Linear Regression, Regularization, Bias-Variance

Data 100 Review Discussion

Slides by Suraj Rampure

Fall 2018

Linear Regression, Regularization, Bias-Variance

Data 100 Review Discussion

Slides by Suraj Rampure

Fall 2018

Regression vs. Classification

Regression is the problem of creating a model that takes in a point and outputs a real number. We've seen regression in the form of Ordinary Least Squares, Ridge Regression, and LASSO Regression.

On the other hand, **classification** is the problem of creating a model that takes in a point and outputs a discrete **label**. In this course, we explored Logistic Regression (which, despite the name, is a classification technique), and previous courses, you may have seen k-Nearest Neighbors and k-Means Clustering, all of which are classification techniques.

A very basic example:

- Regression would allow us to predict a student's final exam grade, given their grades on the midterm and homeworks.
- Classification would allow us to predict whether or not that student will pass the exam

Gradients – Review

Suppose $a, x \in \mathbb{R}^n$ and $A \in \mathbb{R}^{n \times n}$.

Recall the gradients of each of the following functions.

$$f(x) = a^T x$$

$$\nabla f(x) = a$$

$$f(x) = x^T x$$

$$\nabla f(x) = 2x$$

$$f(x) = x^T A x$$

$$\nabla f(x) = (A + A^T)x$$

if A symmetric $\rightarrow$ $2Ax$

$$a = \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_n \end{bmatrix} \qquad x = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}$$

$$a^T x = a_1 x_1 + a_2 x_2 + \cdots + a_n x_n$$

$$\nabla a^T x = \begin{bmatrix} \frac{\partial f}{\partial x_1} \\ \frac{\partial f}{\partial x_2} \\ \vdots \\ \frac{\partial f}{\partial x_n} \end{bmatrix} = \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_n \end{bmatrix} = a$$

Loss Functions – Review

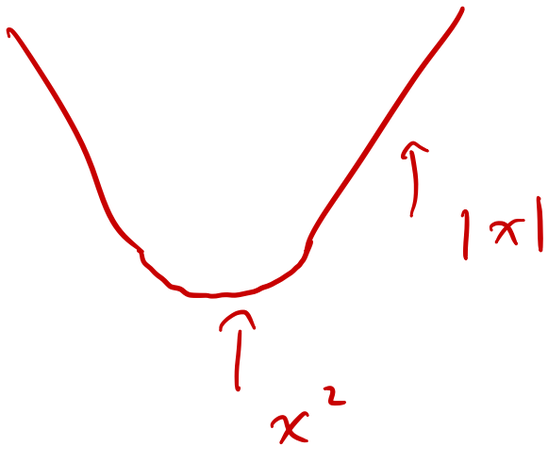
L_1: $y = \theta$ → median

L_2: $y = \theta$ → mean

Suppose y represents the true value of a variable, and θ represents our predicted value.

L1 Loss

$$L_1(y, \theta) = |y - \theta|$$



L2 Loss

$$L_2(y, \theta) = (y - \theta)^2$$

Huber Loss

$$L_\alpha(y, \theta) = \begin{cases} \frac{1}{2}(y - \theta)^2, & |y - \theta| \leq \alpha \\ \alpha\left(|y - \theta| - \frac{1}{2}\alpha\right), & \text{else} \end{cases}$$

$y = mx + b$

Linear Regression in 2D

Suppose we're given $\{(x_1, y_1), (x_2, y_2), ..., (x_n, y_n)\}$, and want to fit a linear model $y = \theta_1 x + \theta_0$, using MSE (i.e. L2) loss.

Our objective function is

$$L(\theta_0, \theta_1) = \frac{1}{n}\sum_{i=1}^{n}(y_i - y_{pred})^2 = \frac{1}{n}\sum_{i=1}^{n}(y_i - \theta_1 x_i - \theta_0)^2$$

To solve: Take partial derivatives with respect to θ_0, θ_1. Solve for θ_0 and θ_1.

Let's try and rewrite this in vector form.

$$L(\theta) = \frac{1}{n}\sum_{i=1}^{n}(y_i - \theta_1 x - \theta_0)^2$$

We can say the following:

$$\theta = \begin{bmatrix}\theta_0 & \theta_1\end{bmatrix}^T$$

$$\phi = \begin{bmatrix}1 & x_1 \\ 1 & x_2 \\ \vdots & \\ 1 & x_n\end{bmatrix}$$

feature matrix

$$y = \begin{bmatrix}y_1 & y_2 & \dots & y_n\end{bmatrix}^T$$

$$L(\theta) = \frac{1}{n}\|y - \phi\theta\|_2^2$$

$$\begin{bmatrix}1 & x_1 \\ 1 & x_2 \\ \vdots & \\ 1 & x_n\end{bmatrix}\begin{bmatrix}\theta_0 \\ \theta_1\end{bmatrix} = \begin{bmatrix}\theta_0 + \theta_1 x_1 \\ \theta_0 + \theta_1 x_2 \\ \vdots \\ \theta_0 + \theta_1 x_n\end{bmatrix}$$

$$\begin{bmatrix}y_1 \\ y_2 \\ \vdots \\ y_n\end{bmatrix}$$

Solution to Normal Equation, using Vector Calculus

$$L(\theta) = \frac{1}{n}\left|\left|y - \phi\theta\right|\right|_2^2 = \frac{1}{n}\left((y-\phi\theta)^T(y-\phi\theta)\right)$$

$$= \frac{1}{n}\left(y^T y - y^T\phi\theta - (\phi\theta)^T y - \theta^T\phi^T\phi\theta\right)$$

$$= \frac{1}{n}\left(y^T y - 2(\phi^T y)^T\theta - (\phi\theta)^T(\phi\theta)\right)$$

Taking the gradient and setting it equal to 0:

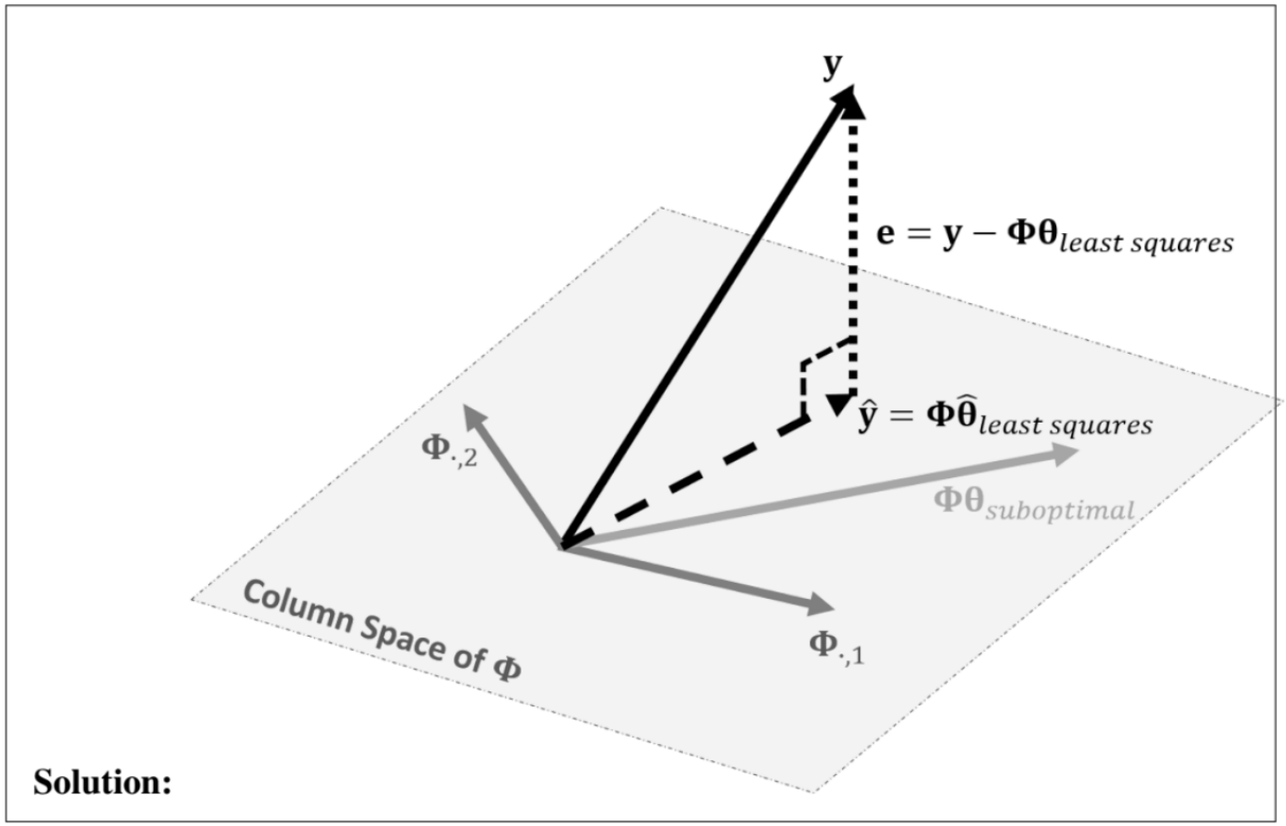
$$\nabla L(\theta) = 0 - 2\phi^T y - 2\phi^T\phi\theta = 0$$

$$\Rightarrow \phi^T\phi\theta = \phi^T y$$

$$\Rightarrow \theta^* = (\phi^T\phi)^{-1}\phi^T y$$

(Notice, we ignored the factor of $\frac{1}{n}$. This yields the same result.

Solution to Normal Equation, using Geometry



$y - \phi\theta$ is orthogonal to span(ϕ)

dot product of every vector in ϕ with $y - \phi\theta = 0$

$$\phi^T(y - \phi\theta) = 0$$

$$\phi^T y - \phi^T \phi\theta = 0$$

$$\phi^T \phi\theta = \phi^T y$$

$$\theta = (\phi^T\phi)^{-1}\phi^T y$$

We see that to minimize e, e must be orthogonal to the column space of ϕ.

Regularization

ordinary least squares

$\text{rank}(\phi) = \text{rank}(\phi^T \phi)$

Issues with "OLS":

- Solution doesn't always exist (if ϕ is not full-rank, $\phi^T \phi$ will not be full rank)
- Numerical issues with inversions
- Potential overfitting to training set – model can be too complex

To fix: Add penalty on magnitude of θ. However, we could use either the L2 norm, or L1 norm!

Using L2: $L(\theta) = \frac{1}{n}||y - \phi\theta||_2^2 + \lambda||\theta||_2^2$ Ridge Regression

Using L1: $L(\theta) = \frac{1}{n}||y - \phi\theta||_2^2 + \lambda||\theta||_1$ LASSO

Recall: $||x||_2 = \sqrt{x_1^2 + x_2^2 + ... + x_n^2}$, and $||x||_1 = |x_1| + |x_2| + ... + |x_n|$.

Regularization – Ridge Regression

When we use the L2 vector norm for the penalty term, our objective function becomes

$$\min_{\theta} \frac{1}{n} ||y - \phi\theta||_2^2 + \lambda ||\theta||_2^2$$

This is called "ridge regression."

Solution can also be determined using vector calculus.

$$\Rightarrow \theta^*_{ridge} = (\phi^T \phi + \lambda I)^{-1} \phi^T y$$

- $(\phi^T \phi + \lambda I)$ is always invertible, for any $\lambda > 0$! For proof, see Discussion 7.
- λ represents the penalty on the size of our model. We will discuss this more later in the review.

Regularization – LASSO Regression

When we use the L1 vector norm for the penalty term, our objective function becomes

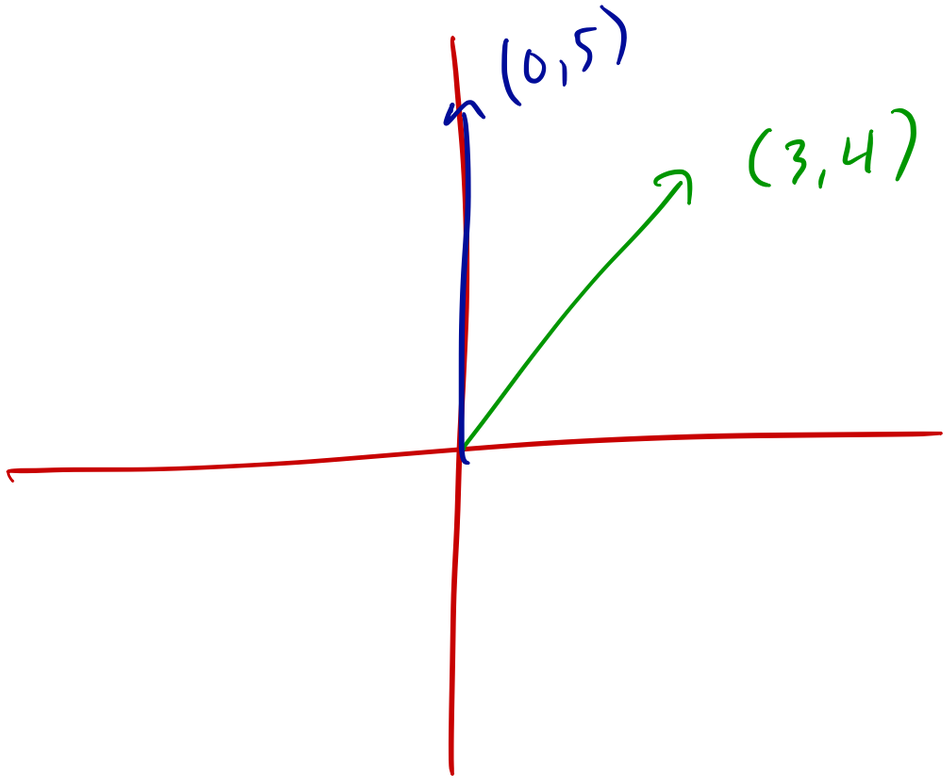
$$\min_{\theta} \frac{1}{n}||y - \phi\theta||_2^2 + \lambda||\theta||_1$$

This is called "LASSO regression." *Fun fact: LASSO stands for Least Absolute Shrinkage and Selection Operator.*

Unlike OLS and Ridge Regression, there is (in general) no closed form solution. Need to use a numerical method, such as gradient descent.

- LASSO regression encourages sparsity, that is, it sets many of the entries in our θ vector to 0. LASSO effectively selects features for us, and also makes our model less complex (many weights set to 0 ——> less features used ——> less complex)
- Again, λ represents the penalty on the size of our model.

L_1 vs. L_2 vector norms: $(3, 4)$ vs. ~~$(5, 0)$~~ $(0, 5)$



L_1 : $|3| + |4| = 7$

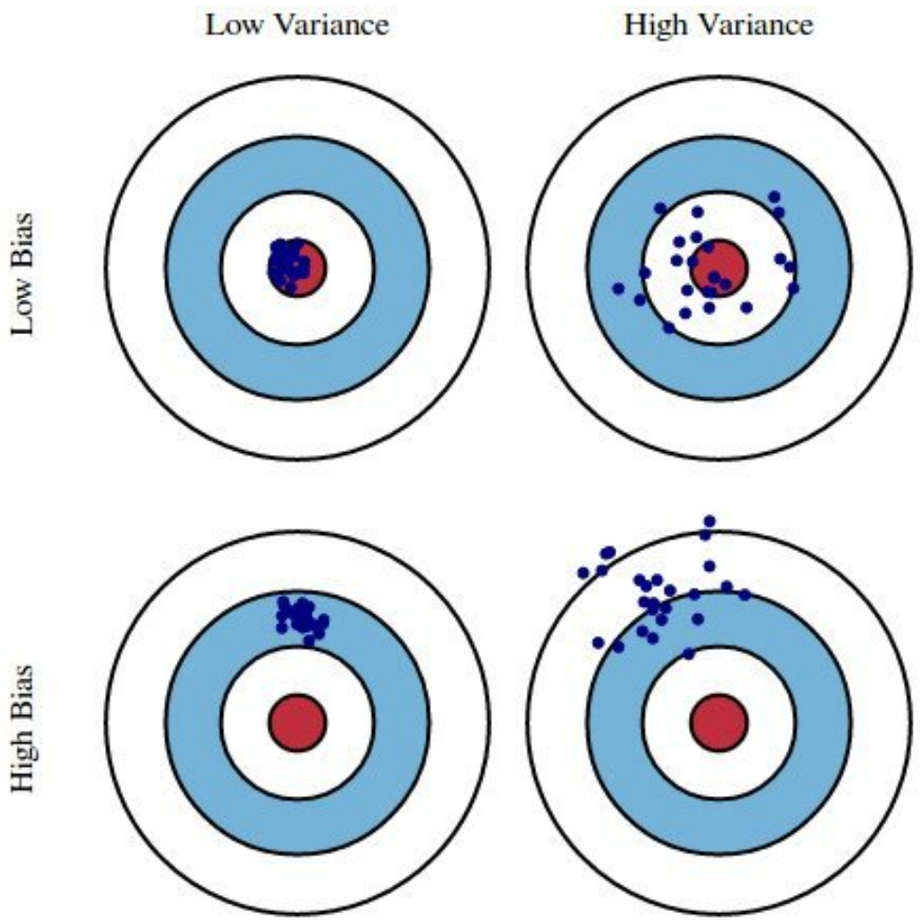
L_2 : 5

L_1 : $|0| + |5| = 5$

L_2 : $\sqrt{0^2 + 5^2} = 5$

L_2 norms are same, but L_1 norm is less when 1 element is set to 0

Bias-Variance



Bias-Variance Decomposition

Suppose ϵ is some random variable such that $\mathbb{E}[\epsilon] = 0$ and $var[\epsilon] = \sigma^2$. Also, suppose we have Y generated as follows:

$$Y = h(x) + \epsilon$$

We collect some sample points $\{(x_i, y_i)\}_{i=1}^n$, and want to fit a model $f_\theta(x)$. We define the model **risk** as $\mathbb{E}[(Y - f_\theta(x))^2]$.

See lecture for derivation

irreducible error / variance

$$\mathbb{E}[(Y - f_\theta(x))^2] = (h(x) - \mathbb{E}[f_\theta(x)])^2 + \mathbb{E}(\mathbb{E}[f_\theta(x)] - f_\theta(x))^2] + \sigma^2$$

bias^2 model variance

This is sometimes referred to as the **bias-variance decomposition.**

$\text{var}(X) = E\left[(E[X] - X)^2\right]$

Let's analyze the objective function for ridge regression (however, the analysis is the same for

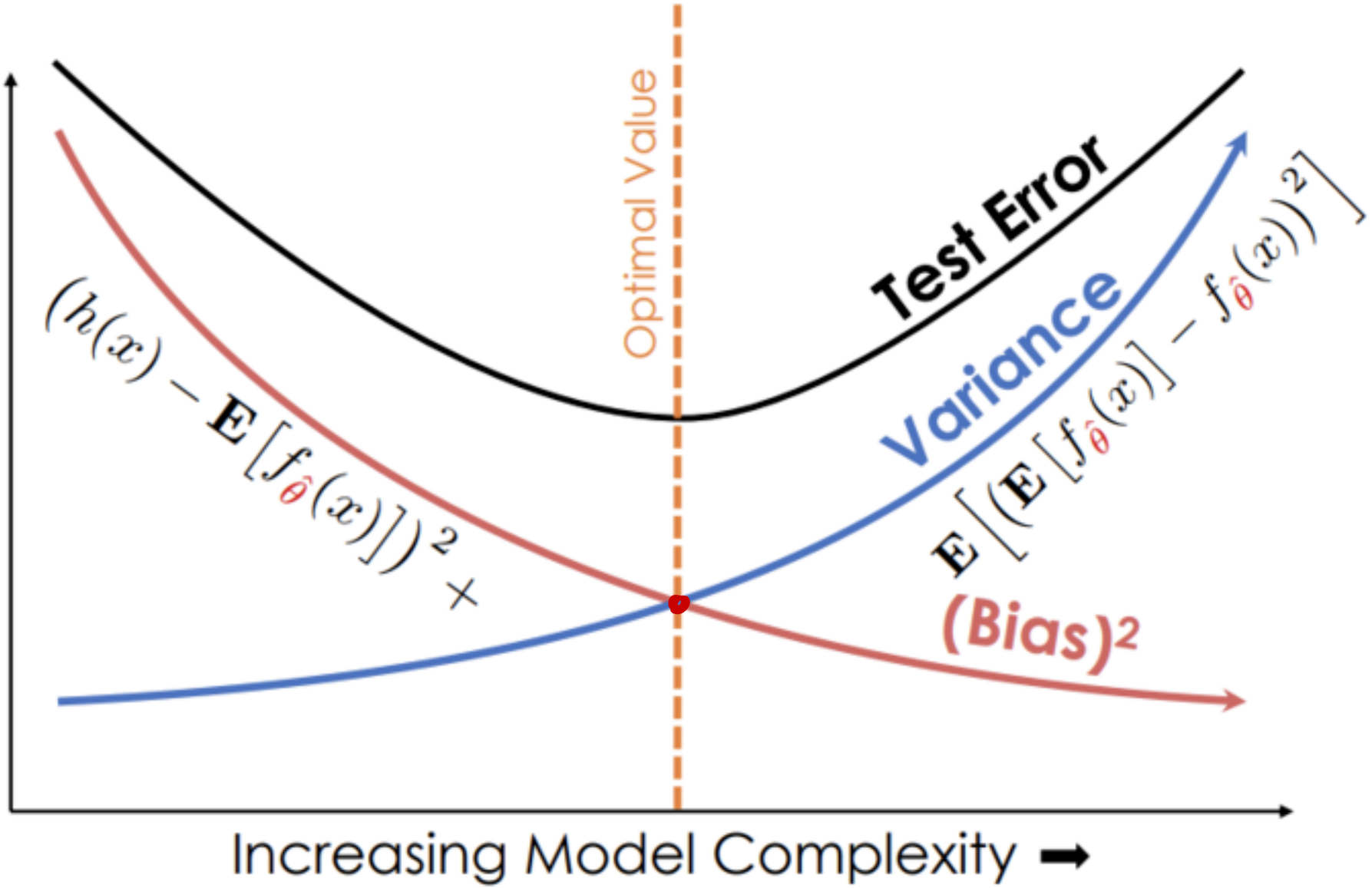
$$\min_{\theta} \frac{1}{n}\left|\left|y - \phi\theta\right|\right|_2^2 + \lambda\left|\left|\theta\right|\right|_2^2$$

As λ increases, model complexity decreases. This is because increasing λ increases the penalty on the magnitude of θ. Since we are trying to minimize the objective, if λ increases, $\left|\left|\theta\right|\right|_2^2$ must decrease.

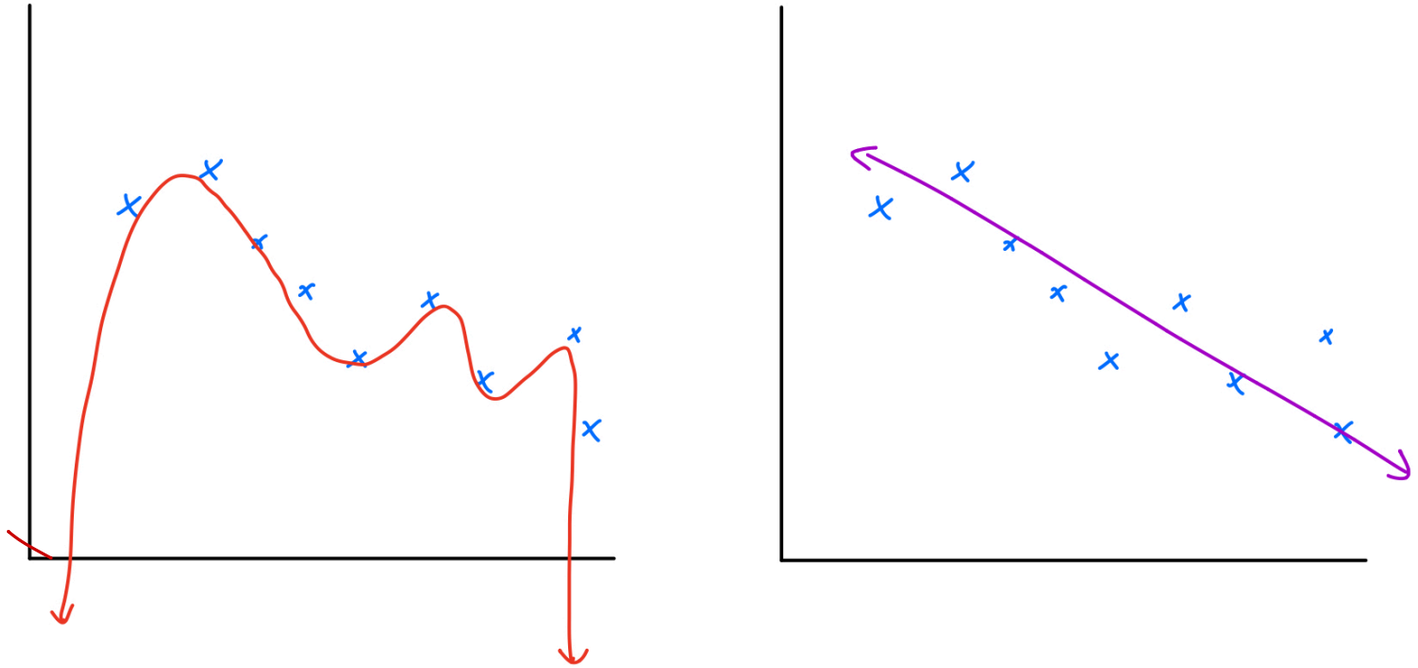
As a result, as λ increases, model bias increases, and variance decreases.

- Bias increases because our model becomes less complex, and thus more general.
- Variance decreases because, again, our model becomes more general.

λ is inversely proportional to model complexity


Optimal Value
Test Error
Variance
$\mathbf{E}\left[\left(\mathbf{E}\left[f_{\hat{\theta}}(x)\right] - f_{\hat{\theta}}(x)\right)^2\right]$
$\left(h(x) - \mathbf{E}\left[f_{\hat{\theta}}(x)\right]\right)^2$ +
(Bias)²
Increasing Model Complexity ➡

Polynomial regression with large d, small d:



The high degree polynomial model has lower bias, but higher variance, than the model on the right.

One way to interpret variance: In the model on the left, if we were to introduce a new point, our polynomial model would change significantly. However, on the right, introducing a new point is unlikely to change our model by much.

More Practice Problems

Some of the following problems come from past exams.

Practice: Regularization, B-V (T/F)

True/False: L_1 reguarlization can help us select a subset of the features that are important. LASSO

True/False: After regularization, we expect the training accuracy to increase and the test accuracy to decrease. opposite!

True/False: In ridge regression, if we let $\lambda \to \infty$, our model will become more and more complex. less, not more

True/False: As we improve our model to reduce bias, we often run the risk of under-fitting. reduce bias → more complex → over-fitting

True/False: Suppose our data is an i.i.d. sample from a population. Collecting a larger sample for use as a training set can help reduce variance. not a good question.... ignore

True/False: Training error is typically larger than test error. typically the opposite

Practice: Regularization

Consider the following general loss formulation.

$$\arg\min_{\theta} \left[\sum_{i=1}^{n} (y_i - x_i^T \theta)^2 + \lambda \sum_{p=1}^{d} \theta_p^2 \right]$$

a) How many data points are there? n

b) What dimension is our data, i.e. how many features are we using? d

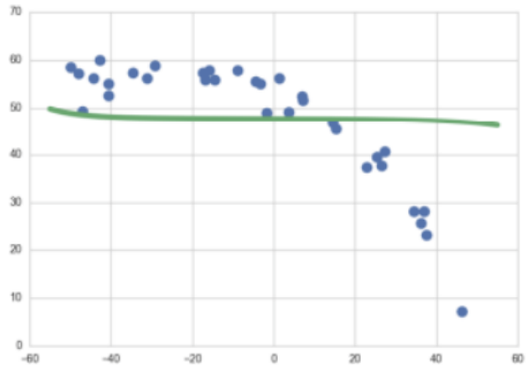
c) Is this a classification or regression problem? regression (ridge)

d) What type of regularization is being used? L_2

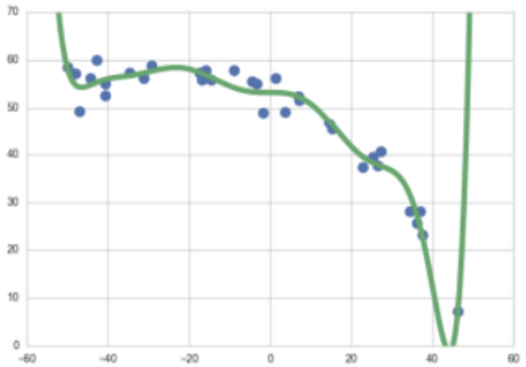
e) As λ increases, what will happen to bias? increase

f) As λ increases,what will happen to variance? decrease

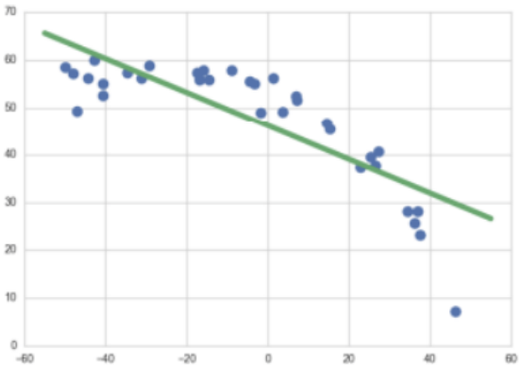
Practice: Identifying types of regression



(a)



(b)



(c)

Determine the plot above that best represents each of the following models.

i. Linear regression c)

ii. Regularized linear regression, using polynomial features and a large λ a)

iii. Linear regression, using degree 10 polynomial features b)

Practice: Orthogonality of error

Suppose X is an $n \times p$ design matrix with full column rank and y is an $n \times 1$ response vector. Let θ^* be the optimal solution to the OLS problem and ϵ be its associated error, i.e. $y = X\theta + \epsilon$.

Prove that $\epsilon \cdot x_i = 0$, where x_i is any column in X.

We can instead show $X^T\epsilon = 0$ $\quad \epsilon = y - X\theta$

$$X^T\epsilon = X^T(y - X\theta)$$
$$= X^T(y - X(X^TX)^{-1}X^Ty)$$
$$= (X^T - X^TX(X^TX)^{-1}X^T)y$$
$$= (X^T - X^T)y = \boxed{0}$$

as required

Didn't get to, not that important

Practice: Loss Functions

This problem is adapted from this semester's midterm.

Consider the following loss function:

$$L(\theta, x, y) = \begin{cases} a(f_\theta(x) - y) & f_\theta(x) \geq y \\ b(y - f_\theta(x)) & f_\theta(x) < y \end{cases}$$

You decide to use the constant model $f_\theta(x) = \theta$, and now need to optimize

and now need to optimize $L(\theta, \mathbf{x}, \mathbf{y}) = \frac{1}{n} \sum_{i=1}^{n} L(\theta, x_i, y_i)$.

In terms of a, b, determine the optimal θ^*.

Sources:

- Discussion 5, 6 and 7
- Spring 2018 Final
- Fall 2017 Practice Final
- http://textbook.ds100.org
- https://www.kdnuggets.com/2016/08/bias-variance-tradeoff-overview.html