Predicting Student Grades using Machine Learning

Naveen Venkat
Dept. of CSIS
BITS Pilani
f2015078*

Sahaj Srivastava
Dept. of CSIS
BITS Pilani
f2015091*

Lakshya Garg
Dept. of Mech. Eng.
BITS Pilani
f2016432*

The code for all the experiments performed here and the dataset is publicly available on **github.com/nmakes/predicting-compre-grades**

I. Data Cleaning

We discover that the data contains a lot of NULL values for each attribute. Table 1 summarizes the number of Non-null records for each attribute. We consider only non-null values for our analysis. Also, we remove the withdrawn (**W**) cases since they do not influence the statistics of the course.

Attribute	Number of non-Null Records
IDNO	203
Year	200
Attendance %	73
M/F	200
CGPA	73
Mid Semester	200
Mid Sem Grade	200
Mid Sem Collection	200
Quiz 1 (30)	202
Quiz 2 (30)	199
Part A (40)	202
Part B (40)	202
Grade	203

Table 1: Meta-Data

* @pilani.bits-pilani.ac.in : Email-ID

We consider 9 attributes for our analysis as given in Table 2.

	Mean	Std
Mid Semester	19.04	8.70
Quiz 1	12.95	6.00
Quiz 2	11.32	5.56
Part A	16.02	6.71
Part B	17.30	7.75
CGPA	8.30	1.17
Year	2.49	0.53
Attendance	4.50	0.74
Grade	6.72	2.06

Table 2: Mean and Standard Deviation of each useful attribute

II. Elementary Analysis using correlation

The correlation of two random variables is defined as:

$$Corr(X,\ Y) = E\left(\frac{(X-\overline{X})(Y-\overline{Y})}{\sqrt{Var(X)\cdot Var(Y)}}\right)$$

A positive correlation implies that both **X** and **Y** increase and decrease together. We do note that correlation does not necessarily imply causation. However, it could provide insight into the behaviour of various variables. These are observed in tables 3-5.

	Mid Semester	Mid Sem Grades	Quiz 1	Quiz 2	Part A	Part B	Grade
Mid Semester	1.00	**0.96 ***	0.64	0.35	0.52	0.53	**0.75 ***
Mid Sem Grades		1.00	0.61	0.37	0.5	0.53	0.72
Quiz 1			1.00	0.47	0.62	0.58	**0.80 ***

Quiz 2				1.00	0.59	0.51	0.63
Part A					1.00	0.66	**0.81 ***
Part B						1.00	**0.76 ***
Grade							1.00

Table 3: Correlation among various scores. The top 5 correlations have been marked in bold with an asterisk (*).

	Attendance	**Mid Sem Grades**	**Grade**
Attendance	1.00	0.03	0.16
Mid Sem Grades		1.00	0.71
Grade			1.00

Table 4: Correlation among attendance and grades. As we see, attendance has very less correlation with Mid Sem Grades and final Grade. One plausible reason could be due to the bucketing of the **Attendance** attribute into 5 levels, due to which the variance in attendance is at least five times as slow as variance in the grades. We also see, **Mid Sem Grades** and final **Grade** have a much higher correlation.

	Mid Sem	**Mid Sem Grade**	**Quiz 1**	**Quiz 2**	**Part A**	**Part B**	**Grade**
Mid Sem Collection	-0.21	-0.2	-0.19	-0.29	-0.29	-0.21	-0.28

Table 5: Correlation among Mid Sem Collection and performance. A weak negative correlation is seen between the Mid Sem Collection and the various scores. This suggests that students who have scored well, had shown lesser tardiness in collecting their mid sem answer scripts.

III. Grade prediction performance of various classifiers

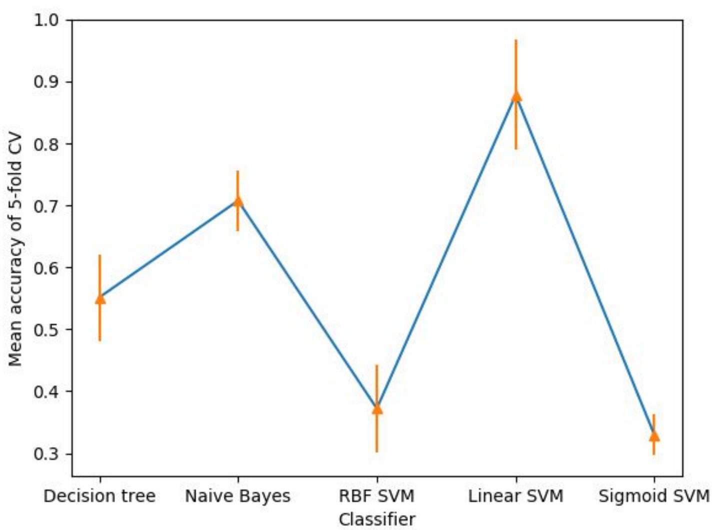
We model the prediction of final **Grade** as a classification problem. Each grade is assumed to be a class. Hence, we have 9 classes (**A** through **E**, and **NC**), since those denote the students who took the course till completion. For these tests we ignore the class withdrawn (**W**) because most of the data is missing for such records.

Since the dataset is very small (197 records for scores, after removal of **W** as mentioned above), we perform a **stratified 5-fold cross-validation** for each of the classifiers to observe and compare their stability. Hence we obtain a **80:20 train-test split**. The stratification aids in handling the **class imbalance problem**.

We use **Decision Tree, Naive Bayes, SVM** (with linear, RBF and sigmoid kernels), and **K-Nearest Neighbor** (k = 1 to 20). In this section, the following experiments are described. We list the scores in **Appendix A**.

EXPERIMENT 1: Predicting final grade using only test scores.

We use five attributes - **Mid sem**, **Quiz 1**, **Quiz 2**, **Part A**, **Part B** to predict the final **Grade**.



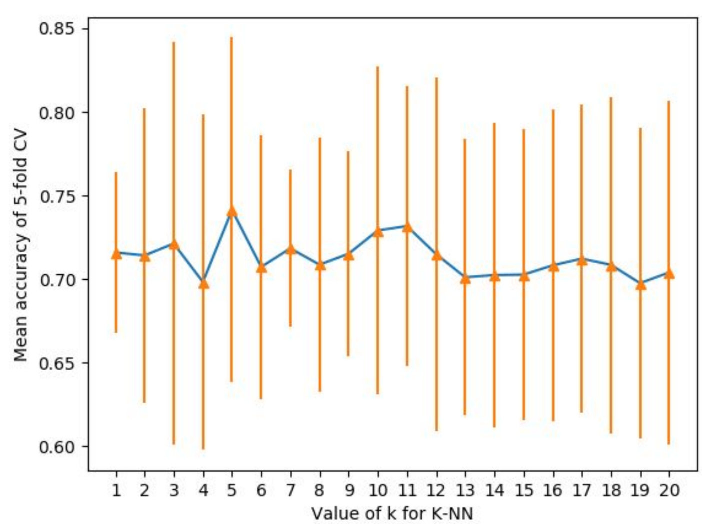


Figure 1: Comparison of prediction accuracy using only test scores. Note that an SVM with a linear kernel obtains the highest accuracy (mean: 0.88, std: 0.08). This indicates that the input data could be linearly separable.

EXPERIMENT 2: Predicting final grade using only additional information.

We observe the prediction capability of the classifiers using only **Year, Attendance, CGPA & MidSem Collection** to identify how well do these attributes distinguish.

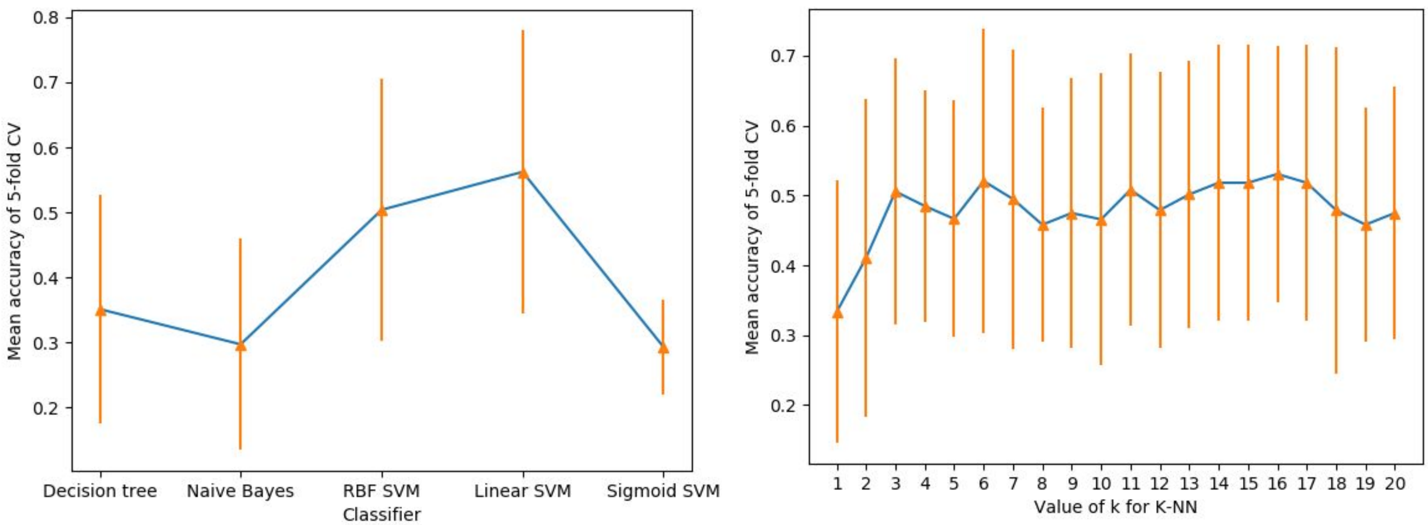


Figure 2: Comparison of prediction accuracy using only additional information. Here too, a linear SVM achieves a good performance (mean: 0.56, std:0.21). However, all the classifiers are not as stable in this data (standard deviation quite high). Also, K-NN performs similar to other classifiers

EXPERIMENT 3: Predicting final grade using both the kinds of attributes.

We combine the attributes in **1** and **2** and compare the results.

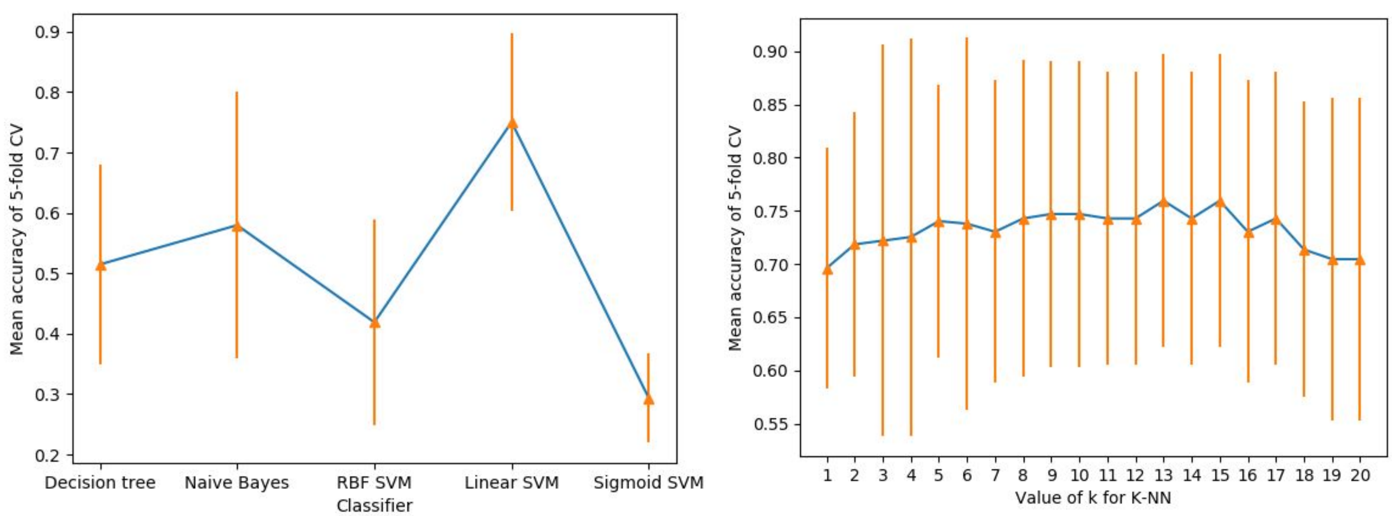


Figure 3: Comparison of prediction accuracy using all the information. Here however, the K-NN classifier performs similar to the SVM with linear kernel.

From the observations in the three tests above, we find that the input space is linearly separable with only the test scores as input. This can be seen by the decrease in performance of the various classifiers on the inclusion of the additional information such as year and attendance. Hence considering only the test scores can help predict the final grade of the student to a good extent.

IV. Performance with Principal Component Analysis

Since we observe good classification with the attributes in section III experiment 1, we ask what could happen if PCA was applied on those dimensions. Hence we run Experiment 1, with PCA applied on the five input dimensions. Though, we estimate that here, PCA will not be a good technique to use since it the scores themselves may not We consider 1, 2, 3, 4, 5 principal components for this purpose. The results obtained are shown in Figures 4-8.

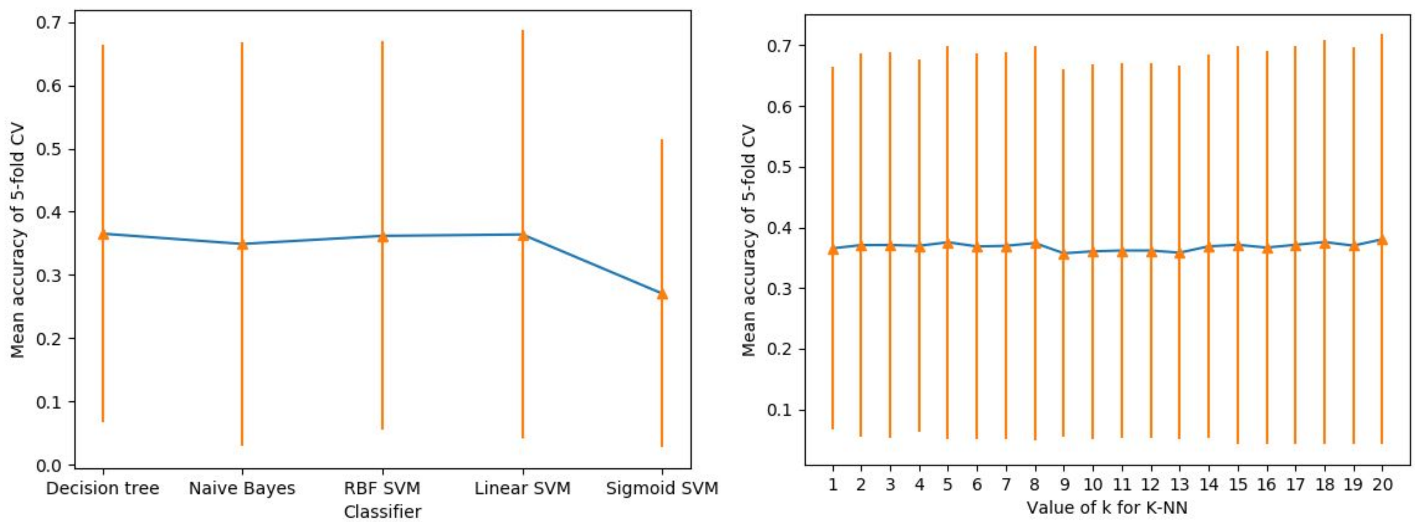


Figure 4: PCA with 1 component

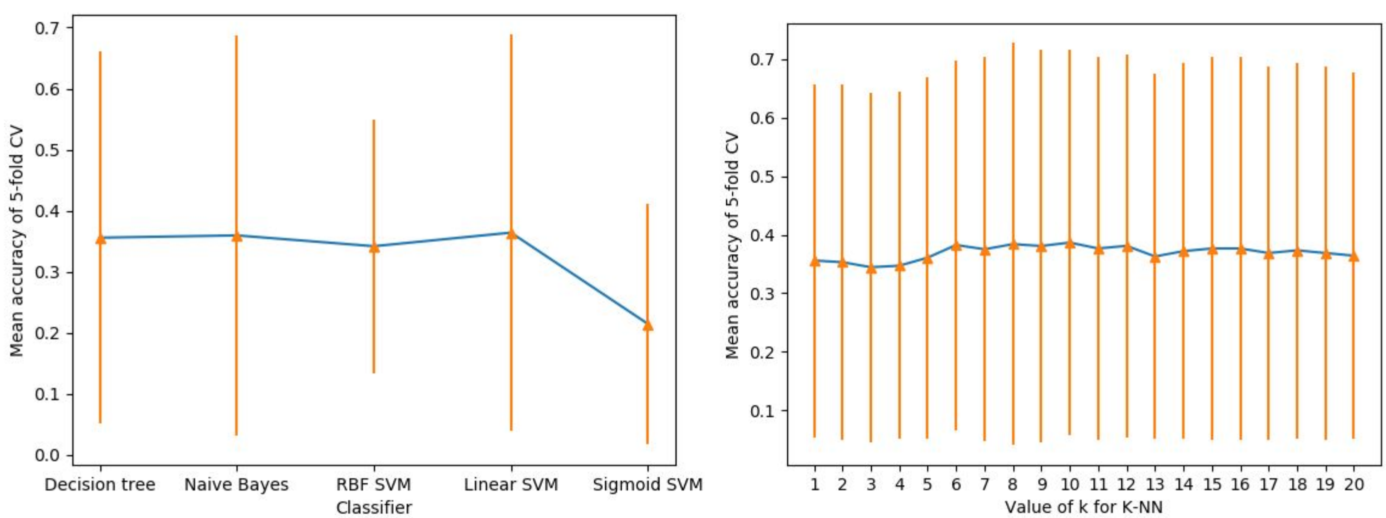


Figure 5: PCA with 2 components

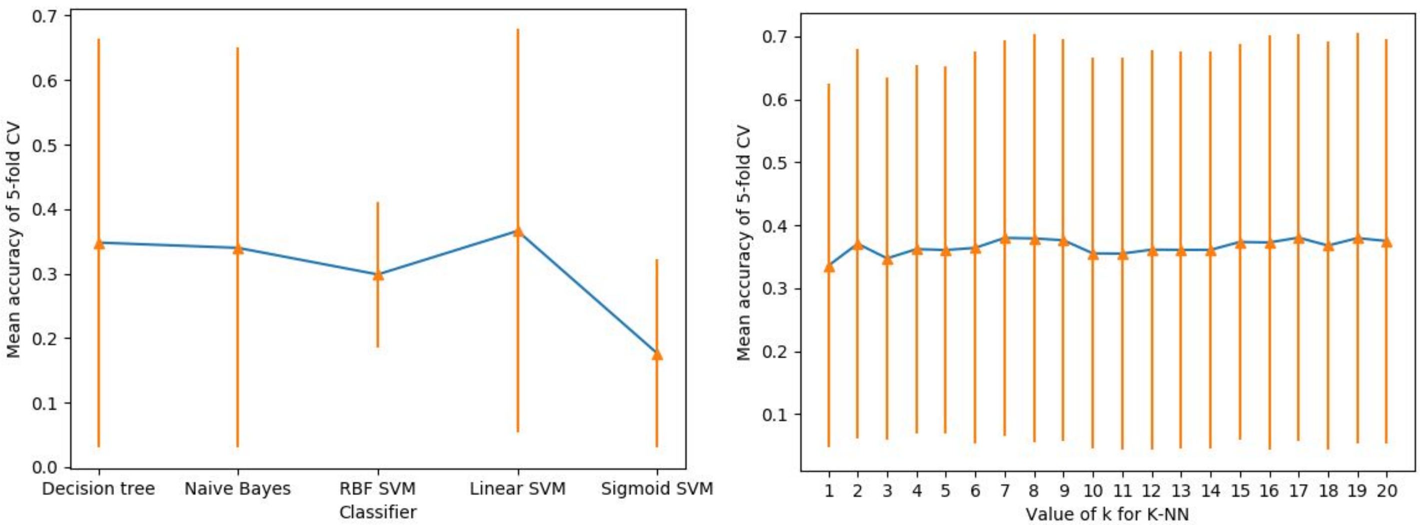


Figure 6: PCA with 3 components

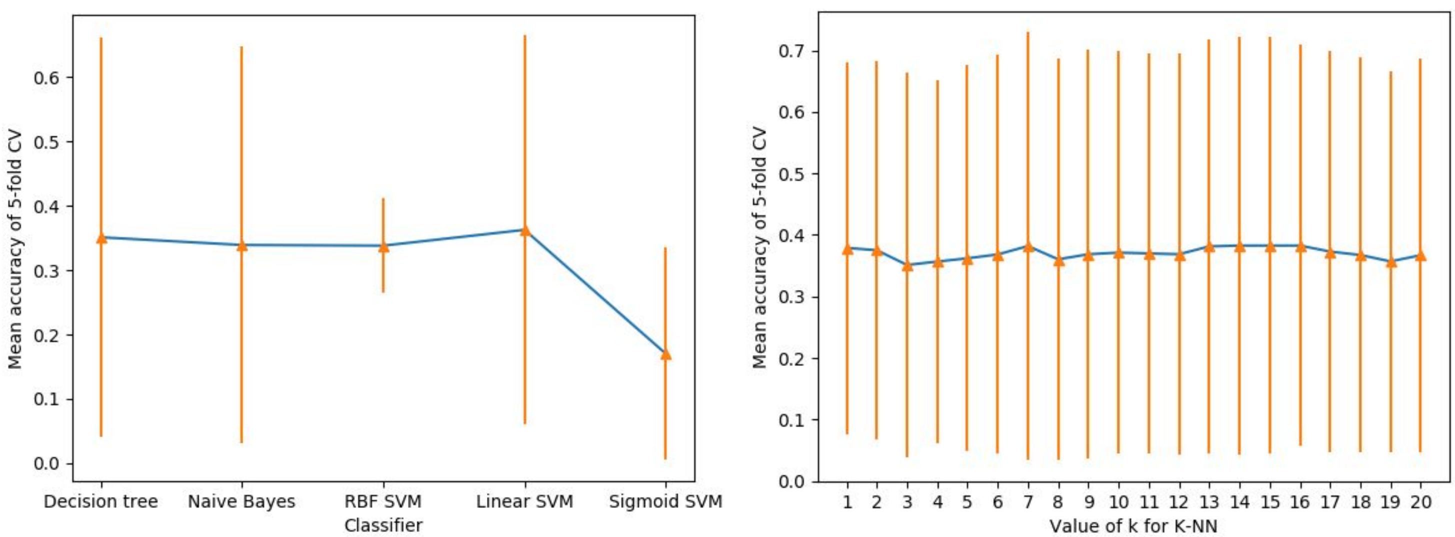


Figure 7: PCA with 4 components

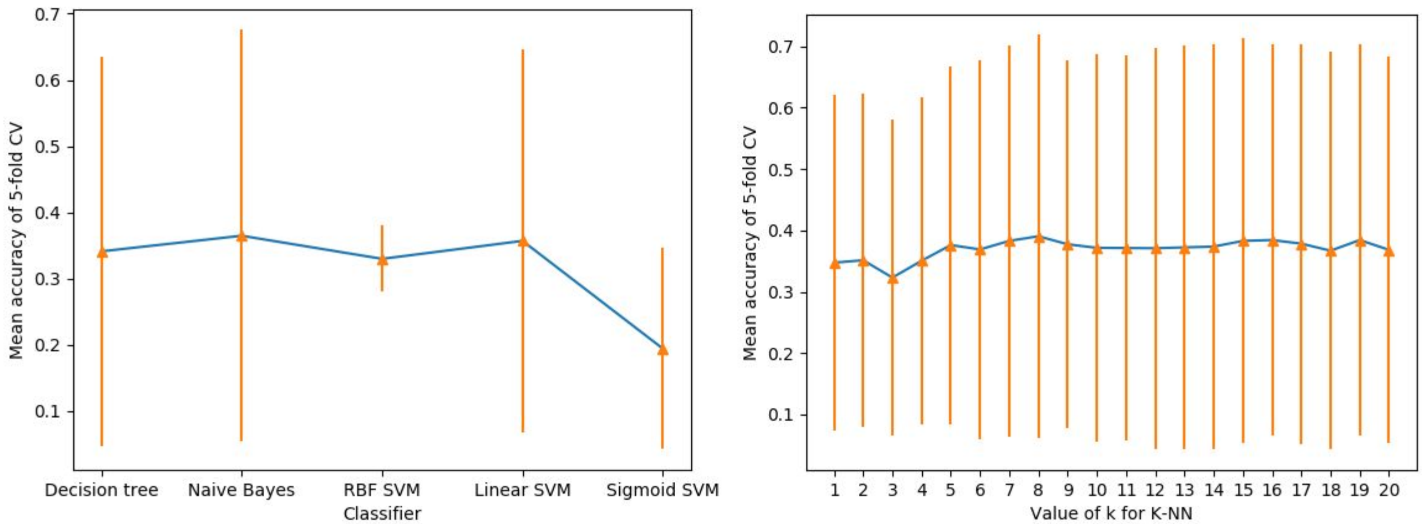


Figure 8: PCA with 5 components

We note that as our hypothesis, all classifiers in the dimensions of the Principal Components perform poorly. Hence PCA is not a good choice for dimensionality reduction. However, Fischer's LDA can be performed to improve results.

V. <u>Results and Conclusions</u>

We find that the data is linearly separable and a good classification accuracy is obtained. Thus we can build a model using the given data to predict the final grade. The most distinguishing attributes are the various test scores (midsem, quiz 1, quiz 2, part a, part b) which clearly separate the data into a linearly separable way. We demonstrate in Section IV that PCA doesn't yield as good performance even when applied on the simple linearly separable data because it doesn't take class labels into account (PCA is class agnostic).

Appendix

Accuracies of various classifiers (mean and standard deviation) on the 5-fold cross-validation experiments

Experiment 1	Experiment 2	Experiment 3
Decision tree mean: 0.551341142517613 std: 0.06993379645831806 **Naive Bayes** mean: 0.7074808590102709 std: 0.048468673989652225 **RBF SVM** mean: 0.3721235888294712 std: 0.07063348703877383 **Linear SVM** mean: 0.8779460147695441 std: 0.08888418888257285 **Sigmoid SVM** mean: 0.3292808759867583 std: 0.03293915411592424 **KNN (k=5)** mean: 0.7413097360156183 std: 0.10323973050486965	**Decision tree** mean: 0.35143939393939394 std: 0.1762787629006127 **Naive Bayes** mean: 0.2973484848484848 std: 0.16251306721494937 **RBF SVM** mean: 0.5040151515151514 std: 0.2018065931926276 **Linear SVM** mean: 0.5623484848484848 std: 0.21741976819385406 **Sigmoid SVM** mean: 0.29318181818181815 std: 0.07371254943972283 **KNN (k= 16)** mean: 0.5306818181818181 std: 0.18373617462028402	**Decision tree** mean: 0.5146969696969697 std: 0.1656775747333154 **Naive Bayes** mean: 0.5796969696969697 std: 0.22109476962006971 **RBF SVM** mean: 0.41901515151515145 std: 0.17083172074761305 **Linear SVM** mean: 0.7503030303030302 std: 0.1466272956172157 **Sigmoid SVM** mean: 0.29318181818181815 std: 0.07371254943972283 **KNN (k=13)** mean: 0.7595454545454545 std: 0.13757216989047844