DS 4400

Machine Learning and Data Mining I
Spring 2021

Alina Oprea
Associate Professor
Khoury College of Computer Science
Northeastern University

February 11 2021

Outline

- Classification
  - K Nearest Neighbors (kNN)
- Cross validation
  - K-fold cross validation
  - Leave-one-out cross validation
- Linear classifiers
- Logistic regression

Gradient Descent vs Closed Form

Gradient Descent

- Initialize θ
- Repeat until convergence

$$\theta_j \leftarrow \theta_j - \alpha \frac{\partial}{\partial \theta_j} J(\boldsymbol{\theta})$$

simultaneous update for j = 0 ... d

Closed form

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

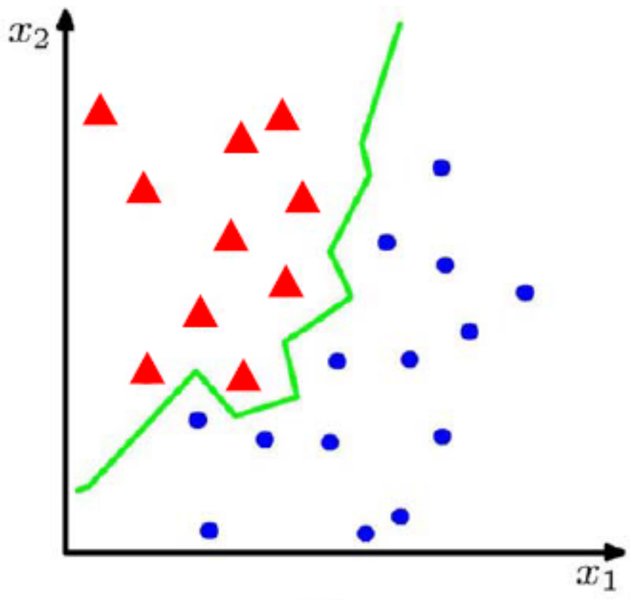
• Gradient Descent	• Closed Form
+ Linear increase in d and N	+ No parameter tuning
+ Generally applicable	+ Gives the global optimum
- Need to choose α and stopping conditions	- Not generally applicable
- Might get stuck in local optima	- Slow computation

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

Classification



Binary or discrete

- Suppose we are given a training set of N observations

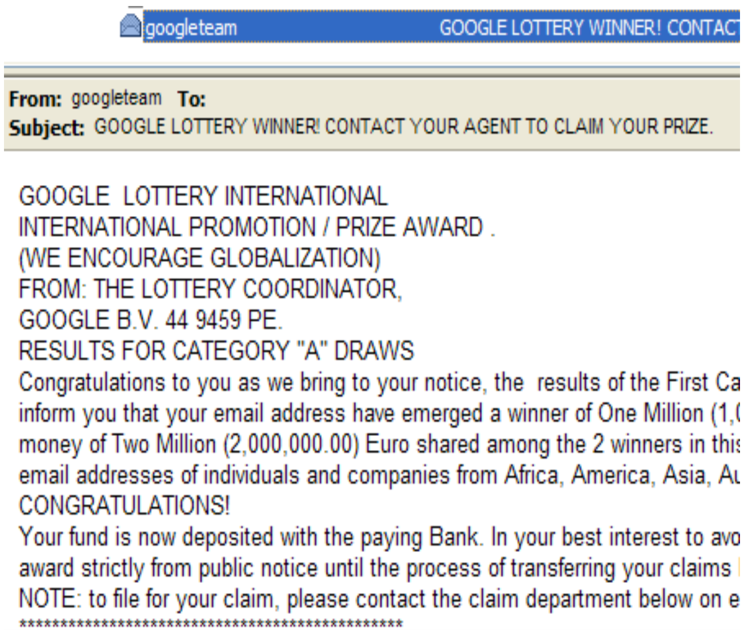
$$\{x_1, \ldots, x_N\} \text{ and } \{y_1, \ldots, y_N\}, x_i \in R^d, y_i \in \{0, 1\}$$

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

$$f(x_i) = y_i$$

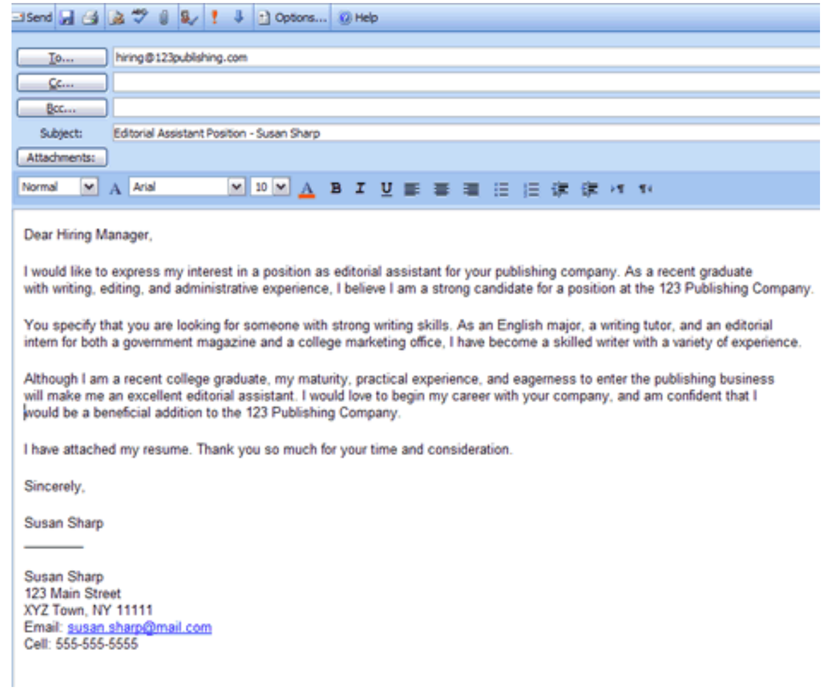
Example 1: Binary classification

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 this
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
****


Send Options... Help

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

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

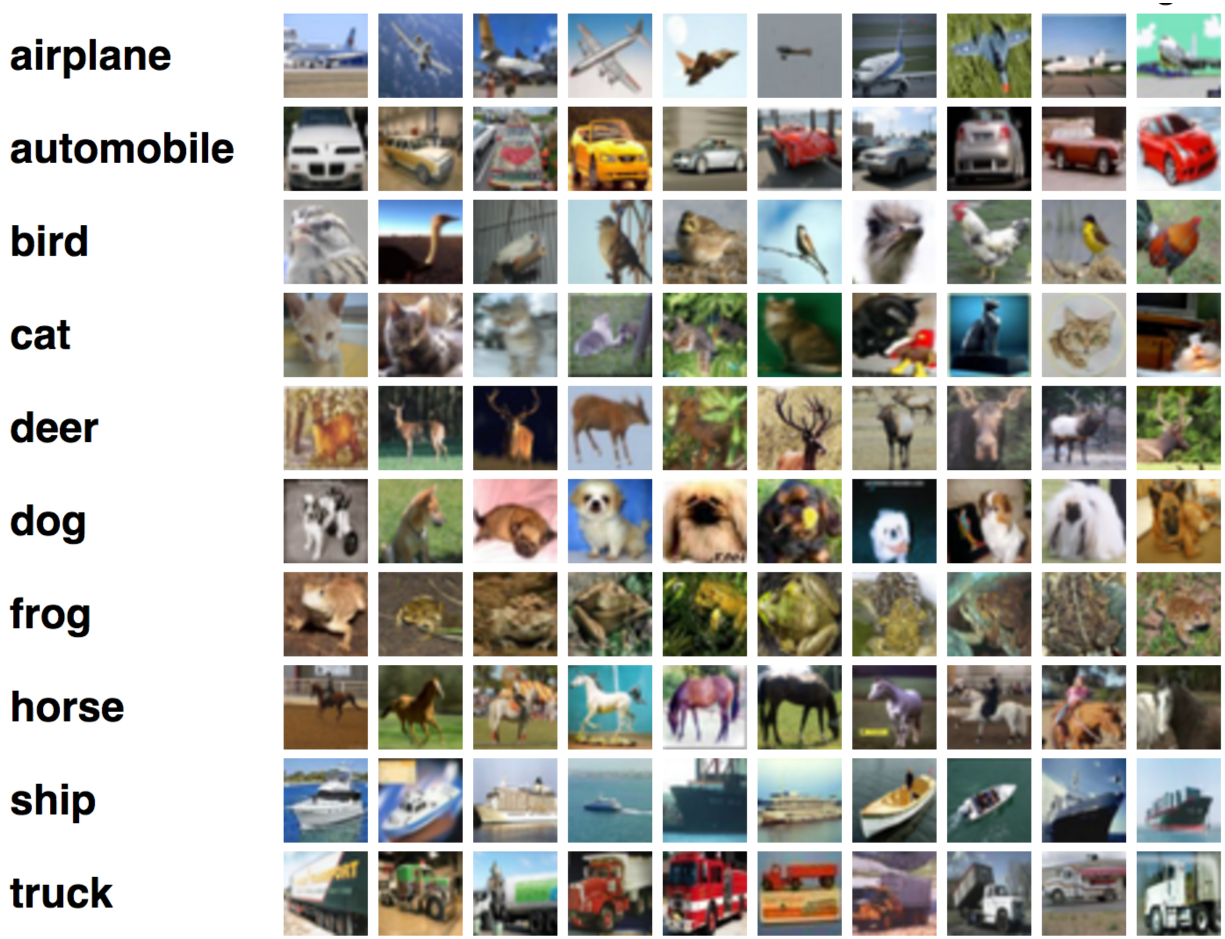
Structural features

- 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

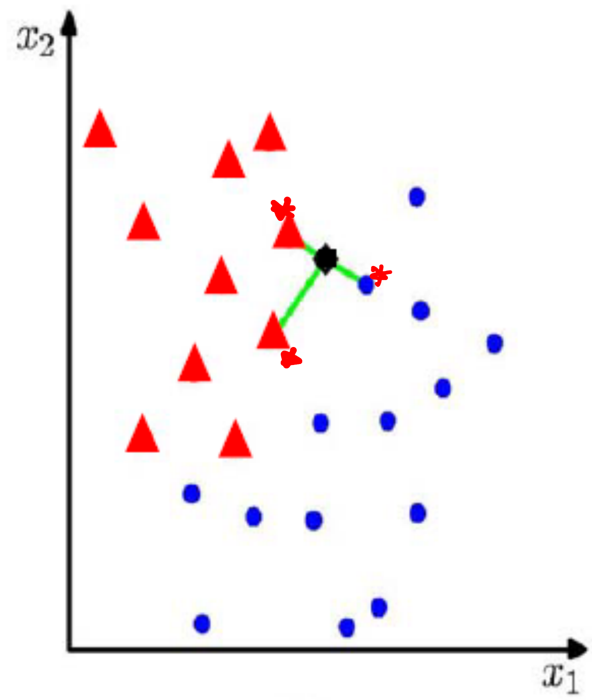
K Nearest Neighbour (K-NN) Classifier

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

DIST(X,Y) = NORM(X-Y)

- Euclidean Distance $$\sqrt{\left(\sum_{i=1}^{k}(x_i - y_i)^2\right)}$$ L_2

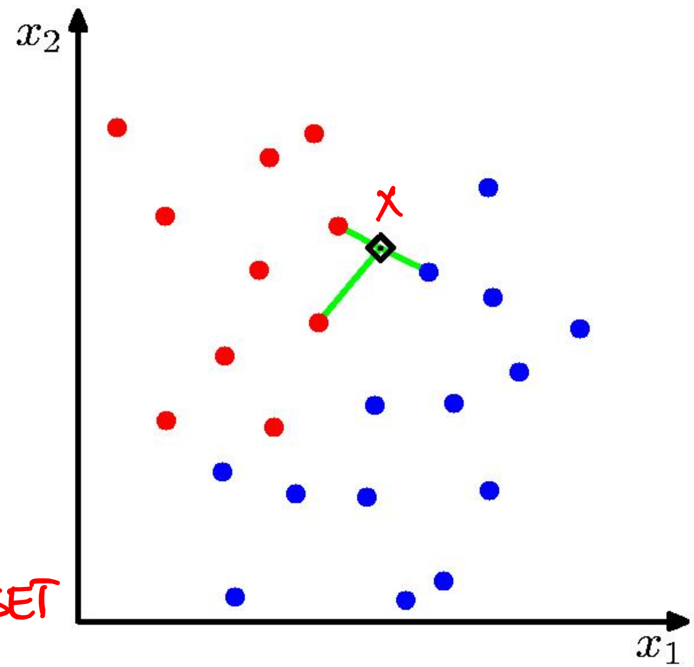
- Manhattan Distance $$\sum_{i=1}^{k}|x_i - y_i|$$ L_1

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

kNN

TRAINING:

- STORE DATA (TRAINING)



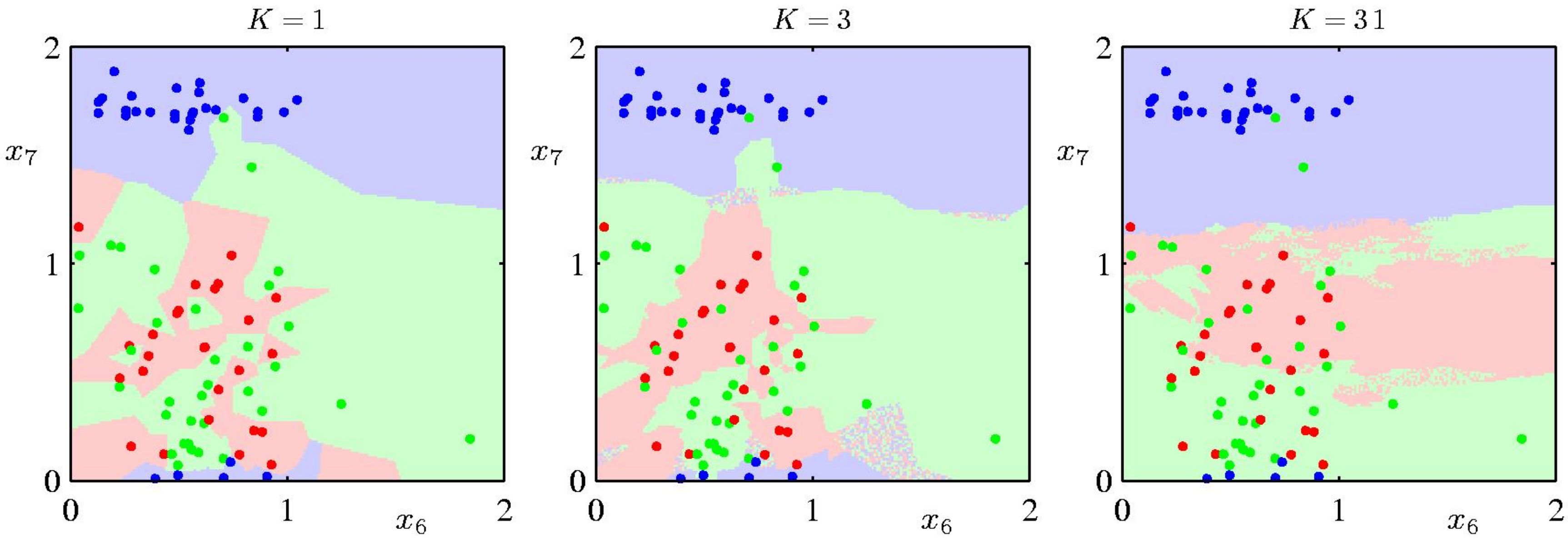
IN TRAING SET

- Algorithm (to classify point x) AT TESTING
  - Find k nearest points to x (according to distance metric)
  - Perform majority voting to predict class of x

PROS:
- SIMPLICITY

CONS:
- COMPUTATIONAL COST
- FEATURES TREATED UNIFORMLY
- INSTANCE LEARNER

K-Nearest-Neighbours for Multi-class Classification

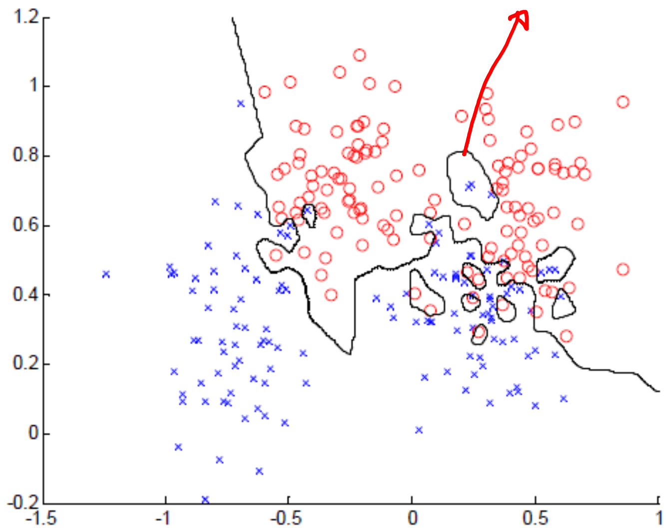


Vote among multiple classes

K = 1

OVERFIT

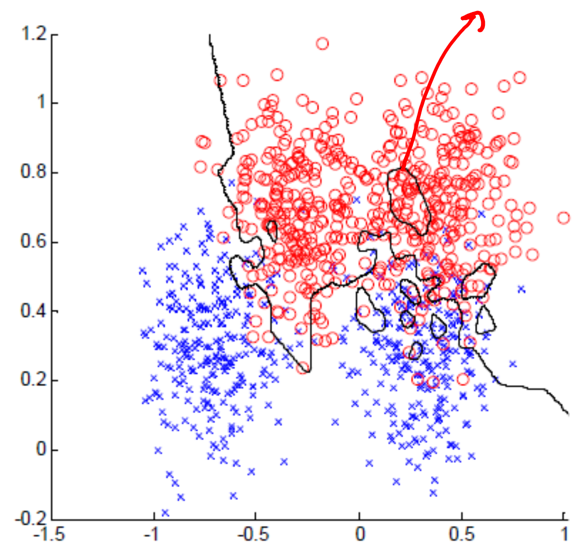
Training data



error = 0.0

Testing data

ERROR

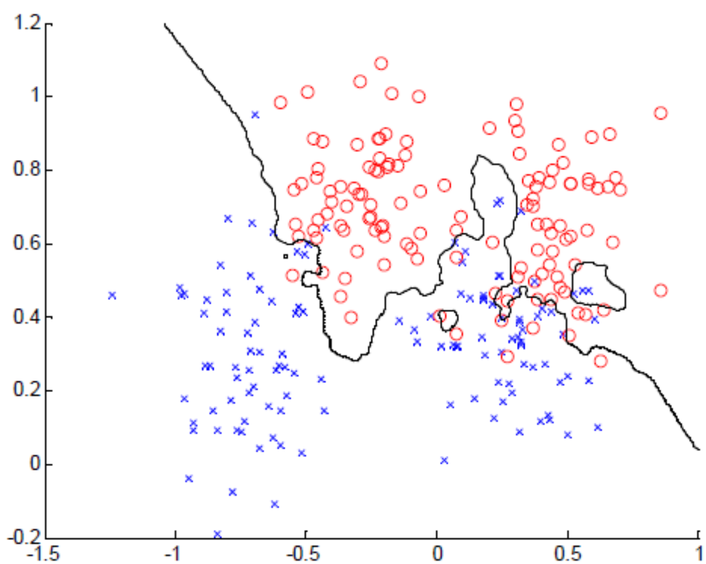


error = 0.15

How to choose k (hyper-parameter)?

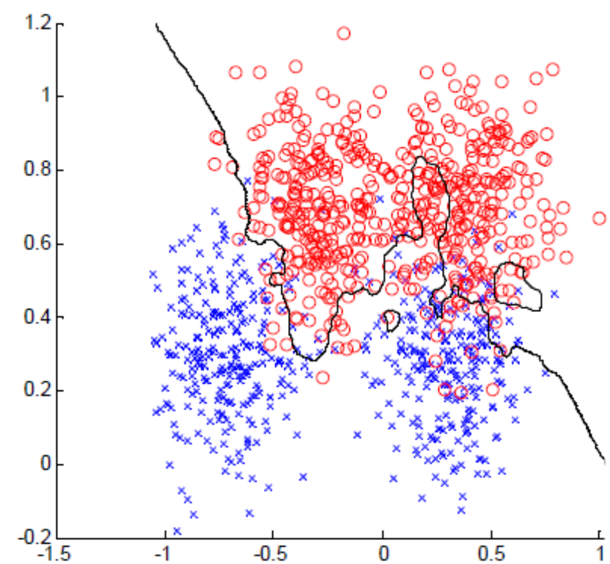
K = 3

Training data



error = 0.0760

Testing data

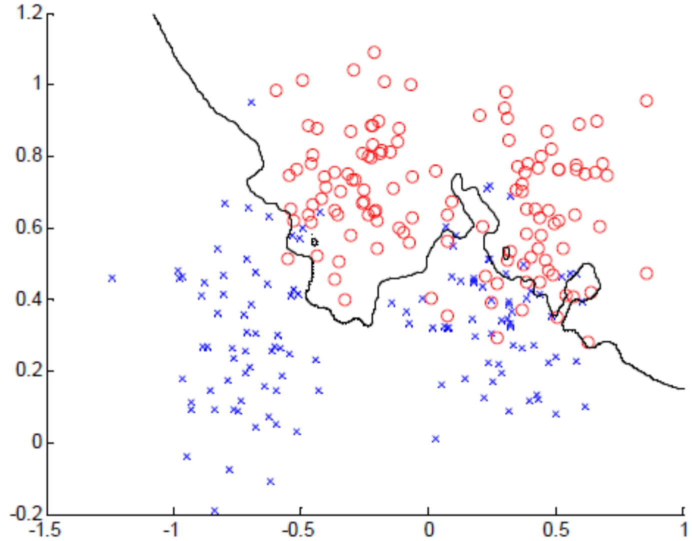


error = 0.1340

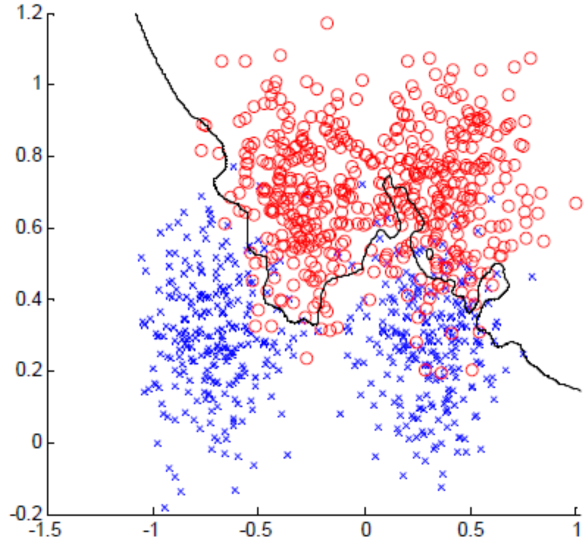
How to choose k (hyper-parameter)?

K = 7

Training data



Testing data

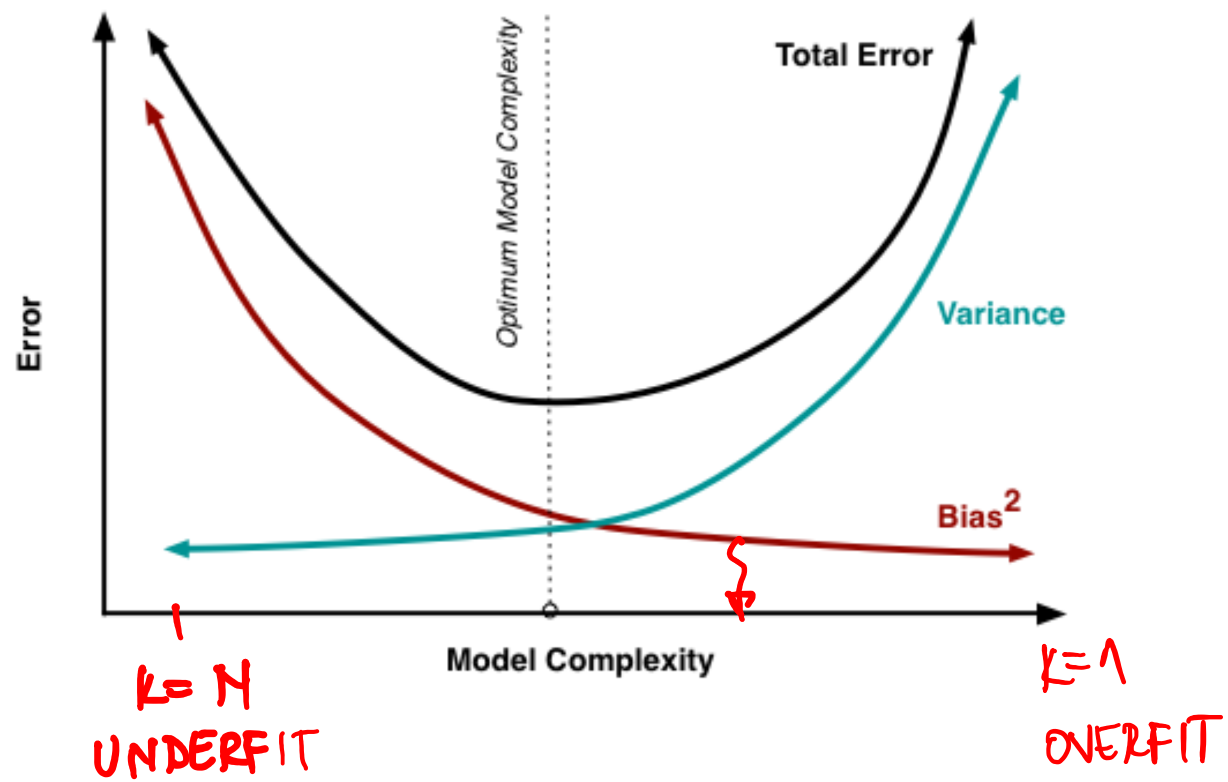


error = 0.1320

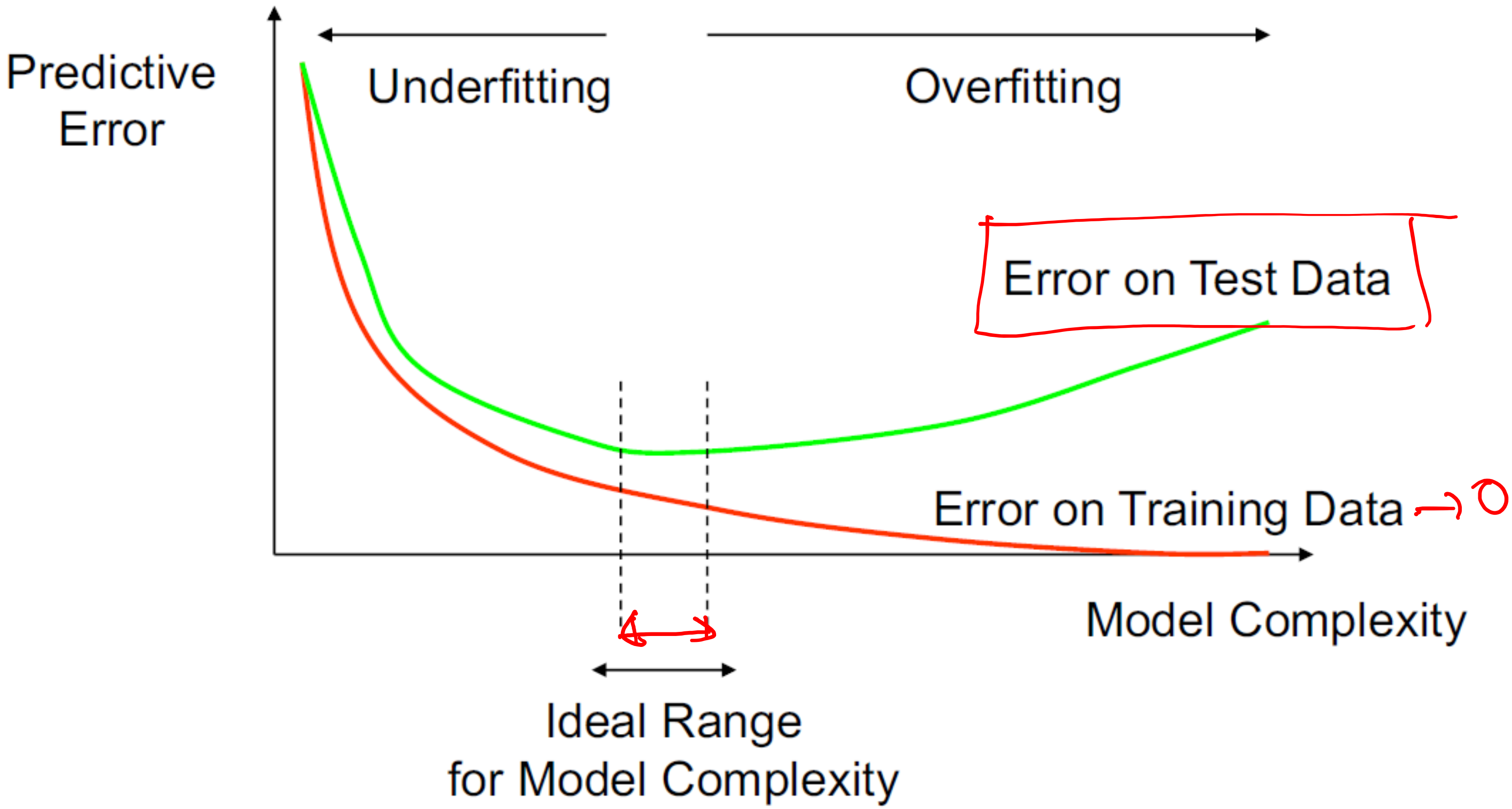
error = 0.1110

How to choose k (hyper-parameter)?

Bias-Variance Tradeoff for kNN



How Overfitting Affects Prediction



How can we avoid over-fitting without having access to testing data?

Cross Validation

As K increases:

- Classification boundary becomes smoother
- Training error can increase

Choose (learn) K by cross-validation

- Split training data into training and validation
- Hold out validation data and measure error on this

ML MODEL

MEASURES ERROR

TRAIN | VAL.

TR. DATA

CV METHOD

- SELECT FRACTION OF TRAINING DATA AT RANDOM & TRAIN ON THIS
- USE THE REST FOR VALIDATION

K=1

EXP1: TRAIN 1 | VAL 1

EXP2:

EXP10: TRAIN 10 | VAL 10

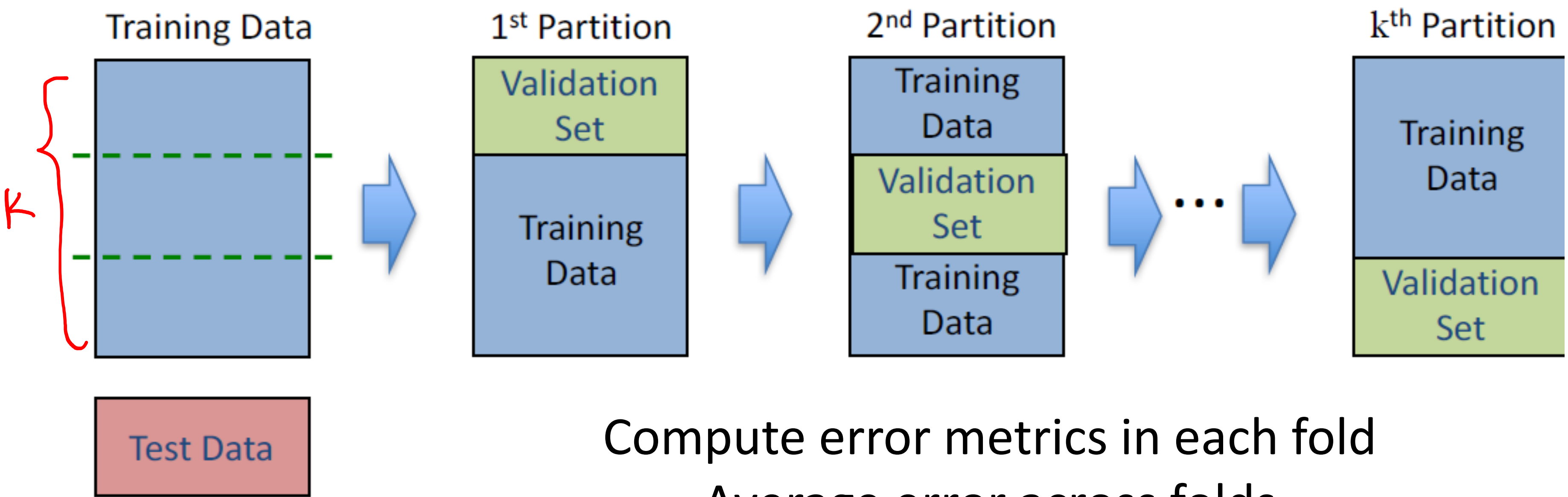
AVG. ERROR K=1

K=3

K=7

PICK K THAT MIN AVG. VALIDATION ERROR

Cross Validation

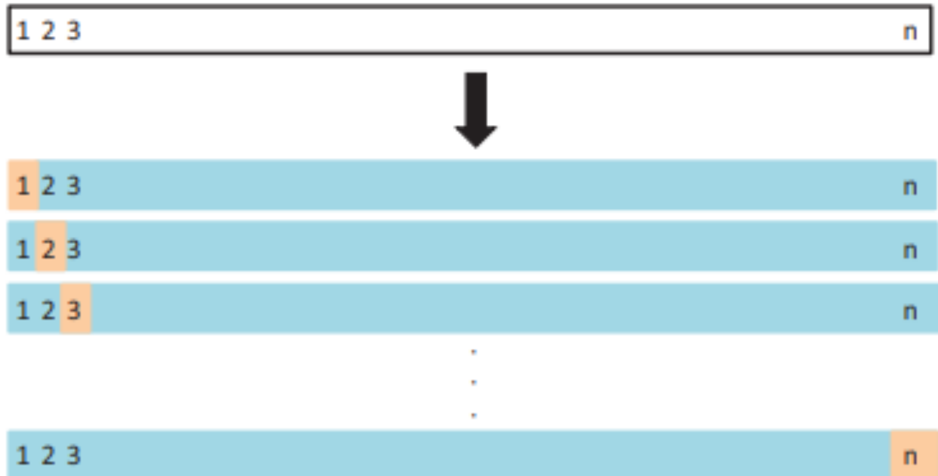


Compute error metrics in each fold
Average error across folds

1. k-fold CV

- Split training data into k partitions (folds) of equal size
- Pick the optimal value of hyper-parameter according to error metric averaged over all folds

Cross Validation



n EXP

TRAIN n-1 POINT
VAL 1 POINT

2. Leave-one-out CV (LOOCV)

– k=n (validation set only one point)

- Pros: Less bias
- Cons: More expensive to implement, higher variance
- Recommendation: perform k-fold CV with k=5 or k=10

Cross-Validation Takeaways

- General method to estimate performance of ML model at testing and select hyper-parameters
  - Improves model generalization
  - Avoids overfitting to training data
- Techniques for CV: k-fold CV and LOOCV
- Compare to regularization

– CV FOR TUNING HYPER-PARAMS

– LASSO & RIDGE ARE APPLICABLE WHEN OPT OBJ

- LINEAR REG
- SVM
- LOGISTIC REG.

Cross Validation Applications

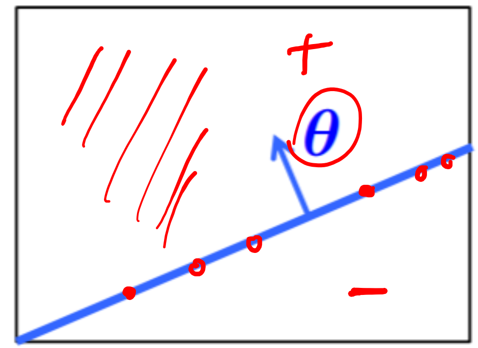
- PICK λ IN REGULARIZATION (LASSO & RIDGE)
  - COMBINING CV + REG.
- FIND DEGREE OF POLY IN POLY REGRESSION
- FIND NUMBER OF KNOTS IN SPLINE REG.

Linear classifiers

- A **hyperplane** partitions space into 2 half-spaces
  - Defined by the normal vector $\boldsymbol{\theta} \in \mathbb{R}^{d+1}$

MODEL PARAM

    - $\boldsymbol{\theta}$ is orthogonal to any vector lying on the hyperplane



  - Assumed to pass through the origin
    - This is because we incorporated bias term θ_0 into it by $x_0 = 1$

θ orthogonal $\Rightarrow \theta^T x = 0$, $\forall x$ ON HYPERPLANE

$h_\theta(x) = f(\theta^T x)$

- Consider classification with +1, -1 labels ...

f = sign.

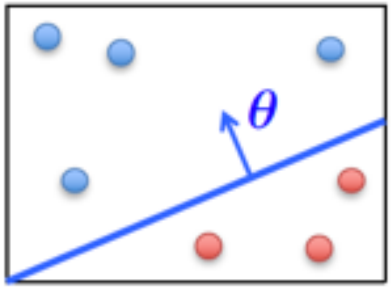
If $\theta^T x > 0 \Rightarrow$ CLASSIFY $+1$

< 0 -1

Linear Classifiers

- **Linear classifiers**: represent decision boundary by hyperplane

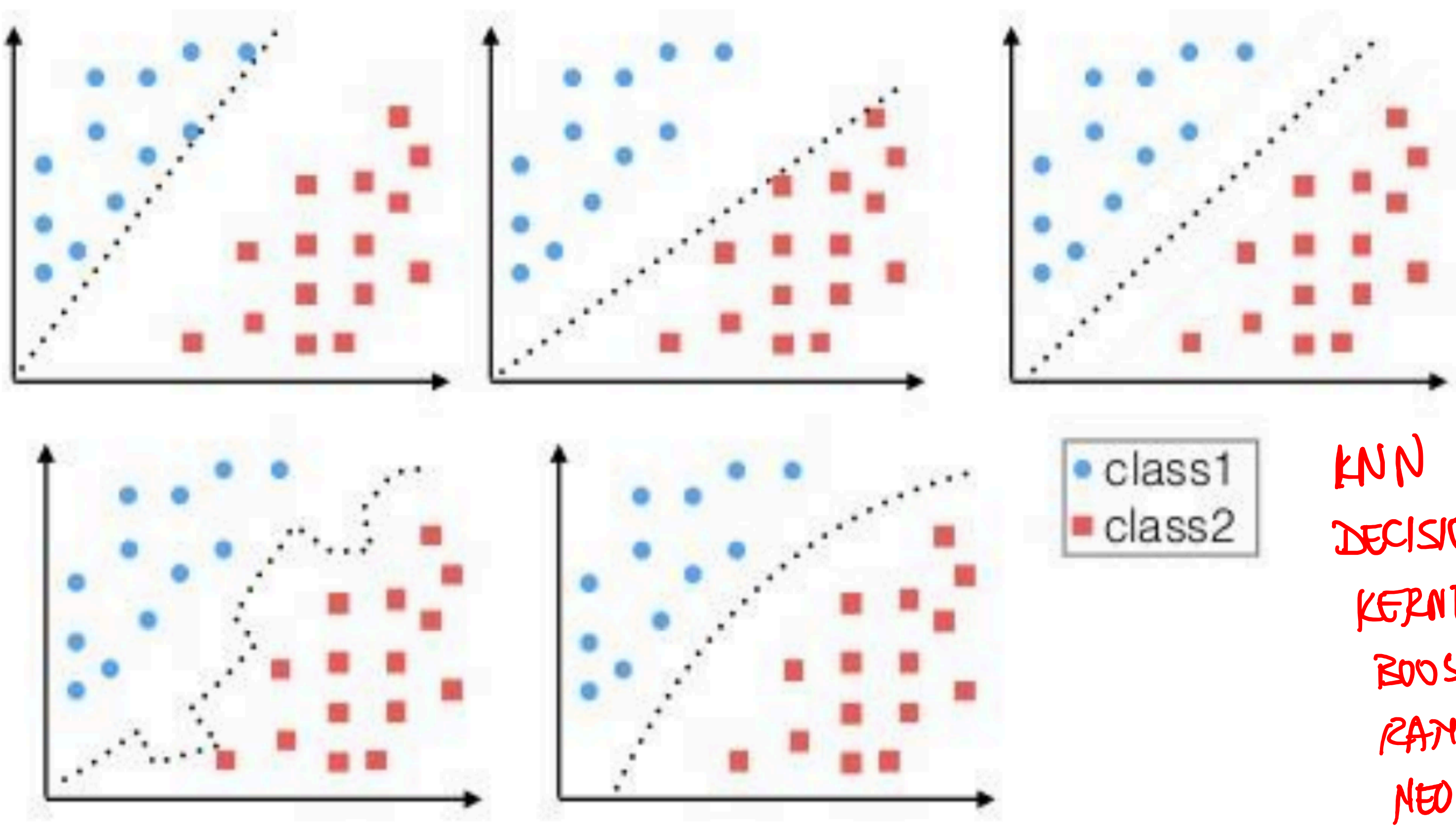
$$\boldsymbol{\theta} = \begin{bmatrix} \theta_0 \\ \theta_1 \\ \vdots \\ \theta_d \end{bmatrix} \quad \boldsymbol{x}^\intercal = \begin{bmatrix} 1 & x_1 & \ldots & x_d \end{bmatrix}$$



$h_\theta(x) = f(\theta^T x)$ linear function
- If $\theta^T x > 0$ classify "Class 1"
- If $\theta^T x < 0$ classify "Class 0"

All the points x on the hyperplane satisfy: $\theta^T x = 0$

Linear vs Non-Linear Classifiers



LOG. REGRESSION
LINEAR SVM
LDA
PERCEPTRON

KNN
DECISION TREES
KERNEL SVM
BOOSTING
RANDOM FOREST
NEURAL NETWORK

Classification Based on Probability

- Instead of just predicting the class, give the *probability of the instance being in that class*

LEARN $P[Y = \overset{1}{\underset{0}{}} \mid X = x]$

- Consider binary classifier with classes 0 and 1

LEARN $\boxed{P[Y=1 \mid X=x]} + P[Y=0 \mid X=x] = 1$

$P[Y=0 \mid X=x] = 1 - P[Y=1 \mid X=x]$

- Advantages: interpretability and confidence of output

Logistic Regression

- Setup
  - Training data: $\{x_i, y_i\}$, for $i = 1, \ldots, N$ $x_i \in \mathbb{R}^d$
  - Labels: $y_i \in \{0,1\}$
- Goals
  - Learn $P(Y = 1 | X = x)$ $= h_\theta(x)$
- Highlights
  - Probabilistic output
  - At the basis of more complex models (e.g., neural networks)
  - Supports regularization (Ridge, Lasso)
  - Can be trained with Gradient Descent

Interpretation of Model Output

$h_{\boldsymbol{\theta}}(\boldsymbol{x})$ = estimated $P(Y = 1|X;\theta)$

Example: Cancer diagnosis from tumor size

$$\boldsymbol{x} = \begin{bmatrix} x_0 \\ x_1 \end{bmatrix} = \begin{bmatrix} 1 \\ \text{tumorSize} \end{bmatrix}$$

$$h_{\boldsymbol{\theta}}(\boldsymbol{x}) = 0.7$$

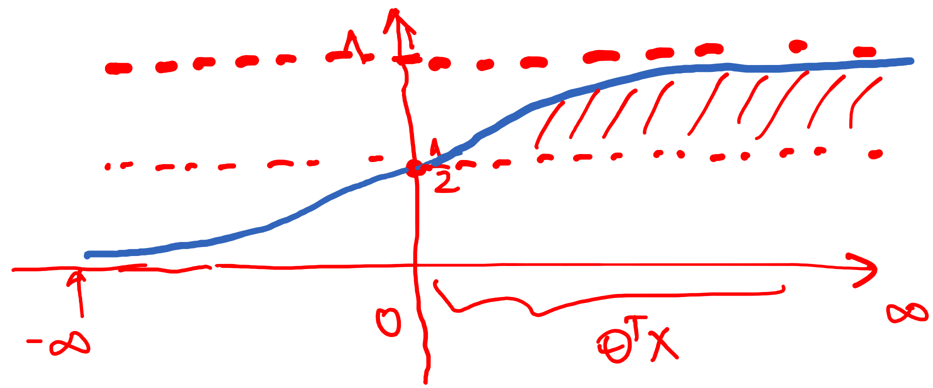
→ Tell patient that 70% chance of tumor being malignant

Note that: $P(Y = 0|X;\theta) + P(Y = 1|X;\theta) = 1$

Therefore, $P(Y = 0|X;\theta) = 1 - P(Y = 1|X;\theta)$

Logistic Regression

- Takes a probabilistic approach to learning discriminative functions (i.e., a classifier)
- $h_{\boldsymbol{\theta}}(\boldsymbol{x})$ should give $P(Y = 1|X;\theta)$
  - Want $0 \leq h_{\boldsymbol{\theta}}(\boldsymbol{x}) \leq 1$



$h_\theta(x) = g(\theta^T x) \in [0,1]$

SIGMOID: $g(z) = \frac{1}{1+e^{-z}}$

$h_\theta(x) = \frac{1}{1+e^{-\theta^T x}}$

LR Predictions

$h_\theta(x) = P[Y=1 \mid X=x] = \frac{1}{1+e^{-\theta^T x}}$

SIGMOID FUNCTION IS NOT LINEAR

- Predict $Y = 1$ if:

$P[Y=1 \mid X=x] > \frac{1}{2}$

$\frac{1}{1+e^{-\theta^T x}} > \frac{1}{2}$

$1 + e^{-\theta^T x} < 2$

$e^{-\theta^T x} < 1$

$e^{\theta^T x} > 1$

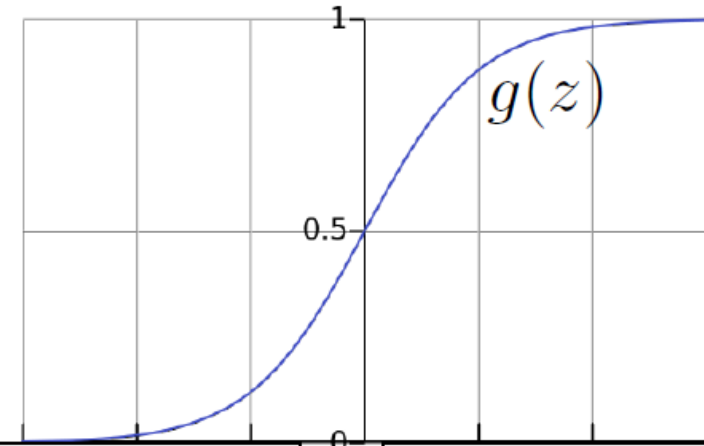
$\boxed{\theta^T x > 0}$

LINEAR CLASSIFIER

Logistic Regression

$$h_{\boldsymbol{\theta}}(\boldsymbol{x}) = g\left(\boldsymbol{\theta}^{\intercal}\boldsymbol{x}\right)$$

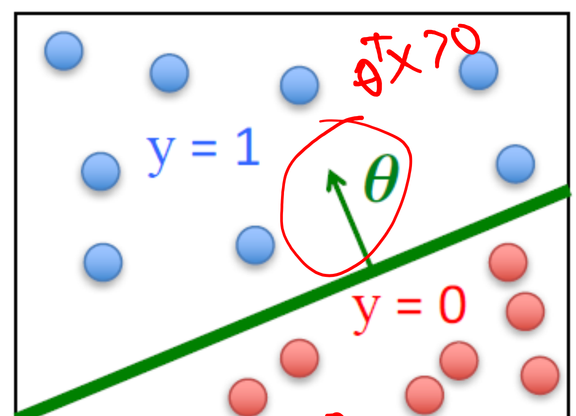
$$g(z) = \frac{1}{1+e^{-z}}$$



$\boldsymbol{\theta}^{\intercal}\boldsymbol{x}$ should be large <u>negative</u> values for negative instances

$\boldsymbol{\theta}^{\intercal}\boldsymbol{x}$ should be large <u>positive</u> values for positive instances

- Assume a threshold and... $\Leftrightarrow \theta^T x \geq 0$
  - Predict $Y = 1$ if $\boxed{h_{\boldsymbol{\theta}}(\boldsymbol{x}) \geq 0.5}$
  - Predict $Y = 0$ if $h_{\boldsymbol{\theta}}(\boldsymbol{x}) < 0.5$



$\theta^T x < 0$

Logistic Regression is a linear classifier!

How to Pick Loss Function?

1) MSE

$$J(\theta) = \frac{1}{N} \sum_{i=1}^{N} [\underbrace{h_\theta(x_i)} - y_i]^2$$

NOT CONVEX.

$$g(\theta^T x) = \frac{1}{1+e^{-\theta^T x}}$$

2) 0-1 LOSS:

$$J(\theta) = \frac{1}{N} \sum_{i=1}^{N} \underbrace{I[h_\theta(x_i) \neq y_i]}$$

$$= \begin{cases} 1, & h_\theta(x_i) \neq y_i \\ 0, & \text{OTHERWISE} \end{cases}$$

ERROR RATE

1) NOT DIFFERENTIABLE

2) DOES NOT MEASURE RATE OF ERROR

Maximum Likelihood Estimation (MLE)

Given training data $X = \{x_1, \ldots, x_N\}$ with labels $Y = \{y_1, \ldots, y_N\}$

What is the likelihood of training data for parameter θ?

Define likelihood function

$$L(\theta) = P[Y \mid X; \theta] = P\underbrace{[y_1, \ldots, y_N \mid x_1, \ldots, x_N; \theta]}$$

Find θ to max $L(\theta)$

$$L(\theta) = \prod_{i=1}^{N} P\underbrace{(Y = y_i \mid X = x_i; \theta)}$$

HOW LIKELY IS TRAINING DATA UNDER MODEL PARAM θ

Log Likelihood

- Max likelihood is equivalent to maximizing log of likelihood

$$\max \log L(\theta) = \sum_{i=1}^{N} \log \underbrace{P[Y=y_i \mid X=x_i; \theta]}$$

$x_i \in \mathbb{R}^D$

TRAINING EXAMPLE

(d- FEATURE)

$y_i \in \{0,1\}$

LABELS

$y_i = 1$

$$P[Y=1 \mid X=x_i; \theta] = h_\theta(x_i)$$

$y_i = 0$

$$1 - h_\theta(x_i)$$

$\log L(\theta) \rightarrow$ DEPENDS ON θ, x_i, y_i

Review

- K nearest neighbors is the first example of classifier
  - Instance learner
- Cross-validation should be performed to
  - Improve generalization and avoid over-fitting
  - Choose hyper parameters (k in kNN)
- Logistic regression is a linear classifier that predicts class probability