}_\tau(x) - \underset{\tau'}{\mathbb{E}}\left[\hat{f}_{\tau'}(x)\right]\right) + \left(\underset{\tau'}{\mathbb{E}}\left[\hat{f}_{\tau'}(x)\right] - f(x)\right) + \left(f(x) - y\right)\right)^2\right]$$

(easy to show that all the cross term expectations are 0)

$$= \underset{\tau}{\mathbb{E}}\left[\left(\hat{f}_\tau(x) - \underset{\tau'}{\mathbb{E}}\left[\hat{f}_{\tau'}(x)\right]\right)^2\right] + \left(\underset{\tau'}{\mathbb{E}}\left[\hat{f}_{\tau'}(x)\right] - f(x)\right)^2 + \underset{y\ x}{\mathbb{E}}\left[\left(f(x) - y\right)^2\right]$$

Variance of the model

i.e. how "unstable" the model is wrt the noise in the training data

Prediction error decomposition

$$\exp.\mathrm{sq}.\mathrm{err}(x) = \mathop{\mathbb{E}}_{\tau,y\,x}\left[\left(\hat{f}_\tau(x) - y\right)^2\right]$$

$$= \mathop{\mathbb{E}}_{\tau,y\,x}\left[\left(\left(\hat{f}_\tau(x) - \mathop{\mathbb{E}}_{\tau'}\left[\hat{f}_{\tau'}(x)\right]\right) + \left(\mathop{\mathbb{E}}_{\tau'}\left[\hat{f}_{\tau'}(x)\right] - f(x)\right) + \left(f(x) - y\right)\right)^2\right]$$

(easy to show that all the cross term expectations are 0)

$$= \mathop{\mathbb{E}}_{\tau}\left[\left(\hat{f}_\tau(x) - \mathop{\mathbb{E}}_{\tau'}\left[\hat{f}_{\tau'}(x)\right]\right)^2\right] + \left(\mathop{\mathbb{E}}_{\tau'}\left[\hat{f}_{\tau'}(x)\right] - f(x)\right)^2 + \mathop{\mathbb{E}}_{y\,x}\left[\left(f(x) - y\right)^2\right]$$

how much the "expected model" differs from the ground truth

Squared bias

Prediction error decomposition

$$\mathrm{exp.sq.err}(x) = \mathop{\mathbb{E}}_{\tau,y\;x}\left[\left(\hat{f}_\tau(x) - y\right)^2\right]$$

$$= \mathop{\mathbb{E}}_{\tau,y\;x}\left[\left(\left(\hat{f}_\tau(x) - \mathop{\mathbb{E}}_{\tau'}\left[\hat{f}_{\tau'}(x)\right]\right) + \left(\mathop{\mathbb{E}}_{\tau'}\left[\hat{f}_{\tau'}(x)\right] - f(x)\right) + \left(f(x) - y\right)\right)^2\right]$$

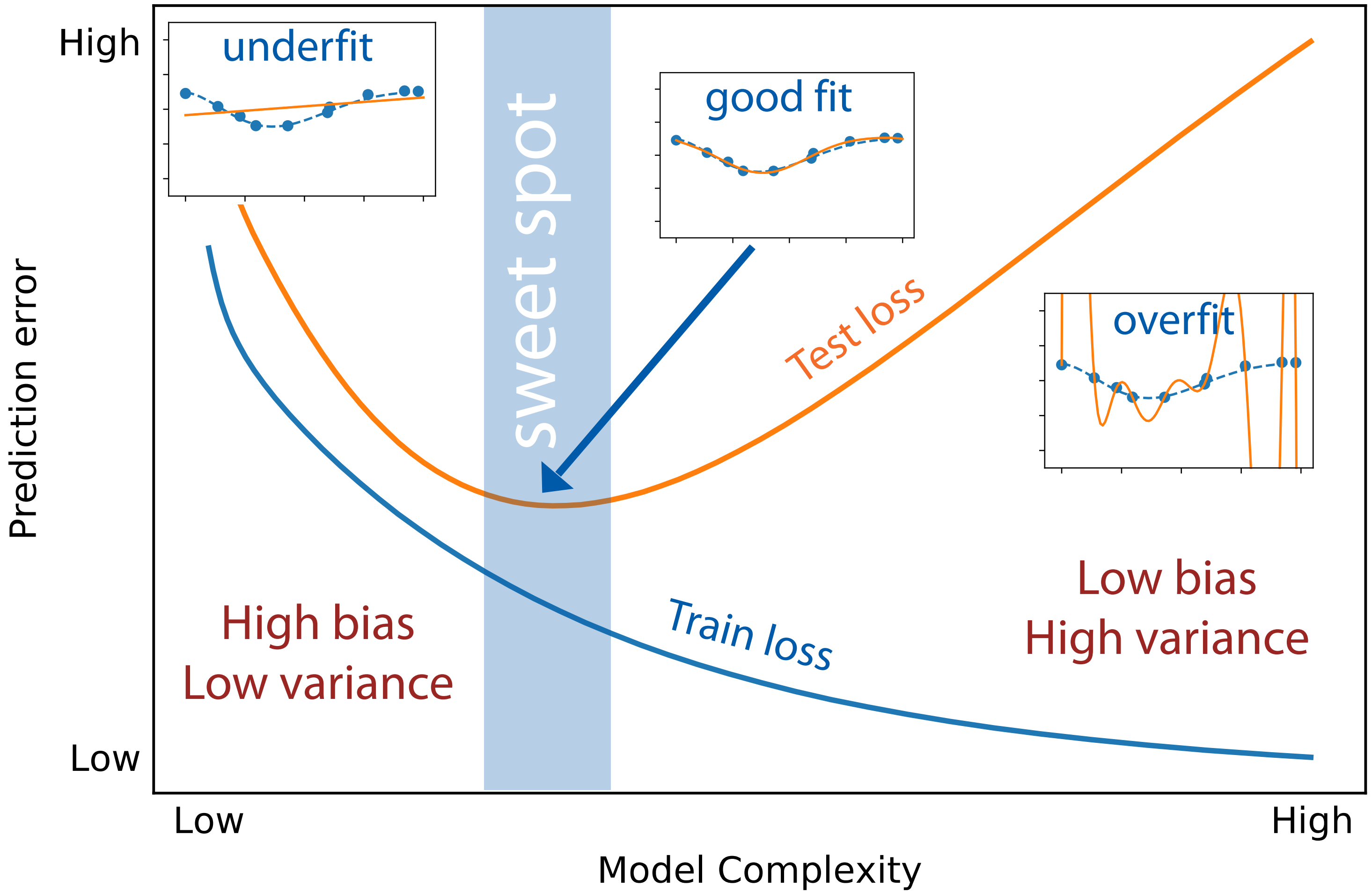
(easy to show that all the cross term expectations are 0)

$$= \mathop{\mathbb{E}}_{\tau}\left[\left(\hat{f}_\tau(x) - \mathop{\mathbb{E}}_{\tau'}\left[\hat{f}_{\tau'}(x)\right]\right)^2\right] + \left(\mathop{\mathbb{E}}_{\tau'}\left[\hat{f}_{\tau'}(x)\right] - f(x)\right)^2 + \mathop{\mathbb{E}}_{y\;x}\left[\left(f(x) - y\right)^2\right]$$

Irreducible error

$\left(= \mathbb{E}\left[\varepsilon^2\right] = \sigma_\varepsilon^2\right)$

Bias-variance tradeoff



Typically there's a tradeoff between the two sources of error

Example: bias and variance of a linear model

Bias and variance error components can be calculated analytically for linear models

Simplification:

for each expectation term $\mathbb{E}_{\tau}$ let's consider the features fixed, i.e. $X_\tau \equiv X$ (the design matrix is constant), and only the target vector y_τ is random)

Example: bias and variance of a linear model

Bias and variance error components can be calculated analytically for linear models

Simplification:

for each expectation term $\underset{\tau}{\mathbb{E}}$ let's consider the features fixed, i.e. $X_\tau \equiv X$ (the design matrix is constant), and only the target vector y_τ is random)

Recall the solution for the linear regression model with the MSE loss:

$$\hat{f}_\tau(x) = \theta_\tau^{\mathrm{T}} x = x^{\mathrm{T}} \theta_\tau$$

$$\theta_\tau = \left(X^{\mathrm{T}} X\right)^{-1} X^{\mathrm{T}} y_\tau$$

Example: bias and variance of a linear model

Let's look at the bias term from the error decomposition:

$$\operatorname{bias}(x) = \mathbb{E}_{\tau}\left[\hat{f}_{\tau}(x)\right] - f(x)$$

Example: bias and variance of a linear model

Let's look at the bias term from the error decomposition:

$$\text{bias}(x) = \mathbb{E}_{\tau}\left[\hat{f}_{\tau}(x)\right] - f(x) = \mathbb{E}_{\tau}\left[x^{\mathrm{T}}\left(X^{\mathrm{T}}X\right)^{-1}X^{\mathrm{T}}y_{\tau}\right] - x^{\mathrm{T}}\theta_{\text{true}}$$

We'll also assume that the true dependence is linear indeed

Example: bias and variance of a linear model

Let's look at the bias term from the error decomposition:

$$\mathrm{bias}(x) = \mathbb{E}_{\tau}\left[\hat{f}_{\tau}(x)\right] - f(x) = \mathbb{E}_{\tau}\left[x^{\mathrm{T}}\left(X^{\mathrm{T}}X\right)^{-1}X^{\mathrm{T}}y_{\tau}\right] - x^{\mathrm{T}}\theta_{\mathrm{true}}$$

$$= x^{\mathrm{T}}\left(X^{\mathrm{T}}X\right)^{-1}X^{\mathrm{T}}\mathbb{E}_{\tau}\left[y_{\tau}\right] - x^{\mathrm{T}}\theta_{\mathrm{true}}$$

Example: bias and variance of a linear model

Let's look at the bias term from the error decomposition:

$$\mathrm{bias}(x) = \mathbb{E}_{\tau}\left[\hat{f}_{\tau}(x)\right] - f(x) = \mathbb{E}_{\tau}\left[x^{\mathrm{T}}\left(X^{\mathrm{T}}X\right)^{-1}X^{\mathrm{T}}y_{\tau}\right] - x^{\mathrm{T}}\theta_{\mathrm{true}}$$

$$= x^{\mathrm{T}}\left(X^{\mathrm{T}}X\right)^{-1}X^{\mathrm{T}}\mathbb{E}_{\tau}\left[y_{\tau}\right] - x^{\mathrm{T}}\theta_{\mathrm{true}}$$

$$= x^{\mathrm{T}}\left(X^{\mathrm{T}}X\right)^{-1}X^{\mathrm{T}}X\theta_{\mathrm{true}} - x^{\mathrm{T}}\theta_{\mathrm{true}}$$

Example: bias and variance of a linear model

Let's look at the bias term from the error decomposition:

$$\mathrm{bias}(x) = \mathbb{E}_{\tau}\left[\hat{f}_{\tau}(x)\right] - f(x) = \mathbb{E}_{\tau}\left[x^{\mathrm{T}}\left(X^{\mathrm{T}}X\right)^{-1}X^{\mathrm{T}}y_{\tau}\right] - x^{\mathrm{T}}\theta_{\mathrm{true}}$$

$$= x^{\mathrm{T}}\left(X^{\mathrm{T}}X\right)^{-1}X^{\mathrm{T}}\mathbb{E}_{\tau}\left[y_{\tau}\right] - x^{\mathrm{T}}\theta_{\mathrm{true}}$$

$$= x^{\mathrm{T}}\left(X^{\mathrm{T}}X\right)^{-1}X^{\mathrm{T}}X\theta_{\mathrm{true}} - x^{\mathrm{T}}\theta_{\mathrm{true}}$$

Example: bias and variance of a linear model

Let's look at the bias term from the error decomposition:

$$\mathrm{bias}(x) = \mathop{\mathbb{E}}_{\tau}\left[\hat{f}_\tau(x)\right] - f(x) = \mathop{\mathbb{E}}_{\tau}\left[x^{\mathrm{T}}\left(X^{\mathrm{T}}X\right)^{-1}X^{\mathrm{T}}y_\tau\right] - x^{\mathrm{T}}\theta_{\mathrm{true}}$$

$$= x^{\mathrm{T}}\left(X^{\mathrm{T}}X\right)^{-1}X^{\mathrm{T}}\mathop{\mathbb{E}}_{\tau}\left[y_\tau\right] - x^{\mathrm{T}}\theta_{\mathrm{true}}$$

$$= x^{\mathrm{T}}\left(X^{\mathrm{T}}X\right)^{-1}X^{\mathrm{T}}X\theta_{\mathrm{true}} - x^{\mathrm{T}}\theta_{\mathrm{true}}$$

$$= x^{\mathrm{T}}\theta_{\mathrm{true}} - x^{\mathrm{T}}\theta_{\mathrm{true}} = 0$$

I.e. linear regression model is unbiased

as long as the true dependence is linear

Example: bias and variance of a linear model

Now let's look at the variance term:

$$\text{variance}(x) = \mathbb{E}_{\tau}\left[\left(\hat{f}_{\tau}(x) - \mathbb{E}_{\tau'}\left[\hat{f}_{\tau'}(x)\right]\right)^2\right]$$

It can then be shown that:

$$\text{variance}(x) = \sigma_\varepsilon^2 x^{\mathrm{T}}\left(X^{\mathrm{T}}X\right)^{-1}x$$

So the variance error component is a quadratic form, defined by the $\left(X^{\mathrm{T}}X\right)^{-1}$ matrix.

Example: bias and variance of a linear model

We can diagonalize $X^{\mathrm{T}}X$:

$$\mathrm{variance}(x) = \sigma_\varepsilon^2 x^{\mathrm{T}}\left(X^{\mathrm{T}}X\right)^{-1}x = \sigma_\varepsilon^2 \tilde{x}^{\mathrm{T}}\Lambda^{-1}\tilde{x}$$

where $\Lambda = \mathrm{diag}\{\lambda_1, \ldots, \lambda_d\}$ is the matrix of eigenvalues of $X^{\mathrm{T}}X$.

Example: bias and variance of a linear model

We can diagonalize $X^{\mathrm{T}}X$:

$$\text{variance}(x) = \sigma_\varepsilon^2 x^{\mathrm{T}}\left(X^{\mathrm{T}}X\right)^{-1} x = \sigma_\varepsilon^2 \tilde{x}^{\mathrm{T}}\Lambda^{-1}\tilde{x}$$

where $\Lambda = \mathrm{diag}\left\{\lambda_1, \ldots, \lambda_d\right\}$ is the matrix of eigenvalues of $X^{\mathrm{T}}X$.

This means that small eigenvalues amplify the model variance.

Example: bias and variance of a linear model

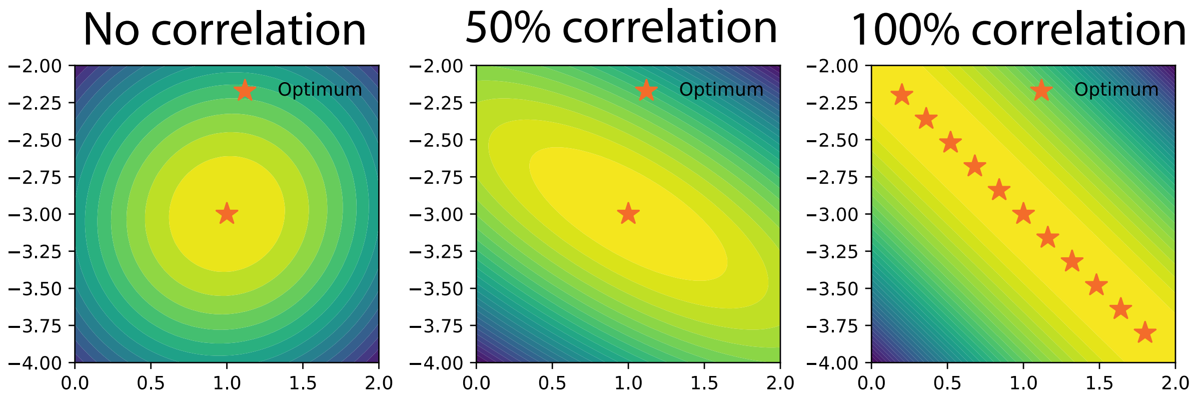
We can diagonalize $X^{\mathrm{T}}X$:

$$\text{variance}(x) = \sigma_\varepsilon^2 x^{\mathrm{T}} \left(X^{\mathrm{T}}X\right)^{-1} x = \sigma_\varepsilon^2 \tilde{x}^{\mathrm{T}} \Lambda^{-1} \tilde{x}$$

where $\Lambda = \mathrm{diag}\left\{\lambda_1, \ldots, \lambda_d\right\}$ is the matrix of eigenvalues of $X^{\mathrm{T}}X$.

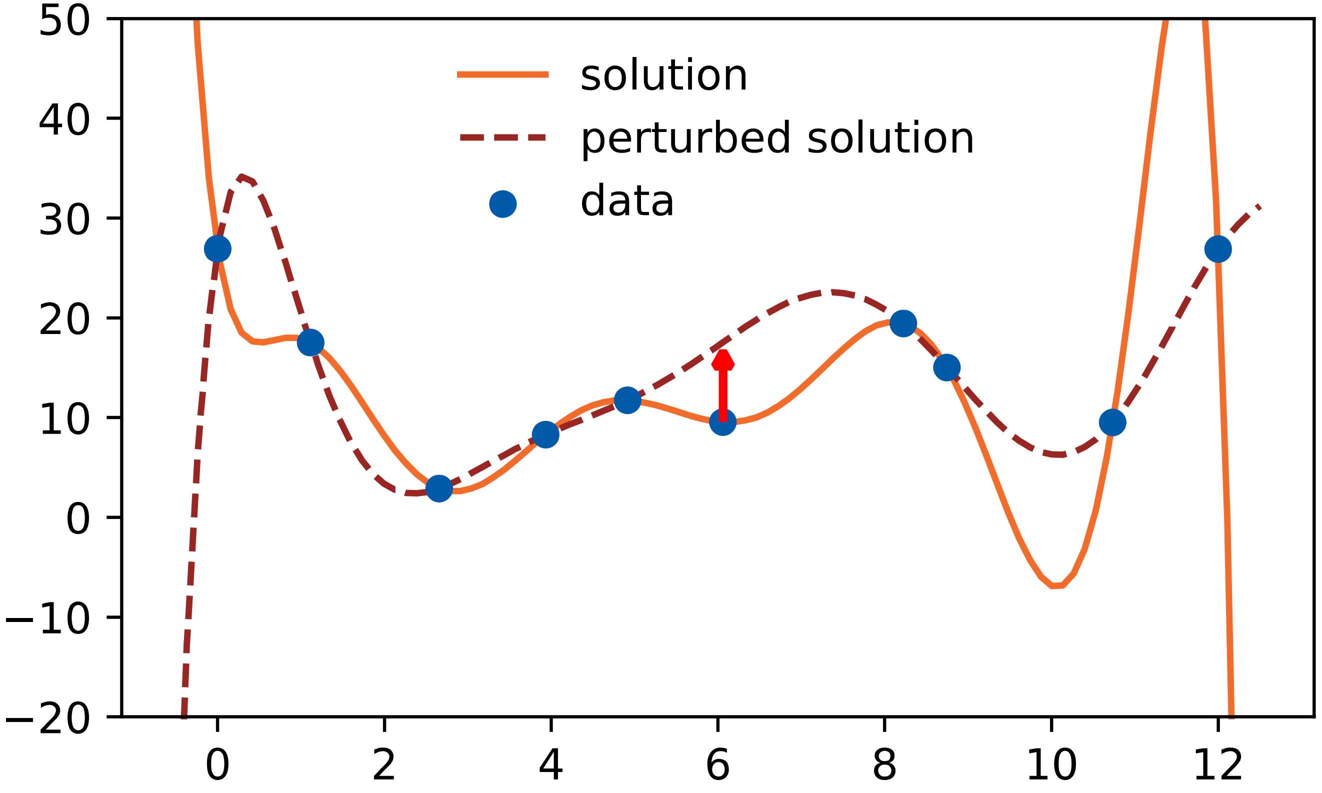
This means that small eigenvalues amplify the model variance.

This happens when $X^{\mathrm{T}}X$ is ill-defined e.g. when the features are correlated



MSE loss values as a function of model parameters

High-variance model



Small perturbation in data

⇩

Large change in prediction

Regularization

How can we reduce the variance?

If only we could increase the eigenvalues of $X^{\mathrm{T}}X\ldots$

How can we reduce the variance?

If only we could increase the eigenvalues of $X^{\mathrm{T}}X\ldots$

In fact, we can do this manually:

$$X^{\mathrm{T}}X \to X^{\mathrm{T}}X + \alpha I,$$

$$\alpha > 0 \in \mathbb{R},$$

I – unit d by d matrix

How can we reduce the variance?

If only we could increase the eigenvalues of $X^{\mathrm{T}}X$…

In fact, we can do this manually:

$$X^{\mathrm{T}}X \to X^{\mathrm{T}}X + \alpha I,$$

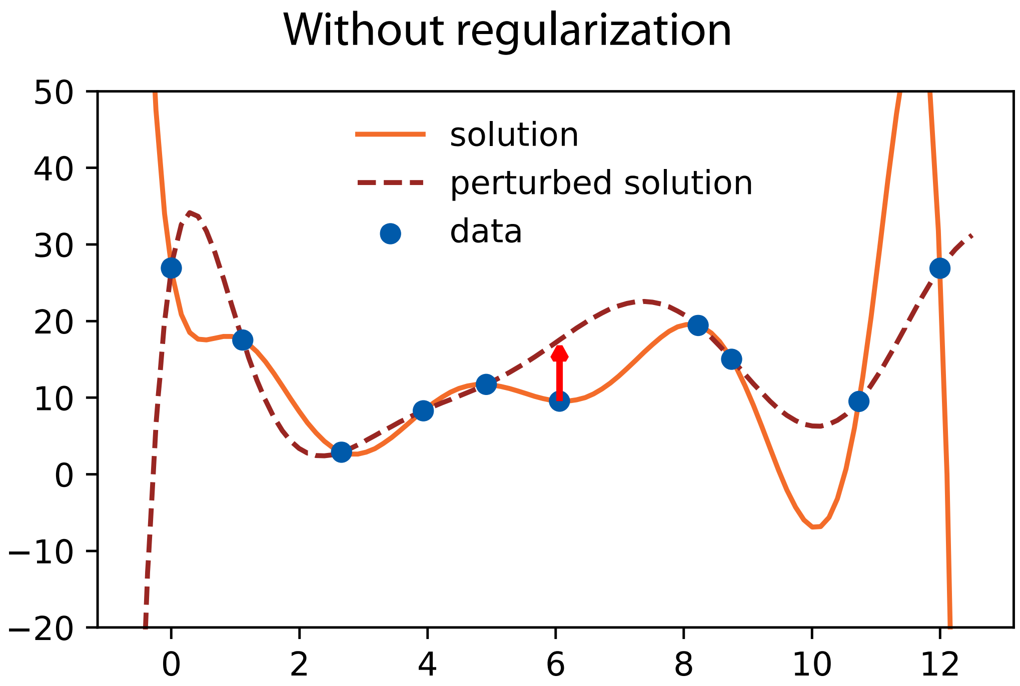
$$\alpha > 0 \in \mathbb{R},$$

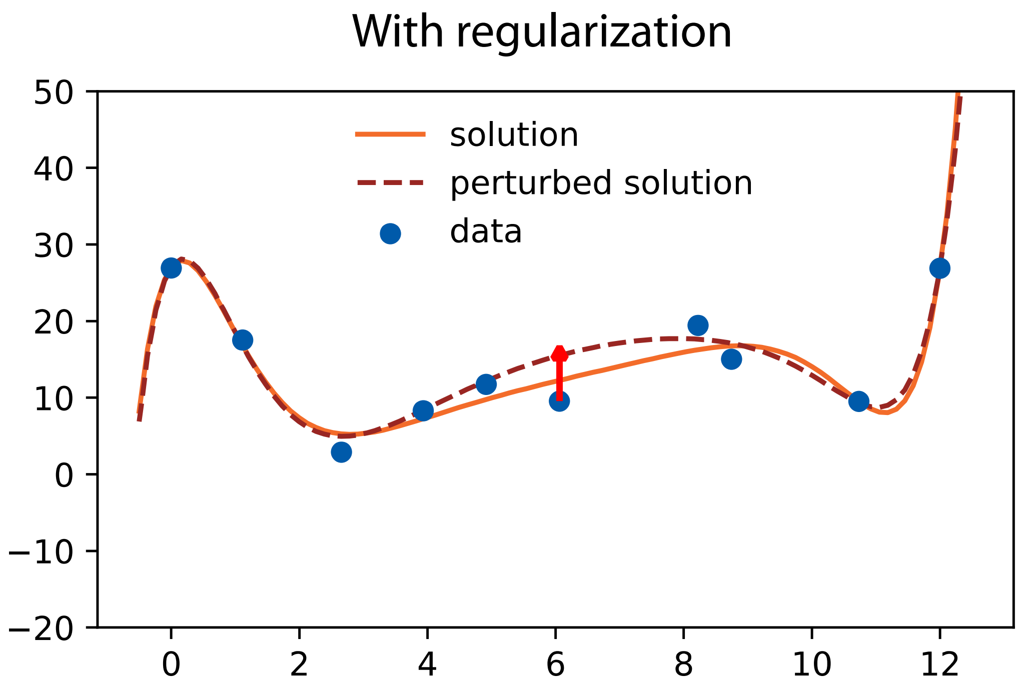
I – unit d by d matrix

I.e. we are changing the solution to:

$$\hat{f}_\tau(x) = x^{\mathrm{T}}\left(X^{\mathrm{T}}X + \alpha I\right)^{-1}X^{\mathrm{T}}y_\tau$$

The effect of regularization





Note: the regularized model is no longer unbiased!

I.e. we increased bias to reduce variance

What problem did we solve?

We have the solution:

$$\hat{f}_\tau(x) = x^{\mathrm{T}}\left(X^{\mathrm{T}}X + \alpha I\right)^{-1} X^{\mathrm{T}} y_\tau$$

Let's reverse engineer the loss function it optimizes:

What problem did we solve?

We have the solution:

$$\hat{f}_\tau(x) = x^{\mathrm{T}}\left(X^{\mathrm{T}}X + \alpha I\right)^{-1}X^{\mathrm{T}}y_\tau$$

Let’s reverse engineer the loss function it optimizes:

$$\theta_\tau = \left(X^{\mathrm{T}}X + \alpha I\right)^{-1}X^{\mathrm{T}}y_\tau$$

What problem did we solve?

We have the solution:

$$\hat{f}_\tau(x) = x^{\mathrm{T}}\left(X^{\mathrm{T}}X + \alpha I\right)^{-1}X^{\mathrm{T}}y_\tau$$

Let's reverse engineer the loss function it optimizes:

$$\theta_\tau = \left(X^{\mathrm{T}}X + \alpha I\right)^{-1}X^{\mathrm{T}}y_\tau$$

$$\left(X^{\mathrm{T}}X + \alpha I\right)\theta_\tau = X^{\mathrm{T}}y_\tau$$

What problem did we solve?

We have the solution:

$$\hat{f}_\tau(x) = x^{\mathrm{T}}\left(X^{\mathrm{T}}X + \alpha I\right)^{-1}X^{\mathrm{T}}y_\tau$$

Let's reverse engineer the loss function it optimizes:

$$\theta_\tau = \left(X^{\mathrm{T}}X + \alpha I\right)^{-1}X^{\mathrm{T}}y_\tau$$

$$\left(X^{\mathrm{T}}X + \alpha I\right)\theta_\tau = X^{\mathrm{T}}y_\tau$$

$$X^{\mathrm{T}}\left(X\theta_\tau - y_\tau\right) + \alpha\theta_\tau = 0$$

What problem did we solve?

We have the solution:

$$\hat{f}_\tau(x) = x^{\mathrm{T}}\left(X^{\mathrm{T}}X + \alpha I\right)^{-1}X^{\mathrm{T}}y_\tau$$

Let's reverse engineer the loss function it optimizes:

$$\theta_\tau = \left(X^{\mathrm{T}}X + \alpha I\right)^{-1}X^{\mathrm{T}}y_\tau$$

$$\left(X^{\mathrm{T}}X + \alpha I\right)\theta_\tau = X^{\mathrm{T}}y_\tau$$

$$X^{\mathrm{T}}\left(X\theta_\tau - y_\tau\right) + \alpha\theta_\tau = 0$$

In fact this is the $\partial/\partial\theta_\tau \mathscr{L} = 0$ equation for:

$$\mathscr{L} = \left\|X\theta_\tau - y_\tau\right\|^2 + \alpha\left\|\theta_\tau\right\|^2$$

What problem did we solve?

$$\mathcal{L} = \left\| X\theta_\tau - y_\tau \right\|^2 + \alpha \left\| \theta_\tau \right\|^2$$

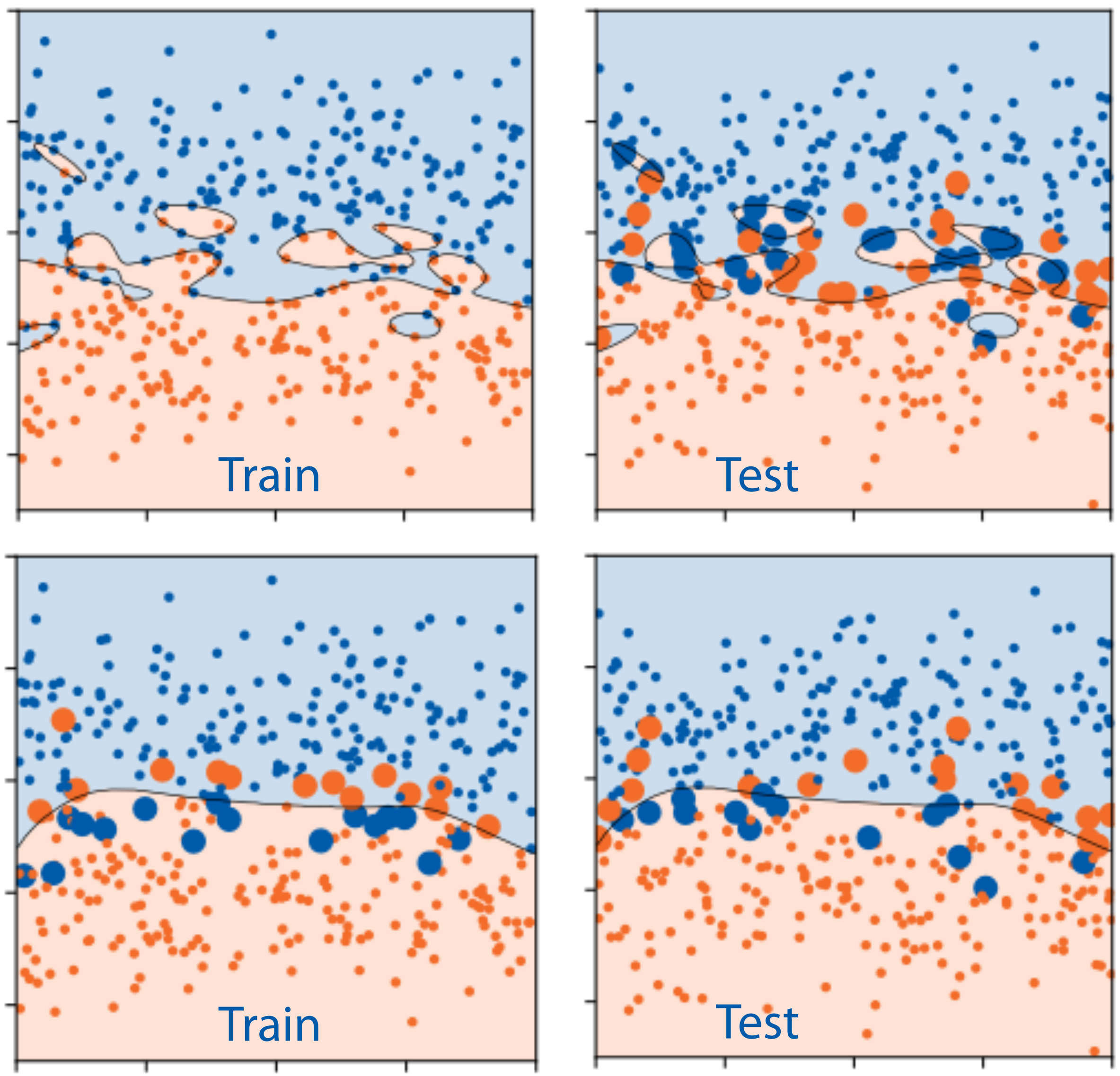
In other words, this linear model:

$$\hat{f}_\tau(x) = x^{\mathrm{T}} \left(X^{\mathrm{T}} X + \alpha I \right)^{-1} X^{\mathrm{T}} y_\tau$$

minimizes MSE loss with L2 penalty term on the model parameters.

Such model is also called
ridge regression

Example: L2-regularized classification



Without regularization

With regularization

By regularizing the model we increase the train loss and decrease the test loss

This improves the generalizability of the model

Various regularization methods

L2 regularization (Ridge):

$$\mathscr{L} = \left\| X\theta_\tau - y_\tau \right\|^2 + \alpha \left\| \theta_\tau \right\|^2$$

L2 norm:

$$\|x\|^2 \equiv \sum_{i=1\ldots d} x_i^2$$

L1 regularization (Lasso):

$$\mathscr{L} = \left\| X\theta_\tau - y_\tau \right\|^2 + \alpha \left\| \theta_\tau \right\|_1$$

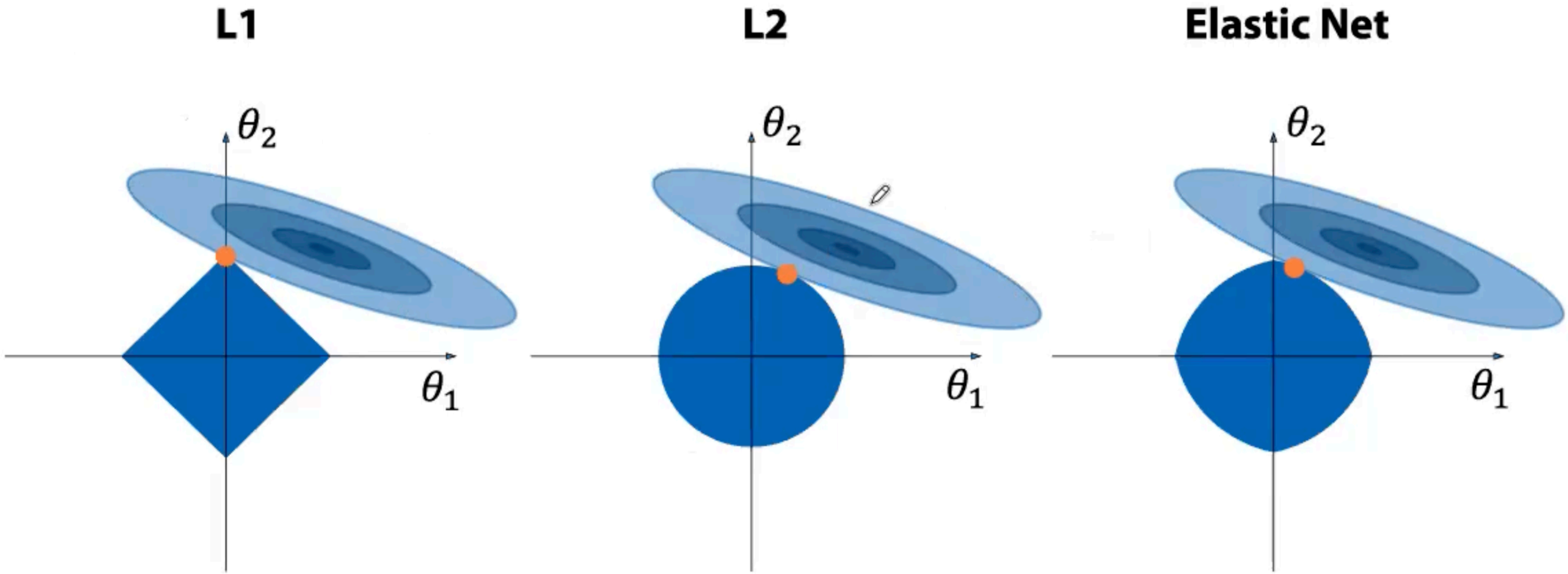
L1 norm:

$$\|x\|_1 \equiv \sum_{i=1\ldots d} \left| x_i \right|$$

Elastic net:

$$\mathscr{L} = \left\| X\theta_\tau - y_\tau \right\|^2 + \alpha \left\| \theta_\tau \right\|^2 + \beta \left\| \theta_\tau \right\|_1$$

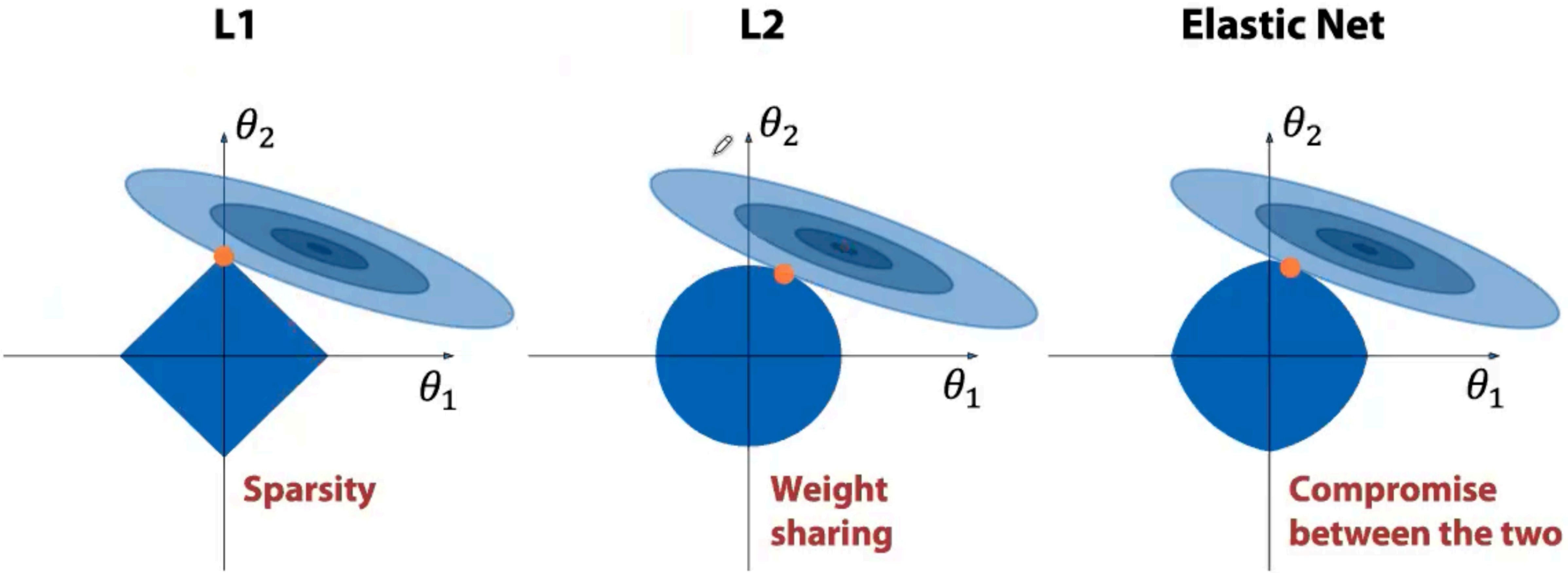
Properties of different regularization methods



They all drive the weights towards smaller values

Yet they induce different properties of the solution

Properties of different regularization methods



Yet they induce different properties of the solution

Probabilistic view

Probabilistic model

Let's revisit our assumption about data:

$$y = f(x) + \varepsilon$$

Now we'll assume that label noise is normally distributed:

$$\varepsilon \sim \mathcal{N}\left(0, \sigma_\varepsilon^2\right)$$

Probabilistic model

Let's revisit our assumption about data:

$$y = f(x) + \varepsilon$$

Now we'll assume that label noise is normally distributed:

$$\varepsilon \sim \mathcal{N}\left(0,\, \sigma_\varepsilon^2\right)$$

This means, that labels are also normally distributed, for a given point x:

$$y \;\; x \sim \mathcal{N}\left(f(x),\; \sigma_\varepsilon^2\right)$$

Probabilistic model

Let's revisit our assumption about data:

$$y = f(x) + \varepsilon$$

Now we'll assume that label noise is normally distributed:

$$\varepsilon \sim \mathcal{N}\left(0,\, \sigma_\varepsilon^2\right)$$

This means, that labels are also normally distributed, for a given point x:

$$y \;\; x \sim \mathcal{N}\left(f(x),\; \sigma_\varepsilon^2\right)$$

We want our model $\hat{f}_\theta(x)$ to fit the true dependence $f(x)$, i.e. we define a probabilistic model:

$$y \;\; x \sim \mathcal{N}\left(\hat{f}_\theta(x),\; \sigma_\varepsilon^2\right)$$

Probabilistic model

Our model can be fitted with the maximum likelihood approach:

$$L = \prod_{i=1\ldots N} \mathcal{N}\left(y_i \,\middle|\, \hat{f}_\theta(x_i),\ \sigma_\varepsilon^2\right) \to \max_\theta$$

Probabilistic model

Our model can be fitted with the maximum likelihood approach:

$$L = \prod_{i=1\dots N} \mathcal{N}\left(y_i \,\middle|\, \hat{f}_\theta(x_i),\ \sigma_\varepsilon^2\right) \to \max_\theta$$

Max. likelihood = min. negative log likelihood

$$-\log L = -\sum_{i=1\dots N} \log \mathcal{N}\left(y_i \,\middle|\, \hat{f}_\theta(x_i),\ \sigma_\varepsilon^2\right)$$

Probabilistic model

Our model can be fitted with the maximum likelihood approach:

$$L = \prod_{i=1\dots N} \mathcal{N}\left(y_i \middle| \hat{f}_\theta(x_i),\ \sigma_\varepsilon^2\right) \to \max_\theta$$

Max. likelihood = min. negative log likelihood

$$-\log L = -\sum_{i=1\dots N} \log \mathcal{N}\left(y_i \middle| \hat{f}_\theta(x_i),\ \sigma_\varepsilon^2\right)$$

$$= -\sum_{i=1\dots N} \left[\log\exp\left(-\frac{\left(y_i - \hat{f}_\theta(x_i)\right)^2}{2\sigma_\varepsilon^2} \right) - \log\sqrt{2\pi\sigma_\varepsilon^2} \right]$$

Probabilistic model

Our model can be fitted with the maximum likelihood approach:

$$L = \prod_{i=1\ldots N} \mathcal{N}\left(y_i \middle| \hat{f}_\theta(x_i),\ \sigma_\varepsilon^2\right) \to \max_\theta$$

Max. likelihood = min. negative log likelihood

$$-\log L = -\sum_{i=1\ldots N} \log \mathcal{N}\left(y_i \middle| \hat{f}_\theta(x_i),\ \sigma_\varepsilon^2\right)$$

$$= -\sum_{i=1\ldots N} \left[\log \exp\left(-\frac{\left(y_i - \hat{f}_\theta(x_i)\right)^2}{2\sigma_\varepsilon^2} \right) - \log\sqrt{2\pi\sigma_\varepsilon^2} \right]$$

$$= C \cdot \sum_{i=1\ldots N} \left(y_i - \hat{f}_\theta(x_i)\right)^2 + const$$

Probabilistic model

Our model can be fitted with the maximum likelihood approach:

$$L = \prod_{i=1\ldots N} \mathcal{N}\left(y_i \,\middle|\, \hat{f}_\theta(x_i),\ \sigma_\varepsilon^2\right) \to \max_\theta$$

Max. likelihood = min. negative log likelihood

$$-\log L = -\sum_{i=1\ldots N} \log \mathcal{N}\left(y_i \,\middle|\, \hat{f}_\theta(x_i),\ \sigma_\varepsilon^2\right)$$

$$= -\sum_{i=1\ldots N} \left[\log\exp\left(-\frac{\left(y_i - \hat{f}_\theta(x_i)\right)^2}{2\sigma_\varepsilon^2} \right) - \log\sqrt{2\pi\sigma_\varepsilon^2} \right]$$

MSE loss $\Longleftrightarrow$ Prob. model with normal label noise!

$$= C \cdot \sum_{i=1\ldots N} \left(y_i - \hat{f}_\theta(x_i)\right)^2 + const$$

Bayessian view

We are going to treat both data (X, y) and model parameters (θ) as random variables

Estimate the parameter distribution given the observed data

Bayessian view

We are going to treat both data (X, y) and model parameters (θ) as random variables

Estimate the parameter distribution given the observed data

(Reminder) Bayes rule:

$$p(\theta \mid X, y) = \frac{p(y \mid \theta, X) \cdot p(\theta)}{\int \left[p(y \mid \theta, X) \cdot p(\theta) \right] d\theta}$$

Bayessian view

We are going to treat both data (X, y) and model parameters (θ) as random variables

Estimate the parameter distribution given the observed data

(Reminder) Bayes rule:

Our prior knowledge about the model parameters

$$p(\theta | X, y) = \frac{p(y | \theta, X) \cdot p(\theta)}{\int \left[p(y | \theta, X) \cdot p(\theta) \right] d\theta}$$

Bayessian view

We are going to treat both data (X, y) and model parameters (θ) as random variables

Estimate the parameter distribution given the observed data

(Reminder) Bayes rule:

Likelihood function

$$p(\theta \mid X, y) = \frac{p(y \mid \theta,\ X) \cdot p(\theta)}{\int \left[p(y \mid \theta, X) \cdot p(\theta) \right] d\theta}$$

Bayessian view

We are going to treat both data $\left(X, y\right)$ and model parameters (θ) as random variables

Estimate the parameter distribution given the observed data

(Reminder) Bayes rule:

$$p\left(\theta \middle| X, y\right) = \frac{p\left(y \middle| \theta,\ X\right) \cdot p(\theta)}{\int \left[p\left(y \middle| \theta, X\right) \cdot p(\theta)\right] d\theta}$$

Posterior knowledge
about the model after
observing the data

Bayessian view

We are going to treat both data (X, y) and model parameters (θ) as random variables

Estimate the parameter distribution given the observed data

(Reminder) Bayes rule:

$$p(\theta \mid X, y) = \frac{p(y \mid \theta, X) \cdot p(\theta)}{\int \left[p(y \mid \theta, X) \cdot p(\theta) \right] d\theta}$$

“Evidence” (probability of observing this data when the parameter uncertainty is integrated out)

Bayessian view

We are going to treat both data (X, y) and model parameters (θ) as random variables

Estimate the parameter distribution given the observed data

(Reminder) Bayes rule:

$$p(\theta \mid X, y) = \frac{p(y \mid \theta,\ X) \cdot p(\theta)}{\int \left[p(y \mid \theta, X) \cdot p(\theta) \right] d\theta}$$

We'll make a point estimate (maximum a posteriori):

$$\hat{\theta} = \underset{\theta}{\operatorname{argmax}}\, p(\theta \mid X, y) = \underset{\theta}{\operatorname{argmax}}\, p(y \mid \theta, X) \cdot p(\theta)$$

Maximum a posteriori

Maximum a posteriori estimate:

$$\hat{\theta} = \underset{\theta}{\operatorname{argmax}}\, p\left(y \,\middle|\, \theta, X\right) \cdot p(\theta) = \underset{\theta}{\operatorname{argmin}} \left[\underbrace{-\log p\left(y \,\middle|\, \theta, X\right)}_{\text{Neg. log likelihood}} \underbrace{- \log p(\theta)}_{\text{Regularizer}} \right]$$

Example

Suppose we model the data with a normal distribution:

$$y \quad x \sim \mathcal{N}\left(\hat{f}_\theta(x),\ \sigma_\varepsilon^2\right)$$

And the prior is normal as well:

$$\theta \sim \mathcal{N}\left(0, \sigma_\theta^2 I\right)$$

Example

Suppose we model the data with a normal distribution:

$$y \quad x \sim \mathcal{N}\left(\hat{f}_\theta(x),\ \sigma_\varepsilon^2\right)$$

And the prior is normal as well:

$$\theta \sim \mathcal{N}\left(0, \sigma_\theta^2 I\right)$$

Then, maximum a posteriori estimate corresponds to minimizing the following loss:

$$\mathcal{L} = -\log p\left(y \,\middle|\, \theta, X\right) - \log p(\theta)$$

Example

Suppose we model the data with a normal distribution:

$$y \;\; x \sim \mathcal{N}\left(\hat{f}_\theta(x),\ \sigma_\varepsilon^2\right)$$

And the prior is normal as well:

$$\theta \sim \mathcal{N}\left(0, \sigma_\theta^2 I\right)$$

Then, maximum a posteriori estimate corresponds to minimizing the following loss:

$$\mathcal{L} = -\log p\left(y \middle| \theta, X\right) - \log p(\theta)$$

$$= C_1 \sum_{i=1\ldots N} \left(\hat{f}_\theta\left(x_i\right) - y_i\right)^2 + C_2 \left\|\theta\right\|^2 + const$$

Example

Suppose we model the data with a normal distribution:

$$y \quad x \sim \mathcal{N}\left(\hat{f}_\theta(x),\ \sigma_\varepsilon^2\right)$$

And the prior is normal as well:

$$\theta \sim \mathcal{N}\left(0,\, \sigma_\theta^2 I\right)$$

Then, maximum a posteriori estimate corresponds to minimizing the following loss:

$$\mathcal{L} = -\log p\left(y \,\middle|\, \theta, X\right) - \log p(\theta)$$

Normal prior $\Longleftrightarrow$ L2 regularization

$$= C_1 \sum_{i=1\ldots N} \left(\hat{f}_\theta\left(x_i\right) - y_i\right)^2 + C_2 \left\|\theta\right\|^2 + const$$

Summary

- Prediction error can be decomposed into components corresponding to model bias and variance

Summary

- Prediction error can be decomposed into components corresponding to model bias and variance
- Linear regression is unbiased, while its variance is large when $X^{\mathrm{T}}X$ matrix is ill-defined

Summary

- Prediction error can be decomposed into components corresponding to model bias and variance
- Linear regression is unbiased, while its variance is large when $X^{\mathrm{T}}X$ matrix is ill-defined
- Typically regularization reduces the variance with the price of increasing the bias

Summary

- Prediction error can be decomposed into components corresponding to model bias and variance
- Linear regression is unbiased, while its variance is large when $X^{\mathrm{T}}X$ matrix is ill-defined
- Typically regularization reduces the variance with the price of increasing the bias
- Different regularization techniques induce different properties of the solution

Summary

- Prediction error can be decomposed into components corresponding to model bias and variance
- Linear regression is unbiased, while its variance is large when $X^{\mathrm{T}}X$ matrix is ill-defined
- Typically regularization reduces the variance with the price of increasing the bias
- Different regularization techniques induce different properties of the solution
- There's a probabilistic model behind the loss function

Summary

- Prediction error can be decomposed into components corresponding to model bias and variance
- Linear regression is unbiased, while its variance is large when $X^{\mathrm{T}}X$ matrix is ill-defined
- Typically regularization reduces the variance with the price of increasing the bias
- Different regularization techniques induce different properties of the solution
- There's a probabilistic model behind the loss function
- Bayessian prior on the model parameters corresponds to some regularization to those parameters

Summary

- Prediction error can be decomposed into components corresponding to model bias and variance
- Linear regression is unbiased, while its variance is large when $X^{\mathrm{T}}X$ matrix is ill-defined
- Typically regularization reduces the variance with the price of increasing the bias
- Different regularization techniques induce different properties of the solution
- There's a probabilistic model behind the loss function
- Bayessian prior on the model parameters corresponds to some regularization to those parameters

- Food for thought: what probabilistic model would correspond to minimizing MAE loss?

Thank you!

al-maeeni@hse.ru

@afdee1c