1. What is conditional independence in a graphic model?
   - X and Y are conditionally independent if
     $P(X,Y) = P(X)P(Y)$
   - And X and Y are conditionally independent given Z if
     $P(X,Y \mid Z) = P(X \mid Z)P(Y \mid Z)$ or
     $P(X \mid Y,Z) = P(X \mid Z)$
   - Three canonical cases of conditional independence:
   - Head-to-tail
     - $X \rightarrow Y \rightarrow Z$
     - $P(X,Y,Z) = P(X)P(Y \mid X)P(Z \mid Y)$
     - Y always blocks (separates) -> X and Z are conditionally independent
   - Tail-to-tail
     - $Y \leftarrow X \rightarrow Z$
     - $P(X,Y,Z) = P(X)P(Y \mid X)P(Z \mid X)$
     - When X is known it blocks (separates) -> Y and Z are independent given X
   - Head-to-head
     - $X \rightarrow Z \leftarrow Y$
     - $P(X,Y,Z) = P(X)P(Y)P(Z \mid X,Y)$
     - When Z or any of its descendants are unknown it blocks (separates) -> X and Y are independent
2. Why does bagging profit from the unstable model? What is the unstable algorithm? Give examples!
   - An unstable algorithm is sensitive to small changes in the training set
   - Bagging trains multiple models on different random subsets of the training set
   - If an unstable algorithm is used with bagging, the different subsets create very different models, resulting in diverse “expert” models that are accurate on different parts of the data
   - Example:
     - A decision tree is unstable, as it chooses different features and produces different splits due to small difference in the training data distribution resulting in different structures that work better on different parts of the data

        - SVMs: SVMs are sensitive to the choice of kernel function and the hyperparameters. Different training subsets or slightly different hyperparameter settings can lead to variations in the decision boundaries generated by SVMs.

3. When do we use LDA (Latent Dirichlet Allocation)?

- When a set of observations needs to be explained with unobserved groups
    - Organise documents into topics based on the words contained in the documents -> We do not know which word corresponds to which topic

4. How to build a regression tree?

- Instead of using a class-based impurity measure, split based on the variance
- The split with the lowest total variance summed across every branch is chosen
- On leaf nodes take the mean (or median) as the output

5. How is the depth of Tree size and the K in Nearest Neighbours affecting bias and variance?

- By increasing the depth of a tree we get more splits which result in more specific models
    - More variance but less bias
- By increasing K we take into account more training samples, thus get a smoother model that averages out the variance of single samples, which is more general
    - Less variance but more bias

6. What is the difference between confidence and support?

- Support is the total number a given itemset or association appears in the training set relative to the training set size
    - N = number of samples, X, Y are random variables
    - Support(X → Y) = P(X,Y) = |{X, Y}| / N
- Confidence is the strength of an association, how many times does Y appear with X relative to X appearing in total (alone or with Y)
    - Confidence(X → Y) = P(Y | X) = P(X,Y) / P(X)

7. Explain the adaptive nearest neighbor method!

- The idea is adjusting the distance metric locally so that neighborhoods stretch out where class labels don’t change much
- This is achieved by defining the distance metric using the covariance matrices of inter-class and intra-class samples

- $D(x, x_0) = (x - x_0)^T \Sigma (x - x_0)$
  - $\Sigma = W^{-1/2}(W^{-1/2} B W^{-1/2} + \epsilon I)W^{-1/2} = W^{-1/2}(B^{*} + \epsilon I)W^{-1/2}$
  - W and B are computed from the Nearest Neighbors of x_0
  - W within-class covariance matrix: $W = \sum_{k=1}^{K} \pi_k W_k$
  - B between-class covariance matrix: $B = \sum_{k=1}^{K} \pi_k (\bar{x}_k - \bar{x})(\bar{x}_k - \bar{x})^T$

8. What is lasso and ridge regularisation?

- They are used with parametric models e.g. linear or polynomial regression
- Lasso adds the sum of the absolute value of the weights to the loss: |w|
- Ridge adds the sum of the squared value of the wights to the loss: w^2

9. Why is the chain of local probable states in the Hidden Markov Model not a good solution?

- The question refers to the evaluation problem of a HMM
- The reason is that evaluating every possible path of the HMM would require to examine every possible sequence of outputs by the Markov chain which would be an N^T algorithm (N number of states, T: length of a sequence)
- This comes from P(O | λ) = $\sum_{Q} P(O, Q \mid \lambda)$
- Solution is the forward-backward procedure

10. What is the soft margin classifier?

- When a training set is non-separable (with the given kernel) the margin can not be empty
- In this case with the introduction of slack variables (ξ), the separating hyperplane which incurs the least error is chosen
- The goal is to find: $r^t(w^T x^t + w_0) >= 1 - \xi^t$
- While adding the error $\sum_t \xi^t$

11. What impurity measures are there?

- Entropy: $\phi(p, 1 - p) = -p \, log_2 p - (1 - p) \, log_2 (1 - p)$

- Gini index: $\phi(p, 1-p) = 2p(1-p)$
- Misclassification error: $\phi(p,\ 1 - p) = 1 - max(p, 1 - p)$
- General entropy: $I_m = -\sum_{i=1}^{K} p_m^i \, log_2(p_m^i)$

12. How are attributes cut in a decision tree?

- Choosing the attribute which after cutting results in the lowest possible entropy summed over all branches
  - If the attribute is categorical, there will be a branch for every possible category value for that attribute
  - If the attribute is numerical, there will be a threshold, which cuts the values into two regions making it a binary decision

13. What is Gibbs sampling? How does it work? What do we use it for?

- Gibbs sampling is an MCMC (Markov Chain Monte Carlo) method used for sampling a complex probability distribution - Gibbs sampling is specifically used for multivariate distributions
- Each sample depends on the last one which means the higher density regions are found faster than with acceptance-rejection sampling, which does not use this information
- It is used when directly sampling from (two variables as an example, but it works with any number) P(X,Y) is difficult but P(X | Y) and P(Y | X) are not
- Algorithm:
  - Start from a sample (x^0,y^0)
  - One iteration:
    - Sample $x^1 \sim P(x \mid y^0)$
    - Sample $y^1 \sim P(y \mid x^1)$
  - Result (x^1,y^1) → goto start

14. What is Naïve Bayes and the Graphical model?

- Graphical model:
  - uses a directed graph to represent a complex probability distribution where
    - Nodes are the random variables P(X)
    - Directed edges represent “direct” influence (conditional dependence) between the two random variables
      X → Y = P(Y | X)
  - With the Bayes rule, the edges can be followed in both directions

        - Forward: Casual (using the cond. dep.)
        - Backward: Diagnostic (using the Bayes rule on the cond. dep.)
- Naïve Bayes
    - A kind of graphical model used for example for classification, where the input features assumed to be independent, and all depend on the class
    - E.g. with two input features: x_0 ($P(x_0 | C)$) ← C ($P(C)$) → x_1 ($P(x_1 | C)$)
    - $P(X | C) = \prod_{j=1}^{d} P(x_j | C)$

15. What are we optimising at SVMs?

- We are maximising the margin between two different classes
    - $X = \{x^t, r^t\}$
    - $r^t = +1$ if $x^t \in C_1$
    - $r^t = -1$ if $x^t \in C_2$
    - $r^t(w^T x^t + w_0) >= +1$ (instead of 0, due to the margin)
    - min $½ ||w||^2$
        - subject to $r^t(w^T x^t + w_0) >= +1\ \forall t$
    - Convert this using Lagrange multipliers to
        - $1/2||w|| - \sum_t \alpha^t r^t (w^T x^t + w_0) + \sum_t \alpha^t$
        - Where w and w_0 are maximised w.r.t. α^t
- Margin: the distance of the closest training examples from the optimally separating hyperplane on both sides

16. What is the HMM and its parts?

- A Markov Model is based on the Markov assumption that the next state only depends on the current state
    - States: $S_1, S_2, \ldots S_N$
    - State at time t: $q_t = S_i$
    - First-order: $P(q_{t+1} = S_j | q_t = S_i, q_{t-1} = S_k, \ldots) = P(q_{t+1} = S_j | q_t = S_i)$
        - The past doesn't count, only the present
    - Transition probabilities: $a_{ij} = P(q_{t+1} = S_j | q_t = S_i)$
        - The prob. of going from state i to state j
        - Given: $a_{ij} \geq 0$ and $\sum_{j=1}^{N} a_{ij} = 1$
    - Initial probabilities: $\pi_i = P(q_1 = S_i)$

        - The prob. of starting in state i
        - Given: $\sum_{j=1}^{N} \pi_i = 1$
    - Observations: e.g.: O = $\{S_i, S_i S_j, S_k\}$
  - It can be thought of as a Stochastic Automaton where the transitions between states are given by probabilities
  - In the case of Hidden Markov Models, the states are not directly observable (they are hidden), we only get observations which are also a probabilistic function of the state
    - Discrete observations: $O = \{v_1, v_2, \dots, v_M\}$
    - Emission probabilities: $b_j(m) = P(O_t = v_m \mid q_t = S_j)$
        - Probability of seeing observation v_m given that we are in state S_j
  - Final elements of an HMM:
    - N: Number of states
    - M: Number of observation symbols
    - $A = [a_{ij}]$: N by N state transition probability matrix
    - $B = b_j(m)$: N by M observation probability matrix
    - $\Pi = [\pi_i]$: N by 1 initial state probability vector
    - $\lambda = (A, B, \Pi)$, parameter set of HMM

17. How to test models at Adaboost?

  - Testing with Adaboost works as a weighted voting algorithm
  - All models, d_j predict on the sample, then a weighted average is taken where w_j is proportional to the model's accuracy on the training set:
    - $w_j = log(1 / \beta_j)$ where β is calculated during training as
    - $\beta_j = \epsilon_j / (1 - \epsilon_j)$ where ϵ is the error rate

18. What is Regularisation?

  - Regularisation is constraining the model's complexity to achieve better generalisation and reduce the variance of the model
  - Due to Occam's razor, simpler models are prefered
  - Examples:
    - Polynomial regression - the degree of the polynomial should be as low as possible while also minimising the training loss
    - Decision tree - the depth of the tree should be as low as possible while also minimising the training loss

19. How can we interpret the Naïve Bayes classifier (with only descrete attributes) as a graphical mode? How is the conditional independence relevant in a Naive Bayes Classifier?

- In graphical model representation of the Naïve Bayes classifier, each attribute is a Node connected to class variable node. The nodes represent discrete random variables, and the connections capture the conditional dependencies. The Naïve Bayes assumption assumes independence between attributes give the class variable.
- In Naïve Bayes classifier, the assumption of conditional independence among the attributes given the class variable is relevant because it simplifies in the model and makes the computation of probabilities more tractable

By assuming conditional independence, the Naïve Bayes classifier simplifies the modeling process, reduces computational complexity, and allows for efficient probability estimation. While the assumption ay not always hold in real-world scenarios. Naïve Bayes can still perform well in practice, especially when the attributes are weakly dependent or when there is a large amount of training data.

20. What is a Hidden Markov Model(HMM), and using which concepts(do we need describe it)?

- A Hidden Markov Model(HMM) is a probabilistic model that is widely used for modeling sequential data with hidden states.
- A HMM using following concepts: Hidden states, Observations, state Transitions, Emission Probabilities, Markov Property, Parameter Estimation, Inference.
- it is important to describe the key concepts of a Hidden Markov Model (HMM) to provide a comprehensive understanding of how the model works. Describing these concepts helps in clarifying the underlying principles and functionalities of an HMM.

21.What is kernel estimator and what are its advantages?

A kernel estimator is a non-parametric method used to estimate the probability density function (PDF) of a random variable. It involves placing a kernel function at each data point and summing them to create a smoothed estimate of the PDF. The choice of kernel function and bandwidth parameter determines the shape and smoothness of the estimate.

Advantages of a kernel estimator:

Flexibility in approximating various probability density functions.
Non-parametric approach without restrictive assumptions.
Smooth estimation of the density function.
Adaptive smoothing through the bandwidth parameter.
Consistency in converging to the true density with increasing sample size.
Robustness in handling irregular or missing data.
Compatibility with other methods for clustering or classification tasks.

22. What is the role of description length in IREP algorithm?

In the IREP rule learning algorithm, the description length guides the selection and pruning of rules, favoring shorter and more interpretable rules to achieve a balance between accuracy and simplicity.

23. Kernel Machine: What is difference between the primal and the dual optimization problem?

The primal and dual optimization problems are two formulations of the optimization problem in kernel machines, such as Support Vector Machines (SVMs).

The primal optimization problem is formulated in the original input space, while the dual optimization problem is formulated in the feature space induced by the kernel function.

The primal problem directly optimizes model parameters, while the dual problem optimizes Lagrange multipliers associated with constraints.

Both formulations can yield the same solution, but one may offer computational advantages over the other in specific cases.

24.What is the Difference of prepruning and postpruning?

prepruning occurs during the tree construction process and stops the growth of the tree based on specific criteria, aiming to prevent overfitting.

Postpruning occurs after the tree is built and involves pruning or simplifying parts of the fully grown tree based on validation set evaluation to improve its generalization performance.

25. What is kernel estimation? when should we use kernel estimation?

Kernel estimation is a non-parametric method used to estimate the probability density function of data. It is employed when the true distribution is unknown or difficult to model parametrically. By placing kernels on data points and summing them, it creates a smooth density estimate. Kernel estimation is useful for exploratory analysis, anomaly detection, non-parametric regression, and statistical testing. It offers flexibility and does not assume a specific distribution shape.

26. Why we use Belief propagation in graphical model? give examples!

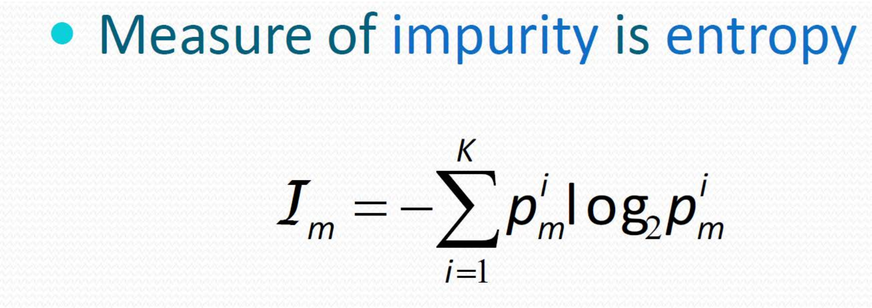
Reason: belief propagation is used because it provides a practical and efficient approach to inference in graphical models. It enables probabilistic reasoning, inference, and decision-making in complex systems with uncertainty, making it applicable to various fields

Examples:

Bayesian Networks: In a Bayesian network, belief propagation can be used to compute the posterior probabilities of unobserved variables given the observed evidence.

Markov Random Fields: In image processing, Markov random fields can model the spatial relationships between pixels. Belief propagation can be used to infer the most likely configuration of unobserved pixels based on the observed neighboring pixels.

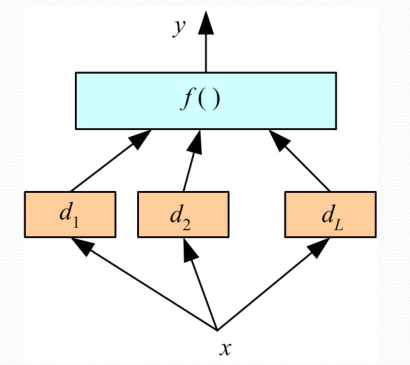
27.How impurity is measured in a node in a decision tree?



- Measure of impurity is entropy

$$I_m = -\sum_{i=1}^{K} p_m^i \log_2 p_m^i$$

28. How stacking works?



Base models are trained on subsets of the data.
Base models make predictions on a validation dataset.
Predictions of the base models serve as input features for a meta model.
The meta model is trained to learn how to combine the base models' predictions.
The trained meta model is used to make final predictions on new data.

29. How hash function is calculated in a locality sensitive?

- For each j = 1 2; . . .. L:

I. Retrieve the points from the bucket obtained by concatenating the k integers in the j-th hash table.

2. For each retrieved point,. compute the distance from q to it, and report the point if it is a correct answer (an R-near neighbor).

3. (optional) Stop as. soon as the number of reported points is more than L'.

30.How is a hash function compute for a data point q in the locality sensitive hashing approach?

hash(q) = p. q
hashp,b(q) = p(q+b)/w

- close points will have a large probability of falling into the same bucket:

Pr(hash (p) = hash(q))>=P1 for ||p-q||<=R1

- points p and q that are far apart have a low probability P2<P1 to fail into the same bucket:

Pr(hash(p) = hash (q)) <=P2 for ||p-q||<=R2,
where R2>R1

31.Naive Bayes rule , what is evidence, likelihood, posterior, prior?

Prior: Initial belief or probability assigned to a class before any evidence is observed. P(A)
Likelihood: Probability of observing the given data given a specific class. P(B|A)
Evidence : Probability of observing the given data irrespective of any specific class.P(B)
Posterior: Updated probability of a class given the observed data. P(A|B)

$$P(A|B)=P(A)*P(B|A) / P(B)$$

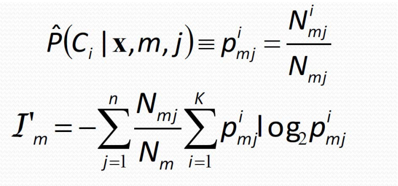
32. What is the difference of causal and diagonostic inference in graphical model?
Causal reasoning in graphical models aims to understand the causal relationships between variables and to determine how changes in one variable directly affect other variables.
Diagnostic reasoning in graphical models focuses on understanding the dependencies and correlations between variables and inferring the relationships between variables in the observed data
In summary, causal reasoning involves understanding causal relationships, while diagnostic reasoning focuses on understanding the associations and dependencies between variables.

33.When we want to do Part Of Speech (POS) tagging that is what is the mostly likely POS tage for each word, how can we describe the task with HMM(hidden state, observation, two probablity matrices, the vector of initial probabilities)
Hidden State: The hidden states represent the POS tags that we want to predict for each word in the given sentence. Each word corresponds to a specific hidden state.
Observation: The observations correspond to the words in the input sentence. These are the visible or observed variables in the HMM.
Two Probability Matrices: The HMM requires two probability matrices: the transition probability matrix and the emission probability matrix.
Vector of Initial Probabilities: This vector represents the initial probabilities of starting in each possible hidden state. Each entry in the vector provides the probability of starting with a specific POS tag.
By using these components, the HMM model can compute the most likely POS tag sequence for a given sentence by considering the transition probabilities, emission probabilities, and initial probabilities. The model utilizes the Viterbi algorithm to find the most probable sequence of hidden states (POS tags) that corresponds to the observed words.

34.How is the best split to be used determined in a decision tree?

·If node m is pure, generate a leaf and stop, otherwise split and continue recursively

·Impurity after split: N_{mj} of N_m take branch j. N^i_{mj} belong to Ci

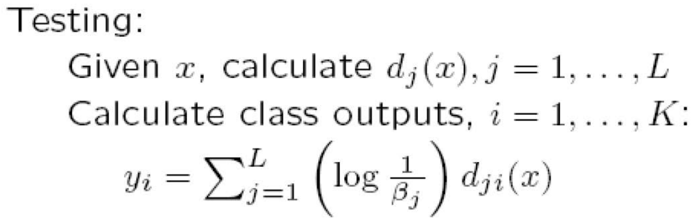


$$\hat{P}(C_i \mid \mathbf{x}, m, j) \equiv p^i_{mj} = \frac{N^i_{mj}}{N_{mj}}$$

$$\mathcal{I}'_m = -\sum_{j=1}^{n} \frac{N_{mj}}{N_m} \sum_{i=1}^{K} p^i_{mj} \log_2 p^i_{mj}$$

·Find the variable and split that min impurity (among all variables -- and split positions for numeric variables)

35.How are various models combined when predicting the label of a test instance in the case of Adaboost?



Testing:

Given x, calculate $d_j(x), j = 1, \ldots, L$

Calculate class outputs, $i = 1, \ldots, K$:

$$y_i = \sum_{j=1}^{L} \left(\log \frac{1}{\beta_j} \right) d_{ji}(x)$$

In AdaBoost, the weak learners are trained sequentially, and their predictions are combined using weighted voting. The final prediction is made by aggregating the predictions of the weak learners based on their weights.

36.What is the main idea behind the soft margin classifier? What is the relation between the value of the complexity constrained C and the model performance?

·The main idea behind the soft margin classifier is to relax the strict margin requirement of the linear SVM and allow for a certain degree of misclassification by introducing slack variables. This enables the classifier to handle data that is not perfectly separable and strike a balance between maximizing the margin and minimizing misclassifications.

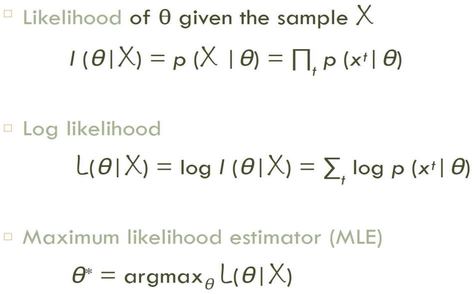
·

The relationship between the value of the complexity constraint (C') and model performance is a trade-off. A smaller C' can help prevent overfitting but may lead to underfitting, while a larger C' allows for more complex models but increases the risk of overfitting. The optimal value of C' depends on the problem, data, and complexity of patterns.

37.What is maximum likelihood estimation of parameters of a model? what is the maximum log likelihood?

Maximum likelihood estimation (MLE) is a method for estimating the parameters of a statistical model. The goal is to find the model parameter values that maximize the likelihood function

The maximum log-likelihood (log-likelihood) refers to the natural logarithm of the likelihood function, that is, the logarithm of the probability of observing the data given the parameters of the model. Including computational convenience and statistical properties, usually using the log-likelihood rather than the likelihood function itself



□ Likelihood of θ given the sample X

$l(\theta|X) = p(X|\theta) = \prod_t p(x^t|\theta)$

□ Log likelihood

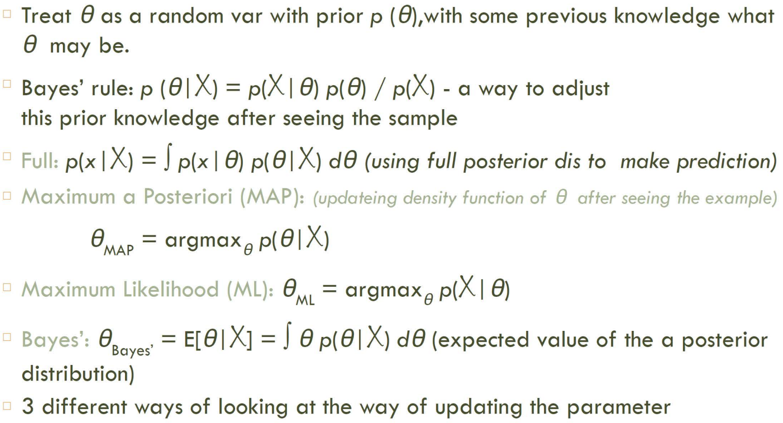
$L(\theta|X) = \log l(\theta|X) = \sum_t \log p(x^t|\theta)$

□ Maximum likelihood estimator (MLE)

$\theta^* = \text{argmax}_\theta L(\theta|X)$

38. What is Bayes estimator?

The Bayes estimator is a method in Bayesian statistics that uses prior knowledge and observed data to estimate unknown parameters. It minimizes the expected loss under the posterior distribution, incorporating both the prior information and the data.



□ Treat θ as a random var with prior $p(\theta)$,with some previous knowledge what θ may be.

□ Bayes' rule: $p(\theta|X) = p(X|\theta)\, p(\theta) / p(X)$ - a way to adjust this prior knowledge after seeing the sample

□ Full: $p(x|X) = \int p(x|\theta)\, p(\theta|X)\, d\theta$ *(using full posterior dis to make prediction)*

□ Maximum a Posteriori (MAP): *(updateing density function of θ after seeing the example)*

$\theta_{MAP} = \text{argmax}_\theta\, p(\theta|X)$

□ Maximum Likelihood (ML): $\theta_{ML} = \text{argmax}_\theta\, p(X|\theta)$

□ Bayes': $\theta_{Bayes'} = E[\theta|X] = \int \theta\, p(\theta|X)\, d\theta$ (expected value of the a posterior distribution)

□ 3 different ways of looking at the way of updating the parameter

38. What is the difference between bias and variance?

Bias refers to the error from incorrect assumptions in a model, while variance refers to the model's sensitivity to fluctuations in the training data.

Bias: bθ(d) = E [d] – θ

(how far is our expected estimator from the real value)

Variance: E [(d–E [d])2]

(how much are these estimations spread out)