1. Classification vs Regression

Your goal is to identify students who might need early intervention - which type of supervised machine learning problem is this, classification or regression? Why?

Answer: This is a classification problem as we have to predict a binary value (yes or no), whether the student will pass or not.

2. Exploring the data:

Total number of students: 395

Number of students who passed: 265

Number of students who failed: 130

Number of features: 30

Graduation rate of the class: 67.09%

4. Training and Evaluating Models:

Models chosen:

- Support Vector Machine
- Gaussian Naïve Bayes
- Random Forest Classifier

I chose these models because we know that this is a classification problem. Gaussian Naïve Bayes is simple, fast and easy to implement and performs well on small data. Support Vector Machine is also robust and efficient for classification problems. And I chose Random Forest Classifier to compare with other two algorithms as generally its accuracy is high, it is very slow and it has the ability to learn the feature interaction (But it performs very bad compared to other models for this given example). And finally I referred to the diagram here http://scikit-learn.org/stable/tutorial/machine_learning_map/ to choose my models.

Support Vector Machine:

The Support Vector Machine (SVM) classifier is a powerful classifier that works well on a wide range of classification problems, even problems in high dimensions and that are not linearly separable. [1]

Advantages:

- Effective in high dimensional spaces.
- Still effective in cases where number of dimensions is greater than the number of samples.
- Uses a subset of training points in the decision function (called support vectors), so it is also memory efficient.

- Versatile: different Kernel functions can be specified for the decision function. Common kernels are provided, but it is also possible to specify custom kernels. [2]

Disadvantages:

- If the number of features is much greater than the number of samples, the method is likely to give poor performances.
- SVMs do not directly provide probability estimates, these are calculated using an expensive five-fold cross-validation (see Scores and probabilities, below). [2]

Applications:

- Text (and hypertext) categorization
- Image classification
- Bioinformatics (Protein classification, Cancer classification)
- Hand-written character recognition. [3]

Training Set Size	100	200	300
Training Time(s)	0.001	0.004	0.006
Prediction Time for Training Set(s)	0.001	0.002	0.006
Predictions Time for Test Set(s)	0.000	0.001	0.002
F1 score for training set	0.8519	0.8664	0.8515
F1 score for test set	0.8466	0.8366	0.8421

Gaussian Naïve Bayes:

It is a classification technique based on Bayes' Theorem with an assumption of independence among predictors. In simple terms, a Naive Bayes classifier assumes that the presence of a particular feature in a class is unrelated to the presence of any other feature. For example, a fruit may be considered to be an apple if it is red, round, and about 3 inches in diameter. Even if these features depend on each other or upon the existence of the other features, all of these properties independently contribute to the probability that this fruit is an apple and that is why it is known as 'Naive'.

Advantages:

- It is easy and fast to predict class of test data set. It also performs well in multi class prediction
- When assumption of independence holds, a Naive Bayes classifier performs better compare to other models like logistic regression and you need less training data.
- It performs well in case of categorical input variables compared to numerical variable(s). For numerical variable, normal distribution is assumed (bell curve, which is a strong assumption). [4]

Disadvantages:

- If categorical variable has a category (in test data set), which was not observed in training data set, then model will assign a 0 (zero) probability and will be unable to make a prediction. This is often known as "Zero Frequency". To solve this, we can use the smoothing technique. One of the simplest smoothing techniques is called Laplace estimation.
- On the other side naive Bayes is also known as a bad estimator, so the probability outputs from *predict_proba* are not to be taken too seriously.
- Another limitation of Naive Bayes is the assumption of independent predictors. In real life, it is almost impossible that we get a set of predictors which are completely independent. [4]

Applications:

- **Real time Prediction:** Naive Bayes is an eager learning classifier and it is sure fast. Thus, it could be used for making predictions in real time.
- **Multi class Prediction:** This algorithm is also well known for multi class prediction feature. Here we can predict the probability of multiple classes of target variable.
- **Text classification/ Spam Filtering/ Sentiment Analysis:** Naive Bayes classifiers mostly used in text classification (due to better result in multi class problems and independence rule) have higher success rate as compared to other algorithms. As a result, it is widely used in Spam filtering (identify spam e-mail) and Sentiment Analysis (in social media analysis, to identify positive and negative customer sentiments)
- **Recommendation System:** Naive Bayes Classifier and Collaborative Filtering together builds a Recommendation System that uses machine learning and data mining techniques to filter unseen information and predict whether a user would like a given resource or not. [4]

Training Set Size	100	200	300
Training Time(s)	0.000	0.001	0.001
Prediction Time for Training Set(s)	0.000	0.001	0.000
Predictions Time for Test Set(s)	0.001	0.000	0.000
F1 score for training set	0.8391	0.8014	0.7821
F1 score for test set	0.7633	0.7633	0.8148

Random Forest Classifier:

Random Forests are an ensemble learning method that operate by building a number of decision trees at training time and outputting the class with the majority vote over all the trees in the ensemble. [5]

Advantages: The Random Forests algorithm is a good algorithm to use for complex classification tasks. The main advantage of a Random Forests is that the model created can easily be interrupted. [5]

Disadvantages: The main limitation of the Random Forests algorithm is that a large number of trees may make the algorithm slow for real-time prediction. [5]

Random Forest can feel like a black box approach for statistical modelers – you have very little control on what the model does. You can at best – try different parameters and random seeds. [6]

Applications: Applications ranging from marketing to healthcare and insurance. [7]

Training Set Size	100	200	300
Training Time(s)	0.016	0.017	0.018
Prediction Time for Training Set(s)	0.001	0.002	0.002
Predictions Time for Test Set(s)	0.000	0.001	0.001
F1 score for training set	0.9781	0.9924	0.9923
F1 score for test set	0.7230	0.75	0.7883

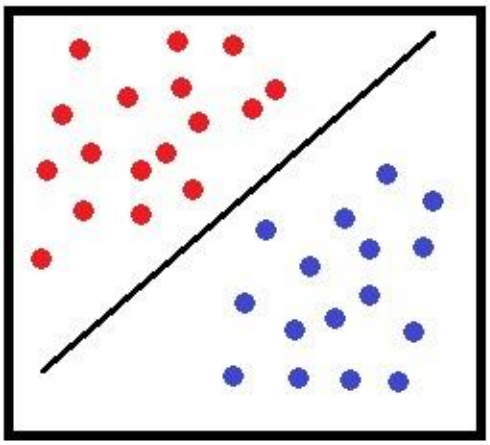
5. Choosing the best model:

I chose the SVC for my final model, as this model has the best f1 score. Also, this model performs the best for the given data. Gaussian Naïve Bayes is somewhat faster than SVC but has lower accuracy. Random Forest works very poorly for the given data, it is very slow as well as its accuracy is also very poor. There is too much overfitting taking place when using Random Forest Classifier. So, the best choice out of these three models is SVC, which has the best f1 score (F1 test SVM=0.8421, GaussianNB=0.8148, RF=0.7883), it is also fast, just a little slower than Gaussian Naïve Bayes.

SVM in Layman's terms:

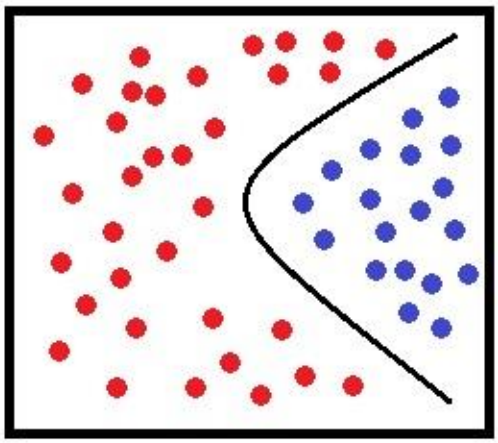
SVM means support vector machine.

1



Linearly Separable

2



Non-linearly Separable

Consider there are points of two colors, blue and red as in box 1. I ask you to draw a line in between them such that the distance between the line and both color of points is maximum and you draw a line. Now if I draw a point on top left of the line without telling you the color and ask you what color it is. Your answer will be red. That is how SVM works on linearly separable data. The points nearest to the drawn line are called support vectors, hence the name Support Vector Machine.

Now consider same red and blue point, but in a different way as in box 2. You are not able to draw a line that separates the red and blue points, but you can draw a curve that separates both the red and blue points. Again I will ask you to draw a curve such that the distance between the nearest points of both colors is maximum. Now if there is a point on the left side of the curve, and I ask you its color, you will say that it is red. That is how SVM works on non-linearly separable data.

Once the points are separated, you know that points on one side of the line or curve corresponds to one color, and points on other side corresponds to other color. This is how training takes place. Now if you draw a point anywhere and ask the machine that what color this point belongs to, it will mathematically check whether the point lies on the left or right of the line and then it will predict what color the point is. This is how SVM does prediction.

Final F1 score of the model:

F1 score for training set: 0.8667

F1 score for test set: 0.84

References:

[1] http://www.nickgillian.com/wiki/pmwiki.php/GRT/SVM

[2] http://scikit-learn.org/stable/modules/svm.html

[3] http://www-labs.iro.umontreal.ca/~pift6080/H08/documents/papers/svm_tutorial.ppt

[4] http://www.analyticsvidhya.com/blog/2015/09/naive-bayes-explained/

[5] http://www.nickgillian.com/wiki/pmwiki.php/GRT/RandomForests

[6] http://www.analyticsvidhya.com/blog/2015/09/random-forest-algorithm-multiple-challenges/

[7] http://blog.yhat.com/posts/random-forests-in-python.html