Chapter 1 Problem Set

1. How would you define machine learning?

I would define machine learning as using algorithms trained on a large amount of data with some sort of features and some sort of target to make decisions or generate a model that can be then be used to make predictions of a target for novel instances given the features of the instances. An engineering-oriented definition (not mine) of machine learning is as follows:

“A computer program is said to learn from experience E with respect to task T and some performance measure P, if its performance on T, as measured by P, improves with experience E.

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

Machine learning is well-suited for tasks that require lots of fine-tuning or hard-coding in rules. Machine learning is also well-suited for complex problems for which there is no existing solution. Further uses include fluctuating environments where new data is constantly entering into the system, and gathering insights about large amounts of data that would be impossible for a single human to parse.

1. What is a labeled training set?

A labeled training set is a number of samples where each sample has features and a label assigned to it. The label can either be discrete (such as a category) or it can be continuous such as a value. Labeled training sets are used in supervised machine learning.

1. What are the two most common supervised tasks?

The two most common supervised tasks are classification, where the labels are discrete, and regression, where the labels are continuous and the model is predicting a value rather than one category. Features in a regression are known as predictors. The targets in classification are known as classes.

1. Can you name four common unsupervised tasks?

Clustering, with either a user-defined number of centroids, or the model selects the number of centroids based on a measurement function. Dimensionality reduction (principal component analysis) where a number of features are compressed into fewer features by selection new features with the greatest variation but also typically losing some amount of information. Visualization of data is another unsupervised task. A fourth common tasks is association based learning. This is where patterns such as how is the weather correlated with sales of a product, or how often do people buy a product after being exposed to advertising. The patterns may not be obvious from the outset but can be identified through machine learning.

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

I would use a reinforcement learning algorithm that provides both rewards, for positive behaviors, and penalties, for unwanted behavior. The agent, or learning system, develops a policy, or strategy based on the rewards and penalties it has experienced.

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

I would use a K-Means clustering algorithm and depending on the situation, I would either define the number of groups that I want, or I would let the algorithm decide the optimal number of groups based on the similarity measure between samples.

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

Spam detection is primarily a supervised learning problem because a set of labeled training samples is provided to the algorithm in the form of users classifying messages as spam in their email inbox.

1. What is an online learning system?

An online learning system is one in which new data can constantly be fed into the model. This is in contrast to a batch learning system that must train on all the data (old plus any new) every time it is trained. An online learning system can take the new data and update its model. It can have high inertia, where it takes many new instances to alter the model, or it can be highly adaptable in which it quickly discards the old data in favor of the new. A more adaptable model would be more susceptible to variations in the data, while a model with large inertia would be slow to adapt to new trends and might miss out on a novel pattern. Once an online learning algorithm has learned about new data, the data can be discarded because the model has already incorporated them into its weighting. This can save memory especially when dealing with streams of data.

1. What is out-of-core learning?

Out-of-core learning is a subset of online learning. It is when small batches of information are fed to the model rather than loading all of the data into memory. Segments of the entire dataset are processed individually to avoid overloading the system memory. This can only be done by online learning systems because the data is not fed in all at once, and each new segment of data must be taken into account.

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

An instance-based learning model relies on a similarity measure to make predictions. Given a new instance, the model will compare it to the samples it has already seen and identify which sample the new instance most closely matches. The K-Nearest neighbors algorithm is an example of instance-based learning. The system essentially memorizes a list of cases and then generalizes to the closet case it has seen previously (using some sort of similarity measure such as distance) when given a new instance.

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

A model parameter is a weight assigned to a feature. When the model makes a prediction, it takes the features and multiples each of them by the model parameter to determine the classification or value of the new instance. A hyperparameter on the other hand is a meta variable that is used to tune the algorithm. It is essentially a setting of the model that can be altered.

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

Model-based algorithms search for a trend in the data whether that be linear or higher order. They then assign weights (model parameters) to each of the features and then when given a new instance, multiple the features of the instances by the weights to arrive at a value or classification. The model uses a utility function (measures how good the model is) or a cost function (measures how off the model is) to find the trend. One example would be a linear regression that uses a cost function and seeks to minimize the sum of the squared errors from the trend line to the supplied data points.

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

Overfitting the data (high variance and low bias), underfitting the data (high bias and low variance), not having enough data or having bad data, and misleading training that suggest a trend that does not actually exist. This last point, nonrepresentative data, is called a sampling bias and there are many different kinds such as the nonresponse bias where the answer supplied by the model is influenced by the subset of the population from which information was drawn.

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?

The model is most likely overfit to the training data. This happens when the model becomes too complex and cannot adapt to novel instances. This can occur when the training data set is too small, or the model has too high variance. The model begins to pick up the noise in the data rather than the signal. Overfitting can be addressed by gathering more training data, reducing the noise in the training data (remove outliers and clean up errors), and reducing the complexity of the model (for example reducing the order of the polynomial in a regression). Constraining a model to make it less complex is known as regularization.

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

A test set is a subset of data that is not used for training a model but it used to assess the performance of the model. It is held out from the training set and then used after the training. The model is given the testing features and makes predictions, and these predictions are checked against the labels of the test set.

1. What is the purpose of a validation set?

The purpose of a validation set is to avoid training a model specifically for the test set. If the test set is used to tune the hyperparameters of the model by choosing parameters that minimize the generalization error, then the model may only perform well on the test set. The validation set is used to select the model and the hyperparameters after several models and configurations have been trained on the training set. The model that performs best on the validation set is then tested a single time on the testing set to determine the generalization error and if there is a need for further tuning.

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

The problem that arises when tuning hyperparameters on the test set is that you may create a model that is very accurate on the test set but cannot generalize to new instances. This is the purpose of a validation set. Rather than testing multiple models and parameter tunes on a test set, they are tested on the validation set with the best model than tested once against the test set. The generalization error from this single test will then be a better measure of the adaptability of the model to new data than if the model had already been run multiple times and tuned for the test set.

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

Cross-validation is splitting the training set into complementary subsets. Each model is then tested against a different combination of these subsets with some data held out to serve as a testing set. This avoids the problem of “wasting” training data by segmenting some as a separate validation set. Essentially, the training data is split multiple times and each model is trained and tested on a different subset. Then, the best performing model can be trained on the full training set and tested a single time against the test set which was segmented from the training data. Cross-validation is one way to avoid overfitting a model to a certain set of training data and to prevent tuning the hyperparameters of the model for a specific testing set.