How does SVM works

SVM works by mapping data to a high-dimensional feature space so that data points can be categorized, even when the data are not otherwise linearly separable. SVM helps in identifying the hyperplane that best separates the input space according to the class labels. Performance of a SVM model often depends on the choice of the kernel selection which helps in separating the data both linearly as well as non-linear.

SVM works by mapping data to a high-dimensional feature space in order to categorize data points that are otherwise not linearly separable. SVM assists in determining the hyperplane that optimally divides the input space based on the class labels. The performance of an SVM model is frequently determined by the kernel selection, which aids in the separation of data both linearly and non-linearly.

How SVM kernels work

The kernel's function is to receive data as input and transform it into the desired form. Different types of kernel functions are used by different SVM algorithms. These functions might be of several forms. The kernel functions compute the inner product of two points in a given feature space. Thus, by establishing a notion of similarity, even in extremely high-dimensional regions, with low computing expense.

Strengths and weaknesses of SVM.

Advantages:

SVM is effective when the number of dimensions exceeds the number of samples.
SVM uses a little amount of memory.

Disadvantages:

The SVM method is not appropriate for huge data sets.
When the data set has more noise, i.e. target classes overlap, SVM does not perform well.
There is no probabilistic justification for the classification because the support vector classifier operates by placing data points above and below the classifying hyperplane.

How Random Forest works

Random forests, also known as random choice forests, are an ensemble learning approach for classification, regression, and other problems that works by generating a large number of decision trees during training. For classification problems, the random forest output is the class chosen by the majority of trees. The mean or average forecast of the individual trees is returned for regression tasks. Random decision forests compensate for decision trees' tendency to overfit to their training set. Random forests outperform decision trees in general, although their accuracy is lower than that of gradient enhanced trees. However, data qualities can have an impact on performance.

How the other 2 algorithms you used work compared to the simple decision tree, your impression of the strengths and weaknesses of these ensemble techniques.

To conclude, the strategies employed in this notebook are two ensemble techniques for strengthening decision tree-based models. In order to achieve more accuracy, Random Forest builds several trees, each with equal weighted leaves within the model. XGboost, on the other hand, employs leaf weighting to punish individuals who fail to enhance model predictability. Both decision tree approaches reduce variance in general, and boosting also decreases bias.