**Analysis**

**Best?**

As per my observation Random Forest algorithm performs better than other 4 which I have used. What I have observed in practice is that Random Decision Forest is very effective in eliminating noise in the model input data.  Given a long list of input variables and a potentially sparse dataset, it is very likely that any predictive model will discover spurious relationships between those inputs and the chosen target variable.  This results in over- fitting and the model does not generalize well enough to future input it has not seen.

Because Random Forest builds many trees using a **subset** of the available input variables and their values, it inherently contains some underlying decision trees that omit the noise generating variable/feature(s).  In the end, when it is time to generate a prediction a vote among all the underlying trees takes place and the majority prediction value wins.

**Worst?**

And as per my observation KNN perform worst than the other classifier, KNN tends to perform well when you have many instances (points) and few dimensions but you have to be very careful about performance because a brute-force version of KNN can be very slow when you have a lot of data. So KNN is good when you have many points but at the same time it becomes too slow a catch-22 scenario.

But again every algorithm requires perform well based on the data we provide. So the important fact is how we organize our data. Recall, though, that better data often beats better algorithms, and designing good features goes a long way. And if you have a huge dataset, your choice of classification algorithm might not really matter so much in terms of classification performance (so choose your algorithm based on speed or ease of use instead).