1. How would you define Machine Learning?
   - I would like to define Machine Learning based on the Engineering perspective.
     "A computer program is said to learn from experience E with respect to some task T and some performance measure P, if its performance on T, as measured by P, improves with experience E."

2. Can you name four types of problems where it shines?
   - ○ Problems that have no well defined mathematical solution.
   - ○ Problems that needs long list of tuning variables.
   - ○ Problems that are accompanied by dynamic environment.
   - ○ Problems with high volume of data and sophisticated pattern

3. What is a labeled training set?
   - Part of the training set that has the desired solution for every features of the training set.

4. What are the two most common supervised tasks?
   - Classification(class predictor) and regression(value predictor)

5. Can you name four common unsupervised tasks?
   - ○ Clustering – example K-means, DBSCAN
   - ○ visualization -
   - ○ Dimensionality reduction
   - ○ association rule learning.
   - ○ Anomaly detection
   - ○ Novelty detection

6. What type of Machine Learning algorithm would you use to allow a robot to walk in various unknown terrains?
   - Reinforcement Learning. Let's frame the problem. We have an agent and we have an environment. The robot doesn't know a prior how to walk in different terrains. And of course it is hard to have a data set with labels that guide the robot walk on different terrains. So, the best course of action is to let the robot move on the different terrains and develop a policy for that rewards the agent if it does walk without falling or stumbling and punish unless otherwise.

7. What type of algorithm would you use to segment your customers into multiple groups?
   - I would use clustering algorithm. I'm assuming that I don't know a prior who is who and I just want to cluster based on the given data. Specifically I could use K-means unsupervised learning to accomplish the task.

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

- In this problem, I already know what constitutes a spam email and what constitutes a ham(not spam) email. So, I would have a list of emails that are spams and hams, which makes it a typical supervised learning.

9. What is an online learning system?
   - If the learning is happening incrementally, it is called online learning. Example, I would have trillions of data set and I would opt to use 10,000 data set for training at a time, which is called mini-batch. Or I might have a changing system and I would like my model to learn as new data set arrives, which make it online as well.

10. What is out-of-core learning?
    - When the memory of the target device(device running the training model) is not capable enough to hold the whole quantity of data, then naturally I have to cluster the data set into small blocks called mini-batch and train the model incrementally. Common on mobile devices or low-resource terminals.

11. What type of learning algorithm relies on a similarity measure to make predictions?
    - Instance-based learning. The reason is that, Instance based learning doesn't have a model of the whole observation, it just simply tries make a correlation between the existing data and new arriving data.

12. What is the difference between a model parameter and a learning algorithm's hyperparameter?
    - A model parameter is a parameter that captures the relationship between the instance(an observation with some features) and its label. Let's in case of linear regression Beta0 and Beta1 are parameters of the model because we can use them to predicate when a new instance is available. On the other hand learning algorithm tries to find the optimal values for the model parameters from the given amount of data set. We can have parameters of the learning algorithm and they are called hyper-parameters.

13. What do model-based learning algorithms search for? What is the most common strategy they use to succeed? How do they make predictions?
    - In model-based learning, the algorithm tries to search for an optimal parameter values that could potentially generalize well to new unseen observations. The goal is not to make a good accuracy on the test data-set, rather beyond that, to make its generalization error as low as possible. The most common strategy is to define a function to optimize. It could be a cost function(As mostly use) or a utility function(or fit function). In cost function, it tries to measure how bad the system is at making predictions and we would like to minimize it. Whereas fit function, it tries to measures how good the system is at making predictions, so we would like to maximize it. After selecting an optimizer(mostly cost function), we would train it on the training data-set and validate it on the test set. If it does work well on the test set, we might feel confident that it could generalize well.

14. Can you name four of the main challenges in Machine Learning?
    - lack of data
    - poor quality data
    - irrelevant features
    - over-fitting the training data
    - Biased 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?
    - This is a typical overfitting the training data problem. My model is capturing every possible pattern of the training data-set, but it is not working well when predicting a new instance. To solve this problem, I can:
        - Get more data so that I would hope that it could see a more generalized patterns
        - Simplify the model: If I used polynomial regressor, I might try a Linear regressor
        - Regularization

16. What is a test set and why would you want to use it?
    - A model that achieves 100% on training data-set doesn’t mean that it is an amazing. It could be the worst, when predicting a new unseen instance. That is why we have a test data-set which tells us how the model trained on the training data-set could generalize on new unseen observations.

17. What is the purpose of a validation set?
    - If we have multiple models to compare with for our system, then it would be in appropriate to tune their hyper-parameter so that they would achieve great on the test data-set. This is a fatal mistake because we are trying the learn them on the test data-set. So, we need to introduce another data-set for comparing, which is called validation set. The modal which is good on the validation set would be finally tested on the test data-set to make sure that it generalize on new unseen instances(observations).

18. What can go wrong if you tune hyperparameters using the test set?
    - Well, you might brag you’ve got 100% accuracy on test data set and launch the model and finally you would get 50% accuracy when tried on new instances. So, the risk is over-fitting the test set.

19. What is repeated cross-validation and why would you prefer it to using a single validation set?
    - Let’s frame a problem where we have a large training set and small validation set. In this case I might mistakenly choose a bad model which for some reason performed well on the validation set. So this is not good.
    - Let’s frame another problem where we have small training set and large validation set. In this case, we don’t have enough information to judge which is good since later the best model on the validation set will be trained on the whole training set(including the validation set), so who knows the best model could be the worst now.

- So, How to solve this problem? We can have multiple tiny validation sets and try to compare based on the average of the accuracy on each validation sets. This is fair.

Interview Questions

ANS 1:

Let me cover first how missing data is handled in Machine learning applications. There are three common practices for handling missing values, those are:

1. get rid of the corresponding instances
2. get rid of the whole attribute
3. set missing value to some value(zero, mean or median). This process is called imputation.

And I have learned two important hints from the problem

A. 15% of my data has missing age value
B. Age is very important for my analysis.

What is requirement A telling me is that the missing value is huge. 15% of 100,000 which is 15,000 instances has missing value. So if I opt out to delete all those 15,000 instances I would risk having less training data. So I will not choose to delete. And since Age is important, I don’t want to delete the whole age attribute. So, I will be having the third choice which is imputation. The common practice in imputation is to set the missing value to mean.

So, my solution is to set the missing age data to the mean of the age attribute.

ANS 2:
There are methods used for detecting outliers in data analysis. Those are

- Z-score
- Inter-quartile range

For this case, I could set that if the data is above or minus mean-2*standard_deviation, I could flagged it as a outlier. That means, If the data is out of the 97% of my data, them immediately flagged as outlier and remove it.

The potential impact of removing those outlier is that the model will not be sensitive to extremes at the consequence of not adapting fast enough.

ANS 3:

We have two features

- property size in square feet and
- local crime rate per 1,000 residents

Now let’s look a typical values for this features

property_size	crime_rate
100	1
10,000	2
20	0

For such data, the machine learning will be influenced much by the property_size than by the crime_rate. So I need to make them so that both have a meaningful impact on the prediction. Since we might have varied property_size ranging from 20 to 10,000, I would opt for standardization since I can have a fairly bell curved function regardless of the extremes. But if I used normalization, I will have distorted distribution even if I squeezed into [0, 1] range.

***** Read Lesson 2 and summarize it.**

The topic of lesson 2’s was about regression. The following are the key points on regression

- How regression analysis works
- Types of regression
  - Simple and multiple
- Normal equation
- Linear, polynomial and non-linear regressors
- Evaluating Linear regression problems
  - Mean squared Error
  - Root Mean Squared Error
- Over-fitting and under-fitting problems
  - Regularization
  - Feature engineering