1. How would you define Machine Learning?

Machine learning is the science of programming computers so they can learn from data. It is about building systems that can learn from data. Learning means getting better at some task, given some performance measure.

1. Can you name four types of problems where it shines?

Machine Learning is great for complex problems for which we have no algorithmic solution, to replace long lists of hand-tuned rules, to build systems that adapt to fluctuating environments, and finally to help humans learn (e.g., data mining).

1. What is a labeled training set?

A labeled training set is a training set that contains the desired solution (a label) for each instance.

1. What are the two most common supervised tasks?

Classification & Regression

1. Can you name four common unsupervised tasks?

Clustering, Visualization, Dimensionality Reduction, Association Rule Learning & Anomaly Detection

1. What type of Machine Learning algorithm would you use to allow a robot to walk in various unknown terrains?

Reinforcement Learning

1. What type of algorithm would you use to segment your customers into multiple groups?

For undefined groups, use clustering algorithm (unsupervised learning). For defined groups, use classification algorithm (supervised learning).

1. Would you frame the problem of spam detection as a supervised learning problem or an unsupervised learning problem?

Supervised learning problem

1. What is an online learning system?

An online learning system can learn incrementally, as opposed to a batch learning system. This makes it capable of adapting rapidly to both changing data and autonomous systems, and of training on very large quantities of data.

1. What is out-of-core learning?

Out-of-core algorithms can handle vast quantities of data that cannot fit in a computer’s main memory. An out-of-core learning algorithm chops the data into mini-batches and uses online learning techniques to learn from these mini-batches.

1. What type of learning algorithm relies on a similarity measure to make predictions?

An instance-based learning system.

1. What is the difference between a model parameter and a learning algorithm’s hyperparameter?

A model has one or more model parameters that determine what it will predict given a new instance (e.g., the slope of a linear model). A learning algorithm tries to find optimal values for these parameters such that the model generalizes well to new instances. A hyperparameter is a parameter of the learning algorithm itself, not of the model (e.g., the amount of regularization to apply).

1. What do model-based learning algorithms search for? What is the most common strategy they use to succeed? How do they make predictions?

Model-based learning algorithms search for an optimal value for the model parameters such that the model will generalize well to new instances. We usually train such systems by minimizing a cost function that measures how bad the system is at making predictions on the training data, plus a penalty for model complexity if the model is regularized. To make predictions, we feed the new instance’s features into the model’s prediction function, using the parameter values found by the learning algorithm.

1. Can you name four of the main challenges in Machine Learning?

Some of the main challenges in Machine Learning are the lack of data, poor data quality, non-representative data, uninformative features, excessively simple models that underfit the training data, and excessively complex models that overfit the data.

1. If your model performs great on the training data but generalizes poorly to new instances, what is happening? Can you name three possible solutions?

If a model performs great on the training data but generalizes poorly to new instances, the model is likely overfitting the training data (or we got extremely lucky on the training data). Possible solutions to overfitting are getting more data, simplifying the model (selecting a simpler algorithm, reducing the number of parameters or features used, or regularizing the model), or reducing the noise in the training data.

1. What is a test set and why would you want to use it?

A test set is used to estimate the generalization error that a model will make on new instances, before the model is launched in production.

1. What is the purpose of a validation set?

A validation set is used to compare models. It makes it possible to select the best model and tune the hyperparameters.

1. What can go wrong if you tune hyperparameters using the test set?

If you tune hyperparameters using the test set, you risk overfitting the test set, and the generalization error you measure will be optimistic (you may launch a model that performs worse than you expect)

1. What is cross-validation and why would you prefer it to a validation set?

Cross-validation is a technique that makes it possible to compare models (for model selection and hyperparameter tuning) without the need for a separate validation set. This saves precious training data.