1. What is the fundamental idea behind Support Vector Machines?

The basic idea underlying Support Vector Machines is to squeeze as much "street" between the classes as feasible. To put it another way, the aim is to have the widest feasible gap between the decision border separating the two classes and the training examples.

2. What is a support vector?

Any instance on the "street," including the "street's" boundary. The support vectors completely determine the decision boundary.

3. Why is it important to scale the inputs when using SVMs?

If the training set is not scaled, the SVM tends to overlook minor characteristics, thus try to fit the widest feasible "street" between the classes.

4. Can an SVM classifier output a confidence score when it classifies an instance? What about a probability?

It can generate a confidence score based on the distance between the test instance and the decision boundary. This score cannot be translated into a probability directly.

5. Should you use the primal or the dual form of the SVM problem to train a model on a training set with millions of instances and hundreds of features?

If there are millions of occurrences, the primal form should be used.

Only linear SVMs are addressed in this question. The primal form of the SVM problem has a computational complexity proportional to the number of training examples m, but the dual form has a computational complexity proportional to a number between m^2 and m^3.

6. 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?

It's conceivable that the regularization is excessive. Increase gamma, C, or both to lower it.

7. How should you set the QP parameters (H, f, A, and b) to solve the soft margin linear SVM classifier problem using an off-the-shelf QP solver?

H′, f′, A′, and b′ are the QP parameters for the hard-margin issue. The soft-margin issue has m more parameters (np = n + 1 + m) and m additional constraints (nc = 2m) in the QP parameters. They can be described as follows:

- H is equal to H′, plus m columns of 0s on the right and m rows of 0s at the bottom
- f is equal to f′ with m additional elements, all equal to the value of the hyperparameter C.
- b is equal to b′ with m additional elements, all equal to 0.
- A is equal to A′, with an extra m × m identity matrix Im appended to the right,

8. Train a LinearSVC on a linearly separable dataset. Then train an SVC and a SGDClassifier on the same dataset. See if you can get them to produce roughly the same model.

Coding is done on Jupyter Notebook

9. Train an SVM classifier on the MNIST dataset. Since SVM classifiers are binary classifiers, you will need to use one-versus-the-rest to classify all 10 digits. You may want to tune the hyperparameters using small validation sets to speed up the process. What accuracy can you reach?

Coding is done on Jupyter Notebook

10. Train an SVM regressor on the California housing dataset.

Coding is done on Jupyter Notebook