Example questions – Digital Multiple Choice Exam 2021

1. Selecting machine learning approach

A machine learning professor wants to test out four new pedagogical approaches, and wants to use the different approaches on four student groups. The professor has information about the students' age, gender, study program, average grade so far and mathematical background, and will use this to create groups so that each group consists of similar students. Next year, the professor will use the results from this years' exam to predict what the results of the new students will be if placed in each of the four groups. A new student shall be placed in the group that would give the highest predicted grade for that student.

Disregard how smart this approach will be (there would for example be problems with bias in the second year). Which machine learning approaches would be most suitable to use?

a) First k-means clustering and then decision tree regression
b) First k-means clustering and then logistic regression
c) First pca dimension reduction and then logistic regression
d) First pca dimension reduction and then decision tree regression

Why: You would not use pca dimension reduction to divide your data points into 4 groups. K-means clustering would help you with that. Decision tree regression would make it possible for you to predict a numerical grade. Logistic regression is a binary classification approach.

2. Learning a Naïve Bayes model

You are running a machine learning process using categorical naïve Bayes on a small training data set.

A	B	Class
1	2	R
2	3	S
3	1	T
4	4	T
2	2	T
3	4	S
4	2	R
2	4	T
1	1	T

For **smoothing** you add 1 to the count of each feature value per target class, and also 1 to each count of target values, i.e., you start with count(A=x, Class=z) = 1 and count(B=y, Class=z) = 1 and count(Class=z) = 1 when building you model. What are the probabilities p(Class = T) and p(B = 1|Class = T) in the learned model?

a) p(Class = T) = 0.556 and p(B = 1|Class = T) = 0.333
b) p(Class = T) = 0.556 and p(B = 1|Class = T) = 0.400
c) p(Class = T) = 0.500 and p(B = 1|Class = T) = 0.400
d) **p(Class = T) = 0.500 and p(B = 1|Class = T) = 0.**333

Why: You start with Count(T) = Count(S) = Count(R) = 1 Then for each datapoint you add 1 to to the correct class. You will end up with count(T) = 6, count(R) = count(S) = 3. So p(class=T) = 6/(6+3+3) = 0.5. You also start with count(B=x | Class = T) = 1. There are 5 instances of Class = T. You get count(B= 1|class = T) = 3, count(B= 2|class = T) = 2, count(B= 3|class = T) = 2 and count(B= 4|class = T) = 3. P(b = 1 | class = T) = 3/(3+2+1+3) = 1/3 = 0.333.

3. Capacity of machine learning model

A machine learning expert says that a model she has designed has very high capacity. You train the model on a rather small data set but with many uncorrelated features. How will this model most likely score in when you compare it to other models?

a) High training set score and low test set score
b) Low training set score and high test set score
c) High training set score and high test set score
d) Low training set score and low test set score

Why: High capacity means that it can represent very many hypotheses, and will have the potential to learn a model that fits data well. But you have few data and many features. This will have the effect that you will most likely overfit a lot. So you will get high training set score. For new data however, you will not get good results, as the model because of its high capacity has adapted too much to the training set.

4. Linear classifiers

You have a data set with two input features x and y with a binary classification (T,F). You have the following data points: (x = 4, y= 2,F), (x=2,y=5,F), (x=6,y=4,T), (x=4,y=6,T). Which of the following lines separates the data best?

a) y = -2x-12
b) y = 2x -12
c) y = 2x + 12
d) y = -2x+12

Why: Linear classifiers divides data in two groups, above and under the line. If you draw the points in a coordinate system, and place these lines in the same coordinate system you would see which one divides the data in two.

5. Random Forest regression

You aim at running a random forest regression algorithm with 1000 decision trees. Each of them returns a value prediction. How would you compute the final output from the model based on these values?

a) Voting
b) Computing R^2
c) Averaging
d) Computing accuracy

Why: Random forest regression is about predicting a numerical value. Averaging is the only way to find a good prediction here. R^2 an accuracy is about scoring models. Voting is useable only if you do classification with random forest.

6. The ReLU activation function

A neural network unit has two inputs x[1] = 0.45, x[2] = 0.85, weights w[1] = -0.92, w[2] = 0.26 and b = 0.12. We run the standard linear computation and use a ReLU activation function. What will be the output from the unit

a) -0.073
b) 0.000
c) 0.073
d) 0.481

Why: The relu activation function is (in this case) max(0.0, w[1]x[1] + w[2]x[2] + b). I fyou compute the value you get 0.000

7. Dimension Reduction

The data you have has 212 features, all of them with values [0.0,10.0], and the target value is a number in the interval 0.0,50.0]. You want to reduce your data set so that each data point is represented by exactly 10 different features, each feature indicating how much each data points fits a particular aspect of the data. What would you use?

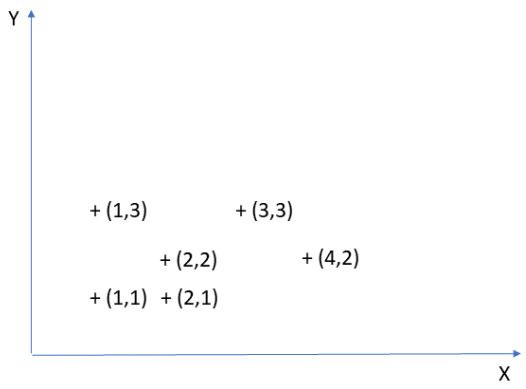
a) PCA
b) NMF
c) t-SNE
d) k-Means

Why: PCA is a candidate here as you may represent a data point with 10 principal components. But as I state that all values are non-negative and I want EXACTLY 10 features, a non-negative matrix factorization (NMF) is in fact a much better choice. Also because each feature then represent something that may be an understandable feature. K-means is for clustering and not so useful here. t-SNE is a visualization technique.

8. DBSCAN

You run a DBSCAN algorithm and starts out with the point (2,2). You start with parameters min_samples = 3 (number of close neighbours including sample itself) and eps = 1.5 (boundary for close). Use Euclidian distance.

How many of the 6 data points in the figure will be so called core samples?



a) 3
b) 4

c) 5
d) 6

Why: A core sample in DBSCAN as applied here is a data point that has at least two (min_samples is 3) neighbours within a distance of 1.5 (eps = 1.5). This goes for the points (1,1), (2,1), (2,2), and (3,3). The tow other points have only 1 neighbour within eps = 1.5. The Euclidian distance is sqrt((x1-x2)^2+(y1-y2)^2). For many pairs of the points this is sqrt(2) = 1.414, bringing the pairs inside the eps=1.5 value.

9. Q-learning

Your robot starts up in state s4, but has no knowledge about the other states of the domain. The q-algorithm is adapted so it allows for new states and actions as they are encountered, and any new state-action pair will get an initial q-value of 1.0. The robot randomly selects action a6, gets a reward of 3, and moves to a new state s5 (having a new set of actions available). With a discount factor of 0.8 and a learning rate of 0.1, what is the q-value for the state-action-pair (s4,a6) after the reward has been received?

a) 1.28
b) 1.5
c) 4.0
d) 2.82

Why: You use the q-learning update formula

$$Q(s_t, a_t) \leftarrow (1-\alpha) \cdot \underbrace{Q(s_t, a_t)}_{\text{old value}} + \underbrace{\alpha}_{\text{learning rate}} \cdot \overbrace{\left(\underbrace{r_t}_{\text{reward}} + \underbrace{\gamma}_{\text{discount factor}} \cdot \underbrace{\max_a Q(s_{t+1}, a)}_{\text{estimate of optimal future value}} \right)}^{\text{learned value}}$$

Any new state action pairs encountered has an initial Q-value of 1 when you see them.

Q(s4,a6) = 1.0 initially and lead to state s5 which we have not seen before, giving a max Q-value for Q(s5,a) = 1.0 (by the information in the question). Discount factor = 0.8 and reward = 3, and learning rate = 0.1 gives a new learned value of 0.38. We multiplyt the old value (1.0) with 1-0.1 = 0.9 and get 0.9 which is added to 0.38. Updated value is 1.28