**1. What does one mean by the term "machine learning"?**

**Machine learning is a branch of artificial intelligence that enables algorithms to uncover hidden patterns within datasets, allowing them to make predictions on new, similar data without explicit programming for each task. Traditional machine learning combines data with statistical tools to predict outputs, yielding actionable insights. This technology finds applications in diverse fields such as image and speech recognition, natural language processing, recommendation systems, fraud detection, portfolio optimization, and automating tasks.**

**For instance, recommender systems use historical data to personalize suggestions. Netflix, for example, employs collaborative and content-based filtering to recommend movies and TV shows based on user viewing history, ratings, and genre preferences. Reinforcement learning further enhances these systems by enabling agents to make decisions based on environmental feedback, continually refining recommendations.**

**Machine learning’s impact extends to autonomous vehicles, drones, and robots, enhancing their adaptability in dynamic environments. This approach marks a breakthrough where machines learn from data examples to generate accurate outcomes, closely intertwined with data mining and data science.**

**2.Can you think of 4 distinct types of issues where it shines?**

**Machine learning algorithms have had good results on problems such has spam detection in email, cancer diagnosis, fraudulent credit card transactions, and automatically driving vehicles.**

**3.What is a labeled training set, and how does it work?**

**In machine learning, a labeled training set is a collection of input data samples, where each sample is associated with one or more output labels that serve as the ground truth for the problem at hand**

**Using a labeled training set is to enable a machine learning model to learn how to map inputs to outputs by learning from examples to make accurate predictions on new, unseen data.**

**To illustrate this concept with an example, let’s consider a simple classification problem of identifying whether an email is a spam or not spam. We can collect a large set of emails, and for each email, we can manually label it as either “spam” or “not spam”. This labeled set will serve as our training data.**

**Each email in our training set is an input sample, and its associated label (spam or not spam) is the output. In this case, the output label is a binary class (spam or not spam), but in other problems, it may be a continuous variable, such as the price of a house or the age of a person.**

**Here is a sample of what a labeled training set might look like:**

**Email Label  
-------------------------------------------------------  
Get rich quick! Buy now! spam  
Discounted prices on electronics spam  
Your invoice is overdue spam  
Meeting at 2pm on Friday not spam  
Reminder: dentist appointment tomorrow not spam**

**In this example, each email is an input sample, and its associated label (spam or not spam) is the output. We can use this training set to train a machine learning model to predict whether a new email is a spam or not spam, based on its content.**

**The process of using a labeled training set to train a machine-learning model typically involves the following steps:**

1. **Splitting the labeled data into a training set and a validation set: The training set is used to train the model, while the validation set is used to evaluate the model’s performance and tune its hyperparameters.**
2. **Preprocessing the data: This involves transforming the input data into a format that can be consumed by the machine learning model. This may involve tasks such as tokenization, normalization, and feature engineering.**
3. **Training the model: The machine learning model is trained on the training set by adjusting its parameters to minimize the difference between its predictions and the ground truth labels.**
4. **Evaluating the model: The model’s performance is evaluated on the validation set by computing metrics such as accuracy, precision, recall, and F1 score.**
5. **Tuning the hyperparameters: The model’s hyperparameters, such as the learning rate and regularization strength, are adjusted to improve its performance on the validation set.**
6. **Testing the model: Finally, the model’s performance is evaluated on a held-out test set to estimate its generalization performance on unseen data.**

**4.What are the two most important tasks that are supervised?**

**The two most common supervised learning tasks are regression and classification. In a regression problem we our prediciton is a scalar value. When we're trying to solve a classification problem, our output is either 1 or 0.**

**5.Can you think of four examples of unsupervised tasks?**

**Common unsupervised tasks include clustering, visualization, dimensionality reduction and association rule learning.**

**6.State the machine learning model that would be best to make a robot walk through various unfamiliar terrains?**

**The best Machine Learning algorithm to allow a robot to walk in unknown terrain is Reinforced Learning, where the robot can learn from response of the terrain to optimize itself.**

**7.Which algorithm will you use to divide your customers into different groups?**

**I would use some sort of clustering algorithm that can find the decision boundaries in the groups automatically. This is an unsupervised approach. However, if I already knew the categories of my customers, then I would choose a supervised approach and go with a classification algorithm.**

**8.Will you consider the problem of spam detection to be a supervised or unsupervised learning problem?**

**I would frame it as a supervised learning problem because humans have a general idea about what spam is and what it isn't. We can use this notion to create a labeled dataset for an algorithm to learn from.**

**9.What is the concept of an online learning system?**

**An online learning system learns from new data on-the-fly. As a result, the system is trained incrementally either by using one example at a time or using a mini-batch approach. This keeps each learning step cheap and memory efficient.**

**10.What is out-of-core learning, and how does it differ from core learning?**

**Out-of-core learning is used when a dataset is too large to fit into a computer's memory. The algorithm loads part of the data, runs a training step, then repeats the process until it has run on all the data.**

**11.What kind of learning algorithm makes predictions using a similarity measure?**

**Instance-based learning algorithms use a measure of similarity to generalize to new cases. In an instance-based learning system, the algorithm learns the examples by heart, then uses the similarity measure to generalize.**

**12.What's the difference between a model parameter and a hyperparameter in a learning algorithm?**

**A hyperparameter is a parameter of the learning algorithm, not the model. For example, in a simple linear regression problem our model is parameterized by theta which is a vector of weights. In order to find the best values for theta we have a cost function which is run repeatedly by the gradient descent algorithm. Gradient descent has a hyperparameter called alpha which is the learning rate of the algorithm.**

**13.What are the criteria that model-based learning algorithms look for? What is the most popular method they use to achieve success? What method do they use to make predictions?**

**The goal for a model-based algorithm is to be able to generalize to new examples. To do this, model based algorithms search for optimal values for the model's parameters, often called theta. This searching, or "learning", is what machine learning is all about. Model-based system learn by minimizing a cost function that measures how bad the system is at making predicitons on new data, plus a penalty for model complexity if the model is regularized. To make a prediction, a new instance's features are fed into a hypothesis function which uses the minimized theta found by repeatedly running the cost function.**

**14.Can you name four of the most important Machine Learning challenges?**

* **Not gathering enough data, or sampling noise. Sampling noise means we'll have non-representative data as a result of chance.**
* **Using a dataset that is not representative of the cases you want to generalize to. This is called sampling bias. For example, if you want to train an algorithm with "cat videos", and all your videos are from YouTube, you're actually training an algorithm to learn about "YouTube cat videos."**
* **Your dataset is full of missing values, outliers, and noise (poor measurments).**
* **The features in your dataset are irrelevant. Garbage in, garbage out.**
  + **Feature selection - choose the most relevant features from your dataset**
  + **Feature extraction - combine features in your dataset to generate a new, more useful feature**
* **When your model performs well on the training data, but not on test data, you've over fit your model. Models that suffer from overfitting do not generalize well to new examples. Overfitting happens when the model is too complex relative to the amount and noisiness of the data.**
  + **Try simplyfying the model by reducing the number of features in the data or constraining the parameters by reducing the degrees of freedom.**
  + **Gather more training data.**
  + **Reduce noise in the training data by fixing errors and removing outliers.**
* **When your model is too simple to learn the underlying structure of the data you've underfit your model.**
  + **Select a more powerful model with more parameters**
  + **Use feature engineering to feed better features to the model**
  + **Reduce the constraints of the model (increase degrees of freedom, reduce regularization parameter, etc.)**

**15.What happens if the model performs well on the training data but fails to generalize the results to new situations? Can you think of three different options?**

**This is a case where the model is overfitting the training data. To couteract overfitting, we can reduce the complexity of the model by removing features or constraining the parameters. We could gather more data. Finally we can reduce noisiness in the data by fixing errors and removing outliers.**

**16.What exactly is a test set, and why would you need one?**

**When we want to know how well our model generalizes to new cases we prefer to use a test set instead of actually deploying the system. To build the test set we split the training data (50-50, 60-40, 80-20 are common splits) into a training set and test set. Our model is training with the training set. Then we use the model to run predictions on the test set. Our error rate on the test set is called the generalization error or out-of-sample error. This error tells us how well our model performs on examples it has never seen before.**

**If the training error is low, but the generalization error is high, it means we're overfitting our model.**

**17.What is a validation set's purpose?**

**Let's say we have a linear model and we want to perform some hyperparameter tuning to reduce the generalization error. One way to do this 100 different models with 100 different hyperparameter values using the training set and finding the generalization error with the test set. You find the best hyperparameter value gives you 5% generalization error.**

**So you launch the model into production and find you're seeing 15% generalization error. This isn't going as expected. What happened?**

**The problem is that for each iteration of hyperparameter tuning, you measured the generalization error then updated the model using the same test set. In other words, your produced the best generalization error for the test set. The test set no longer represents cases the model hasn't seen before.**

**A common solution to this problem is to have a second holdout set called the validation set. You train multiple models with various hyperparameters using the training set, you select the model and hyperparameters that perform best on the validation set, and when you are happy about your model you run a single final test against the test set to get an estimate of the generalization error**

**18.What precisely is the train-dev kit, when will you need it, how do you put it to use?**

**Cross-validation is a tool to compare models without needing a separate validation set. It is preferred over validation set because we can save from breaking of part of the training set to create a validation set, as having more data is valuable regardless.**

**19.What could go wrong if you use the test set to tune hyperparameters?**

**If you tune hyperparameters using the test sets, then it may not perform well on the out-of-sample data because the model is tuned just for that specific set.**