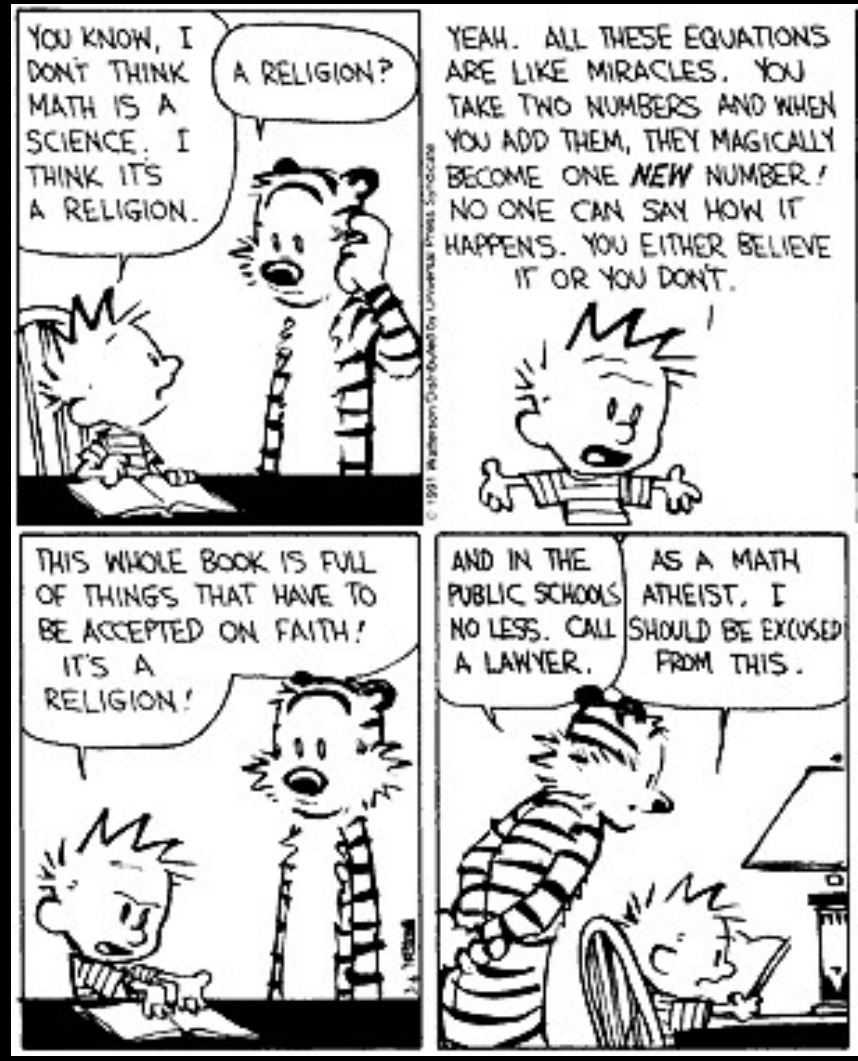


INFO 251: Applied Machine Learning

Linear Models

Announcements

- Assignment 3 due Monday
- Quiz 1 scheduled for March 2, first ~40 minutes of class
  - 10-15 multiple choice and short-answer questions
  - See piazza for details on quiz timing

Course Outline

- Causal Inference and Research Design
  - Experimental methods
  - Non-experiment methods
- Machine Learning
  - Design of Machine Learning Experiments
  - **Linear Models and Gradient Descent**
  - Non-linear models
  - Neural models
  - Unsupervised Learning
  - Practicalities, Fairness, Bias
- Special topics

Key Concepts (last lecture)

- Cost Functions
- Gradient Descent
- Local and global minima
- Convex functions
- Incremental vs. Batch GD
- Learning rates
- Feature scaling

Outline

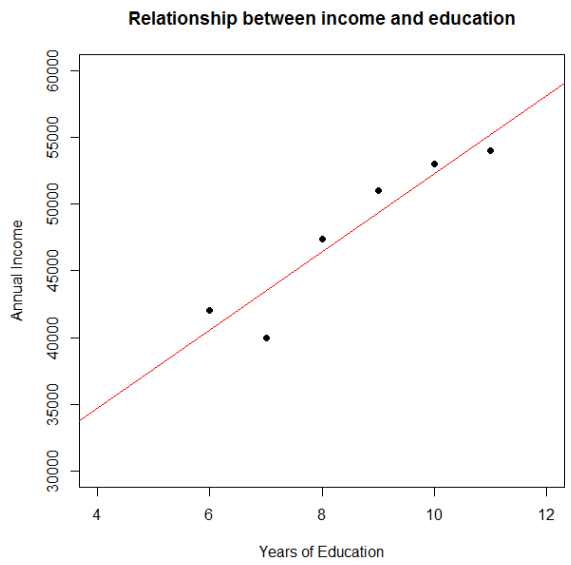
- Regularization
- Ridge and Lasso
- Logistic regression (inference)
- Logistic regression (prediction)
- Support vector machines
- Kernels

Key Concepts (this lecture)

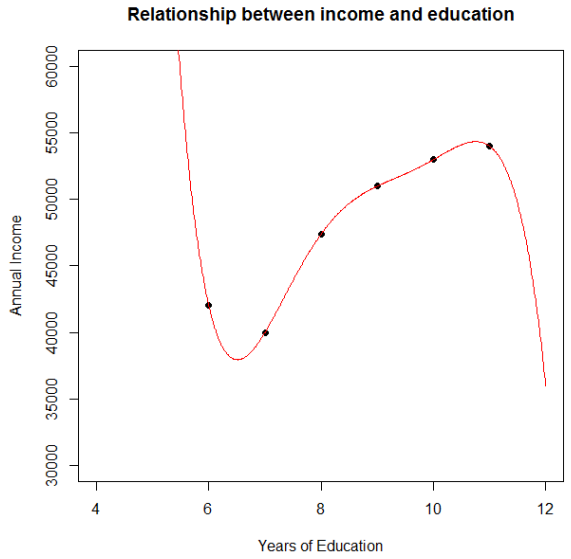
- Regularization
- Ridge
- Lasso
- Logistic regression
- Simplified sigmoid cost function
- Odds ratios
- Overfitting revisited
- Support vector machines
- Hard vs. soft margins
- Kernel functions

Overfitting revisited

- Overfitting: If we have too many features, our model may fit the training set very well, but fail to generalize to new examples



$$wages_i = \alpha + \beta * educ_i + error_i$$



$$wages_i = \alpha + \beta_1 * educ_i + \ldots + \beta_5 * educ_i^5 + error_i$$

Overfitting: Solutions

- Later in the course:
  - Feature selection
  - Model selection
  - Dimensionality reduction
- Now: Regularization
  - Keep all the features, but reduce magnitude of additional parameters

Regularization: Intuition

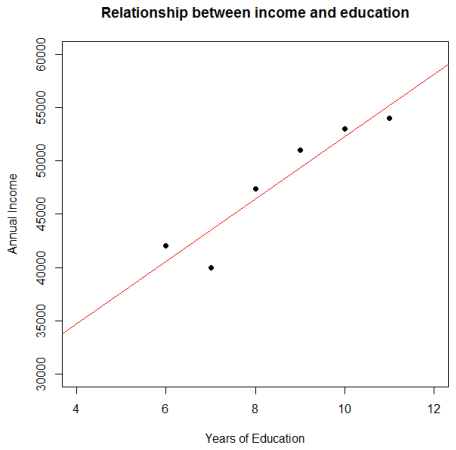
- Occam's Razor
  - A principle of parsimony, economy, or succinctness used in problem-solving. It states that among competing hypotheses, the hypothesis with the fewest assumptions should be selected.

Ockham chooses a razor

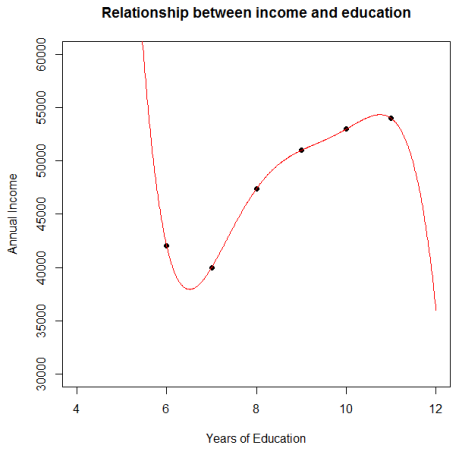
See: Domingos, P. The role of Occam's razor in knowledge discovery. *Data Mining and Knowledge Discovery 3* (1999), 409–425.

Regularization: Intuition

- Idea: Add a cost penalty for additional complexity in the model
- Example: polynomial regression
  - Model: $Y_i = \theta_0 + \theta_1 X_i + \ ... + \theta_k X_i^k$
  - Parameters: $\theta_0, ..., \theta_k$
  - Original "Cost": $J(\theta) = \frac{1}{2N} \sum_{i=1}^{N} \left(\theta_0 + \theta_1 X_i + \ ... + \theta_k X_i^k - Y_i\right)^2$



$wages_i = \alpha + \beta * educ_i + error_i$



$wages_i = \alpha + \beta_1 * educ_i + ... + \beta_5 * educ_i^5 + error_i$

Regularization: Intuition

- Original Cost
  - $J(\theta) = \frac{1}{2N}\sum_{i=1}^{N}\left(\theta_0 + \theta_1 X_i + \ \ldots + \theta_k X_i^k - Y_i\right)^2$
- Intuitive Goal
  - $J(\theta) = \frac{1}{2N}\sum_{i=1}^{N}\left(\theta_0 + \theta_1 X_i + \ \ldots + \theta_k X_i^k - Y_i\right)^2 + C(\theta_1, \ldots, \theta_k)$
- Penalized (Regularized) Cost
  - $J(\theta) = \frac{1}{2N}\sum_{i=1}^{N}\left(\theta_0 + \theta_1 X_i + \ \ldots + \theta_k X_i^k - Y_i\right)^2 + \lambda \sum_{j=1}^{k} \theta_j^2$

penalty

"Ridge" coefficient

Regularization parameter

Regularization and Linear Regression

- Original Gradient Descent
  - Repeat until convergence:

    α <- $\alpha - R\frac{\partial}{\partial\alpha}J(\alpha,\beta)$

    β <- $\beta - R\frac{\partial}{\partial\beta}J(\alpha,\beta)$
  - Original derivative of J (in linear regression, $Y_i = \alpha + \beta X_i$)

    α <- $\alpha - R\frac{1}{N}\sum_{i=1}^{N}(\alpha + \beta X_i - Y_i)$

    β <- $\beta - R\frac{1}{N}\sum_{i=1}^{N}(\alpha + \beta X_i - Y_i) \quad X_i$
- Regularized version has new partial derivatives:

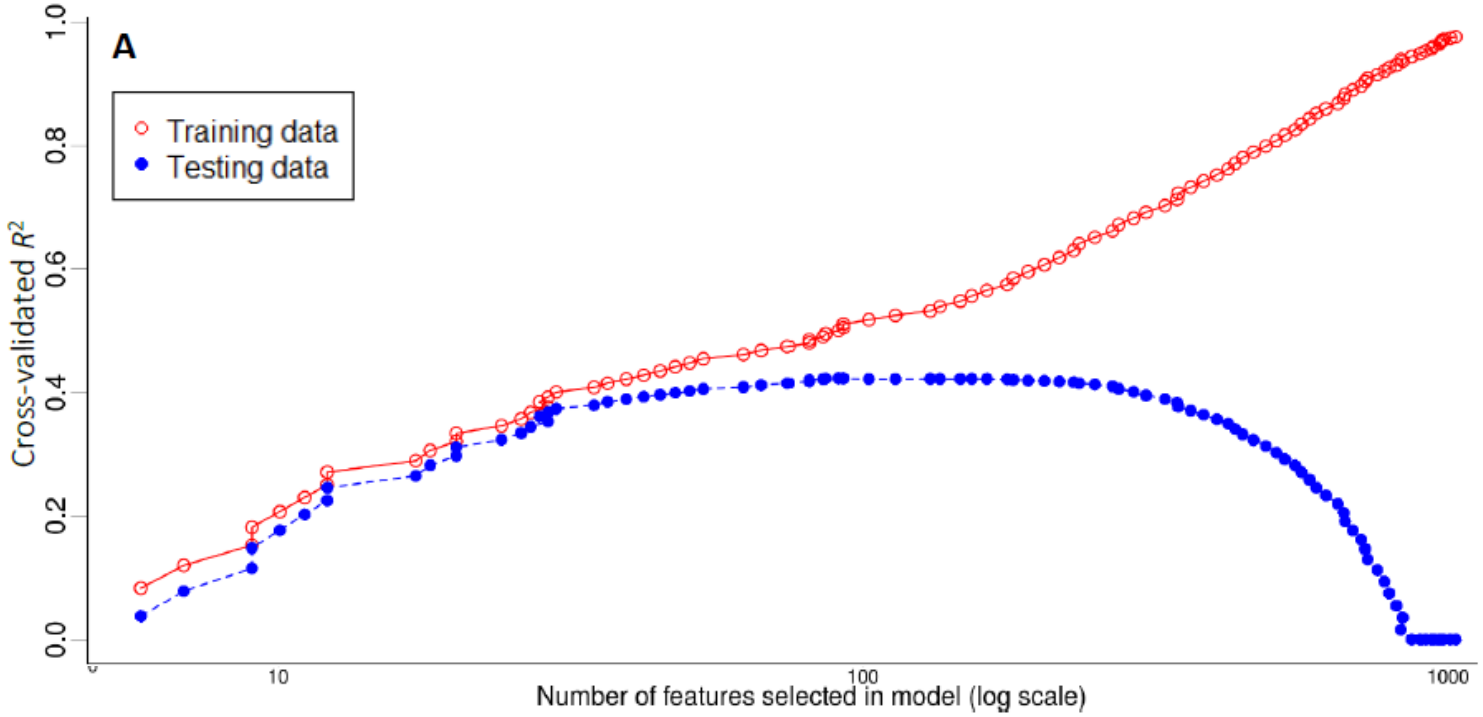
  β <- $\beta - R\left[\frac{1}{N}\sum_{i=1}^{N}(\alpha + \beta X_i - Y_i) \quad X_i \; + \frac{\lambda}{N}\beta\right]$
- Rewritten:

  β <- $\beta\left(1 - R\frac{\lambda}{N}\right) - R\frac{1}{N}\sum_{i=1}^{N}(\alpha + \beta X_i - Y_i) \quad X_i$

Regularization: Some notes

$$J(\theta) = \frac{1}{2N}\sum_{i=1}^{N}\left(\theta_0 + \theta_1 X_i + \ \ldots + \theta_k X_i^k - Y_i\right)^2 + \lambda \sum_{j=1}^{k} \theta_j^2$$

- How to select λ?
  - Cross validation!



Regularization: Some notes

$$J(\theta) = \frac{1}{2N}\sum_{i=1}^{N}\left(\theta_0 + \theta_1 X_i + \ \ldots + \theta_k X_i^k - Y_i\right)^2 + \lambda \sum_{j=1}^{k} \theta_j^2$$

- What happens in regularization if features are in different units?
  - Penalty on different scales
  - Solution: Scale features

Outline

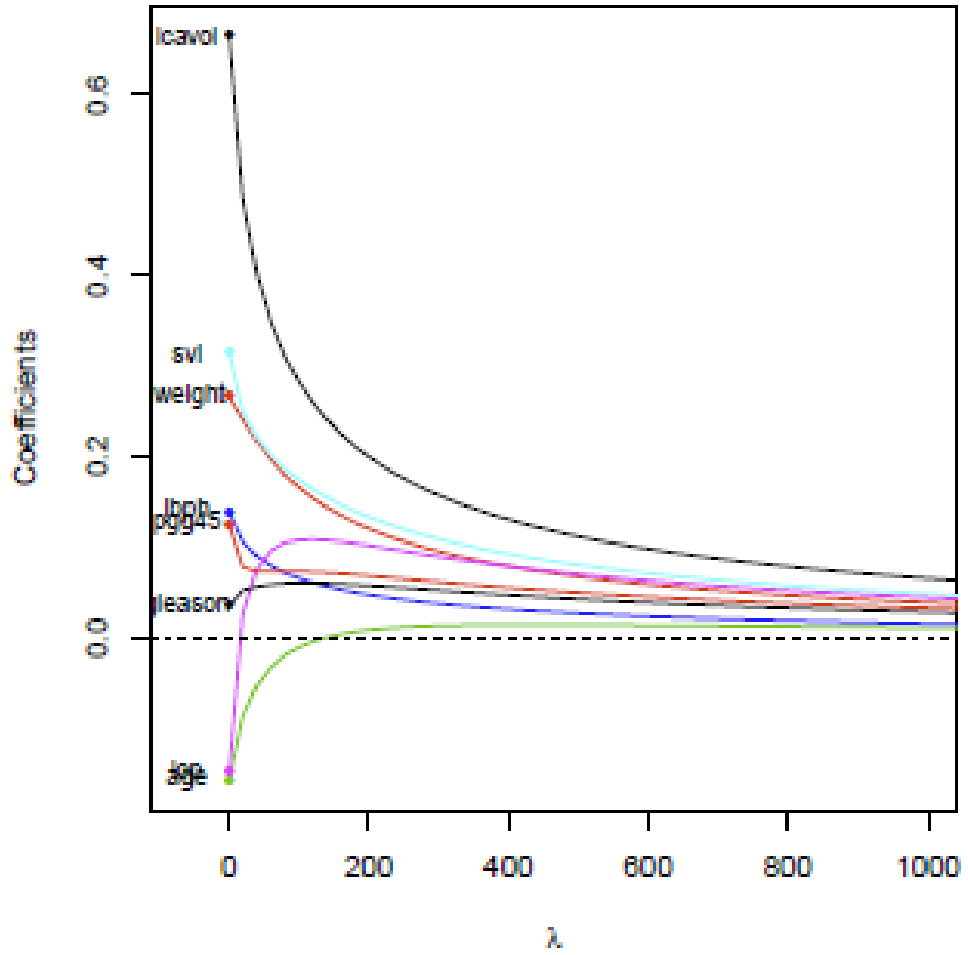
- Regularization
- **Ridge and Lasso**
- Logistic regression (inference)
- Logistic regression (prediction)
- Support vector machines
- Kernels

"Ridge"

$$J(\theta) = \frac{1}{2N}\sum_{i=1}^{N}\left(\theta_0 + \theta_1 X_i + \ \ldots + \theta_k X_i^k - Y_i\right)^2 + \lambda \sum_{j=1}^{k} \theta_j^2$$

- L_2 norm (ridge regression): penalty proportional to θ^2
  - Works best when a subset of the true coefficients are small
  - Will never set coefficients to zero exactly
  - Cannot perform variable selection in the linear model
  - Coefficients harder to interpret

Ridge: Coefficient plot



Source: Ryan Tibshirani

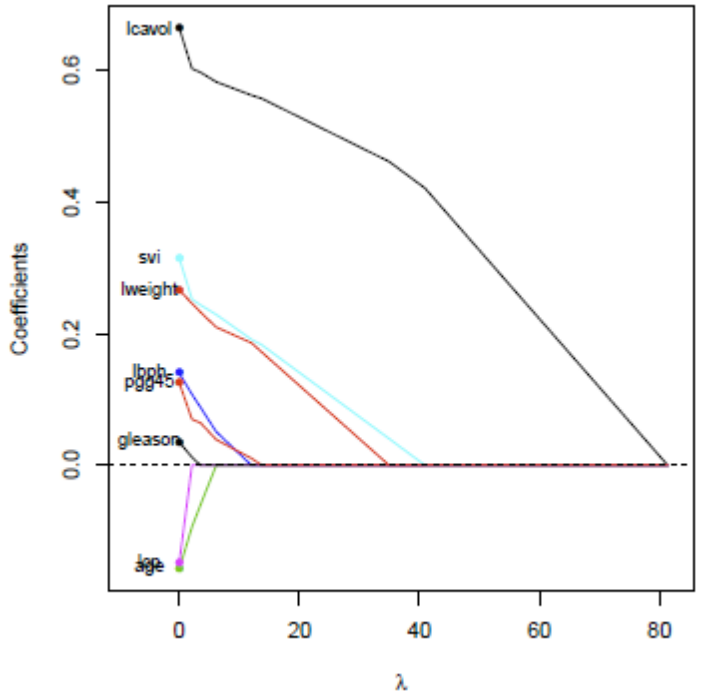
LASSO

$$J(\theta) = \frac{1}{2N}\sum_{i=1}^{N}\left(\theta_0 + \theta_1 X_i + \ \ldots + \theta_k X_i^k - Y_i\right)^2 + \lambda \sum_{j=1}^{k} |\theta_j|$$

- L_1 norm (lasso regression): penalty proportional to θ
  - Selects more relevant features and discards the others, vs. Ridge regression which reduces parameters but doesn't drive to zero
    - See ESL pp. 68
    - Andrew, Galen; Gao, Jianfeng (2007). "Scalable training of L_1-regularized log-linear models". *Proceedings of the 24th International Conference on Machine Learning*
  - Not differentiable
  - Coefficients still difficult to interpret, though "post-lasso" versions can reduce bias (e.g., Belloni & Chernozhukov)

LASSO: Coefficient plot

- Least Absolute Selection and Shrinkage Operator
  - See ESL section 3.4
  - Tibshirani (1996), "Regression Shrinkage and Selection via the Lasso"



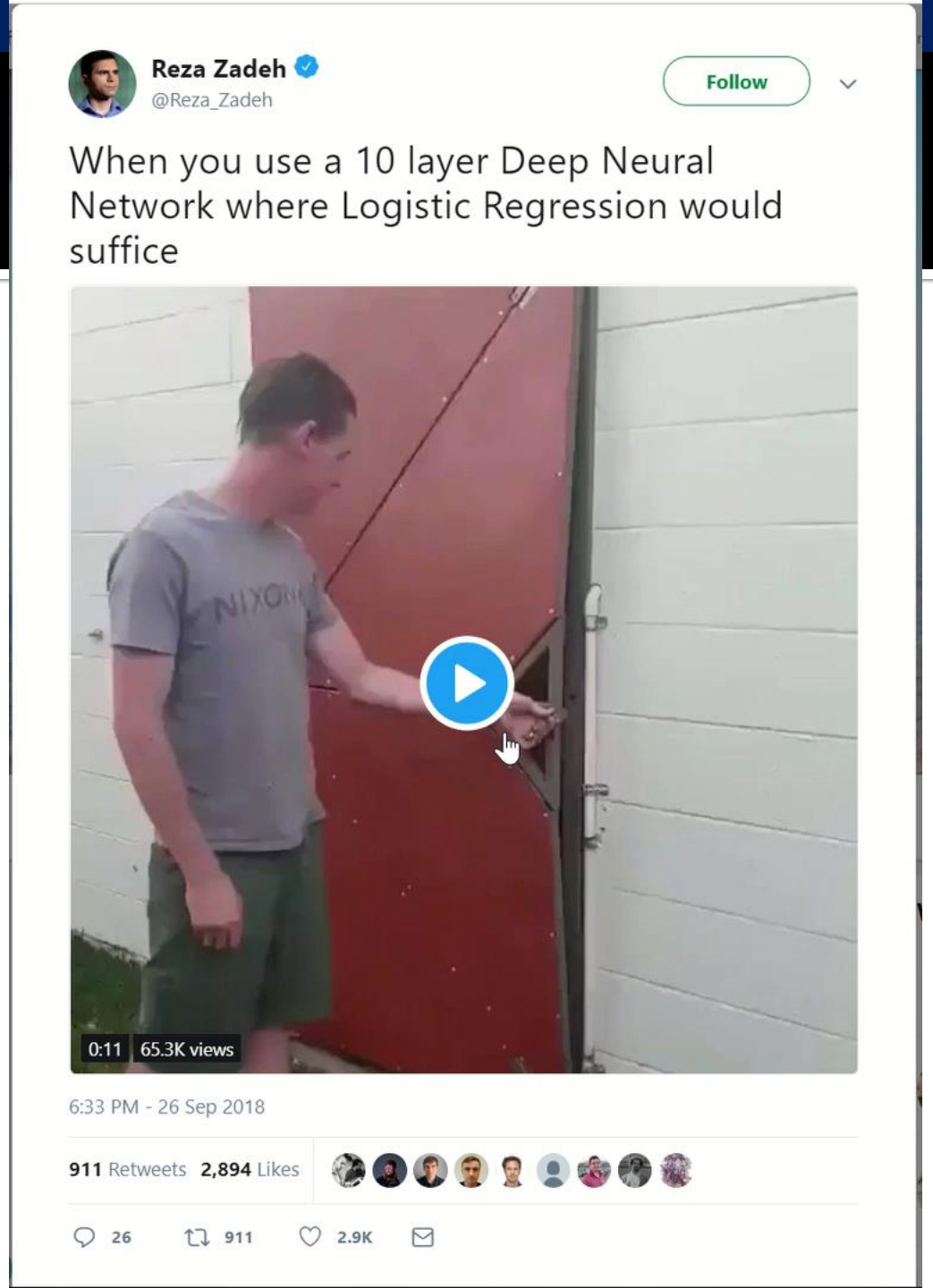
Other forms of Regularization

Model	Fit measure	Entropy measure[4][5]
AIC/BIC	$\|Y - X\beta\|_2$	$\|\beta\|_0$
Ridge regression	$\|Y - X\beta\|_2$	$\|\beta\|_2$
Lasso[6]	$\|Y - X\beta\|_2$	$\|\beta\|_1$
Basis pursuit denoising	$\|Y - X\beta\|_2$	$\lambda\|\beta\|_1$
Rudin-Osher-Fatemi model (TV)	$\|Y - X\beta\|_2$	$\lambda\|\nabla\beta\|_1$
Potts model	$\|Y - X\beta\|_2$	$\lambda\|\nabla\beta\|_0$
RLAD[7]	$\|Y - X\beta\|_1$	$\|\beta\|_1$
Dantzig Selector[8]	$\|X^\top(Y - X\beta)\|_\infty$	$\|\beta\|_1$
SLOPE[9]	$\|Y - X\beta\|_2$	$\sum_{i=1}^{p} \lambda_i \|\beta\|_{(i)}$

A linear combination of the LASSO and ridge regression methods is elastic net regularization.

Outline

- Regularization
- Ridge and Lasso
- **Logistic regression (inference)**
- Logistic regression (prediction)
- Support vector machines
- Kernels


Reza Zadeh
@Reza_Zadeh
Follow
When you use a 10 layer Deep Neural Network where Logistic Regression would suffice
0:11
65.3K views
6:33 PM - 26 Sep 2018
911 Retweets 2,894 Likes
26
911
2.9K

Logistic regression: Basics

- Logistic regression
  - Models the (linear) relationship between one or more independent variables and one binary dependent variable
  - As with linear regression, can be used for inference and prediction; used to predict (and classify) binary outcomes

Inference	**Prediction**
What is the effect of an additional year of schooling on whether an individual is eligible for welfare?	Do we predict that an individual with 6 years of education will be eligible for welfare?
What caused the server to go down last week?	Will the server go down this week?
How big a factor is "home court advantage" in whether our team will win or lose?	Are we going to win this week?

Logistic Regression: Idea

- Logistic Regression: Model
  - The logistic regression model assumes that the independent variables have a linear relationship with the logit transformation of the dependent variable

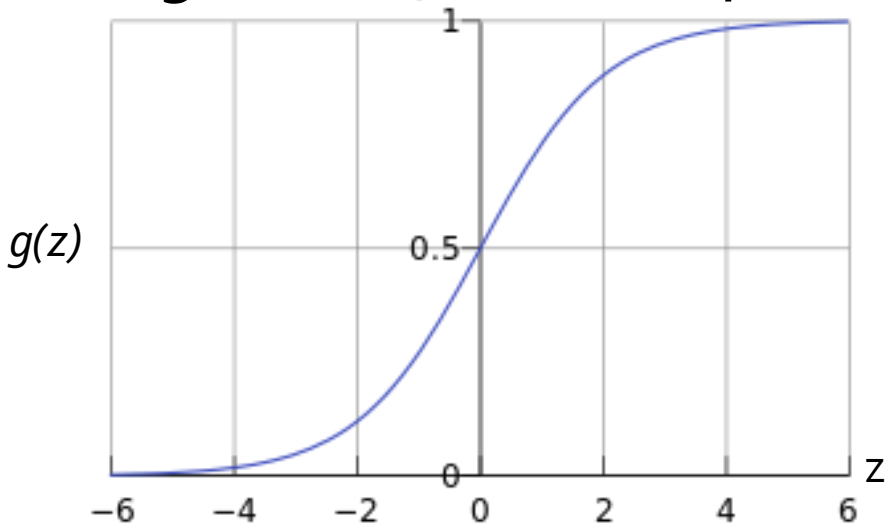
Logistic Regression: Idea

- Logit transformation maps probabilities to log of odds ratios
  - Odds ratio: probability success / probability failure
  - Example: Probability success = 0.8
    - Odds ratio is 4
    - "Odds of success are 4 to 1"

- In other words:
  - $\text{logit}(p) = \log\left(\frac{p}{1-p}\right) = \alpha + \beta X + \cdots$
  - $p = \frac{e^{\alpha+\beta X+\cdots}}{1+e^{\alpha+\beta X+\cdots}}$

Logistic Regression: The logistic function

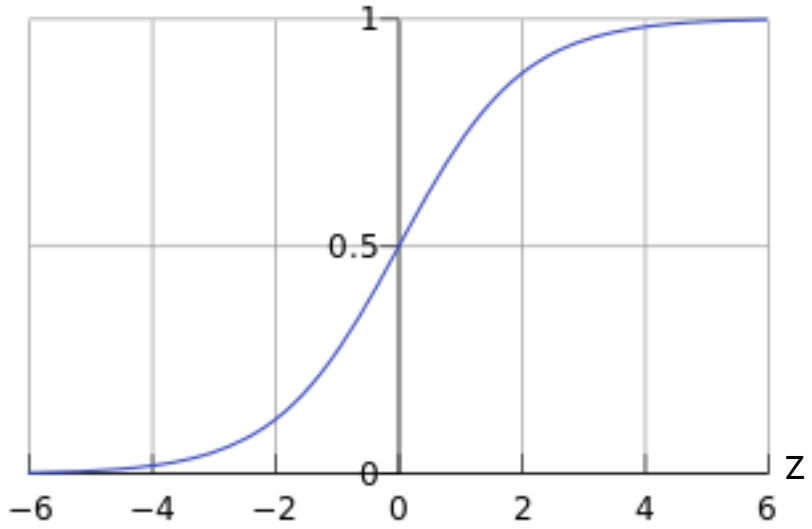
- Logistic (sigmoid) function: $g(z) = \frac{e^z}{e^z+1} = \frac{1}{1+e^{-z}}$
  - Transforms [-∞,+∞]=>[0,1]
  - Constrains output of our model between 0 and 1
  - In logistic regression, $z = \alpha + \beta X + \cdots$



$$g(z) = \frac{1}{1 + e^{-(\alpha+\beta X+\cdots)}}$$

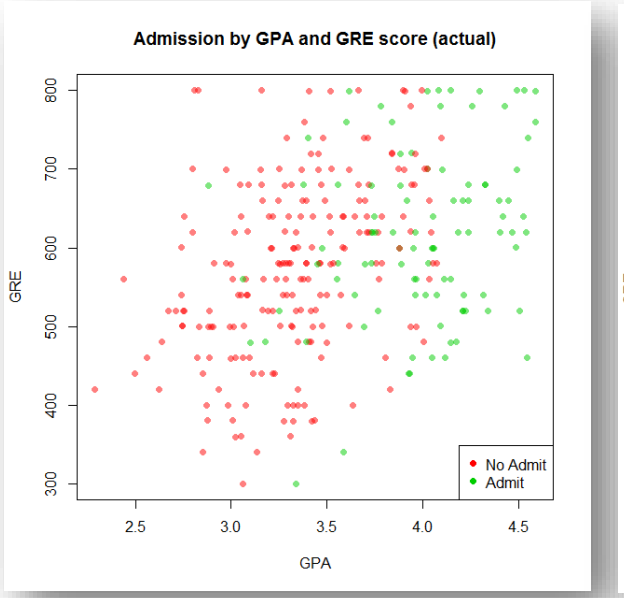
Logistic Regression: Decision Boundary

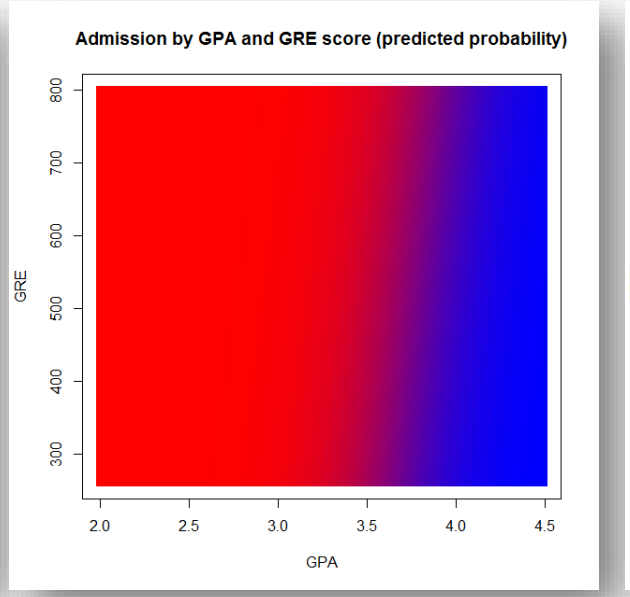
- Interpretation of g(z)?
  - Probability that y=1
  - $P(y = 1|x: \alpha, \beta)$
- Simple classifier
  - Predict y=1 if g(z)≥0.5
  - Predict y=0 if g(z)<0.5
- How does this relate to values of z?
  - Predict y=1 if z≥0
  - Predict y=0 if z<0
  - Typically, $z = \alpha + \beta X + \cdots$

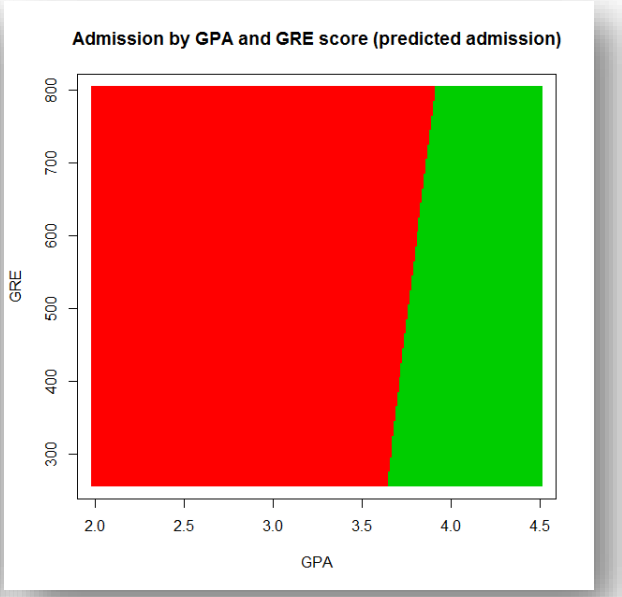


Logistic Regression: Example

- Example: admission vs. GRE and GPA
  1. Start with raw data
  2. Fit logistic regression
  3. Threshold converts g(z) to classification







Logistic Regression: Coefficients

- How do we interpret the coefficients from a logistic regression?
  - The coefficient tells you what change to expect in the *log odds ratio* of your dependent variable, for a one-unit increase in your independent variable.

- Ways to make this more intelligible
  - Convert from log odds ratio to odds ratio
    - $\exp(\beta)$
  - Convert from odds ratio to probability
    - $\frac{odds}{1+odds}$

Logistic Regression: Coefficients

- Example with no predictor variables
  - Likelihood of being honor student
    - $\text{logit}(honor_i) = \alpha + \epsilon_i$
    - i.e., log(p/(1-p)) = -1.12546
  - Note that p = exp(-1.12546)/(1+exp(-1.12546)) = .245

```
Logistic regression                                     Number of obs   =        200
                                                        LR chi2(0)      =       0.00
                                                        Prob > chi2     =          .
Log likelihood = -111.35502                             Pseudo R2       =     0.0000

------------------------------------------------------------------------------
         hon |      Coef.   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
   intercept |   -1.12546   .1644101    -6.85   0.000    -1.447697   -.8032217
------------------------------------------------------------------------------
```

hon	Freq.	Percent	Cum.
0	151	75.50	75.50
1	49	24.50	100.00
Total	200	100.00	

Logistic Regression: Coefficients

- Example with single predictor variable
  - Likelihood of honor student, by major
    - $\text{logit}(honor_i) = \alpha + \beta STEM_i + \epsilon_i$
  - exp(0.593) = 1.809
    - (this is the odds ratio)
    - (corresponds to p=0.644)
  - The odds radio can also be seen in the cross-tabs:
    - Odds for non-STEM: 0.23 (17/74)
    - Odds for STEM: 0.42 (32/77)
    - Odds for STEM 81% higher
      - 0.42 / 0.23 = 1.809
      - 0.644 / (1-0.644) = 1.809

```
Logistic regression                               Number of obs   =        200
                                                  LR chi2(1)      =       3.10
                                                  Prob > chi2     =     0.0781
Log likelihood = -109.80312                       Pseudo R2       =     0.0139

------------------------------------------------------------------------------
         hon |      Coef.   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
        stem |    .5927822   .3414294     1.74   0.083    -.0764072    1.261972
   intercept |   -1.470852   .2689555    -5.47   0.000    -1.997995   -.9437087
------------------------------------------------------------------------------
```

```
           |        stem
       hon |       yes        no |     Total
-----------+---------------------+----------
         0 |        74        77 |       151
         1 |        17        32 |        49
-----------+---------------------+----------
     Total |        91       109 |       200
```

Outline

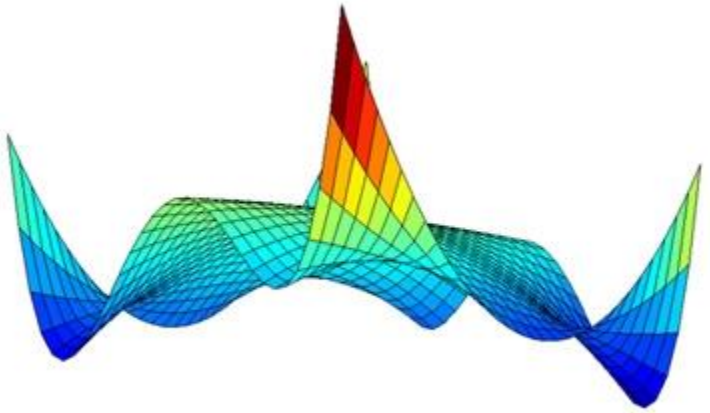
- Regularization
- Ridge and Lasso
- Logistic regression (inference)
- **Logistic regression (prediction)**
- Support vector machines
- Kernels

Logistic Regression: General formulation

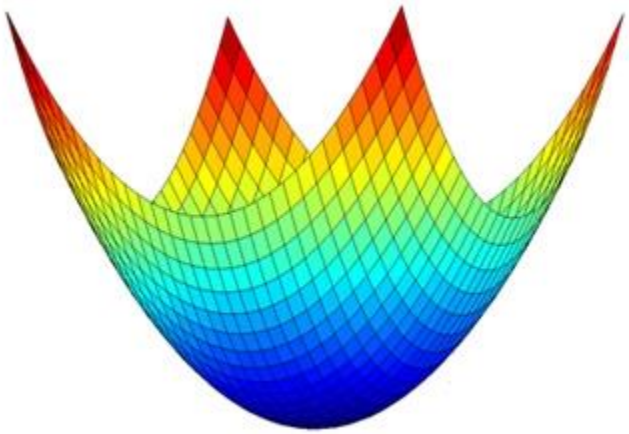
- Model ("hypothesis")
  - $P(Y_i = 1|x; \theta) = g(z) = \frac{1}{1+e^{-z}}$
- Parameters
  - θ are the parameters, often α, β
  - If $\theta = (\alpha, \beta)$, $P(Y_i = 1) = \frac{1}{1+e^{-(\alpha+\beta X_i)}}$
- Cost Function
  - $J(\theta) = \frac{1}{N}\sum_{i=1}^{N} \text{Cost}(\hat{Y}_i, Y_i)$
  - (more on this shortly)
- Objective
  - $\min_{\theta} J(\theta)$

Logistic Regression: Cost function

- Cost Function
  - Linear regression: $J(\alpha, \beta) = \frac{1}{2N}\sum_{i=1}^{N}(Y_i - \alpha - \beta X_i)^2$
  - Why not $J(\alpha, \beta) = \frac{1}{2N}\sum_{i=1}^{N}\left(Y_i - \frac{1}{1+e^{-\alpha-\beta X_i}}\right)^2$

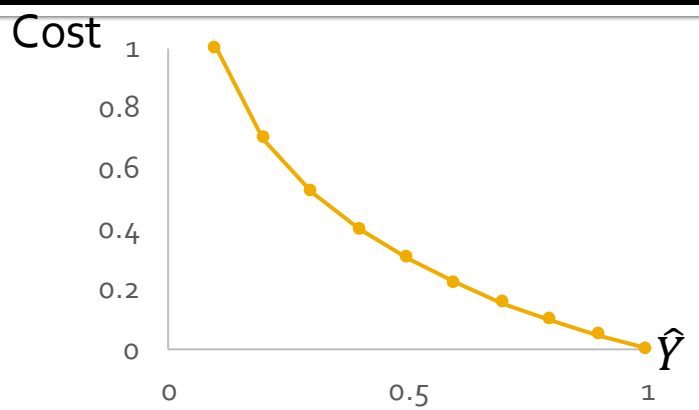


- Not convex ☹
  - Sigmoid function is complex, $J(\alpha, \beta)$ is not convex…
  - Susceptible to local minima, want to convert to something convex

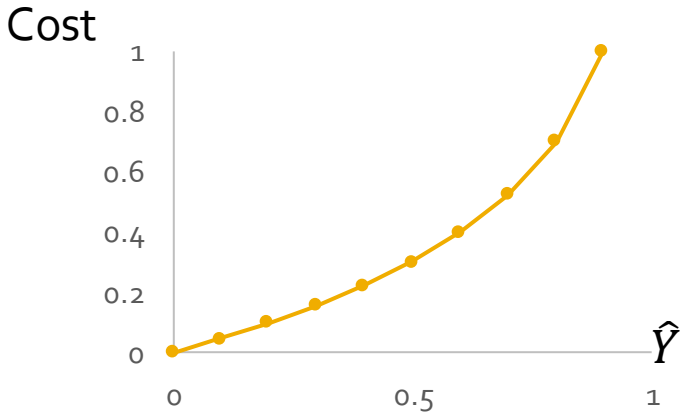


Logistic Regression: Cost function

- Cost Function (think of $\hat{Y}_i = \frac{1}{1+e^{-(\alpha+\beta X_i)}}$)
  - $\text{Cost}(\hat{Y}_i, Y_i) = \begin{cases} -\log(\hat{Y}_i) & \text{if } Y_i = 1 \\ -\log(1-\hat{Y}_i) & \text{if } Y_i = 0 \end{cases}$
  - $\text{Cost}(\hat{Y}_i, Y_i) = -Y_i \cdot \log(\hat{Y}_i) - (1-Y_i) \cdot \log(1-\hat{Y}_i)$



- This is convex:
  - If $Y_i = 1$, what is cost if $\hat{Y}_i = 1$? What if $\hat{Y}_i = 0$?
    - No cost if model predicts 1
    - Penalizes mistakes
  - If $Y_i = 0$, what is cost if $\hat{Y}_i = 1$? if $\hat{Y}_i = 0$?
    - No cost if model predicts 0
    - Penalizes mistakes



Logistic Regression: Gradient Descent

- How to minimize $J(\theta)$?
  - $J(\theta) = -\frac{1}{N}\sum_{i=1}^{N} Y_i \cdot \log \hat{Y}_i + (1 - Y_i) \log\left(1 - \hat{Y}_i\right)$
- Gradient Descent!
  - $\theta \leftarrow \theta - R\frac{\partial}{\partial\theta}J(\theta)$
- With revised cost function, $\frac{\partial}{\partial\theta}J(\theta) = -\frac{1}{N}\sum_{i=1}^{N}(Y_i - \hat{Y}_i)X_i$
  - Note similarities to linear regression! But not identical:
  - Logistic regression: $\hat{Y}_i = \frac{1}{1+e^{-(\alpha+\beta X_i)}}$
- Gradient Descent Algorithm (logistic regression)
  - Repeat until convergence:
  - $\beta \leftarrow \beta + R\frac{1}{N}\sum_{i=1}^{N}(Y_i - \hat{Y}_i)X_i$
  - in other words: $\beta \leftarrow \beta + R\frac{1}{N}\sum_{i=1}^{N}\left(Y_i - \frac{1}{1+e^{-(\alpha+\beta X_i)}}\right) \quad X_i$

Gradient descent: Example Quiz

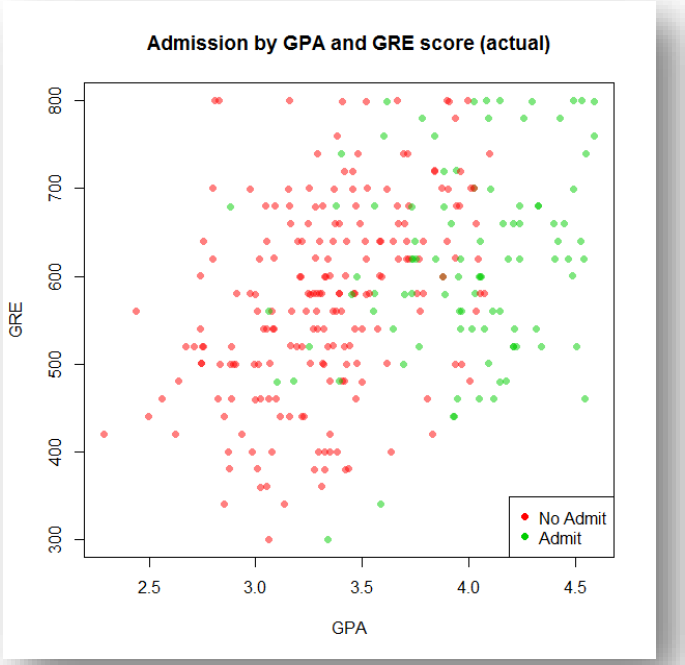
- To ensure that gradient descent is working properly
  1. Plot $J(\theta)$ as a function of θ, and ensure $J(\theta)$ is decreasing
  2. Plot $J(\theta)$ as a function of number of iterations, and ensure $J(\theta)$ is decreasing
  3. Plot $J(\theta)$ as a function of θ, and make sure $J(\theta)$ is convex
  4. Plot $J(\theta)$ as a function of learning rate R, and make sure $J(\theta)$ in monotonic (either constantly increasing or constantly decreasing) in R

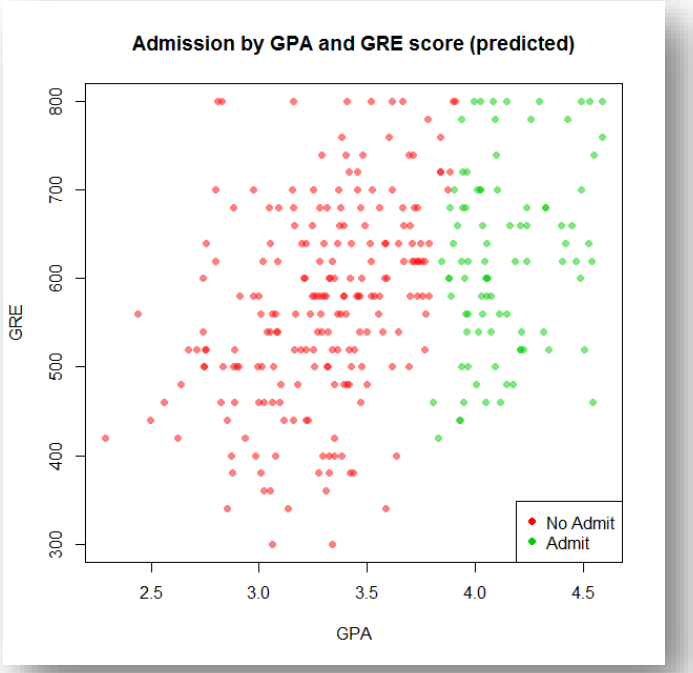
Outline

- Regularization
- Ridge and Lasso
- Logistic regression (inference)
- Logistic regression (prediction)
- **Support vector machines**
- Kernels

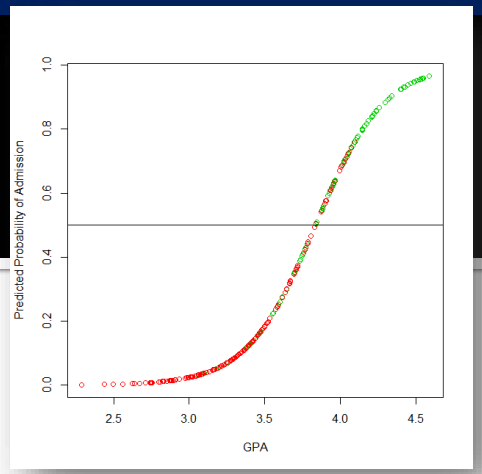
Logistic Regression: Recap

- Compare actual vs. predicted values from our logistic regression

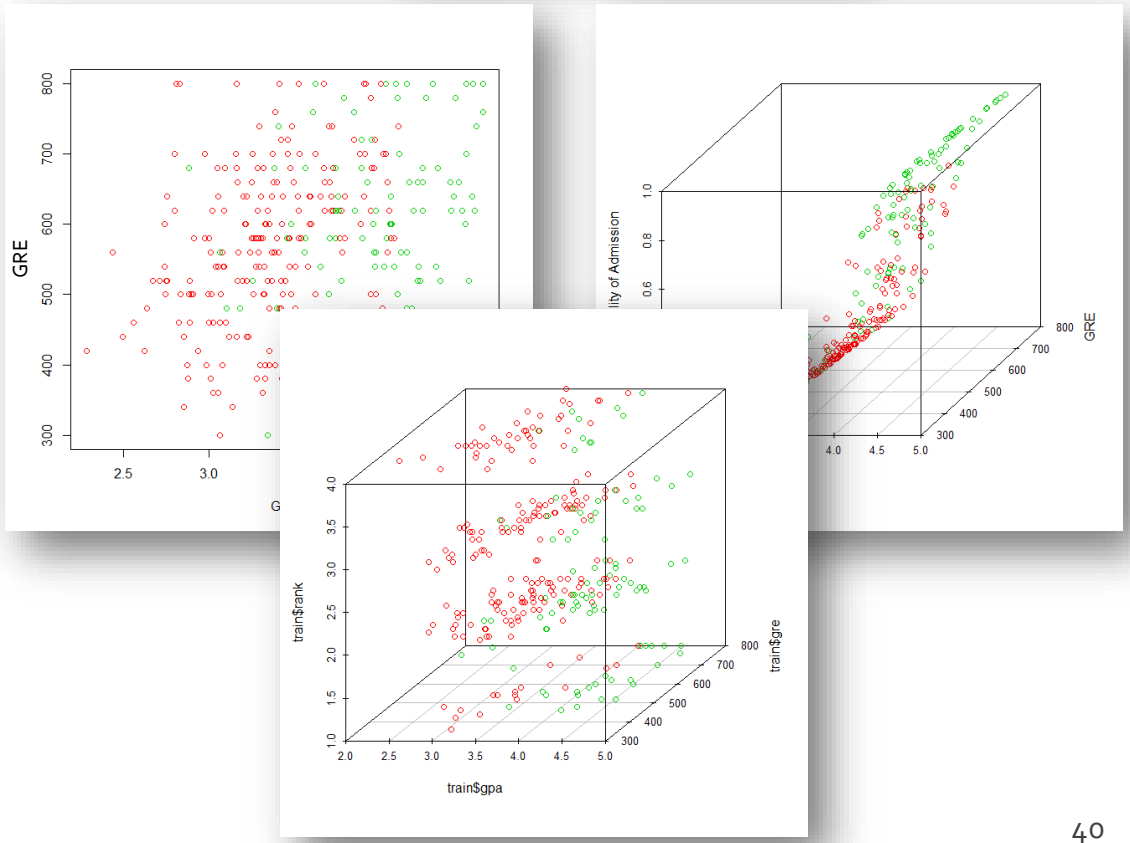




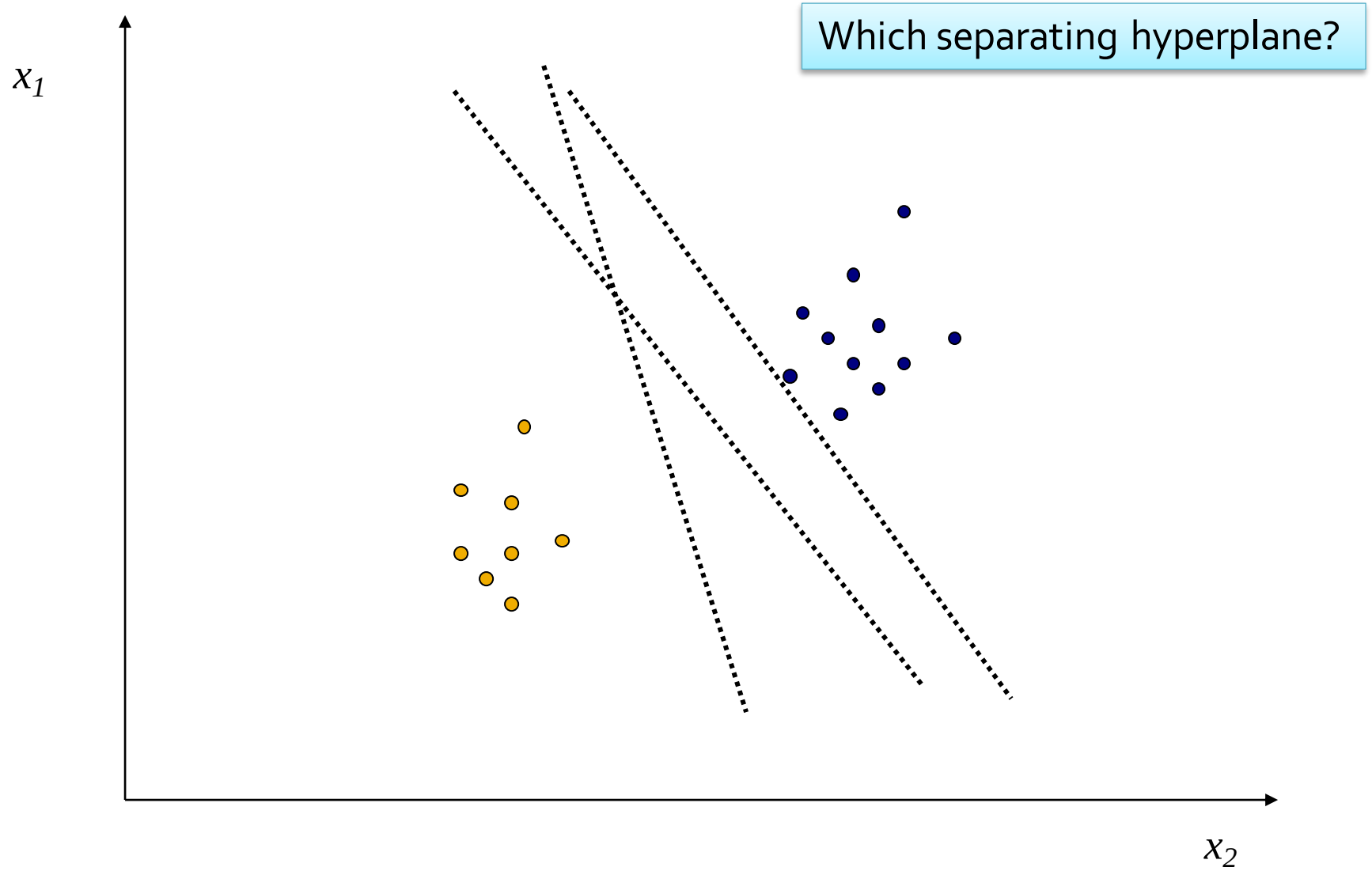
Support Vector Machines



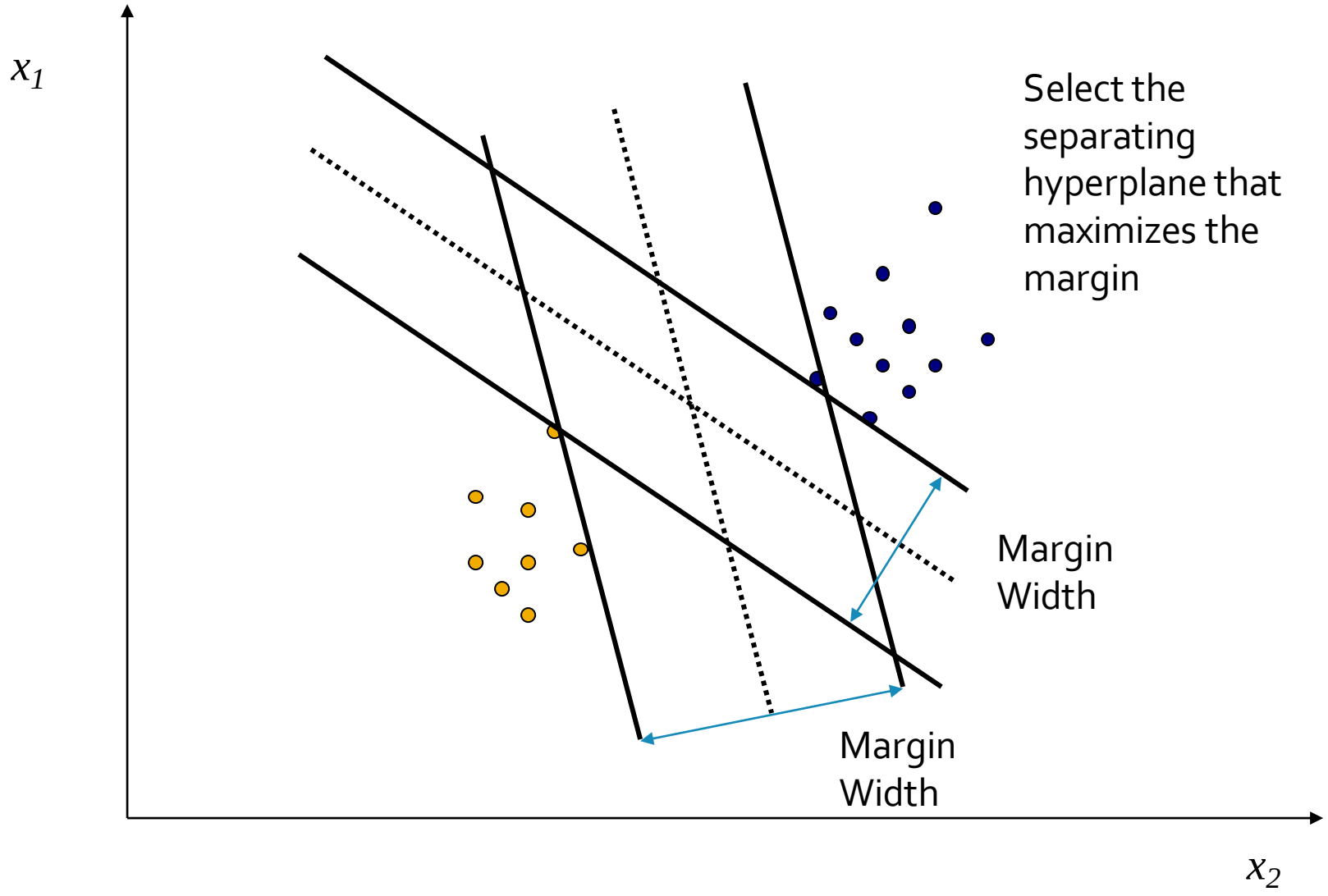
- Generalizes the "linear discriminant"
- Starting point: Data as vectors
  - 1 variable: points on a (1-D) line
    - Discriminant is a number, a threshold
  - 2 variables: points on a (2-D) plane
    - Discriminant is a line
  - 3 variables: points in (3-D) space
    - Discriminant is a plane



SVMs: Hyperplanes



Maximizing the Margin



SVM Definition

- SVM defined by a separating plane
  - Represented by a weight vector *w*, and an intercept *b*
- Classifier function: $f(x) = sign(w^T x + b)$
- We can find an SVM classifier by solving the system of constraints (a quadratic programming problem):
  - $\max_{w,b}(\alpha)$ maximize the margin
  - where $w^T x - b \geq \alpha$ for points *x* in the first class
  - and $w^T x - b \leq -\alpha$ for points *x* in the second class
  - with $w^T w = 1$

- See Daume chapter 7

Soft-Margin SVM

- What if there is no separating hyperplane?
  - Introduce penalties ξ_i to mis-classifications
  - Helps prevent overfitting

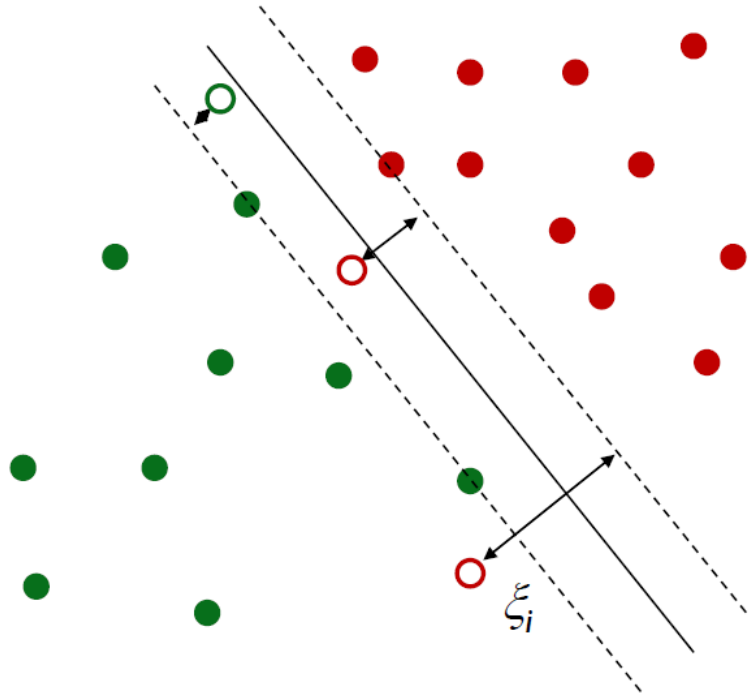


Image: John Canny

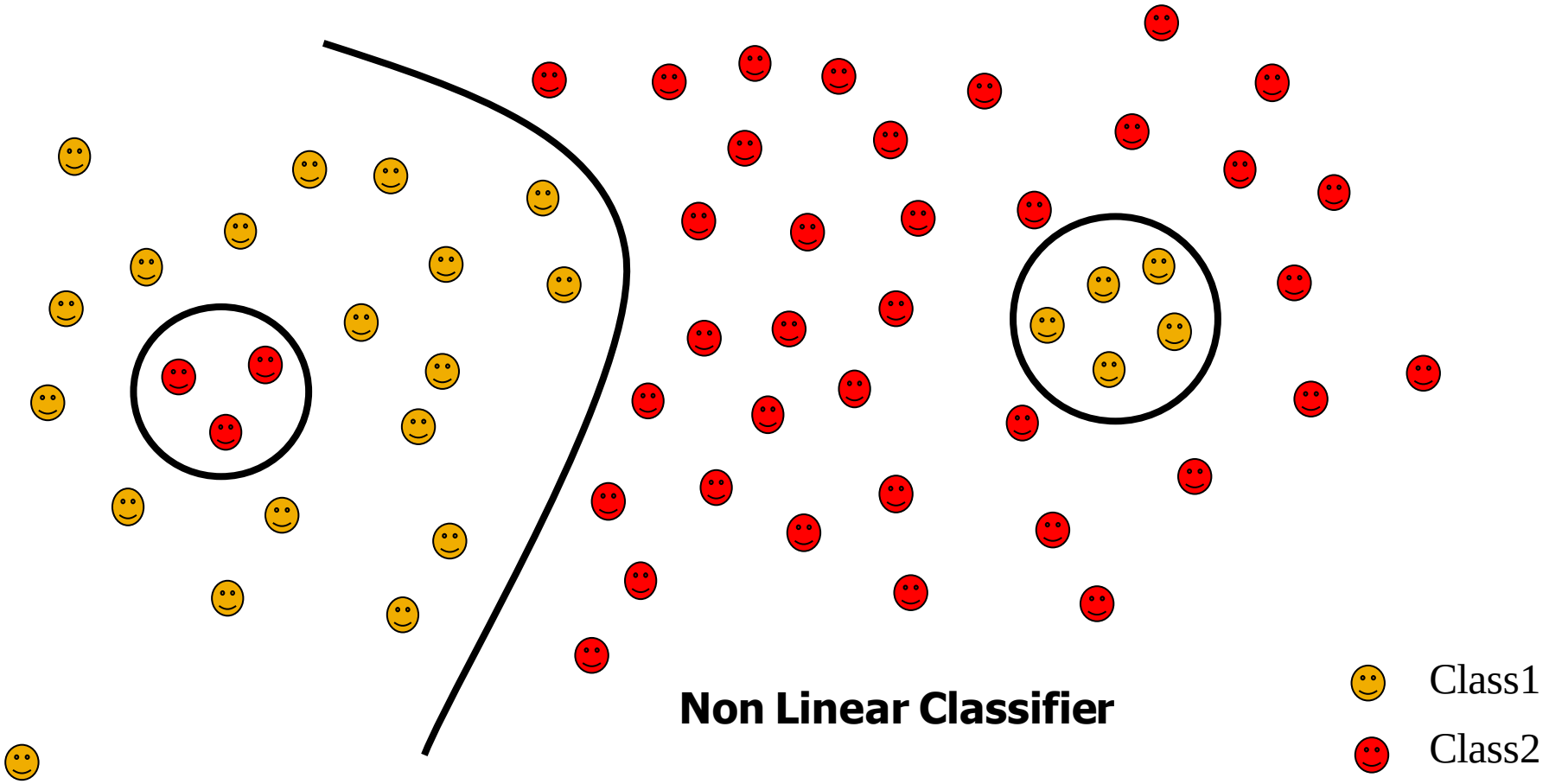
Linear models: Recap

- Linear models rely on some notion of a linear boundary (i.e., a hyperplane)
- But real-world data are typically not linearly separable
- Some classifiers just make a decision as to which class an object is in; others estimate class probabilities

Outline

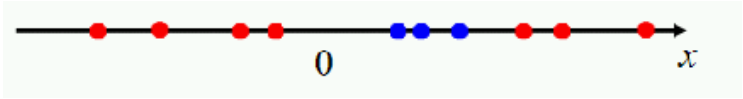
- Regularization
- Ridge and Lasso
- Logistic regression (inference)
- Logistic regression (prediction)
- Support vector machines
- **Kernels**

Nonlinearly separable data

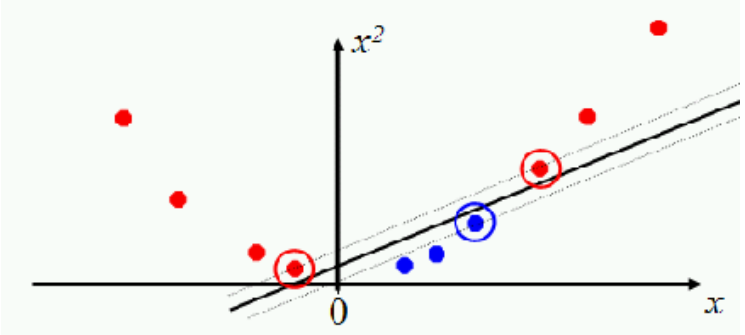


Extending linear models

- We are modeling y with feature x

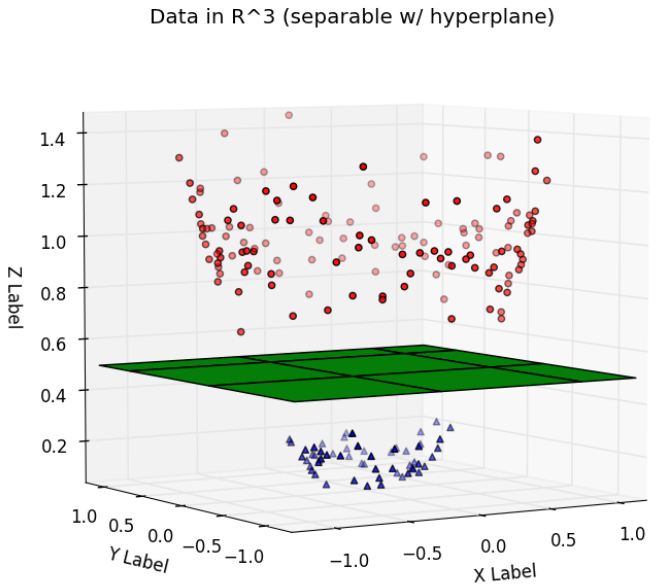


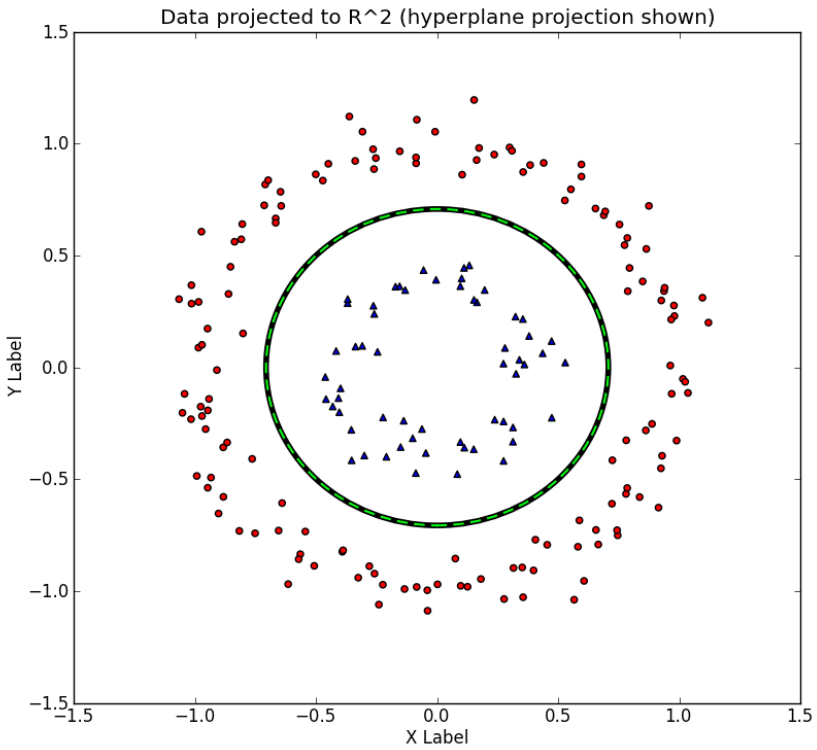
  - Classes are not separable with this feature
- One solution: non-linear classifier
- Another solution: add features!
  - E.g., x^2



Kernel SVM

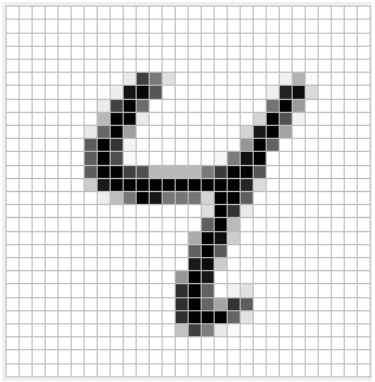
Kernel Methods: Example





Feature combinations

- Recall our feature space in digit classification
  - 28 x 28 pixels = 784 features
  - with 2nd order features: ~615k features
  - with 3rd order features: ~480m features



- Remember the "curse of dimensionality"?
  - We don't have enough data to train
- Adding interactions can help, but adding too many can hurt

Key Concepts (this lecture)

- Regularization
- Ridge
- Lasso
- Logistic regression
- Simplified sigmoid cost function
- Odds ratios
- Overfitting revisited
- Support vector machines
- Hard vs. soft margins
- Kernel functions

For Next Class:

- Read:
  - Chapters 5 and 6 of Daume