**MACHINE LEARNING**

1. Ans:- The computational complexity of linear regression is 0 (n).

2. Ans:- Logistic Regression

3. Ans:- Gradient Descent

4. Ans:- Lasso

5. Ans:- All of the above.

6. Ans:- False.

7. Ans:- it does not matter whether half is there or not

8. Ans:- Correlation

9. Ans:- Following statements are true regarding Normal Equation used to compute linear regression:
   a) We don't have to choose the learning rate.
   b) It becomes slow when number of features are very large.

10. Ans:- Following statements are true:
   a) Linear Regression will have high bias and low variance.
   b) Polynomial with degree 5 will have low bias and high variance.

11. Ans:- It discovers causal relationship

12. Ans:- We can use batch gradient descent, stochastic gradient descent (SGD), or mini-batch gradient descent (MBGD). SGD and MBGD would work the best because neither of them need to load the entire data-set into memory in order to take 1 step of gradient descent. Batch would be ok with the caveat that you have enough memory to load all the data.
The normal equations method would not be a good choice because it is computationally inefficient. The main cause of the computational complexity comes from inverse operation on an (n x n) matrix.

13. Ans:- The normal equations method does not require normalizing the features, so it remains unaffected by features in the training set having very different scales. Feature scaling is required for the various gradient descent algorithms. Feature scaling will help gradient descent converge quicker.