Kernel and Ensemble Methods

How SVM works

Support vector machines attempt to find hyperplanes to separate different classes in the dataset [1]. (Alternatively, the planes can be used to model the dataset similarly to linear regression and SVM can be used for regression [2], but this seems to be less common). The SVM has two important parameters: cost, which sets a tolerance for how much data is allowed to appear on incorrect sides of the separating planes (inversely, so higher cost decreases tolerance), and gamma, which specifies how exactly the kernel should try to fit the training data set (large values may lead to overfitting) [1]. A unique feature of SVM is that the separating hyperplanes don't necessarily need to be actual lines/planes, but the chosen kernel defines what shape will be fit to the data. Linear, polynomial, and radial are common kernel types.

SVM strengths and weaknesses

Adjustable kernel types and hyperparameters are a strength of support vector machines; there is significant room to optimize the model for a particular dataset and task. Nevertheless, this strength is in some situations a weakness, as the model requires an inconveniently long amount of training time on large datasets, and the training time is multiplied for each combination of parameters you choose to test. Similarly to kNN models, SVM is scalable to highly dimensional datasets and does not depend on the structure of the data [1].

How Random Forest works

The Random Forest algorithm builds many diverse decision trees from a training dataset. When the model is used for prediction, each data point is evaluated by all of the models, and the models "vote" on which classification is most likely correct [3]. In order to ensure variation between each decision tree, the training dataset and selected features are randomized for each individual tree. The concept behind the Random Forest algorithm is simple but surprisingly effective.

Random Forest strengths and weaknesses

Intuitively, the primary weakness with the Random Forest algorithm is the necessity of building and predicting on multiple decision trees. This multiplies the training and prediction times by a factor of the number of trees used in the model. Additionally, Random Forest models are impractical for regression, as they are prone to overfitting and bound to the range of the training data [4]. In many situations, however, these weaknesses are offset by the increasingly high accuracy brought by combining multiple uncorrelated models [3]; the weaknesses of one tree do not necessarily affect the other trees.

References

[1] https://www.analyticsvidhya.com/blog/2017/09/understaing-support-vector-machine-example-code/

[2] https://towardsdatascience.com/an-introduction-to-support-vector-regression-svr-a3ebc1672c2

[3] https://towardsdatascience.com/understanding-random-forest-58381e0602d2

[4] https://blog.citizennet.com/blog/2012/11/10/random-forests-ensembles-and-performance-metrics https://techvidvan.com/tutorials/svm-kernel-functions/