CSCI-4/5588, Fall-2020
Machine Learning II
Study Guide for Final Exam

(Please don't distribute this study guide. The guide is for your study purpose only)

Note: The Final Exam is on **Dec 7 (Monday), 5:30 PM to 7:30 PM.**

1. While considering separable classes (see Figure (A)) for support vector machine development, we formulate our goal as:

$$\min_{\beta,\beta_0} \frac{1}{2}\|\beta\|^2 \qquad (1)$$

subject to $y_i(\beta^T x_i + \beta_0) \geq 1$, ***i* =1, ..., *N*.**

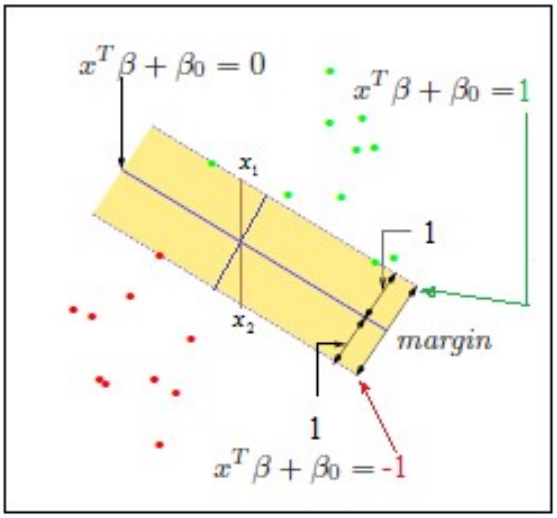


Figure A: Support vector classifier. The decision boundary is the solid line, while broken lines bound the shaded maximal margin of width, 1+1 = 2.

Show that the *Lagrange* (primal, L_P) function, to be minimized w.r.t. β and β_0 can be written as:

$$L_p(\beta,\beta_0,\alpha_i) = \frac{1}{2}\|\beta\|^2 - \sum_{i=1}^{N}\alpha_i\left[y_i(x_i{}^T\beta + \beta_0) - 1\right] \qquad (2)$$

Here, α_i is the Lagrange multiplier and $\alpha_i \geq 0$.

From (2), derive the dual

$$L_D(\alpha_i) = \sum_{i=1}^{N}\alpha_i - \frac{1}{2}\sum_{i=1}^{N}\sum_{k=1}^{N}\alpha_i\alpha_k y_i y_k x_i{}^T x_k$$

Ans: It is to be noted that there is a minus sign in front of the ***Lagrange multiplier*** (equation #2) term because we minimize with respect to β and β_0 but maximizing with respect to α_i.

Setting the derivatives to zero, we obtain:

$$\beta = \sum_{i=1}^{N} \alpha_i y_i x_i \tag{3}$$

$$0 = \sum_{i=1}^{N} \alpha_i y_i \tag{4}$$

[Important: **Note here you need to show how we can obtain Equation #3 and #4, check class notes**]
and substituting these in (2) we obtain the so-called Wolfe dual

$$\begin{aligned} L_p(\beta, \beta_0, \alpha_i) &= \frac{1}{2}\|\beta\|^2 - \sum_{i=1}^{N} \alpha_i \left[y_i (x_i^T \beta + \beta_0) - 1 \right] \\ &= \frac{1}{2}\|\beta\|^2 - \sum_{i=1}^{N} \alpha_i y_i x_i^T \beta - \sum_{i=1}^{N} \alpha_i y_i \beta_0 + \sum_{i=1}^{N} \alpha_i \\ &= \frac{1}{2}\|\beta\|^2 - \sum_{i=1}^{N}\sum_{k=1}^{N} \alpha_i \alpha_k y_i y_k x_i^T x_k - \beta_0 \sum_{i=1}^{N} \alpha_i y_i + \sum_{i=1}^{N} \alpha_i \\ &= \frac{1}{2}\sum_{i=1}^{N}\sum_{k=1}^{N} \alpha_i \alpha_k y_i y_k x_i^T x_k - \sum_{i=1}^{N}\sum_{k=1}^{N} \alpha_i \alpha_k y_i y_k x_i^T x_k + \sum_{i=1}^{N} \alpha_i \qquad \left[\because \sum_{i=1}^{N} \alpha_i y_i = 0 \right] \\ &= \sum_{i=1}^{N} \alpha_i - \frac{1}{2}\sum_{i=1}^{N}\sum_{k=1}^{N} \alpha_i \alpha_k y_i y_k x_i^T x_k \end{aligned}$$

Therefore, we can now express dual equation as a function of α_i as:

$$L_D(\alpha_i) = \sum_{i=1}^{N} \alpha_i - \frac{1}{2}\sum_{i=1}^{N}\sum_{k=1}^{N} \alpha_i \alpha_k y_i y_k x_i^T x_k$$

2. Show that the redial basis function (RBF) or Gaussian kernel, i.e., $\exp(-\gamma\|x - x'\|^2)$, can be expressed as an inner product in infinite-dimensional space. Assuming, $x \in \Re^1$ and $\gamma > 0$.

Ans:

$$\begin{aligned} K(x, x') &= \exp(-\gamma\|x - x'\|^2) \\ &= \exp(-\gamma(x - x')^2) \\ &= \exp(-\gamma x^2 + 2\gamma x x' - \gamma x'^2) \\ &= \exp(-\gamma x^2).\exp(-\gamma x'^2).\exp(2\gamma x x') \end{aligned}$$

$$= \exp(-\gamma x^2).\exp(-\gamma x'^2).\left(1 + \frac{2\gamma x x'}{1!} + \frac{(2\gamma x x')^2}{2!} + \frac{(2\gamma x x')^3}{3!} + \ldots\right)$$

$$\left[\because e^x = 1 + \frac{x}{1!} + \frac{x^2}{2!} + \frac{x^3}{3!} + \ldots\right]$$

$$= \exp(-\gamma x^2).\exp(-\gamma x'^2).\left(1.1 + \sqrt{\frac{2\gamma}{1!}}x.\sqrt{\frac{2\gamma}{1!}}x' + \sqrt{\frac{(2\gamma)^2}{2!}}x^2.\sqrt{\frac{(2\gamma)^2}{2!}}x'^2 + \sqrt{\frac{(2\gamma)^3}{3!}}x^3.\sqrt{\frac{(2\gamma)^3}{3!}}x'^3 + \ldots\right)$$

$$= \left(\exp(-\gamma x^2).[1 \quad \sqrt{\frac{2\gamma}{1!}}x \quad \sqrt{\frac{(2\gamma)^2}{2!}}x^2 \quad \ldots]\right) \left(\exp(-\gamma x'^2).\begin{bmatrix} 1 \\ \sqrt{\frac{2\gamma}{1!}}x' \\ \sqrt{\frac{(2\gamma)^2}{2!}}x'^2 \\ \vdots \end{bmatrix}\right)$$

$$= h(x)^T\, h(x')$$

3. What are the usual steps to be followed for prediction using SVM.

Ans:

Steps to follow:

- Transform data to the format of an SVM package
- Conduct simple scaling on the data
- Consider the RBF kernel $K(x; y) = \exp(-\gamma\|x - y\|^2)$
- Use cross-validation to find the best parameter C and γ
- Use the best parameter C and γ to train the whole training set
- Test

4. What is unsupervised learning? (Chapter #5, Page 1-2)

5. Write down the *k*-means clustering algorithm. (Chapter #5, ~page 4)

Ans:

Algorithm: *K-means Clustering*

Inputs: Parameter K, dataset D $\{(x_1), \ldots, (x_N)\}$.

1. Initialize cluster centroids $\{m_1, m_2, \ldots, m_K\}$ randomly.

2. For $\forall i$ compute the cluster assignment,

$$C(i) = \arg\min_{1 \le k \le K} \|x_i - m_k\|^2$$

3. For $\forall k$ compute,

$$m_k = \frac{\sum_{i=1}^{N} I\{C(i) = k\}\, x_i}{\sum_{i=1}^{N} I\{C(i) = k\}}$$

where, I is the indicator function. *Note, for any m_k if $\sum_{i=1}^{N} I\{C(i) = k\} = 0$ then we reset m_k randomly.

4. Goto step 2 to repeat, unless cluster assignments ($C(i)$) do not change.

6. How can we find the optimal clusters (K^*) using the *K*-means clustering algorithm?

Ans: (Hints: See "Practical Issues", pages 6 to 7 in the lecture notes)

7. What are the basic ideas of a bootstrap method? (Chapter#6, page 1-2)

Ans:

A bootstrap is a general tool for assessing statistical accuracy.

Suppose we have a model fit to a set of training data. We denote the training set by $Z = (z_1, z_2, \ldots, z_N)$ where $z_i = (x_i, y_i)$. The basic idea is to

(1) randomly draw datasets ***with replacement*** from the training data,
(2) this is done B times ($B = 100$ say), producing B bootstrap datasets.
(3) Then we refit the model to each of the bootstrap datasets and examine the behavior of the fits over the B replications.

8. What is bagging? (Chapter #6, page 2-3)

9. Write down the steps of AdaBoost.M1 Algorithm (Chapter#6, page 4)

Ans:

Algorithm: AdaBoost.M1.

1. Initialize the observation weights $w_i = 1/N$, i = 1, 2, …, N.
2. For m = 1 to M
   (a) Fit a classifier $G_m(x)$ to the training data using weights w_i.
   (b) Compute
$$err_m = \frac{\sum_{i=1}^{N} w_i I(y_i \neq G_m(x_i))}{\sum_{i=1}^{N} w_i}$$
   (c) Compute $\alpha_m = \log((1 - err_m)/err_m)$.
   (d) Set $w_i \leftarrow w_i \cdot \exp[\alpha_m \cdot I(y_i \neq G_m(x_i))],\ i = 1,2,...,N$.
3. Output $G(x)$ = sign $[\sum_{m=1}^{M} \alpha_m G_m(x)]$.

---

10. Write down the steps of Random Forest for regression and classification (chapter#6, page 10)

Ans:

Algorithm: *Random Forest for Regression or Classification.*

1. For b = 1 to B:
   (a) Draw a bootstrap sample Z^* of size N from the training data.
   (b) Grow a random-forest tree T_b to the bootstrapped data by recursively repeating the following steps for each terminal node of the tree until the minimum node size n_{min} is reached.
      i) Select m variables at random from the p variables.
      ii) Pick the best variable/split-point among them.
      iii) Split the node into two daughter nodes.
2. Output the ensemble of trees $\{T_b\}_1^B$.

To make a new prediction at a new point x:

Regression: $\hat{f}_{rf}^B(x) = \frac{1}{B}\sum_{b=1}^{B} T_b(x)$.

Classification: Let $\hat{C}_b(x)$ be the class prediction of the b^{th} random-forest tree.

Then $\hat{C}_{rf}^B(x)$ = majority vote $\{\hat{C}_b(x)\}_1^B$.

---

11. Write down the motivation behind the development of the Random Forest algorithm.

Ans: The essential idea in bagging is to average many noisy but approximately unbiased models and hence reduce the variance. Trees are ideal candidates for bagging since they

can capture complex interaction structures in the data, and if grown sufficiently deep, have a relatively low bias. Since trees are notoriously noisy, they benefit greatly from the averaging. Moreover, since each tree generated in bagging is identically distributed (*i.d.*), the expectation of an average of B such trees is the same as the expectation of any one of them. This means the bias of bagged trees is the same as that of the individual trees, and the only hope of improvement is through variance reduction.

12. Write down the steps for k-fold cross-validation for model selection.
Ans:

1. Randomly split data (D) into *k* disjoint subsets as: $D_1, D_2, ..., D_k$.
2. For $\forall i$, model M_i is evaluated as:
   For j =1 to k:
   Train the M_i using data: ($D - D_j$) and get the predictor P_{ij}
   Test P_{ij} using D_j and get the error E_{ij}.
   EndFor j

   E_i = average of E_{ij} for $\forall j$, which is the generalized error of M_i.
3. Pick the best model, M_i having the lowest generalized error E_i.
4. Retrain $M_{i=\text{best}}$ using full dataset D.

13. Draw the characteristic curve of (a) sigmoid function and (b) hyperbolic tangent function (see class note or find it by yourself)

14. For the given Artificial Neural Network below, write the vector-equations involved in forward propagation:

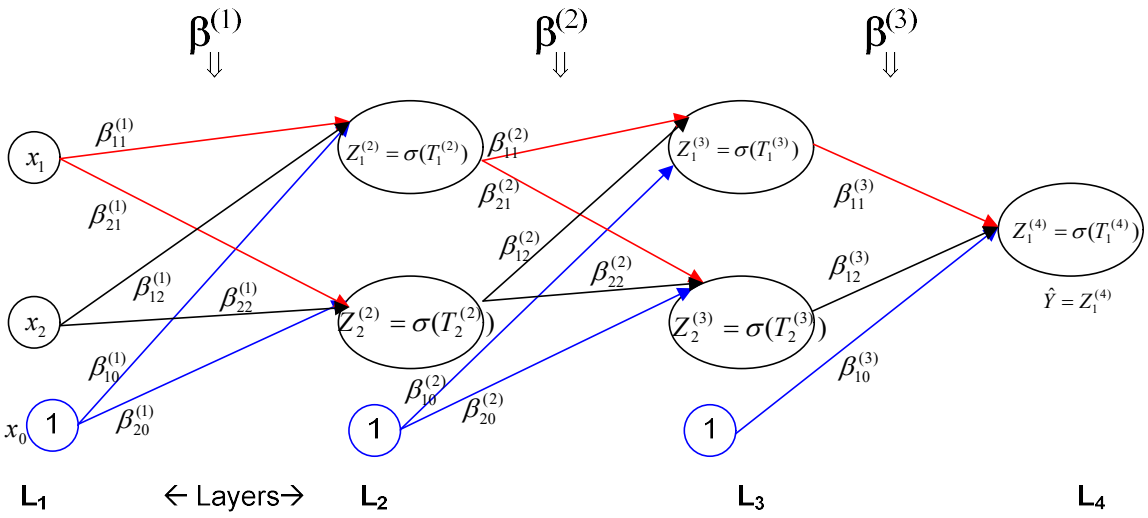


Figure: A multilayer neural network demonstrating the notations.
Assume that the transformation function σ = f_{sig} and bias unit (input "1") in each layer is presented as $Z_0^{(l)}$, where *l* is the layer number.

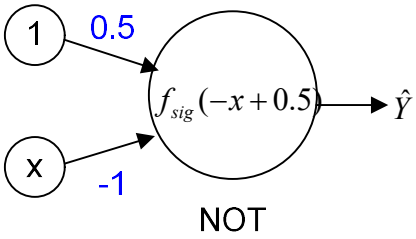
Ans:

$$Z^{(1)} = X$$
$$T^{(2)} = \beta^{(1)^T} Z^{(1)}$$
$$Z^{(2)} = f_{sig} \bullet (T^{(2)}) \quad \text{and add } Z_0^{(2)}$$
$$T^{(3)} = \beta^{(2)^T} Z^{(2)}$$
$$Z^{(3)} = f_{sig} \bullet \left(T^{(3)}\right) \text{ and add } Z_0^{(3)}$$
$$T^{(4)} = \beta^{(3)^T} Z^{(3)}$$
$$Z^{(4)} = \hat{Y} = f_{sig} \bullet (T^{(4)})$$

[Note: Don't forget the '.'s (dots) in the above equations, they indicate element-wise operations**]**

15. Design (i) NOT, (ii) OR and (iii) AND gates using a single neuron model and show the operation using truth table.

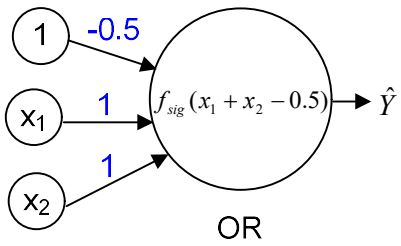
Ans:



x	$f_{sig}(-x+0.5)$	$\hat{Y}$
0	$f_{sig}(+0.5)$	1
1	$f_{sig}(-0.5)$	0

(a) (b)

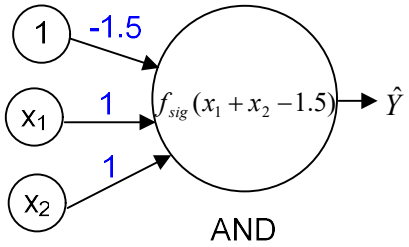
Figure A: Modeling (a) NOT and the corresponding (b) truth-table.



x_1	x_2	$f_{sig}(x_1+x_2-0.5)$	$\hat{Y}$
0	0	$f_{sig}(-0.5)$	**0**
0	1	$f_{sig}(+0.5)$	**1**
1	0	$f_{sig}(+0.5)$	**1**
1	1	$f_{sig}(+1.5)$	**1**

(a) (b)

Figure B: Modeling (a) OR and the corresponding (b) truth-table.



x_1	x_2	$f_{sig}(x_1+x_2-1.5)$	$\hat{Y}$
0	0	f_{sig}(-1.5)	**0**
0	1	f_{sig}(-0.5)	**0**
1	0	f_{sig}(-0.5)	**0**
1	1	f_{sig}(+0.5)	**1**

(a) (b)

Figure C: Modeling (a) AND and the corresponding (b) truth-table.

16. (a) What is supervised learning? (b) What is the overfitting problem?

17. (a). What are the differences between 'regression' and 'classification' problems? [Chapter #1, Page 8]

(b) Describe Newton's method for deriving the equation for finding the minimum (or maximum) of a given equation.

Ans:

Say, we have an equation $f(x) = 0$

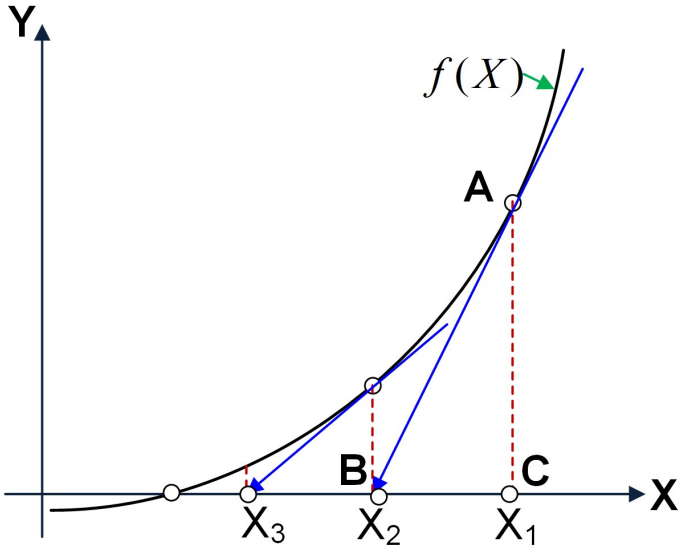


Figure: Newton's method is used for finding the solution of an equation.

For the solution of the equation (i.e., to find for what value of x, $f(x) = 0$), assume our initial point is x_1, which intersects the x-axis at C and $f(x)$ at A (see Figure above). Also, assume that the tangent at A intersects the x-axis at B where the value of x is x_2. From ΔABC and the definition of the slope of an equation, we can write:

$$f'(x_1) = \frac{f(x_1) - 0}{x_1 - x_2} \qquad (i)$$

$$x_2 = x_1 - \frac{f(x_1)}{f'(x_1)} \qquad (ii)$$

In general we can write,

$$x_{t+1} = x_t - \frac{f(x_t)}{f'(x_t)} \qquad (iii)$$

Equation (*iii*) can be iteratively used to get the solution of the equation.

Now, for a minimization problem, if minimum exists then, we actually need to find the value of x for which $f'(x) = 0$.

We can think of going for the solution of the equation, $f'(x) = 0$ using (*iii*).

Therefore, we can similarly write,

$$x_{t+1} = x_t - \frac{f'(x_t)}{f''(x_t)} \qquad (iv)$$

Now, we can use equation (*iv*) to find the minimum (or, maximum) of equation $f(x) = 0$.

.

18. Write down the pseudocode for the Genetic Algorithm.

Ans: Genetic Algorithm (GA) is a population-based optimization algorithm. The formation was inspired by natural evolution. Pseudocode for GA is given below:

1. Form the initial population (usually random)
2. Compute the fitness to evaluate each chromosome (member of them population)
3. Select pairs to mate from best-ranked individuals and replenish population
    a. - Apply crossover operator
    b. - Apply mutation operator
4. Check for termination criteria, else go to step #2

Figure: Pseudocode for Genetic Algorithm

19. Write down the pseudocode of the iterative hill-climbing algorithm.

Ans:

```
procedure iterated hillclimber
begin
  t ← 0
  repeat
    local ← FALSE
    select a current string v_c at random
    evaluate v_c
    repeat
      select 30 new strings in the neighborhood of v_c
        by flipping single bits of v_c
      select the string v_n from the set of new strings
        with the largest value of objective function f
      if f(v_c) < f(v_n)
        then v_c ← v_n
        else local ← TRUE
    until local
    t ← t + 1
  until t = MAX
end
```

20. Write down the pseudocode of the simulated annealing algorithm.

Ans:

```
procedure simulated annealing
begin
  t ← 0
  initialize temperature T
  select a current string v_c at random
  evaluate v_c
  repeat
    repeat
      select a new string v_n
        in the neighborhood of v_c
        by flipping a single bit of v_c
      if f(v_c) < f(v_n)
        then v_c ← v_n
        else if random[0, 1) < exp{(f(v_n) − f(v_c))/T}
            then v_c ← v_n
    until (termination-condition)
    T ← g(T, t)
    t ← t + 1
  until (stop-criterion)
end
```

21. Name ten non-deterministic algorithms (Excluding Genetic Algorithms, Hill-Climbing, Simulated Annealing).

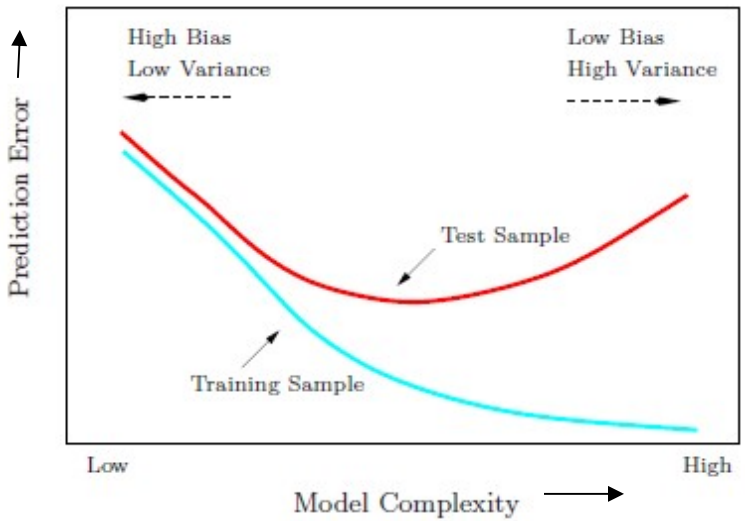
Ans:

Particle Swarm Optimization (PSO),
Artificial Colony Optimization (ACO),

Artificial Bee Colony (ABC),
Glowworm Swarm Optimization (GSO),
Cuckoo Search Algorithm (CSA),
Firefly Algorithm (FA),
Bat Algorithm (BA),
Monkey Algorithm (MA),
Krill Herd Algorithm (KHA),
Wind Driven Optimization (WDO),
Social Spider Algorithm (SSA),
Artificial Immune Systems,
Conformational Space Annealing, …

22. (a) Draw a figure to show the relationship among *Prediction Error*, *Model Complexity*, *Bias*, *Variance*, *Training dataset,* and *Test dataset*, (b) What is the relation between, the Residual Sum of Squares (RSS) and the Mean Squared Error (MSE), (c) What is the relationship among MSE, Bias Error, and Variance Error?

Ans:
(a)



(b) The relation between the Residual Sum of Squares (RSS) and the Mean Squared Error (MSE) is given by the following equation:

$$MSE = \frac{1}{N} RSS$$

Here, *N* is the number of actual and predicted data pairs (pair of actual and its corresponding predicted data points) in RSS.

(c) The relationship among MSE, Bias-Error, and Variance-Error are given by the following equation:

$$MSE = (\text{Bias-Error})^2 + \text{Variance-Error}$$

---------- Recommender System / Chapter 07 ----------

23. What is the long tail phenomenon? Explain.
Ans**:** See slide 6 and slide 7.

24. What are the major classes of recommender systems, and how do they work?
Ans: see slide 10.

25. What is the utility matrix, and how would you formally define it?
Ans: see slide 12 and 11.

26. What are the key problems that a recommender systems address?
Ans: Slide 15.

27. How will you gather a rating for the utility matrix? Explain.
Ans: Slide 16.

28. How will you build the profile for (a) item containing the movie and (b) item containing text – explain.

Ans: Slide 21-26.

29. Suppose the only features of movies are the set of actors and the average rating. Consider two movies with five actors each. Two of the actors are in both movies. Also, one movie has an average rating of 3 and the other an average of 4. The two vectors are given below (row-wise):

0	1	1	0	1	1	0	1	3α
1	1	0	1	0	1	1	0	4α

Here, the last component shown represents the average rating, which has been shown to have an unknown scaling factor α. Explain how you will compute the similarity of the two movies and discuss the impact of the possible values of α.

Ans: Slide 27-28.

30. Describe the pros and cons of the content-based recommender system.
Ans: Slide 33-34.

31. Given the following utility matrix, among the 03 similarity measurement approaches: Jaccard similarity, cosine similarity, normalized or centered cosine similarity – which method will be most appropriate to apply and why? Explain.

	HP1	HP2	HP3	TW	SW1	SW2	SW3
A	4			5	1		
B	5	5	4				
C				2	4	5	
D		3					3

Ans: Slide 37-43.

32. Provide and explain two possible options to compute rating prediction in collaborating filtering.

Ans: Slide 45.

33. How will you use the item-item collaborative filtering to compute the rating of user *x* on item *i*? Explain.

Ans: Slide 46.

34. Describe the pros and cons of collaborative filtering.

Ans: Slide 52.

35. Describe the common practice in collaborative filtering where global baseline estimation is linearly combined to compute the ratings of user *x* on item *i*.

Ans: Slide 56, 57.

---- x ----