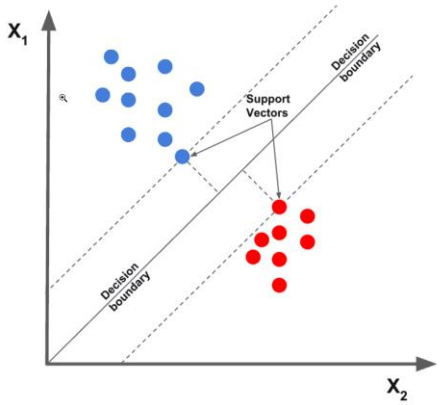
Q1: Please briefly and precisely answer the following questions (a short paragraph or a couple of sentences shall do):

1. What are the two most common supervised tasks?
2. Can you name four common unsupervised tasks?
3. Which Linear Regression training algorithm can you use if you have a training set with millions of features?
4. Suppose the features in your training set have very different scales. Which algorithms might suffer from this, and how? What can you do about it?
5. Do all Gradient Descent algorithms lead to the same model, provided you let them run long enough?
6. Suppose you use Batch Gradient Descent and you plot the validation error at every epoch. If you notice that the validation error consistently goes up, what is likely going on? How can you fix this?
7. Suppose you are using Ridge Regression and you notice that the training error and the validation error are almost equal and fairly high. Would you say that the model suffers from high bias or high variance?
8. What is a support vector?
9. Can an SVM classifier output a confidence score when it classifies an instance? What about a probability?
10. Say you've trained an SVM classifier with an RBF kernel, but it seems to underfit the training set. Should you increase or decrease γ (gamma)? What about C?

Q2: Computation (summarize your results)

Load the MNIST data (introduced in Chapter 3), and split it into a training set, a validation set, and a test set (e.g., use 50,000 instances for training, 10,000 for validation, and 10,000 for testing). Then train the following classifiers, (1) a logistics regression classifier and (2) an SVM classifier. Please compare the performance (computation time, accuracy, balance between variance and bias, and AUC) of these two classifiers on this dataset.

1. Most two common supervise tasks are regression and classification. The classification is used when we are predicting 0 or 1 and regression is used when the value is in the scalar.
2. Four common unsupervised tasks are
    - clustering,
    - visualization,
    - Anomaly detection
    - association rule,
    - dimensionality reduction.
3. I would use Stochastic gradient descent, mini-batch gradient descent, or gradient descent.
4. I would say the normal equations method because it does not need to normalize the features, therefore it remains unaffected by features in the training set having very different scales. Feature scaling will help gradient descent converge quicker and it is required for the various gradient descent algorithms.
5. No, they don't do. This is due to the fact that it sometimes hits a local minimum or local optima point. The model may diverge if the learning rate is too high. Additionally, depending on where the initialization is, it can only reach the local minimum. The problem is that mini-batch gradient descent and stochastic gradient descent both have randomness built into them. This indicates that they can converge close to the global optimum, but they rarely do.
6. If the validation error consistently goes up, that means the model could be diverging because of the high learning rate. If the training error also goes up, that is an indication of diverging. You can fix that by lowering the learning rate and then re-training. If the training error is not increasing, then your model is overfitting and you have to retrain with a different model.
7. The model suffers from high bias, because the errors are both high, indicating wrong assumptions and therefore underfitting. In order to reduce high bias, you have to decrease alpha.
8. Support vector machines in terms of machine learning are supervised machine learning models with associated learning algorithms that analyze data for regression and classification analysis. Support vector machines can solve leaner and non-leaner problems. The algorithm creates a line or a hyperplane which separates data into classes.



9. An SVM classifier can output the distance between the decision boundary, and the test instance, and you can use this as a confidence score. However, this score cannot be directly converted into an estimation of the class probability. While building an SVM in Scikit-Learn, if you set the probability parameter to True, it will use the SVM's scores to calibrate the probabilities after training (trained by an additional five-fold cross-validation on the training data). This will add the predict_proba() and predict_log_proba() methods to the SVM.
10. The underfit data set might be because of too much regularization. To reduce underfit need to increase Gamma or C or Both.

References:

1) Support Vector Machine - Introduction to Machine Learning Algorithms

https://bit.ly/2zgoEVq