**1. What is the underlying concept of Support Vector Machines?**

**Ans:** SVM is a powerful supervised learning algorithm used for classification or regression tasks. The key concept is to find the optimal hyperplane that separates different classes while maximizing the margin between the classes. It aims to find the best decision boundary that can classify data points.

**2. What is the concept of a support vector?**

**Ans:** Support vectors are the data points that lie closest to the decision boundary, also known as the margin. These points play a crucial role in defining the decision boundary and maximizing the margin in SVM.

**3. When using SVMs, why is it necessary to scale the inputs?**

**Ans:** Scaling the inputs in SVMs is necessary to ensure that all features contribute equally to the distance computation. As SVM relies on distance calculations, features with larger scales can dominate the distance measures, leading to biased results.

**4. When an SVM classifier classifies a case, can it output a confidence score? What about a percentage chance?**

**Ans:** SVM classifiers can output a confidence score, which is the distance from the test instance to the decision boundary. However, they do not directly provide a probability or percentage chance like some other classifiers. Techniques such as Platt scaling can be applied to convert the distance into a probability estimation.

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

**Ans:** When dealing with large datasets, the dual form of the SVM problem is generally preferred as it can handle a large number of training instances more efficiently than the primal form. The dual form allows the use of the kernel trick, making it suitable for handling complex data and non-linear decision boundaries.

**6. Let's say you've used an RBF kernel to train an SVM classifier, but it appears to underfit the training collection. Is it better to raise or lower (gamma)? What about the letter C?**

**Ans:** If an SVM classifier with an RBF kernel underfits the training data, increasing the gamma parameter can make the model more complex, potentially leading to better fitting. Similarly, increasing the C parameter can also help in reducing underfitting by allowing more violations of the margin.

**7. To solve the soft margin linear SVM classifier problem with an off-the-shelf QP solver, how should the QP parameters (H, f, A, and b) be set?**

**Ans:** For the soft margin linear SVM classifier problem, the QP parameters should be set as follows:

H: Matrix that penalizes deviations from the margin.

f: Vector that contains the cost function coefficients.

A: Matrix that defines the constraints on the decision function.

b: Vector that holds the constraints for the decision function.

**8. On a linearly separable dataset, train a LinearSVC. Then, using the same dataset, train an SVC and an SGDClassifier. See if you can get them to make a model that is similar to yours.**

**Ans:** LinearSVC, SVC, and SGDClassifier are all linear models, but they may use different optimization techniques. By using the same dataset, we can compare the models and find the one that best fits the data.

**9. On the MNIST dataset, train an SVM classifier. You'll need to use one-versus-the-rest to assign all 10 digits because SVM classifiers are binary classifiers. To accelerate up the process, you might want to tune the hyperparameters using small validation sets. What level of precision can you achieve?**

**Ans:** An SVM classifier can be trained on the MNIST dataset using the one-versus-the-rest strategy for multi-class classification. By tuning the hyperparameters using small validation sets, a high level of precision can be achieved, often comparable to other advanced models.

**10. On the California housing dataset, train an SVM regressor.**

**Ans:** An SVM regressor can be trained on the California housing dataset to predict housing prices based on various features. This can help in understanding the relationship between the input features and the target variable, allowing for accurate price predictions.