CSCI-4/5587, Fall 2023, Machine Learning I
Study Guide for Test#1

(Based on Chapter 1)

(Please don't distribute this study guide.
The guide is provided for your study purpose only)

1. How would you briefly define Machine Learning in contrast to the traditional programming model?

1. Machine Learning (ML) is about building systems that can learn from data. Learning means getting better at some task, given some performance measures. Eventually, ML gives computers the ability to learn without being explicitly programmed.

Traditional programming models versus the ML models are compared using the figures below:

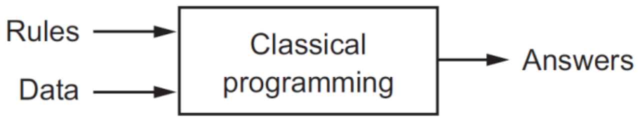

Fig. 1: Traditional programming Model.

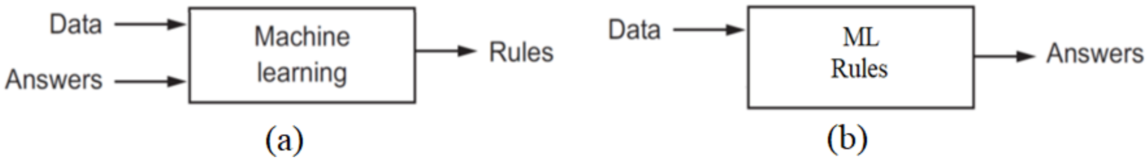

Fig. 2: Machine learning: a new and powerful programming paradigm.

2. Can you name four types of problems where Machine Learning shines?

2. 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).

3. What is a labeled training set?

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

4. What are the two most common supervised tasks?

4. The two most common supervised tasks are regression and classification.

5. Can you name four common unsupervised tasks?

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

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

6. Reinforcement Learning is likely to perform best if we want a robot to learn to walk in various unknown terrains since this is typically the type of problem that Reinforcement Learning tackles. It might be possible to express the problem as a supervised or semisupervised learning problem, but it would be less natural.

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

7. If you don't know how to define the groups, then you can use a clustering algorithm (unsupervised learning) to segment your customers into clusters of similar customers. However, if you know what groups you would like to have, then you can feed many examples of each group to a classification algorithm (supervised learning), and it will classify all your customers into these groups.

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

8. Spam detection is a typical supervised learning problem: the algorithm is fed many emails along with their labels (spam or not spam).

9. What is an online learning system?

9. 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.

10. What is out-of-core learning?

10. 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.

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

11. An instance-based learning system learns the training data by heart; then, when given a new instance, it uses a similarity measure to find the most similar learned instances and uses them to make predictions.

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

12. 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 learning rate in the gradient descent algorithm, the amount of regularization to apply).

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

13. 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.

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

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

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

15. 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.

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

16. 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.

17. What is the purpose of a validation set?

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

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

18. 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).

19. How do you save and load a machine learning model in (a) weka and (b) python? Describe using codes for both cases?

19. (a) Assume mlp is an instance of the Multi-layer Perceptron model. We can the trained model using weka's write method of the SerializationHelper as the following line:

```
import weka.core;
  ...
          weka.core.SerializationHelper.write("saved_mlp_model", mlp);
```

We can load the saved model using the read method of the SerializationHelper as the following line:

```
import weka.core;
  ...
      mlp = (MultilayerPerceptron)weka.core.SerializationHelper.read("saved_mlp_model");
      // Note that typecasting is needed here while reading the model.
```

(b) Python pickle module is used for serializing and de-serializing a Python object structure. You can also use joblib. joblib is optimized to be fast and robust on large data in particular. To write, use ‘joblib.dump’ and to read, use ‘joblib.load’.

To save a kNN model, instantiated as knn, using pickle, as an example, the following lines can be used:

```
import pickle
            ...
      pickle.dump(knn, open('iris_saved_knn_model','wb'))
```

To load the save kNN model, knn, using pickle, the following line can be used:

```
import pickle
      ...
loaded_model = pickle.load(open('iris_saved_knn_model', 'rb'))
#  Note: here typecasting is not needed and the variable type of ‘loaded_model’ is automatically and dynamically determined.
```

-- x --