Machine Learning Nanodegree Project-02:Build a Student Intervention System

Sanjiv Lobo,2016

As education has started to rely a lot on technology, a lot of data is generated for analysis and the results are available in real time. To meet this goal, we are making this project in which we are
analysis student data to decide if they need intervention to lower the number of failures in the system, thus increasing the quality of education provided.

Classification vs Regression.

The given dataset is clearly labelled and the output needed is if the person is going to pass or fail. This is a **classification** problem since the data is categorically divided with labels and we have to predict a certain fixed output.

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%**

Preparing the data..

Feature columns:

- school - student's school (binary: "GP" or "MS")
- sex - student's sex (binary: "F" - female or "M" - male)
- age - student's age (numeric: from 15 to 22)
- address - student's home address type (binary: "U" - urban or "R" - rural)
- famsize - family size (binary: "LE3" - less or equal to 3 or "GT3" - greater than 3)
- Pstatus - parent's cohabitation status (binary: "T" - living together or "A" - apart)
- Medu - mother's education (numeric: 0 - none, 1 - primary education (4th grade), 2 – 5th to 9th grade, 3 – secondary education or 4 – higher education)

- Fedu - father's education (numeric: 0 - none, 1 - primary education (4th grade), 2 – 5th to 9th grade, 3 – secondary education or 4 – higher education)
- Mjob - mother's job (nominal: "teacher", "health" care related, civil "services" (e.g. administrative or police), "at_home" or "other")
- Fjob - father's job (nominal: "teacher", "health" care related, civil "services" (e.g. administrative or police), "at_home" or "other")
- reason - reason to choose this school (nominal: close to "home", school "reputation", "course" preference or "other")
- guardian - student's guardian (nominal: "mother", "father" or "other")
- traveltime - home to school travel time (numeric: 1 - <15 min., 2 - 15 to 30 min., 3 - 30 min. to 1 hour, or 4 - >1 hour)
- studytime - weekly study time (numeric: 1 - <2 hours, 2 - 2 to 5 hours, 3 - 5 to 10 hours, or 4 - >10 hours)
- failures - number of past class failures (numeric: n if 1<=n<3, else 4)
- schoolsup - extra educational support (binary: yes or no)
- famsup - family educational support (binary: yes or no)
- paid - extra paid classes within the course subject (Math or Portuguese) (binary: yes or no)
- activities - extra-curricular activities (binary: yes or no)
- nursery - attended nursery school (binary: yes or no)
- higher - wants to take higher education (binary: yes or no)
- internet - Internet access at home (binary: yes or no)
- romantic - with a romantic relationship (binary: yes or no)
- famrel - quality of family relationships (numeric: from 1 - very bad to 5 - excellent)
- freetime - free time after school (numeric: from 1 - very low to 5 - very high)
- goout - going out with friends (numeric: from 1 - very low to 5 - very high)
- Dalc - workday alcohol consumption (numeric: from 1 - very low to 5 - very high)

- Walc - weekend alcohol consumption (numeric: from 1 - very low to 5 - very high)
- health - current health status (numeric: from 1 - very bad to 5 - very good)
- absences - number of school absences (numeric: from 0 to 93)

Target columns:

- passed - did the student pass the final exam (binary: yes or no)

Processed feature columns:

['school_GP', 'school_MS', 'sex_F', 'sex_M', 'age', 'address_R', 'address_U', 'famsize_GT3', 'famsize_LE3', 'Pstatus_A', 'Pstatus_T', 'Medu', 'Fedu', 'Mjob_at_home', 'Mjob_health', 'Mjob_other', 'Mjob_services', 'Mjob_teacher', 'Fjob_at_home', 'Fjob_health', 'Fjob_other', 'Fjob_services', 'Fjob_teacher', 'reason_course', 'reason_home', 'reason_other', 'reason_reputation', 'guardian_father', 'guardian_mother', 'guardian_other', 'traveltime', 'studytime', 'failures', 'schoolsup', 'famsup', 'paid', 'activities', 'nursery', 'higher', 'internet', 'romantic', 'famrel', 'freetime', 'goout', 'Dalc', 'Walc', 'health', 'absences']

Training set: 300 samples

Test set: 95 samples

Training and Evaluation Methodology

Decision Tree Classifier:
Reasons for choosing this classifier:-

1. It can easily be visualized by a layman and one can understand the process being done better.
2. Predictions can be done in O(T logT).
3. It does binary classification which is useful since we need to know if a student passes or fails.

Advantages:-

1. The decision tree can be easily visualized[2](check output in the notebook.)
2. Data input can be given without much preprocessing.

Disadvantages:-

1. They can result in overfitting to the training data which causes test data errors..
2. Slight change in input dataset calls for a new tree to be generated which is problematic.
3. Biased tree are formed if certain features are ignored or given extra importance.

Applications:-

To uncover flaws in a Boeing manufacturing process.[1]

Support Vector Machines

Reasons for choosing this classifier:-

1. They work excellently when there are a lot of features relative to the size of data available.
2. SVMs work quite well when the dataset size is small.
3. Our dataset does not contain a lot of noise, making it a perfect fit for it.

Advantages:-

1. It can still provide good prediction when the number of features is greater than the number of samples.
2. Uses a subset of training points in the support vectors, making it memory efficient.
3. Different Kernel functions can be specified for the deciding function.

Disadvantages:-

1. It could perform poorly if the number of features is greater than the samples to a large extent, due to overfitting.

Applications:-

Image processing, Object recognition, text processing and classification.

Naive Bayes(Bernoulli) Classifier:-

Reasons for choosing this classifier:-

1. It does not generate a relation between different features and can be given independent labelled data to predict from easily.
2. In spite of their oversimplified assumptions, naive Bayes classifiers have many applications, like document classification and spam filtering.

Advantages:-

1. Small training dataset needed to estimate the necessary parameters.
2. Fast to train (single scan).and fast to classify.
3. Not sensitive to irrelevant features.
4. Handles real and discrete data.
5. Handles streaming data well.

Disadvantages:-

1. Assumes independence of features and doesn't realise related features.
2. Takes longer to train than to predict.

Applications:-

Spam classification for email inboxes.

Tables:-

Decision Tree Classifier

	Training set size		
	100	200	300
Training time (secs)	0.017	0.009	0.008
Prediction time (secs)	0.001	0.001	0.004
F1 score for training set	1.0	1.0	1.0
F1 score for test set	0.6616	0.5812	0.6666

Average F1 score for test set = **0.6366**

Support Vector Machines

	Training set size		
	100	200	300
Training time (secs)	0.006	0.014	0.020
Prediction time (secs)	0.003	0.005	0.006
F1 score for training set	0.8537	0.8497	0.8650
F1 score for test set	0.7843	0.7862	0.7973

Average F1 score for test set = **0.7893**

Naive Bayes(Bernoulli) Classifier

	Training set size		
	100	200	300
Training time (secs)	0.009	0.008	0.018
Prediction time (secs)	0.002	0.008	0.027
F1 score for training set	0.8456	0.7900	0.8073
F1 score for test set	0.7681	0.7230	0.7761

Average F1 score for test set = **0.7557**

Choosing the right model..

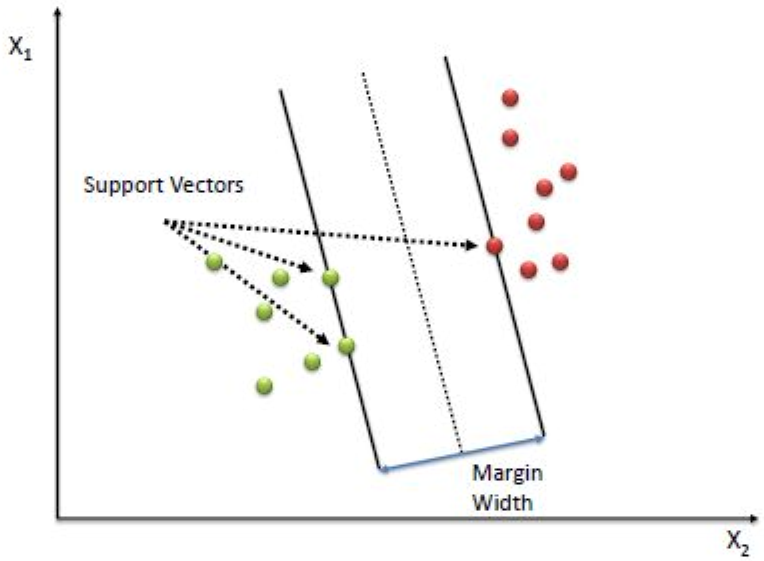
Based on the analysis done above, SVM provides the best performance without overfitting the data. Now the choice is either better performance or lower training time. It's better to choose higher performance as we need accurate results as it is the student's future at stake, as failing the final exam can have an adverse affect on their psychological state.

The time taken to train and predict decision trees is the lowest but its F1 score is awful when compared to what level we need it to be(F1 score = 0.6366). Now, NB classifier takes about the same time as the SVM and has a lower F1 score of 0.7577. SVM has the best F1 score = 0.7893 which means that it is the best model when it comes to predicting results properly and since we have established the fact that accurate results are of paramount importance in comparison to computational time, the SVM model seems the best fit to our needs.

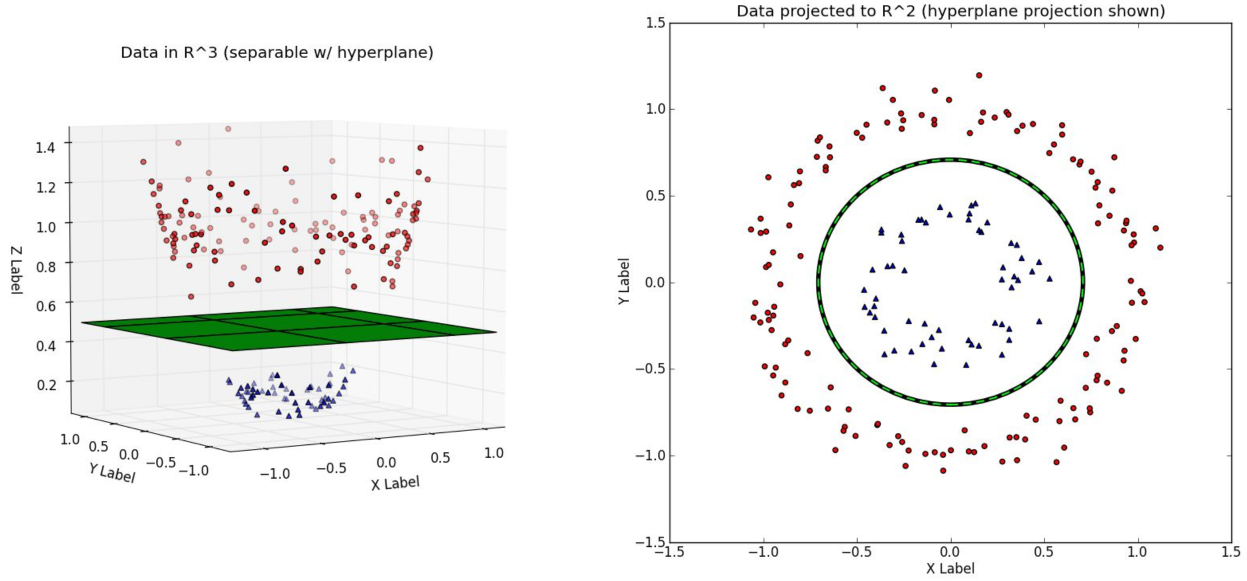
Support Vector Machines come from the concept of decision lines that define decision boundaries. A 'decision line' is one that separates different types of objects. In other words, labeled training data as input, the algorithm outputs a division that categorizes new test data. SVM chooses the best decision line or division such that the distance between that line and the nearest observations of different classes is maximum.

What the SVM does is that it separates linearly separable data into its own groups and finds a hyperplane to divide the groups. In the figure given below two groups of data are separated with the help of a hyperplane that helps in classifying the input test dataset. As shown in the figure below, the margin width is the width between the two different types of training datasets inputs given. Our aim is to maximize this width so as to ensure that the test data can be classified properly.

The data points on the very edge of the margin or decision boundary are the most important ones. They are called support vectors and have a direct impact on the decision surface and thus, the margin. A change in the position of the vectors change the position of the decision surface.



The kernel trick is used when datasets have more than two dimensions. What it essentially does is that it converts it into a linear model where the data can be separated.



Let’s take the right figure as an example.

The data is not linearly separable even though the distinction is pretty obvious. What the kernel trick does is that it takes the x-axis and y-axis input and sums their squares which is effectively the equation of a circle(mapping to a radial coordinate system). Also, a radial based function can be used. This value is then plotted and its plot turns out to be linear and the data can be separated easily. This was done by modifying the kernel in a smart and effective way to convert the data into a linear format. The same is done for the left example also.

This linearly separable data is then classified as described earlier.

The chosen model was tuned using Grid Search CV because the data set is small and unbalanced. Also, in such a case, F1 score is better than accuracy as a metric of evaluation. The parameters optimized were gamma and C. Attempts to use kernel as a parameter were made, but results were erratic and caused problems in the notebook, However, a F1 score of 0.8082 on the test sets were obtained, which was a bit better than the default model. With an increased dataset, the F1 score will definitely improve with the GridSearchCV parameter tuning method.

References:-

[1] P. RIDDLE, R. SEGAL, AND O. ETZIONI. Representation design and brute-force induction in a Boeing manufacturing domain. Applied Artificial Intelligence, 8(1):125--147, January-March 1994.

[2] Scikit-learn documentation-http://scikitlearn.org/stable(helped a lot for the coding part of the project.)