Some of questions I got in ML last year's final exam, if related

1. What is the difference of causal and diagonostic inference in graphical model?

1. Causal Inference: Predicts effects given a cause. It follows the direction of the arrows in the graphical model. Example: If it's sunny, people may go outside.
2. Diagnostic Inference: Determines probable causes given an effect. It goes against the direction of the arrows in the graphical model. Example: If people are outside, it's likely sunny.

Essentially, causal inference goes from cause to effect, while diagnostic inference goes from effect to cause.

2.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 probability matrices, the vector of initial probabilities)

1. Hidden States: These represent the POS tags, such as Noun, Verb, Adjective, etc.
2. Observations: These are the words in the sentence. Each word corresponds to one observation in the sequence.
3. Transition Probability Matrix (A): This is a square matrix where each element Aij represents the probability of moving from state i (a POS tag) to state j (another POS tag). For instance, Aij could represent the probability of a verb following a noun in a sentence.
4. Emission Probability Matrix (B): This is a matrix where each element Bij represents the probability of an observation (word) being generated from a given state (POS tag). For instance, Bij could represent the probability of the word 'run' being a verb.
5. Initial Probability Vector (π): This represents the probabilities of the different states (POS tags) being the start state. For instance, it could represent the probability of a sentence starting with a noun, verb, etc.

3.Difference of prepruning and postpruning

techniques used to reduce the problem of overfitting.

- **Pre-pruning**: Stops growth of the decision tree early to prevent overfitting, based on set criteria like maximum tree depth or minimum samples per leaf. It's computationally efficient, but risks underfitting (being too simple).
- **Post-pruning**: involves building the decision tree fully first, then removes insignificant branches (overfit on the pruning set). It’s computationally intensive, it typically gives more accurate models, as it avoids underfitting.

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

Kernel estimation is a statistical technique used to estimate the probability density function of a random variable, typically through Kernel Density Estimation (KDE). It's useful when

- you don't know your data's distribution,
- want a flexible data representation,
- or are dealing with small, continuous data.

However, the choice of bandwidth and kernel type can significantly impact the results, requiring careful application.

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

Locality-Sensitive Hashing (LSH) helps find similar items in a high-dimensional space.

1. A set of hash functions is created to ensure similar data points map to the same hash bucket.
2. q is passed through each hash function to produce a hash value.
3. These hash values are combined to create a composite hash code for q.
4. q is placed into a hash bucket corresponding to its composite hash code.

When a query comes in, it's hashed in the same way. Similar data points are likely to be in the same hash bucket.

why we use Belief propagation in graphical model give examples

in case of a chain:

We are interested in the distribution of an internal node X, given the evidence at the root and the leaf -propagate evidence up and down the chain

In case of a tree:

each node receive evidence from its children nodes and its parents

we can answer queries about any node, given evidence anywhere in the tree

Naive Bayes rule, what is evidence likelihood posterior prior?

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

- **P(A|B)**: This is the posterior probability. It's the probability of event A given that event B has occurred. It's what we're trying to compute.
- **P(B|A)**: This is the likelihood. if A has occurred, what is the probability of event B.
- **P(A)**: This is the prior probability. It's our initial belief about event A before we have any other information.

- **P(B)**: This is the evidence or marginal likelihood. It's the total probability of event B happening.

Lasso regularization and ridge regularization?

These techniques are used to prevent overfitting by adding a penalty to the loss function.

Lasso Regression (L1 regularization) adding a penalty term equal to the sum of the absolute value of the coefficients multiplied by the regularization parameter, λ to the loss function. It can reduce some coefficients to zero, performing feature selection.

Ridge Regression (L2 regularization) adding a penalty term equal to the sum of square of the coefficients multiplied by the regularization parameter, λ to the loss function. It shrinks coefficients but doesn't reduce them to zero.

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

Impurity is usually measured using **Entropy.** it's the amount of information needed to decide whether a new instance should be classified as positive or negative.

How stacking works?

Stacking combines multiple different models (the base models) to improve prediction accuracy. It uses a meta-model to make predictions based on the outputs of the base models.

1. Splitting the training set into two subsets.
2. Training multiple base models (like linear regression, decision trees, etc.) on the first subset.
3. Making predictions on the second subset with these base models. These predictions are used as inputs for a second-level model, the meta-model.
4. Training the meta-model on these inputs.
5. Making a final prediction on a new instance by first using the base models, then the meta-model.

1.What is conditional independence in graphic model?

Conditional independence in a graphical model means that two variables become independent given the value of a third variable. If variables A and B are conditionally independent given C, it means once we know C, A and B become independent.

If you have three variables - A, B, and C - and A is conditionally independent of B given C, then the probability distribution of A given B and C is the same as the probability distribution of A given just C. Mathematically, this relationship can be written as:

$P(A \mid B, C) = P(A \mid C)$

What is unstable model and give an example?
An unstable model is one where small change in the input data can lead to significant changes in the output. Decision trees, for example, are unstable models because a slight change in the training data can result in a very different tree structure.

What is Bagging?
Bagging is an ensample method to solve the instability of a model by using bootstrapping to create many different versions of the original training dataset, and then training a separate model on each of these datasets. The final prediction is obtained by averaging the predictions (in case of regression) or voting (in case of classification) from all individual models.

2.Why the unstable model profit from bagging, what is unstable algorithm (model), give examples

Reduction of Overfitting By averaging predictions from multiple models trained on different subsets of the data, bagging effectively reduces the chance of overfitting

Error Reduction: The models built from the bootstrapped datasets are likely to make different errors when predicting the output for a new input. When their predictions are combined, some of these errors are likely to cancel each other out, resulting in a reduction in the overall error of the bagged model compared to any individual model.

Improvement in Accuracy: Since bagging uses the wisdom of the crowd, it often results in improved accuracy compared to a single model.

3.When we use LDA

4.How to build regression tree?

1. **Prepare Data**: Clean, normalize if needed, handle missing values.
2. **Choose Split**: Begin at root, divide data into two nodes. Pick the variable and split-point that the resulting child nodes are as pure as possible for regression trees, the measure of impurity is typically the residual sum of squares (RSS). So, the split that minimizes the RSS is chosen.

3. **Split Recursively**: For each child node, repeat Step 2. Keep splitting until a stopping condition is met, such as maximum tree depth or minimum node size.
4. **Prune Tree**: To avoid overfitting, prune the tree using a validation set and complexity parameter.

5.How the depth of tree size and the k in nearest neighbour affect bias and variance?

Tree Depth in Decision Trees:

- Deep trees can overfit, having low bias but high variance. it is very likely that the model will capture the noise in the data
- Shallow trees can underfit, having high bias but low variance. it may not capture important information

the goal is to find a balance in the tree depth that minimizes the total error with respect to bias and variance.

'K' in K-Nearest Neighbors:

- Small 'k' can overfit, leading to low bias and high variance. the prediction becomes more sensitive to noise in the data.
- Large 'k' can underfit, leading to high bias and low variance. The model might become too generalized

the goal is to find a balance in the number of neighbors that minimizes the total error with respect to bias and variance.

6.what is the difference between confidence and support?

1. **Support**: It measures how frequently an itemset appears in the dataset. if the itemset is {milk, bread}, then the support of this itemset is the proportion of all transactions in which milk and bread are bought together.
2. **Confidence**: It measures how often a rule (like item X leading to item Y) is true. It is the probability of seeing the Y given X

7.what is adaptive nearest neighbor method

The idea of the adaptive nearest neighbor method is to adjust the distance metric locally, so that the resulting neighborhoods stretch out in directions for which the class probabilities don't change much.

suppose we have data points that represent people, and the features are height and income. We're trying to classify people into two classes: "buys luxury goods" or "doesn't buy luxury goods". In this case, income might be a lot more important than height. A traditional nearest-neighbor method wouldn't take this into account and would consider big differences in height just as important as big

differences in income. This might lead to errors, like classifying a tall person with a low income as "buys luxury goods".

Adaptive nearest-neighbor methods try to solve this problem by adjusting how they measure "nearest" based on the local structure of the data. In regions where a particular feature (like height) doesn't matter much for the class (like "buys luxury goods" or not), they'll "stretch" the neighborhood along that feature. This is like saying "even if someone is a lot taller or shorter, they're still a 'near neighbor' as long as their income is similar".

So, in simple terms, adaptive nearest-neighbor methods are a smarter version of nearest-neighbor methods. They adjust their understanding of "nearest" based on what's important in the local area of the data point they're trying to classify.

9.why the chain of local probable state in HMM is not a good solution

- HMM assumes that the observed variables are conditionally independent given the hidden state. In other words, given the hidden state, the observation at a time point does not depend on the observations at other time points.
- it does not guarantee a globally optimal solution. Viterbi algorithm can find the most probable state sequence given the observations. However, this is a local optimum and does not guarantee the globally best state sequence.

10.what is soft margin classifier?

is a type of Support Vector Machine (SVM) that allows some examples to be misclassified to better generalize to unseen data.

A soft margin classifier introduces a parameter called "C", which controls the trade-off between the width of the margin and the misclassification error. If C is small, then the width of the margin is considered more important than classifying all points correctly. If C is large, correct classification of all points is considered more important.

The main idea behind the soft margin classifier is to find a good balance between keeping the width of the margin as large as possible and limiting the margin violations (i.e., instances that end up in the middle of the street or even on the wrong side). This way, the classifier can better generalize to unseen data.

Questions from today.

What are unstable models, how bagging useful with unstable model

What is HMM, its main concepts

It's a statistical model used in machine learning and data analysis. It's based on the theory of Markov processes, which are mathematical models for systems that transition between states over time.

1. **States**: An HMM has a finite number of states, each of which represents some condition or situation. For example, in an HMM used to model the weather, the states might be "sunny", "rainy", and "cloudy".
2. **Observations**: These are the actual data points you have. In the weather example, an observation might be the type of clothing a person is wearing.
3. **Transitions**: the system transitions from one state to another over time. Each possible transition has a probability, which is represented in a transition matrix.
4. **Emission probabilities**: These are the probabilities of observing each possible observation from each state. This is represented in an emission matrix.
5. **Initial state probabilities**: These are the probabilities in the start of the system before any transitions have occurred.

What are the three main problems usually considered in the context of HMMs?

1. **Evaluation problem**: Given the model parameters (transition and emission probabilities) and an observation sequence, compute the likelihood of the sequence.
2. **Finding the State Sequence problem**: Given the model parameters and an observation sequence, find the most likely sequence of states (the one that most probably would generate the observed data).
3. **Learning problem**: Given just an observation sequence, estimate the model parameters (transition and emission probabilities).

What is difference between primary and dual optimization problems in Kernel Machines

1. **Primal Problem:** In SVMs, the primal problem aims to find the best hyperplane that separates the classes with the largest margin in the feature space. It's often posed as a minimization problem with certain inequality constraints.
2. **Dual Problem:** The dual problem arises from the primal problem through a process called Lagrangian duality. It allows the use of kernel functions, enabling SVMs to handle high-dimensional and non-linearly separable data. The dual problem is often a maximization problem, and its solution also solves the primal problem.

What is Naive Bayes and how it is related to conditional independence

Naive Bayes is a machine learning algorithm based on Bayes' theorem for classification tasks. It works by predicting a class based on features of a given sample. The "naive" part of Naive Bayes

comes from its assumption that all features are independent of each other, given the class. This assumption simplifies the computation, making the algorithm fast and scalable, despite not always being accurate in real-world scenarios.

So, Naive Bayes is linked to conditional independence because it assumes that its features are conditionally independent given the class of a sample.

How do we get naive bayes classifier using graphical models and what is the use of conditional independence in the naive Bayes classifier

A graphical model represents probabilistic relationships between variables. In the case of Naive Bayes, this is represented by a directed graph (Bayesian Network). The graph has one root node (the class) and multiple leaf nodes (the features). Edges from the class to the features reflect the model's assumption that features are conditionally independent given the class.

What is Kernel Estimator and its advantages and drawbacks?

A kernel estimator is a statistical tool used to estimate patterns in data. Rather than fitting a specific type of curve to the data (like a straight line or a parabola), it places a "window" over each data point and averages the data inside to create a smoother trend line.

Advantages:

1. **Flexibility**: Can fit many different data shapes.
2. **Simplicity**: Easy to understand and use.
3. **Robustness**: Handles irregularities in data well.
4. **Non-parametric**: Doesn't assume a specific underlying distribution.

Drawbacks:

1. The choice of kernel and its size (bandwidth) can have a large impact on the results
2. can be computationally intensive for large data sets
3. they may not perform well with high-dimensional data.

What is a role of description length in IREP in Rule Learning

Description length is a fundamental concept in IREP and similar rule learning algorithms. It refers to the complexity of the rule set. The aim is to find a set of rules that correctly classifies the data with the smallest description length. By minimizing description length, IREP prevents overfitting

Therefore, description length in IREP is used to balance accuracy and simplicity, to prevent overfitting and to produce a rule set that is as simple as possible while still accurately classifying the data.

How can we measure the quality of the estimator?

1. **Bias**: The bias of an estimator is the difference between this estimator's expected value and the true value of the parameter being estimated. An estimator is said to be unbiased if its bias is zero. Less bias indicates a better quality of an estimator.

2. **Variance**: The variance of an estimator is a measure of how much the estimates vary around the expected value. A good estimator should have a small variance.

3. **Mean Square Error (MSE)**: MSE is the average of the squares of the errors—that is, the average squared difference between the estimated values and the actual value. It's one of the most widely used metrics for measuring the quality of an estimator because it combines both bias and variance (MSE = Bias^2 + Variance).