

MACHINE LEARNING

1. $O(n^3)$ – (Option D)
2. Polynomial Regression – (Option C)
3. Gradient Descent – (Option B)
4. Lasso – (Option C)
5. Stochastic Gradient Descent – (Option A)
6. True – (Option A)
7. presence of half makes it easy to do grid search – (Option B)
8. Correlation – (Option B)
9. Both Option A and B
10. Linear Regression will have high bias and low variance – (Option A)
11. It discovers causal relationship – (Option C)

12. We could use batch gradient descent, stochastic gradient descent, or mini-batch gradient descent. SGD and MBGD would work the best because neither of them need to load the entire dataset 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 $(n \times n)$ matrix.

$O(n^2)$ to $O(n^3)$

13. 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.