Machine Learning Engineer Nanodegree
Project 2- Building a student intervention system

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: The most fundamental question in model building is determining what you would like the model to predict. Since we are trying to predict a discrete binary outcome, whether the student will pass or fail this is classification problem.

2. Exploring the Data

Can you find out the following facts about the dataset?

Total number of students	**395**
Number of students who passed	**265**
Number of students who failed	**130**
Graduation rate of the class (%)	**30**
Number of features (excluding the label/target column)	**67.09%**

3. Preparing the Data

Execute the following steps to prepare the data for modeling, training and testing:

- **Identify feature and target columns**
- **Preprocess feature columns**
- **Split data into training and test sets**

Code is available in student_intervention.ipynb

4. Training and Evaluating Models

For the student intervention problem I chose to train and evaluate following classifiers:

- Decision Tree Classifier
- Support Vector Machines Classifier
- Randomized Forest

Decision Trees Classifier

What are the general applications of this model

A decision tree is the minimum number of yes/no questions that one has to ask to assess the probability of making a correct decision, or we can also say that a decision tree is a set of rules used to classify data into categories. It looks at the variables in a data set, determines which are most important, and then

comes up with a tree of decisions which best partitions the data. The tree is created by splitting data up by variables and then counting to see how many are in each bucket after each split.

Strengths

- Simple to understand and to interpret. Trees can be visualized.
- If the decision tree is short, it is easy for a human to interpret it.
- Able to handle both numerical and categorical data.
- Able to handle multi-output problems.
- Ability to deal with irrelevant features. The algorithm selects "relevant" features first, and generally ignores irrelevant features.
- Requires little data preparation.
- Decision trees combined into an ensemble create some of the best binary classifiers.

Weaknesses

- Sometimes decision tree learners can create over complex trees that do not generalize the data well. This is called overfitting and it is one of their main disadvantage.
- Decision trees can be unstable because small changes in the data might result in large changes in the tree. This problem is mitigated by using decision trees within an ensemble.
- Decision tree learners create biased trees if some classes dominate.
- Another disadvantage is that they don't support online learning, which means that you have to rebuild your tree when new examples come on.

Why did you choose this model to apply?

I thought it would be good potential candidate for modeling the student intervention system because:

- Is less complex so should be very time efficient.
- A simpler model that is easier to interpret and explain to the audience. It delivers the best humanly understandable results.
- Predictive power, relative simplicity.

SVM classifier

What are the general applications of this model

SVM is a non-probabilistic parametric classifier with a broad range of applications. It uses a linear hyperplane for separating the data points, which can also be used as a nonlinear classifier through the use of kernels.

- Especially popular in text classification problems where very high-dimensional spaces are the norm.
- Hand-written characters can be recognized using SVM.
- SVMs are helpful in text and hypertext categorization as their application can significantly reduce the need for labeled training.
- Classification of images can also be performed using SVMs. Experimental results show that SVMs achieve significantly higher search accuracy than traditional query refinement schemes after just three to four rounds of relevance feedback.

- SVMs are also useful in medical science to classify proteins with up to 90% of the compounds classified correctly.

Strengths

- SVMs have good generalization performance. High accuracy.
- Effective in cases where number of dimensions is greater than the number of samples.
- Uses a subset of training points in the decision function, so it is also memory efficient.
- With an appropriate kernel they can work well even if the data isn't linearly separable in the base feature space.

Weaknesses

- The major downside of SVMs is that they can be painfully inefficient to train.
- Expensive training and testing phase, both in speed and size.
- It is not recommend for any problem where you have many training examples.
- SVMs are not recommend for most "industry scale" applications. Anything beyond a toy/lab problem might be better approached with a different algorithm.
- SVMs do not directly provide probability estimates, these are calculated using an expensive five-fold cross-validation

Why did you choose this model to apply?

SVM does not require a significant amount of data to make a reasonable prediction. Since the student intervention dataset is small and SVM provides good generalization performance, high accuracy and is less computationally intensive, I thought it would be another good potential candidate for modeling the student intervention system. Also SVM is a powerful and flexible classifier that could be adjusted to specific applications through fine tuning its parameters.

Random Forest Classifier

What are the general applications of this model

Random Forest Classifier is an ensemble learning method in which a number of relatively naive hypotheses are aggregated up to create a more robust hypothesis.

Strengths

- Random forest is robust to outliers.
- Give you a really good idea of which features in your data set are the most important.
- Almost always have lower classification error and better f-scores than decision trees.
- Deal really well with uneven data sets that have missing variables.
- It can be used to generate very good classifiers if techniques (such as bagging) are used to “prune” the forest so it generalizes better.

Weaknesses

- This algorithm is computationally intensive and should use a relatively large amount of computing power.
- For better accuracy, need more trees. This can slow down the training performance.
- Will not work as well with a small dataset.
- Prone to overfitting.

Why did you choose this model to apply?

Although the dataset is small, I decided to try the Random forest model. I also wanted to see if a computationally intensive algorithm is viable on a small data set given the budget constraints.

Decision Trees Classifier vs SVM vs Random Fores

DecisionTreeClassifier	**Training set size**		
	100	**200**	**300**
Trainig Time	0.000	0.000	0.000
Prediction Time	0.000	0.000	0.000
F1 score for training set	1.000	1.000	1.000
F1 score for testing set	0.723	0.750	0.748

SVC	**Training set size**		
	100	**200**	**300**
Trainig Time	0.004	0.004	0.012
Prediction Time	0.000	0.008	0.004
F1 score for training set	0.883	0.887	0.870
F1 score for testing set	0.792	0.789	0.803

RandomForestClassifier	**Training set size**		
	100	**200**	**300**
Trainig Time	0.028	0.043	0.047
Prediction Time	0.004	0.000	0.000
F1 score for training set	0.993	1.000	0.998
F1 score for testing set	0.800	0.808	0.789

5. Choosing the Best Model

Which model is generally the most appropriate?

Based on the experiments you performed earlier, in 2-3 paragraphs explain to the board of supervisors what single model you choose as the best model. Which model has the best test F1 score and time efficiency? Which model is generally the most appropriate based on the available data, limited

resources, cost, and performance? Please directly compare and contrast the numerical values recorded to make your case.

All models performed well in case of time efficiency (training and predicting time). Only Random Forests was a bit slower while training the model comparatively to our other models. However the time difference is so insignificant that it cannot be the main factor to choose the best model for this problem. So the model in this case should be chosen based on the F1 scores that they produce.

Even though Decision trees did perfect job on classifying training data correctly, where the F1 score for all training sizes is 1.0, they were not able to reproduce the same F1 score for the testing data that means that Decision Trees are performing poorly on unseen data. Also it seems that Random Forest Classifier got the best F1 score 0.808 for training set at 200 training examples. In terms of the testing F1 scores it seems that SVM produced the best F1 score (0.803) for 300 training examples.

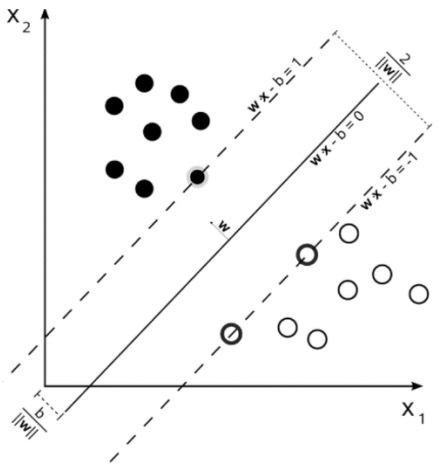
Considering all this I've choose SVM as the best model for this problem. SVM's F1 score were consistent and changed very little with varying training dataset sizes. The F1 scores produced by SVM for testing data were quite good for all training sizes.

Therefore, I chose SVM's SVC as the best model to describe the data in this problem.

How the final model chosen is supposed to work (layman's terms)?

In 1-3 paragraphs explain to the board of supervisors in layman's terms how the final model chosen is supposed to work (for example if you chose a decision tree or support vector machine, how does it learn to make a prediction).

SVM is a type of linear separator. Suppose we want to split black circles from the white ones by drawing a line. Typically there are an infinite number of lines that will accomplish this task. SVMs, in particular, find the "maximum-margin" line - this is the line "in the middle". Intuitively, this works well because it allows for noise and is most tolerant to mistakes on either side.



In case of a two dimensional problem, SVM will try to draw a curve between the different features to separate the outcomes. In our case we have multidimensional problem, so the SVM is going to try to make a surface, instead of a single curve, between all of those dimensions that best separates the students that

graduated and those that did not. The best surface or curve is the one that maximizes the distance between the different points of that feature with the different outcomes.

Fine-tune the model

Fine-tune the model. Use GridsSarch with at least one important parameter tuned and with at least 3 settings. Use the entire training set for this.

Code is available in student_intervention.ipynb.

Final F1 score

What is the model's final F1 score?

In statistical analysis of binary classification, the F1 score (also F-score or F-measure) is a measure of a test's accuracy. F1 score is summarizing the number of correct positives and correct negatives out of all possible cases.

The final F1 score that the tuned model can produce is **0.819**, which is slightly better that what the default SVM could produce.