CSC 422 Final Review

# Pre-midterm Concepts

## Cross Validation

Divide Data into 3 Parts:

Training set to fit the model

Validation set to select model

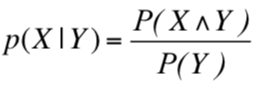
Testing set to assess prediction error

* Sample is too small => no validation set

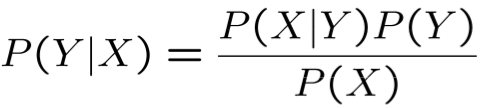
CV: k-fold - > when k = n, LOOCV

* each data is tested once in testing. Error = sum/N instead of sum/K
* large K => few testing points => large variance of error estimation => overfitting
* small K => large testing points => small variance of error estimation => underfitting

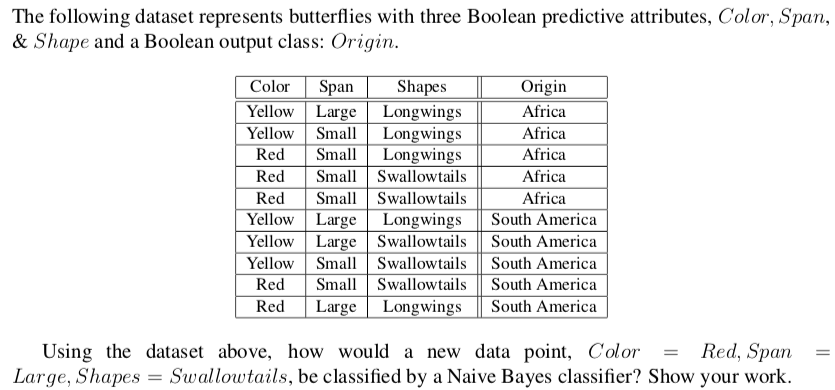
## Conditional Probabilities



## Bayesian Rule



Naive Bayesian Classifier Example P(A|Y=Yi)



Divide into groups based on A value

Goal: compute P(Origin| Color, Span, Shapes) and compare

P(Origin = AF) = 0.5, P(Origin = SA) = 0.5

Compute the conditional probability in each group

P( Color, Span, Shape, Origin)

= P(Origin) P(Color|Origin) P(Span|Origin) P(Shape|Origin)

AF

P(Color = Red | Origin=AF)= 3/5 = 0.6

P(Span=Large | Origin=AF)= 1/5 = 0.2

P(Shape=Swallowtails | Origin=AF)= 2/5 = 0.4

P(Color, Span, Shape, Origin) = 0.5\*0.6\*0.2\*0.4 = 0.024

SA

P(Color = Red | Origin=SA)= 3/5 = 0.4

P(Span=Large | Origin=SA)= 1/5 = 0.6

P(Shape=Swallowtails | Origin=SA)= 2/5 = 0.6

P(Color, Span, Shape, Origin) = 0.5\*0.4\*0.6\*0.6 = 0.072

Compare

0.072 > 0.024 => SA

## 

# New Objectives

## Bayesian Network

Bayesian Network

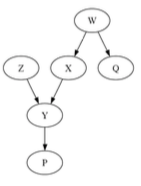
P(A = true|B = false) + P(A = false|B = false) = 1

Ex. P(A = true, B = true, C = true, D = true) A -> B -> (C D)

= P(A= true) \* P(B = true |A= true) \* P(C = true | B = true) P( D = true | B = true)

= (0.4)\*(0.3)\*(0.1)\*(0.95)

D-separation( X1, X3 | X2)

Serial X1 -> X2 -> X3 blocked if Z is in X2

Diverging X1 <- X2 -> X3 blocked if Z is in X2

Converging X1 -> X2 <- X3 blocked if Z + its descendants are NOT in X2

* every path has to be blocked

Ex. d-sep ({Z}, {X, W, Q} | {P}) ? NO

Z -> Y <- {X, W, Q}, and Y has a child P

Y is converging but its child P is in X2

It’s NOT blocked

* d-sep({Z}, {X, W, Q}|Ø) is YES

Joint distribution

With chain rule of probability: P(C, S, R, W) = P(C) \* P(S|C) \* P(R| C,S) \* P(W|C,S,R)

1 + 2 + 4 + 8 = 15

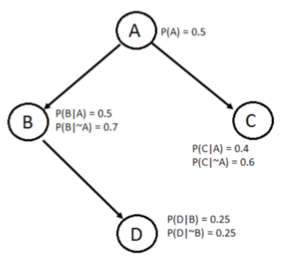
With conditional independence: P(C, S, R, W) = P(C) \* P(S|C) \* P(R| C) \* P(W|S,R)

1 + 2 + 2 + 4 = 9

Bayesian Network Inference

Determine independence(blocked) -> Compute

Ex. compute P(C|D), P(C|B)



Determine independence(blocked) of C and D?

P(C|D) = P(C) = 0.5 ????

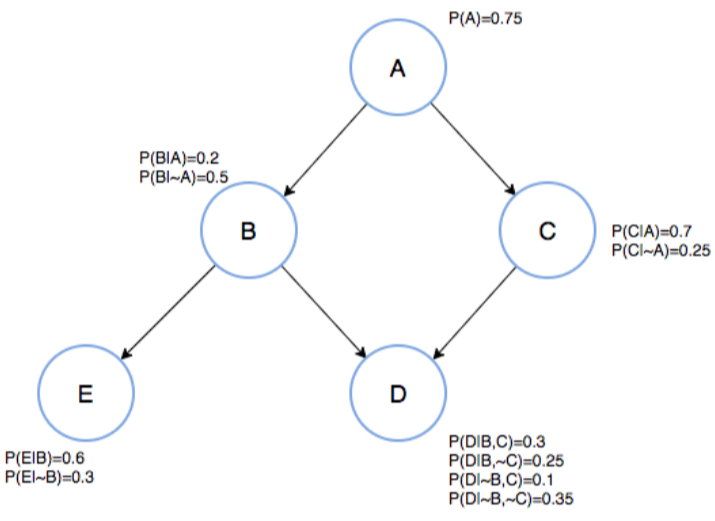
compute P(C|B)

P(C, B) = P(A) \* P(B|A) \* P(C|A) + P(~A) \* P(B|~A) \* P(C|~A) = 0.1 + 0.21 = 0.31

P(B) = 0.5\*0.5 + 0.5\*0.7 = 0.6

P(C|B) = 0.31/0.6 = 0.516

Ex. compute P(B), P (∼ B, C, D, E), P (D | A)



P(E) = P(E|B)P(B) + P(E| ∼ B)P(∼ B) and P(B) will be computed by P(A)

P(∼ B,C,D,E) = P(A) P(∼ B|A) P(C|A) P(E| ∼ B) P(D| ∼ B,C)+

P(∼ A) P(∼ B| ∼ A) P(C| ∼ A) P(E| ∼ B) P(D| ∼ B,C)

P(D|A) = P(A,D)/P(A)

P(A, D) = P(A,B,C,D) + P(A,B,∼ C,D) + P(A,∼ B,C,D) + P(A,∼ B,∼ C,D)

## Linear Regression

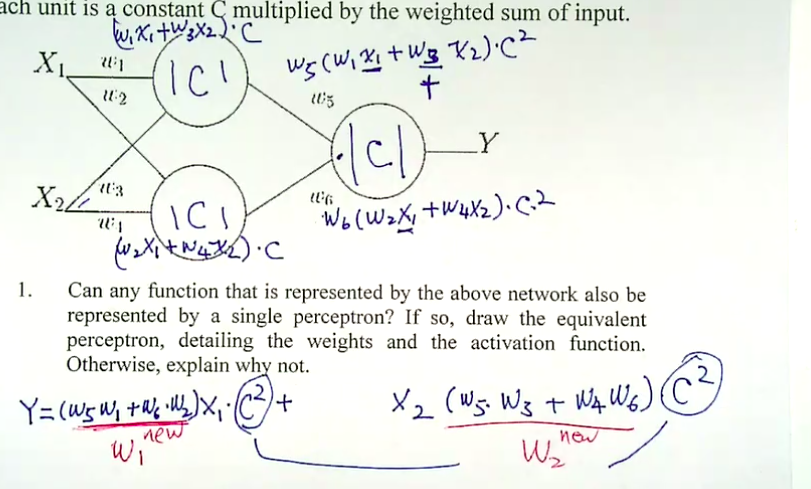
Minimize sum of squared errors

compute the sum of squared error with given y = f(x)

take partial derivatives and set each to 0 => compute the constants in f(x)

## ANN

Explain and evaluate neural network classification methods.



Can the space of functions that is represented by the above ANN also be represented by linear regression? YES because of linear activation units

Example

Consider the XOR function Y.

Draw a fully connected three unit ANN that has binary inputs X1, X2, 1 and output Y. Select weights from (100, -100, 10, -10) to implement Y = (X1 XOR X2). For this questions, assume the sigmoid activation function: f(x) = 1/(1+e^(-x)).

X1 X2 Y

0 0 0

0 1 1

1 0 1

1 1 0

Second layer: Y = A OR B

A B Y

0 0 0

0 1 1 (as B)

1 0 1 (as A)

1 1 1

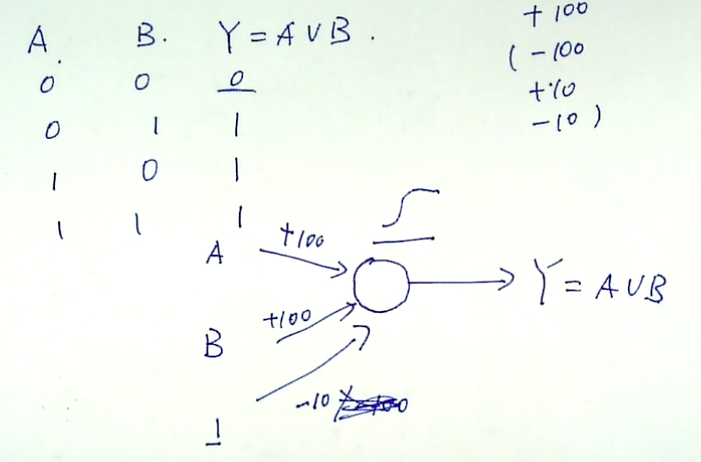
Y = (X1) \* A + (X2) \* B + (1) \* 1 = 100\*A + 100\*B -10

A = B = 0, (1) has to be negative [-10, -100]

A = 0, B = 1, Y = 1. So (X2) has to be a larger positive # to cancel out the (1)\* 1

(X2) = 100, (1) = -10

A = 1, B = 0, Y = 1. Similarly, (X1) = 100



First layer: Y = A + B

X1 X2 A B

0 0 0 0

0 1 0 1

1 0 1 0

1 1 0 0

For A = (X1)\*X1 + (X2)\*X2 + (1)\*1 = 100\*X1 -100\*X2 -10

X1 = X2 = 0, A = 0 csae: (1) should be [-100, -10]

X1 = 1, X2 = 0, A = 1 case: (X1) = 100, (1) = -10

X1 = 1, X2 = 1, A = 0 case: (X2) = -100

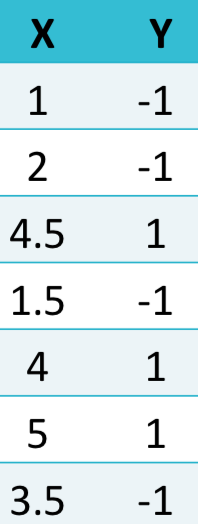
For B = (X1)\*X1 + (X2)\*X2 + (1)\*1 = -100\*X1 + 100\*X2 -10

X1 = X2 = 0, A = 0 csae: (1) should be [-100, -10]

X1 = 0, X2 = 1, A = 1 case: (X2) = 100, (1) = -10

X1 = 1, X2 = 1, A = 0 case: (X1) = -100

## SVM

Ex(Linear SVM). Consider the following dataset. We are going to learn a linear SVM from it of the form f(x) = sign(wx+w0).

support vectors are (4, 1) and (3.5, -1)

3.5w + w0 = -1

4w + w0 = 1

solve two equations:

w = 4

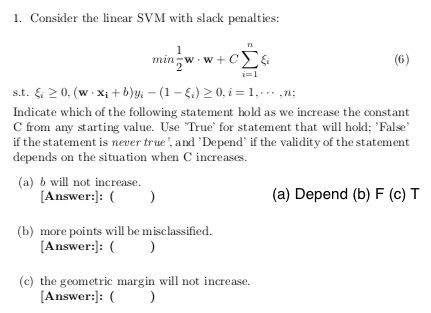
w0 = -15

Ex(Soft). Suppose that we want to learn a so'-margin linear SVM for this data set

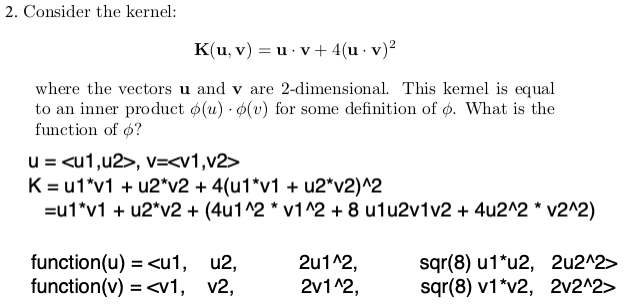
if C = 0, no penalty => all of the support vectors

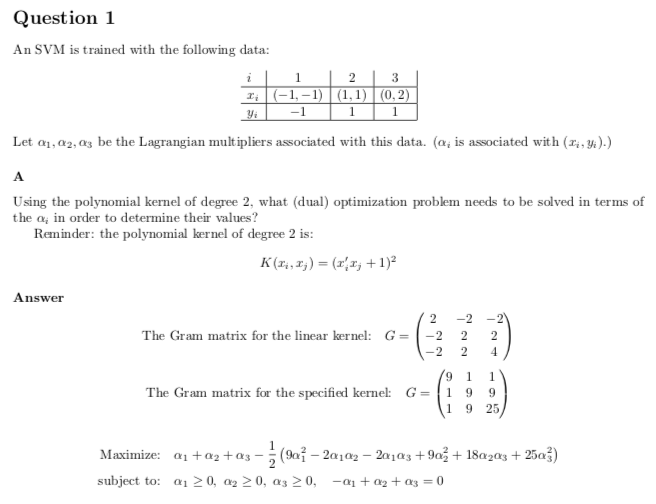
If C -> ∞, we will have 2 support vectors

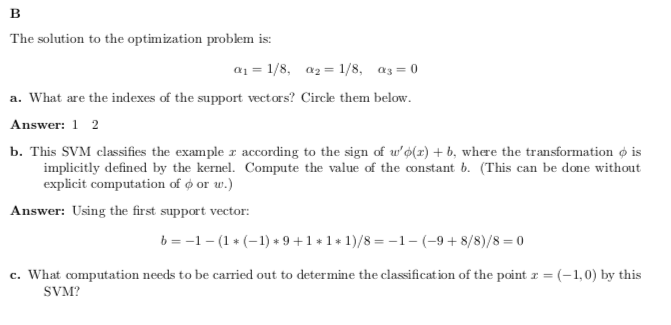
Ex(Soft Margin)

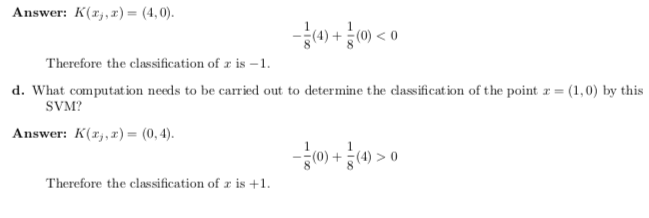


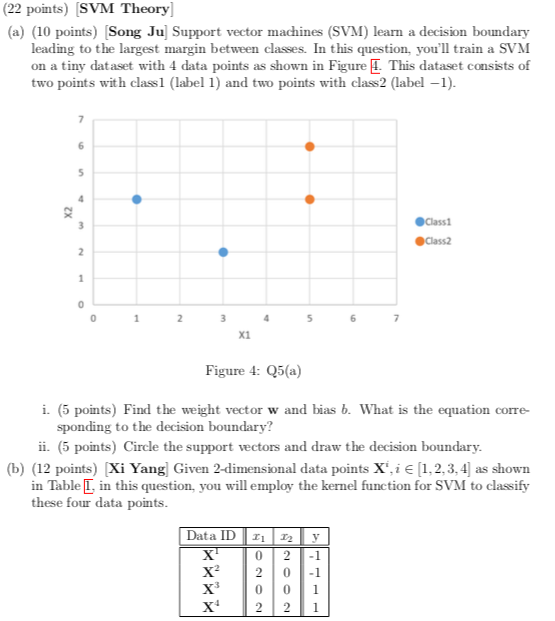
Ex(Kernel)

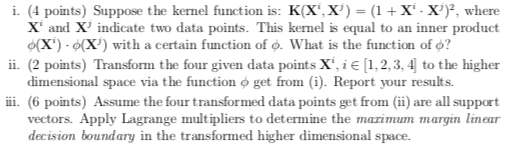


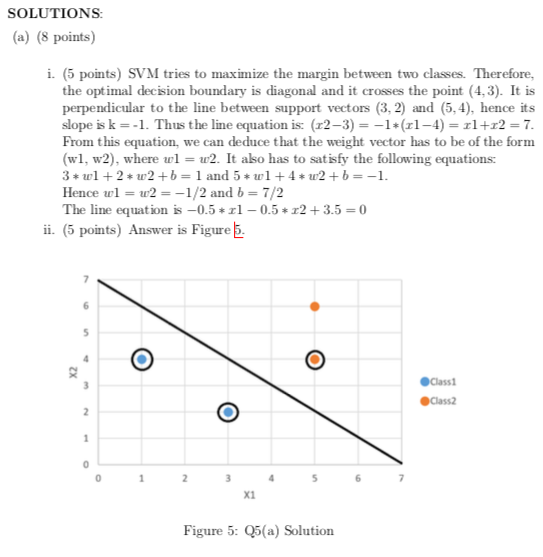
Ex(Kerne)l



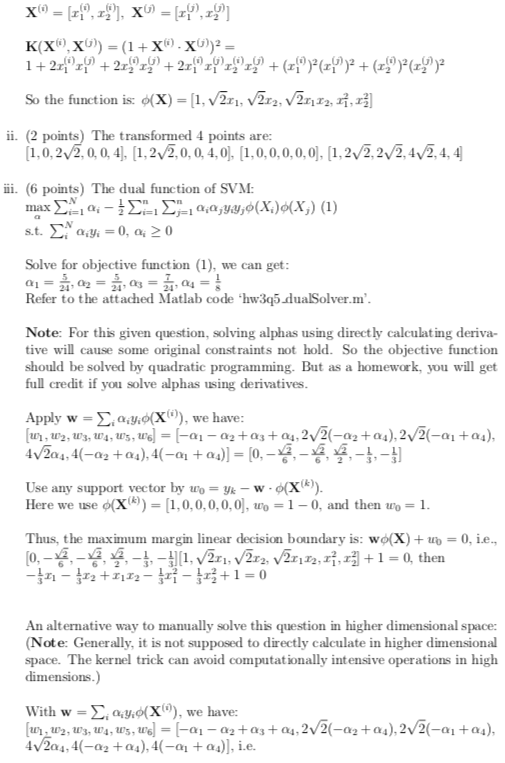


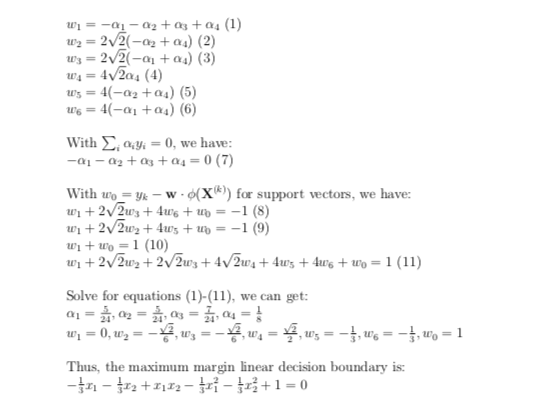
Ex (Linear and Kernel)





(b)





## Clustering

### K-Means

select the closest center -> compute the new center and repeat

### Hierarchical

single(refill with smallest) and complete(refill with largest) example

### Density-Based DBSCAN

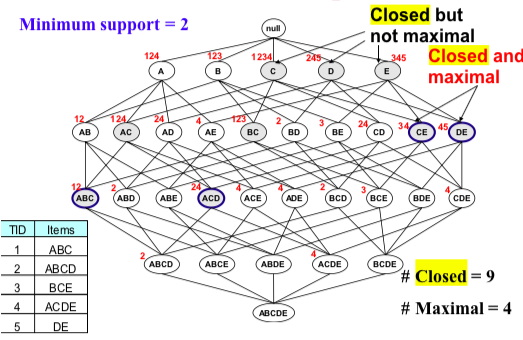
core point: > min\_point within the certain radius

border point: neighbor(in certain radius) of the core points

## Associate Analysis

Support and confidence of association rules

Maximal and closed itemsets



closed = the next level doesn’t have the same support count

maximal = the next level is not frequent

Apriori principle in identifying association rules.

Generate and calculate frequent itemsets using the Apriori algorithm.

## Hash-Table and Transaction Comparison

storage and transaction comparison

# FP Tree

Order the itemset based on the frequency/support count

Build the FP Tree

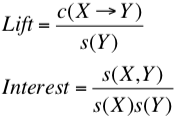
Find the frequent itemset from the least frequent item

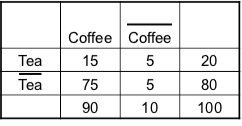
## Apriori Principle

start from {1}, {2}, {3}, prune the infrequent itemset => combine {1, 2} and repeat

## Interestingness of Associations and Association rules

Interest = 1 independence, interest < 1 negative associated, interest > 1 positive associated



Ex. 

confidence(Tea->Coffee) = 15/20 = 0.75

P(Coffee) = 0.9

Lift = 0.75/0.9 = 0.8333 < 1 (negatively associated)

Even the confidence is relatively high, it’s not enough to measure the interestingness of the rulle Tea->Coffee since the P(Coffee) is 0.9. The chance that Coffee occurs alone is very high.

# Formula

