# Reviewing

## Week 1a: linear/logistic models, training/testing, Gradient Descending

+ linear model is to capture the linear relationship between two variable x and y, or we can write y = wx + b for prediction

+ logistic model is to capture the probability of Bernoulli distribution of y and w\*x + b for binary classification.

+ learning the model by gradient descending.

+ training set is to build the model and test if the model fit; testing set is to test if the model can handle new input.

### A. Determine if the statement is about training (A) or testing (E) or cross-validation (V) or coin/Bernoulli splitting (S)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| A1. We can repeat the test many times with the fixed size for the test set with this method | A | E | **C** | S |
| A2. We can fit a model using it | **A** | E | C | S |
| A3. When we are given a model, we can use it to test if it is over-fitting | A | **E** | C | S |
| A4. We use it to split the dataset into two parts, the larger part is for training and the smaller part is for testing. | A | E | C | **S** |
| A5. We use it to split the dataset into training and testing sets but we don't have the same size of the test set all the time depending on the randomization seed | A | E | C | **S** |

### B. Determine if the statement is about linear model (L) or logistic regression model (G) or SVM classification model (S)

|  |  |  |  |
| --- | --- | --- | --- |
| B1. We can classify new pattern x by the formula y = sign(w\*x+b) where w is the sum of alphaj\*yj\*xj | L | G | **S** |
| B2. We use it to learn the linear relationships between variables in the given data table, for example food price increases when gas price increases | **L** | G | S |
| B3. We use it to solve the binary classification problem and we can learn the model parameters with Gradient Descending | L | **G** | S |
| B4. We have to use training examples (we cannot stay away from training examples) to classify the new input | L | G | **S** |
| B5. We can have the slack variable or the control parameter to adjust the confusion area | L | G | **S** |
| B6. We can get rid of training examples in binary classification using this method | L | **G** | S |

## Week 1b: mixture model, GMM, kmeans, EM

+ mixture model is to combine simple distribution to capture complicated dataset.

+ Gaussian mixture model is a mixture model with Gaussian distributions as mixture components.

+ using kmeans or EM to learn GMM

### C. Complete the equations for the GMM

Mix the two Gaussian distributions p1(x) = N(-10, 1) and p2(x) = N(10, 1) with the same weights.

|  |
| --- |
| C1. The GMM model will be p(x) = …**0.5**… p1(x) + …**0.5**….. p2(x) |

Choosing the closest value to complete the equation

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| C2. Compute  = ………. | 0 | 0.5 | **0.75** | 1 |
| C3. Compute  = ……… | 0 | 0.5 | **0.75** | 1 |
| C4. Compute p(0) = ……… | **0** | 0.5 | 0.75 | 1 |
| C5. When we set w1=0 and w2=1, assuming that w1/p1 and w2/p2 are in pairs, compute  = ……… | 0 | **0.5** | 0.75 | 1 |

### D. Determine if the statement is about EM method (E) or kmeans (K) or neither of them (N)

|  |  |  |  |
| --- | --- | --- | --- |
| D1. We try to find circular clusters with this method | **K** | E | N |
| D2. Many methods, including Baum-Welch, pLSA, are built based on this method | K | **E** | N |
| D3. We have the E-step and the K-step in this method | K | E | **N** |
| D4. We try to find elliptic clusters with this method | K | **E** | N |
| D5. We ignore the size (the variance) of each cluster using this method | **K** | E | N |

## Week 2a: SVM

+ to classify by separating the samples with line w\*x + b

+ w can be built based on the support vectors.

+ we learn support vectors by SMO

+ we can use kernel to classify non-linear datasets (by transforming feature with kernel).

### E. Determine if the statement is about linear SVM (L) or kernel SVM (K) or soft-margin SVM (S) or none (O) of them

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| E1. We use it to separate complicate datasets, such as quadric dataset, xor datasets | L | **K** | S | O |
| E2. We use one pair of (w, b) to classify the dataset with three output classes (categories) | L | K | S | **O** |
| E3. We can adjust the confusion area using this type of SVM, we can use slack variable in learning it | L | K | **S** | O |
| E4. When we see that we have two separating clusters for two classes, we can use this type of SVM | **L** | K | S | O |
| E5. When we see that we have the two clusters. One class is within the inner circle, the other is the outer circle, we have to use this kind of SVM | L | **K** | S | O |

## Week2b: boosting, bagging, random forest

+ combining methods

+ we have to tell the differences between the methods

+ bagging: to remove (bootstrap) the rows only

+ random forest: to remove (bootstrap) the rows and columns and growing tree

+ boosting: to build the new classifier based on the error sample only (emphasizing on error)

+ tree: to divide the samples into homogeneous regions to reduce error (more homogeneous, fewer errors). Tree is very fast classifier.

### F. Determine if the statement is about Boosting (B) or bagging (A) or random forest (F) or classification tree (T)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| F1. We use output labels to divide the sample space into homogeneous regions in order to reduce the classification error and quickly locate the answer | B | A | F | **T** |
| F2. We use weights to emphasize on the misclassified samples on the current classifier to build the next. | **B** | A | F | T |
| F3. We build the weak classifiers independently with randomized dataset via row bootstrapping | B | **A** | F | T |
| F4. We build the weak classifiers independently with randomized dataset via row/column bootstrapping | B | A | **F** | T |
| F5. We build linearly-combined classifier, the linear coefficients of which are based on the current classification error | **B** | A | F | T |

## Week 3b: HMM

+ to capture the relationship between symbol (observed) sequences and the state sequences.

+ for state Y and symbol sequence X, we will have the model p(X, Y) given by (A, B, π).

+ A is the probability state transition matrix. A(i,j) is the probability that we move from state i to state j.

+ B is the emission matrix. B(j, k) is the probability that state j produces symbol k. When we hear the cellphone number in the first place j=1 we expect to hear the word k=0 and the second j=2 we expect to hear k=9.

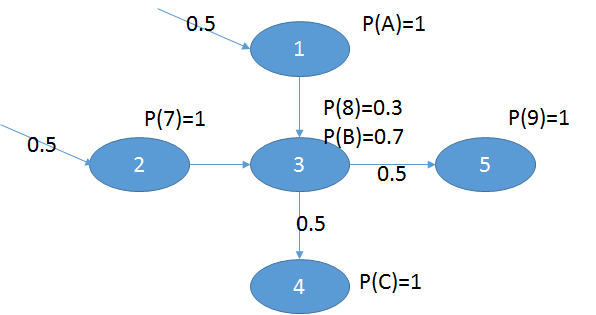
+ π(j) is the probability that state j will be the initial state (or the starting point of evaluation).

+ learning the HMM model with Baum-Welch procedure (E-M procedure with different E step and M step); Viterbi training algorithm

+ find the best state sequence given the symbol sequence: Viterbi

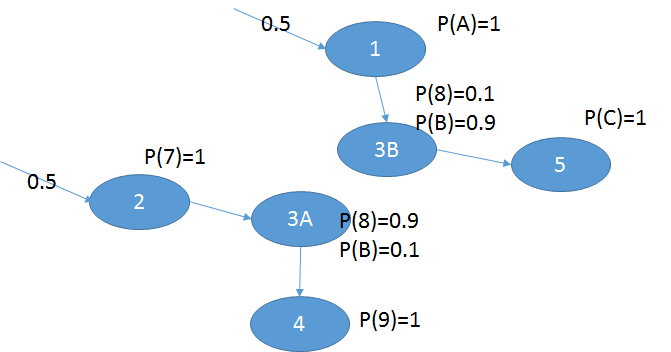
### G. Compute the probability of the following sequence

You are given the model



|  |
| --- |
| G1. The probability of observing ABC in the model is …**0.5\*0.7\*0.5**….. |
| G2. The probability of observing 789 in the model is …**0.5\*0.3\*0.5**….. |
| G3. The probability of observing 78C in the model is …**0.5\*0.3\*0.5**….. |
| G4. The probability of observing AB9 in the model is ……**0.5\*0.7\*0.5**…. |

### H. Compute the sequence based on the following model



|  |
| --- |
| H1. The probability of observing ABC in the model is …**0.5\*0.9\*1**….. |
| H2. The probability of observing 789 in the model is …**0.5\*0.9\*1**….. |
| H3. The probability of observing 78C in the model is …**0**….. |
| H4. The probability of observing AB9 in the model is …**0**….. |

## Week 3b: sampling, optimization

+ sampling method: to draw samples from a distribution, if we draw enough, we can reconstruct the original distribution.

+ some methods we have learn: uniform U(a, b), box-muller for N(mu, sigma), 12U-6 for N(mu, sigma), rejection for complicated distribution, importance sampling also for complicated distribution without rejection.

+ exploring methods (to find a path from initial point to the optimal point): gibbs sampling, simulated annealing, enumeration, uniform sampling.

+ optimization: to find the best (or the optimal, or the maximal, or the minimal) point of the function f(x) given the range or the constraint x ∈R

+ kmeans, gradient descending, gibbs sampling, simulated annealing are all to find the optimal solution from one initial point.

### K. Determine if the statement is about Gibbs sampling (G), Simulated Annealing (A), rejection sampling (R) or Box-Muller method (B)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| K1. We can draw N(mu, sigma) samples from this method | G | A | R | **B** |
| K2. We discard sample xk drawing from q(x) using the value of (xk) with this method | G | A | **R** | B |
| K3. We fix all dimensions except for one current dimension to draw the sample with this method | **G** | A | R | B |
| K4. We use the control variable to determine whether to move forward even when we meet lower value of the function we want to optimize | G | **A** | R | B |
| K5. We can sample within a circle with this method like we turn right four time around the city block and return to the same place | **G** | A | R | B |

## Week 4: clustering

+ different methods to divide the samples into groups (clusters): kmeans, GMM, hierarchal clustering, spectral clustering.

+ hierarchical clustering: look at the tree and read it.

+ to evaluate the error for kmeans.

+ to understand the E-M process (initializing, labeling, averaging)

+ to tell the differences between EM and kmeans

### L. Determine if the statement is about hierarchical (H) or spectral (S), or kmeans (K)

|  |  |  |  |
| --- | --- | --- | --- |
| L1. We need Laplacian matrix of the similarity matrix to find clusters | H | **S** | K |
| L2. We find clusters by building the tree first then we cut the tree to smaller similar groups | **H** | S | K |
| L3. We start from randomized clusters and reduce the error by adjust the randomized clusters | H | S | **K** |

### M. Hierarchical clustering

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | M1. When we build the tree from the dataset  (1, 2, 7, 7.1, 9, 10) the label will be   |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | A | B | C | D | E | F | |  |  |  |  |  |  |   M2. And when we cut the tree into 2 classes the labels will be   |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | 1 | 2 | 7 | 7.1 | 9 | 10 | |  |  |  |  |  |  | |

## Week 5: deep learning

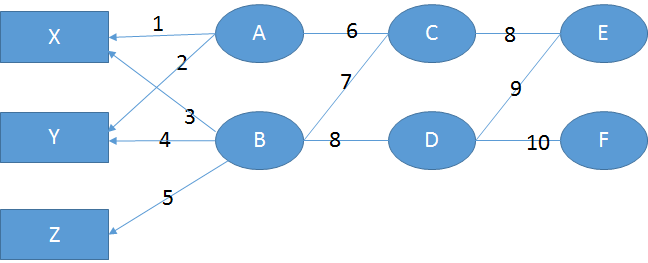
+ motivation to deep learning?

+ concept of latent/hidden variables and how to work with them

+ from model to equations and from equations to model

+ the 3 typical types of DL model

### N. Write the linear combination for each layer



|  |
| --- |
| N1. First layer ……**1A+3B** and **3A+4B** and **5B**... |
| N2. Second layer …**6AC+7BC+8BD**…... |
| N3. Last layer …**9CE+9DE+10DF**…... |