DS 4400

Machine Learning and Data Mining I
Spring 2022

Alina Oprea
Associate Professor
Khoury College of Computer Science
Northeastern University

February 9 2022

Announcements

- Will start grading HW 1
- Will release HW 2 this week
- Midterm exam on March 2
- Project proposal on March 7
  - Will release some resources (datasets, project ideas)

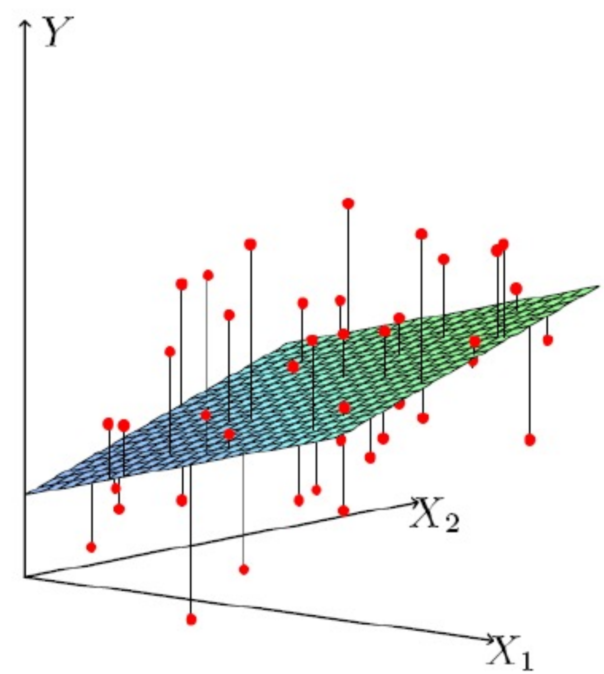
Outline

- Non-linear regression
  - Polynomial regression
  - Cubic, spline regression
- Regularization
  – Ridge regression
  – Lasso regression
- Classification
  – K Nearest Neighbors (kNN)
  – Bias-Variance tradeoff

Multiple Linear Regression

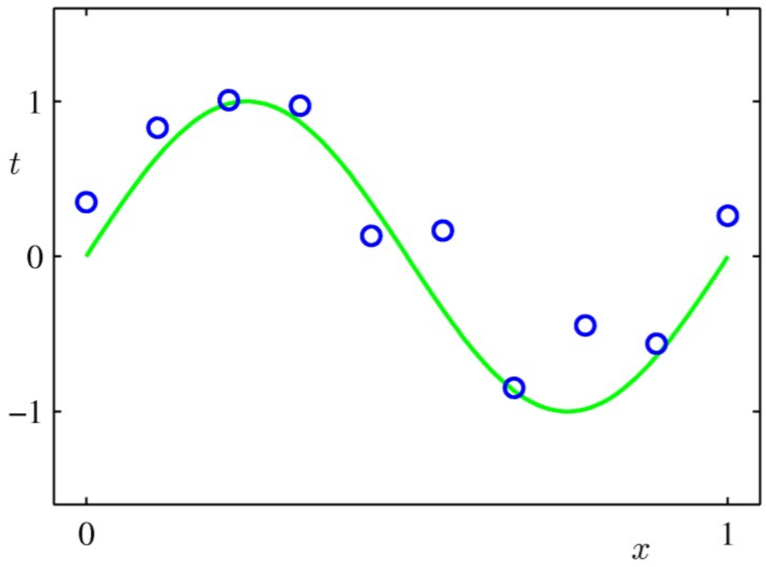
- Dataset: $x_i \in R^d, y_i \in R$
- Hypothesis $h_\theta(x) = \theta^T x$
- MSE $= \frac{1}{N}\sum (\theta^T x_i - y_i)^2$ Loss / cost

$$\theta = (X^\top X)^{-1} X^\top y$$

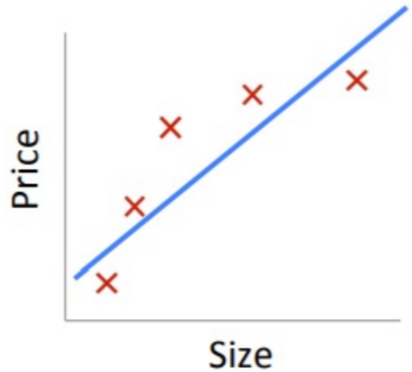


Polynomial Regression

- Polynomial function on single feature
  - $h_\theta(x) = \theta_0 + \theta_1 x + \cdots + \theta_p x^p$

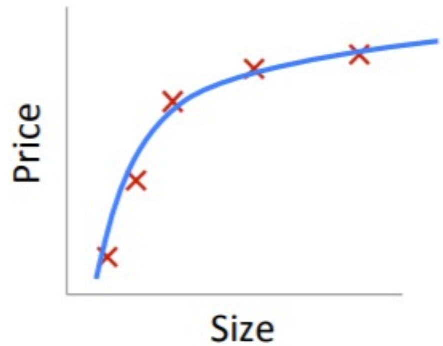


Polynomial Regression



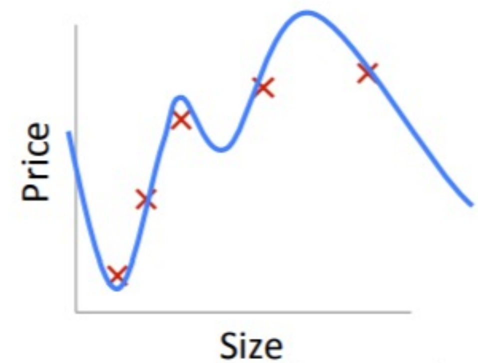
$\theta_0 + \theta_1 x$

Underfitting
(high bias)



$\theta_0 + \theta_1 x + \theta_2 x^2$

Correct fit



$\theta_0 + \theta_1 x + \theta_2 x^2 + \theta_3 x^3 + \theta_4 x^4$

Overfitting
(high variance)

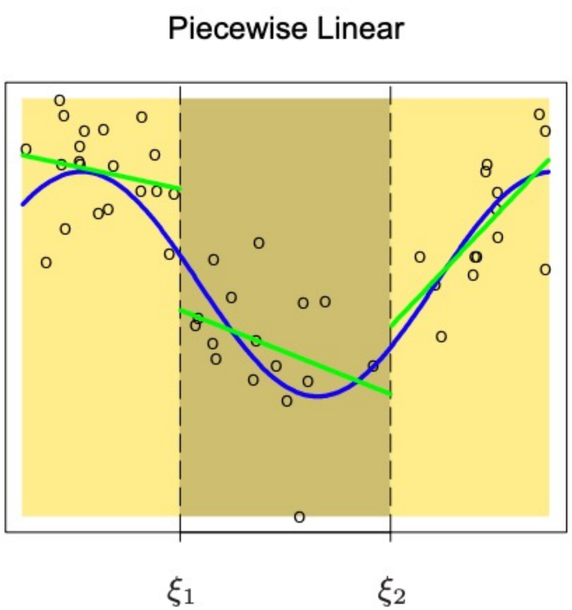
- Typically to avoid overfitting $d \leq 4$

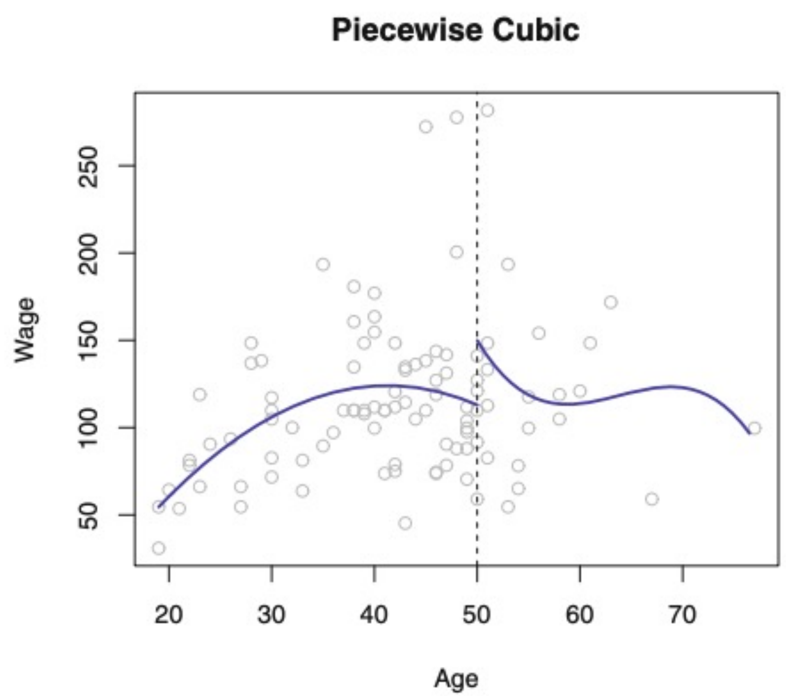
Polynomial Regression Training

- Simple Linear Regression
- $h_\theta(x) = \theta_0 + \theta_1 x + \cdots + \theta_p x^p$
- How to train model?

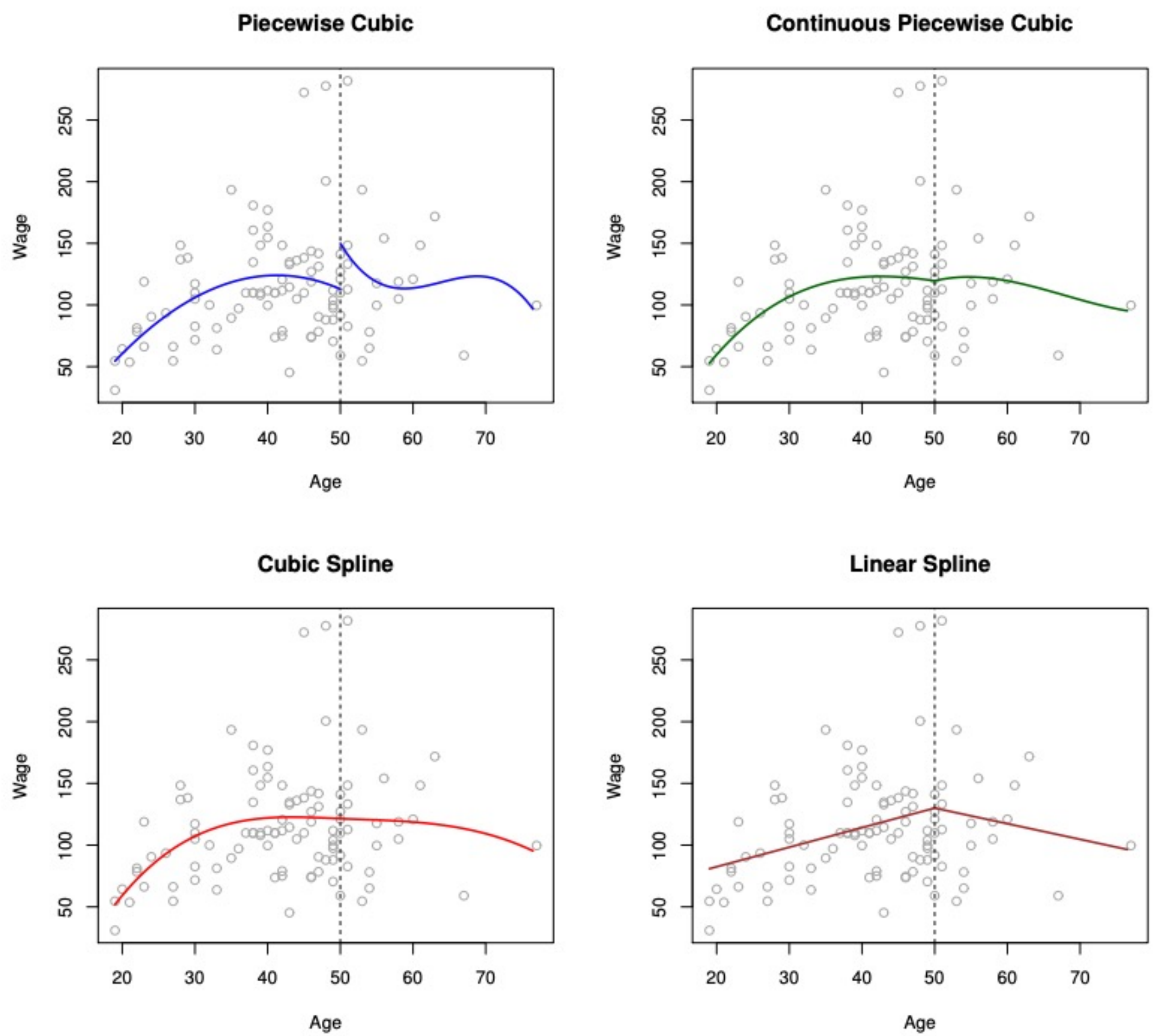
Piecewise Polynomial

- Divide the space into regions
- Polynomial regression on each region
  - Linear piecewise (degree 1), quadratic piecewise (degree 2), cubic piecewise (degree 3)

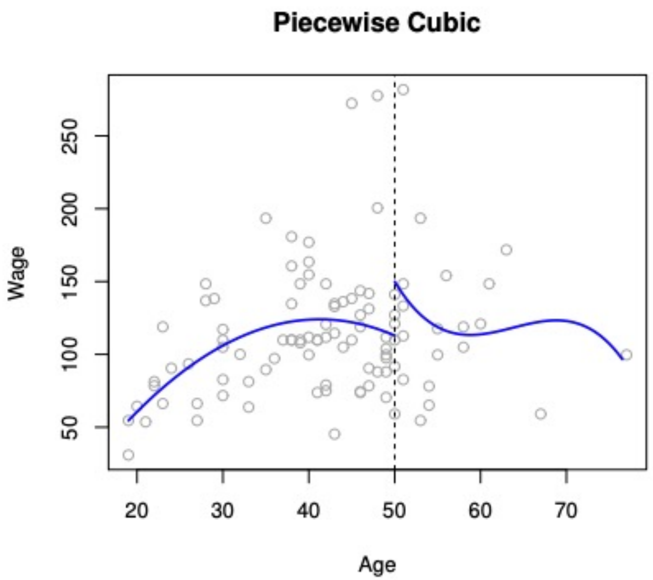




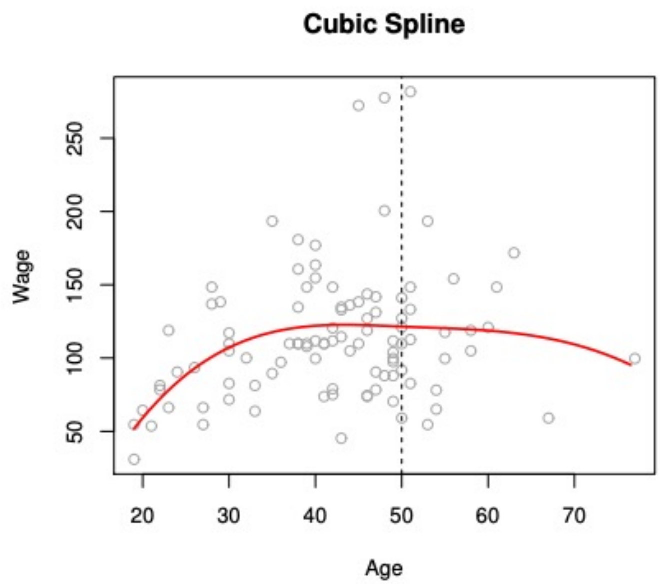
Piecewise and spline regression



Piecewise polynomial vs Regression spline



1 break at $\texttt{Age} = 50$

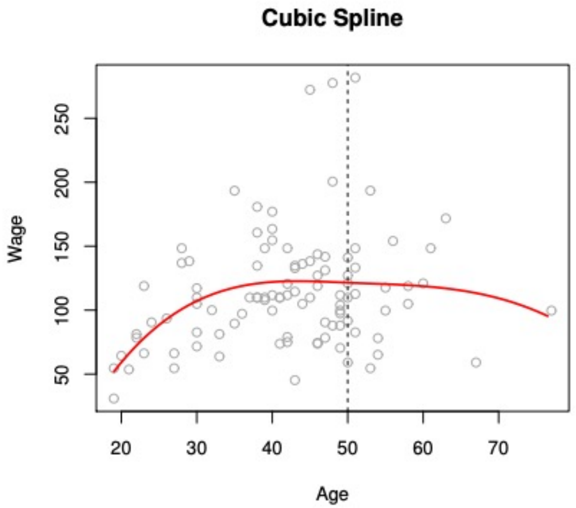


1 knot at $\texttt{Age} = 50$

Definition: Cubic spline

A cubic spline with knots at x-values $\xi_1, \ldots, \xi_K$ is a continuous piecewise cubic polynomial with *continuous derivates* and *continuous second derivatives* at each knot.

Cubic splines

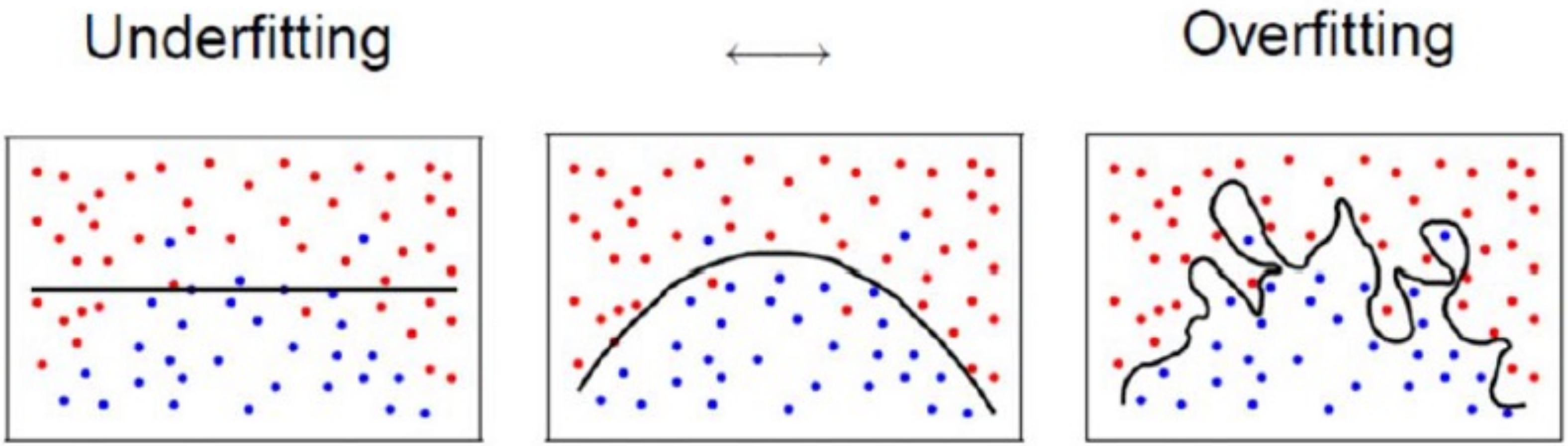


- Turns out, cubic splines are sufficiently flexible to *consistently* estimate smooth regression functions f
- You can use higher-degree splines, but *there's no need to*
- To fit a cubic spline, we just need to pick the knots

Additive Models

- Multiple Linear Regression Model
  - $y_i = \theta_0 + \theta_1 x_1 + \cdots + \theta_d x_d$
- Additive Models
  - $y_i = \theta_0 + f_1(x_1) + \cdots + f_d(x_d)$
- Can instantiate functions f with:
  - Linear functions: $f_i(x_i) = \theta_i x_i$
  - Quadratic: $f_i(x_i) = \theta_i^1 x_i + \theta_i^2 x_i^2$
  - Cubic: $f_i(x_i) = \theta_i^1 x_i + \theta_i^2 x_i^2 + \theta_i^3 x_i^3$

Generalization in ML



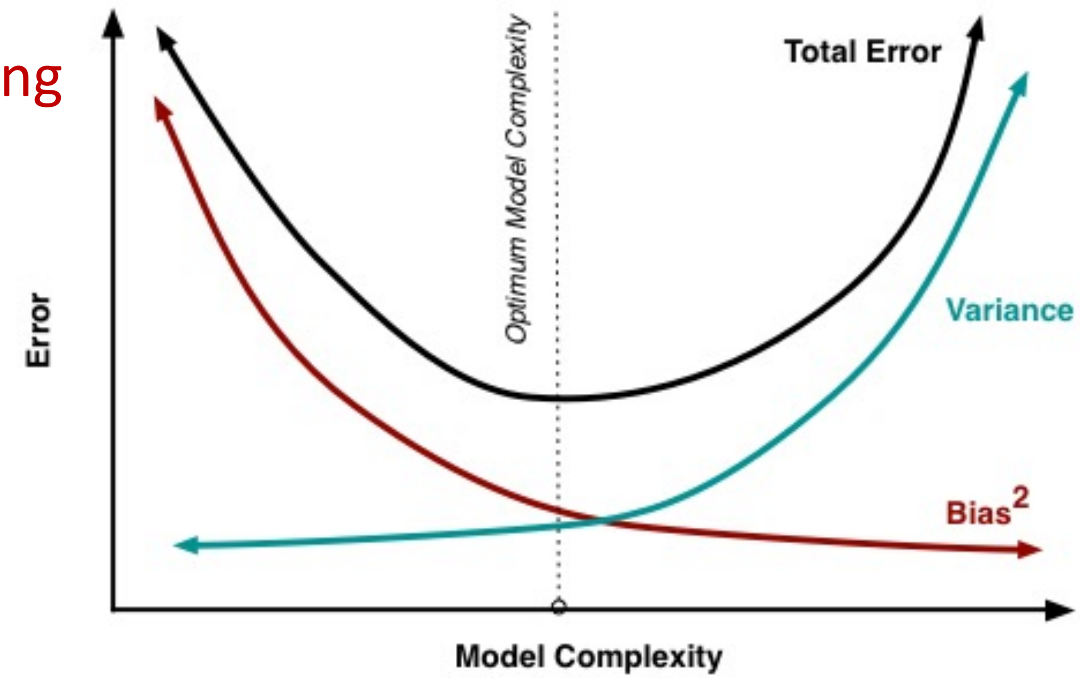
Simple model

Complex model

- Goal is to generalize well on new testing data
- Risk of overfitting to training data

Bias-Variance Tradeoff

Under-fitting



Over-fitting

- Bias = Difference between estimated and true models
- Variance = Model difference on different training sets

MSE is proportional to Bias + Variance

Regularization

- A method for automatically controlling the complexity of the trained model
- Goals
  - Reduce model complexity
  - Reduce variance
  - Mitigate the bias-variance tradeoff
- Main techniques
  - Modify loss function to account for regularization term (Ridge, Lasso)
  - Perform feature selection and fit model on subset of features

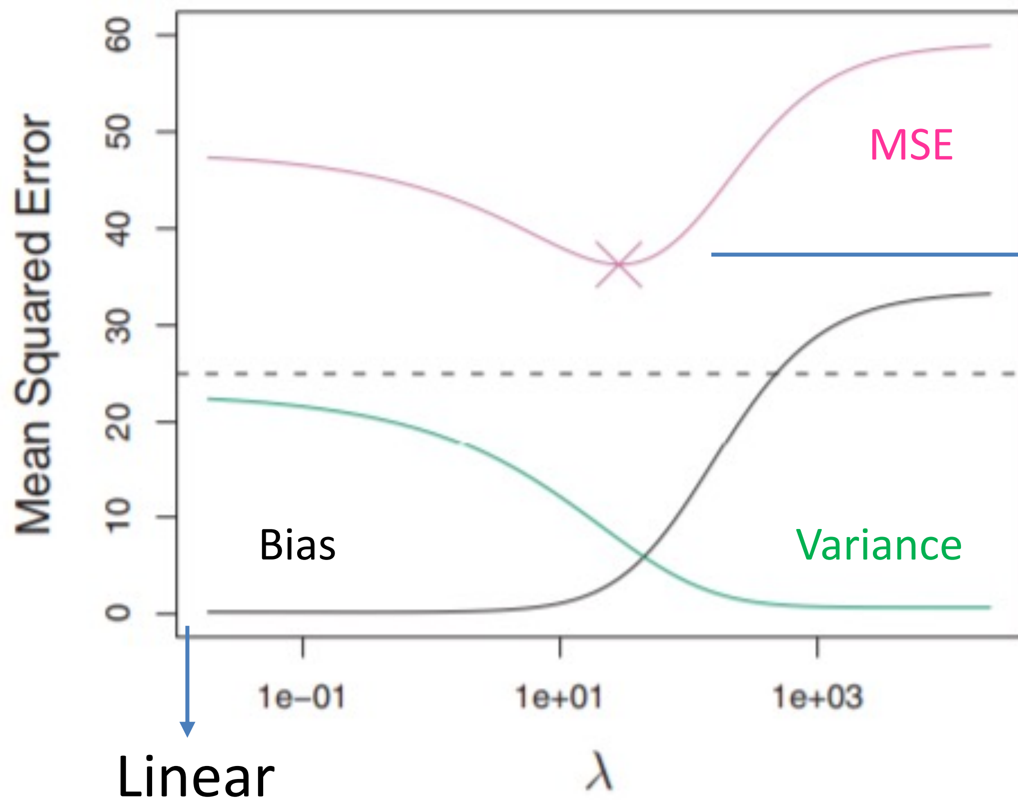
Ridge regression

- Linear regression objective function

$$J(\theta) = \underbrace{\sum_{i=1}^{N} [h_\theta(x_i) - y_i]^2}_{\text{model fit to data}} + \underbrace{\lambda \sum_{j=1}^{d} \theta_j^2}_{\text{regularization}}$$

  - λ is the regularization parameter ($\lambda \geq 0$)
  - No regularization on θ_0!
    - If λ = 0, we train linear regression
    - If λ is large, the coefficients will shrink close to 0

Bias-Variance Tradeoff

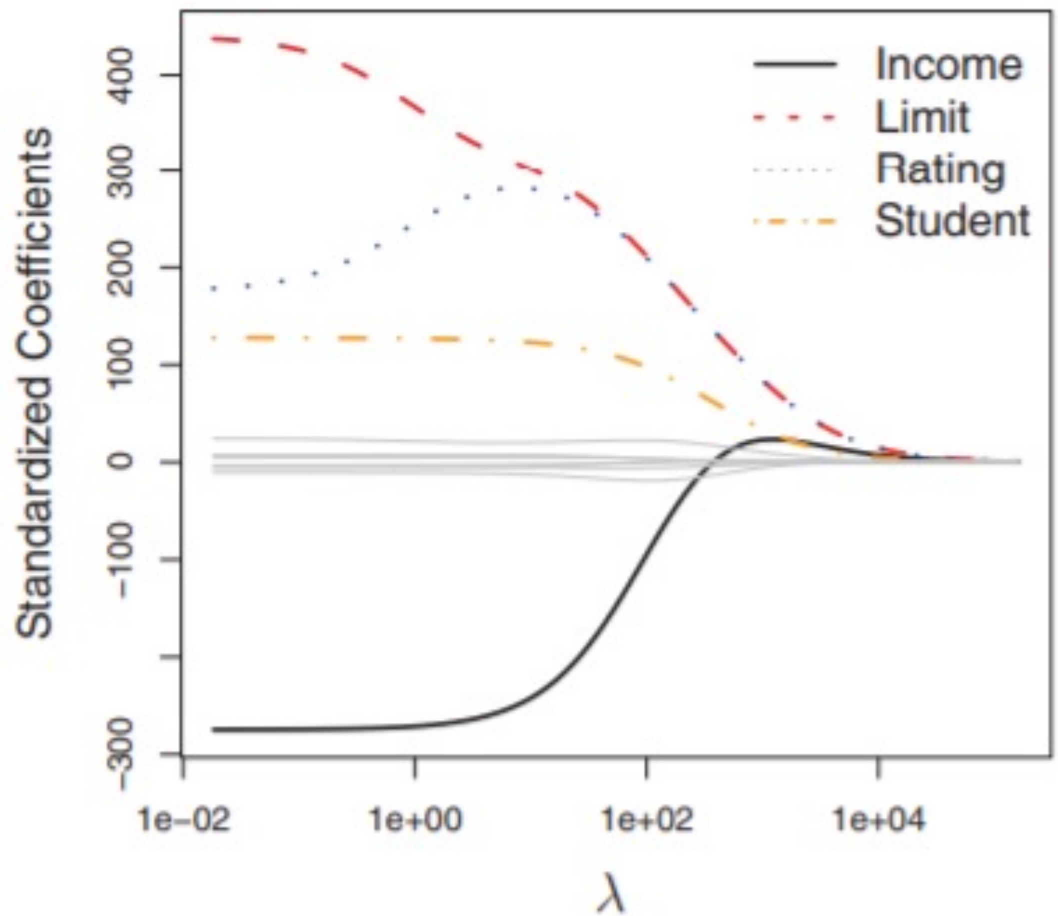


Optimal
Ridge regression

Linear
regression

Reduced model
complexity

Coefficient shrinkage



Predict credit card balance

GD for Ridge Regression

Min Loss

$$J(\theta) = \sum_{i=1}^{N} [h_\theta(x_i) - y_i]^2 + \lambda \sum_{j=1}^{d} \theta_j^2$$

Gradient update: $\theta_0 \leftarrow \theta_0 - \alpha \sum_{i=1}^{N} (h_\theta(x_i) - y_i)$

$$\theta_j \leftarrow \theta_j - \alpha \sum_{i=1}^{N} (h_\theta(x_i) - y_i) x_{ij} - \underbrace{\alpha\lambda\theta_j}_{\text{Regularization}}$$

$$\boxed{\theta_j \leftarrow \theta_j (1 - \alpha\lambda) - \alpha \sum_{i=1}^{N} (h_\theta(x_i) - y_i) x_{ij}}$$

Lasso Regression

$$J(\theta) = \underbrace{\sum_{i=1}^{N} (h_\theta(x_i) - y_i)^2}_{\text{Squared Residuals}} + \lambda \underbrace{\sum_{j=1}^{d} |\theta_j|}_{\text{Regularization}}$$

- L1 norm for regularization
- Results in sparse coefficients
- Issue: gradients cannot be computed around 0
- Method of sub-gradient optimization

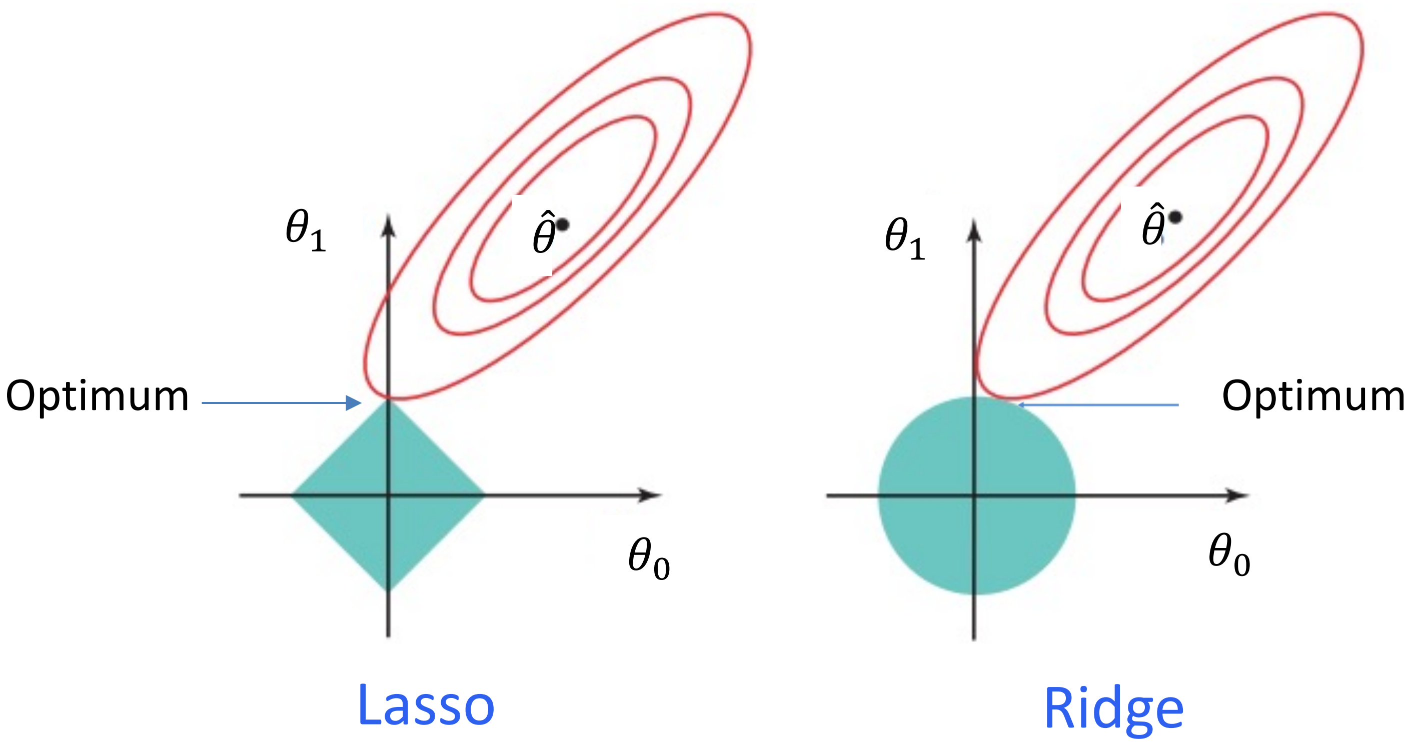
Alternative Formulations

- Ridge
  - L2 Regularization
  - $\min_{\theta} \sum_{i=1}^{N} [h_\theta(x_i) - y_i]^2$ subject to $\sum_{j=1}^{d} |\theta_j|^2 \leq \epsilon$

- Lasso
  - L1 regularization
  - $\min_{\theta} \sum_{i=1}^{N} [h_\theta(x_i) - y_i]^2$ subject to $\sum_{j=1}^{d} |\theta_j| \leq \epsilon$

Lasso vs Ridge

- Ridge shrinks all coefficients
- Lasso sets some coefficients at 0 (sparse solution)
  - Perform feature selection

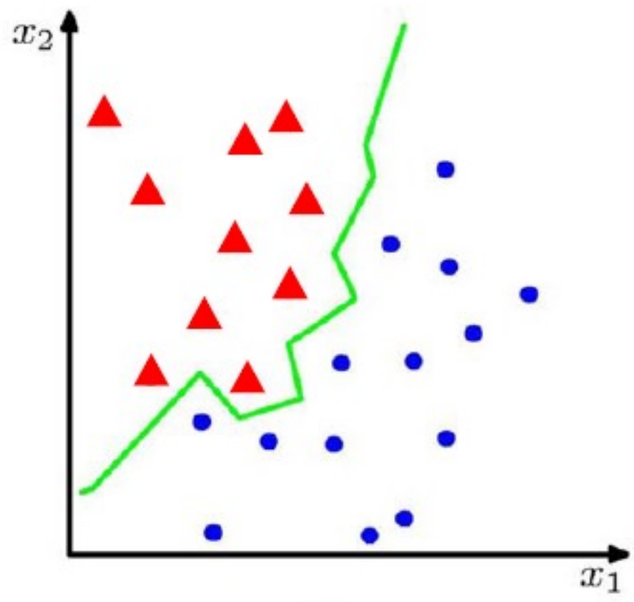


Ridge vs Lasso

- Both methods can be applied to any loss function (regression or classification)
- In both methods, value of regularization parameter λ needs to be adjusted
- Both reduce model complexity

• Ridge	• Lasso
+ Differentiable objective + Gradient descent converges to global optimum - Shrinks all coefficients	- Gradient descent needs to be adapted + Results in sparse model + Can be used for feature selection in large dimensions

Classification



Binary or discrete

- Suppose we are given a training set of N observations

$$\{x_1, \dots, x_N\} \text{ and } \{y_1, \dots, y_N\}, x_i \in R^d, y_i \in \{0, 1\}$$

- Classification problem is to estimate f(x) from this data such that

$$f(x_i) = y_i$$

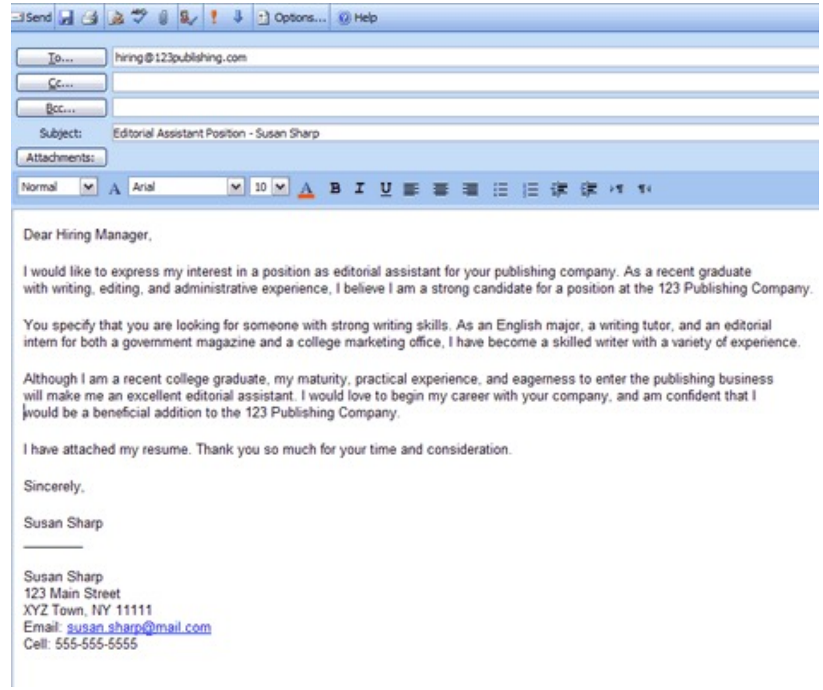
Example 1: Binary classification

Classifying spam email


googleteam GOOGLE LOTTERY WINNER! CONTAC

From: googleteam **To:**
Subject: GOOGLE LOTTERY WINNER! CONTACT YOUR AGENT TO CLAIM YOUR PRIZE.

GOOGLE LOTTERY INTERNATIONAL
INTERNATIONAL PROMOTION / PRIZE AWARD .
(WE ENCOURAGE GLOBALIZATION)
FROM: THE LOTTERY COORDINATOR,
GOOGLE B.V. 44 9459 PE.
RESULTS FOR CATEGORY "A" DRAWS
Congratulations to you as we bring to your notice, the results of the First Ca
inform you that your email address have emerged a winner of One Million (1,(
money of Two Million (2,000,000.00) Euro shared among the 2 winners in thi
email addresses of individuals and companies from Africa, America, Asia, Au
CONGRATULATIONS!
Your fund is now deposited with the paying Bank. In your best interest to avo
award strictly from public notice until the process of transferring your claims
NOTE: to file for your claim, please contact the claim department below on e
****



To: hiring@123publishing.com
Subject: Editorial Assistant Position - Susan Sharp

Dear Hiring Manager,

I would like to express my interest in a position as editorial assistant for your publishing company. As a recent graduate with writing, editing, and administrative experience, I believe I am a strong candidate for a position at the 123 Publishing Company.

You specify that you are looking for someone with strong writing skills. As an English major, a writing tutor, and an editorial intern for both a government magazine and a college marketing office, I have become a skilled writer with a variety of experience.

Although I am a recent college graduate, my maturity, practical experience, and eagerness to enter the publishing business will make me an excellent editorial assistant. I would love to begin my career with your company, and am confident that I would be a beneficial addition to the 123 Publishing Company.

I have attached my resume. Thank you so much for your time and consideration.

Sincerely,

Susan Sharp

Susan Sharp
123 Main Street
XYZ Town, NY 11111
Email: susan.sharp@mail.com
Cell: 555-555-5555

Content-related features

- Use of certain words
- Word frequencies
- Language
- Sentence

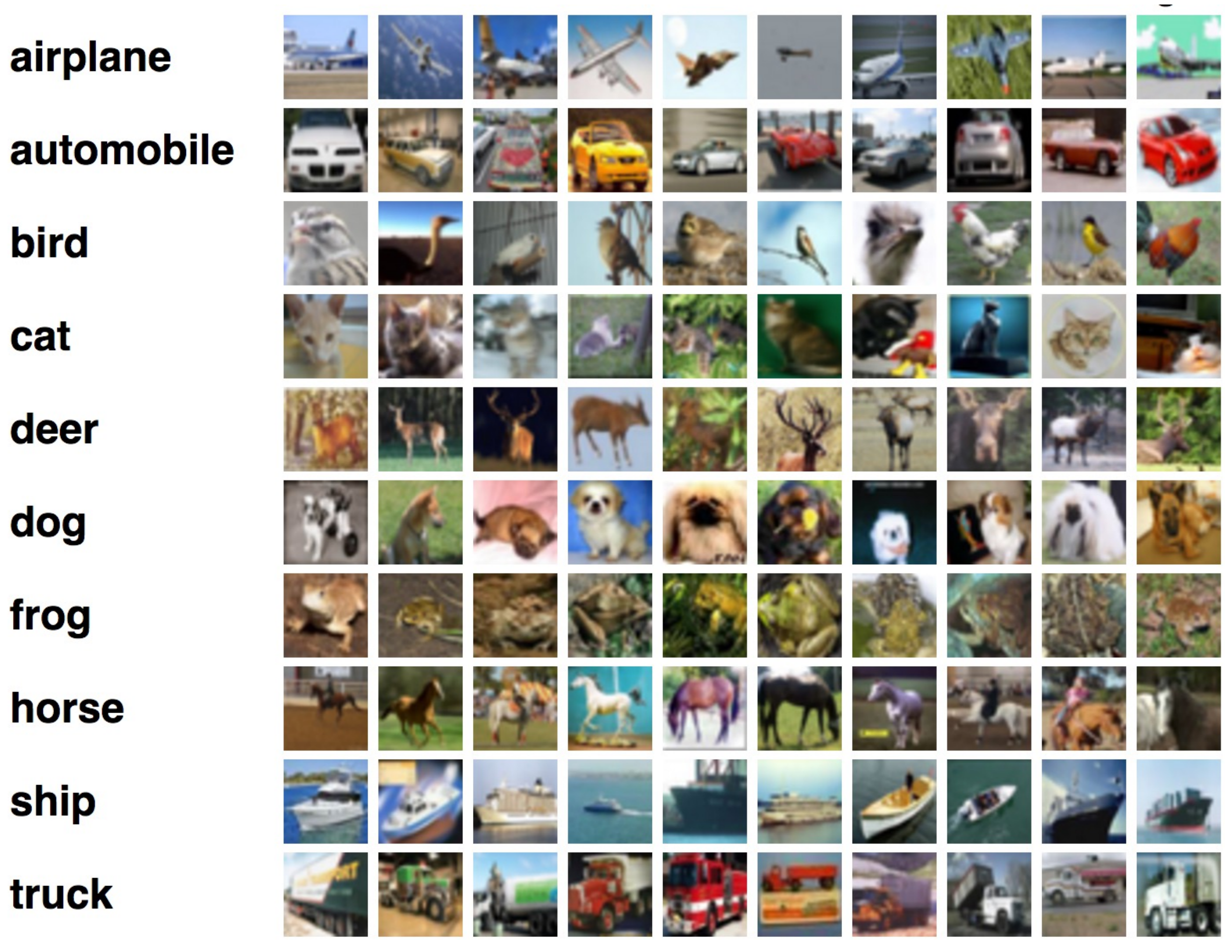
Structural features

- Sender IP address
- IP blacklist
- DNS information
- Email server
- URL links (non-matching)

Binary classification: SPAM or HAM

Example 2: Multi-class classification

Image classification



Multi-class classification

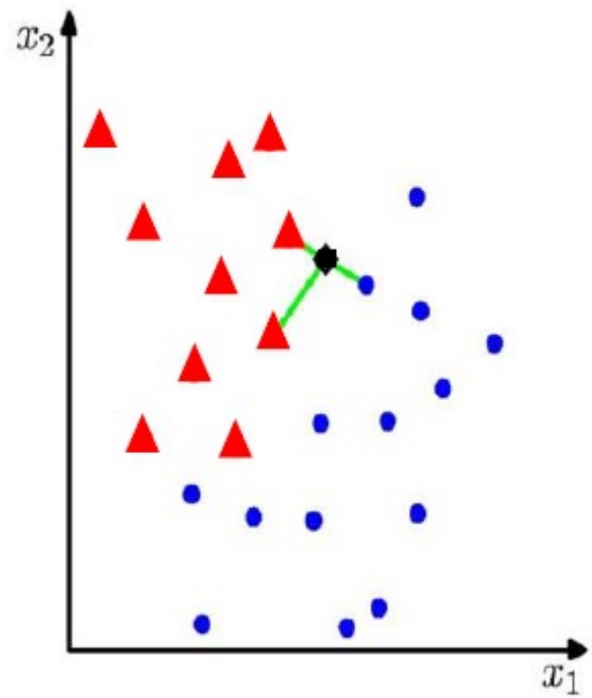
K Nearest Neighbour (K-NN) Classifier

Algorithm

- For each test point, x, to be classified, find the K nearest samples in the training data
- Classify the point, x, according to the majority vote of their class labels

e.g. K = 3

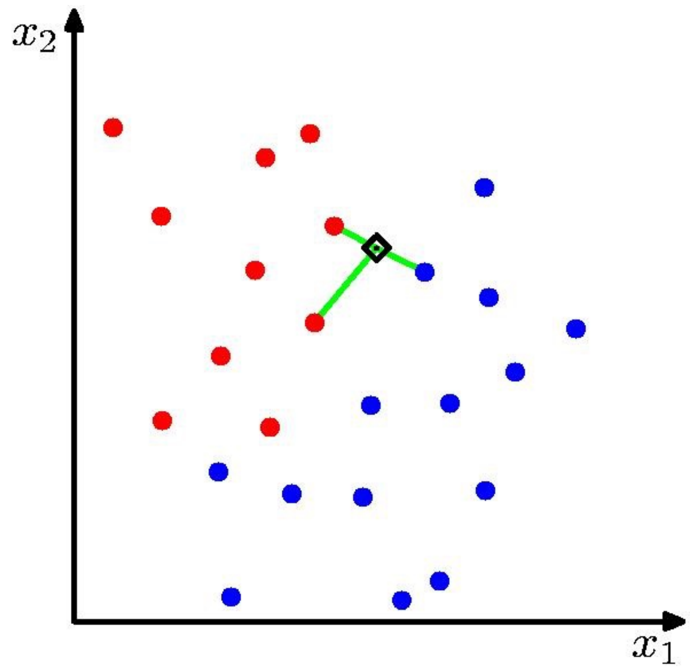
- applicable to multi-class case



Distance Metrics

- Euclidean Distance $$\sqrt{\left(\sum_{i=1}^{k}(x_i - y_i)^2\right)}$$
- Manhattan Distance $$\sum_{i=1}^{k}|x_i - y_i|$$
- Minkowski Distance $$\left(\sum_{i=1}^{k}(|x_i - y_i|)^q\right)^{\frac{1}{q}}$$

kNN



- Algorithm (to classify point x)
  - Find k nearest points to x (according to distance metric)
  - Perform majority voting to predict class of x
- Properties
  - Does not learn any model in training!
  - Instance learner (needs all data at testing time)

Acknowledgements

- Slides made using resources from:
  - Andrew Ng
  - Eric Eaton
  - David Sontag
- Thanks!