Classifications for Business Graduate Admissions

Aqdas Juya

2024-03-17

Table of contents

Executive Summary

This report analyzes different classification models to predict business graduate admissions based on GPA and GMAT scores. Using 10-fold cross-validation, we compared the Support Vector Classifier (SVC) with a linear kernel, the Support Vector Machine (SVM) with polynomial kernels, and SVM with a radial basis function (RBF) kernel. The Polynomial SVM (Degree 3) achieved the highest accuracy (93.33%), while the SVC and RBF models both performed well at 80% accuracy. Overall, SVC offers a simple yet effective approach, but the Polynomial (Degree 3) model provides the best classification performance.

Introduction and Background

Admissions decisions in business schools rely on multiple factors to assess an applicant's potential for success. A common approach involves using academic benchmarks such as grade point average (GPA) and graduate management admission test (GMAT) scores to situate applicants into different decision groups: admit, do not admit, and borderline. In this study, we analyze business school admission data to evaluate the success of these predictors in classifying applicants and determining the most appropriate classification method.

The primary objective of this report is to develop and compare different classification methods that provide the most accurate results based on the GPA and GMAT scores. This involved conducting an exploratory data analysis (EDA) to understand the distribution of the predictors and their relationships with admission decisions, the use of support vector classifier (SVC) and evaluating its performance using a 10-fold cross validation, comparing different support vector machines (SVM) including polynomial kernel of degree 2 and radial basis function (RBF) kernel with response(Y) and cost of parameter to determine the most effective path.

A specific constraint we were held upon was to ensure that the classification model generalizes well to new applicants. To address this, the dataset was divided into training and test sets, with only the last five observations served for testing.

Through comparing different SVC and SVM methods, our results showed the model whose classification method best serves the differentiation for the applicants and provides a dependable basis for admission decisions.

Data Description

The dataset used in this report consists of business school admission records, where applicants are categorized into one of the three groups based on their academic standing: admit (group 1), do not admit(group 2), and borderline (group 3). The data specifically reported two noteworthy predictors: undergraduate grade point average (GPA) and graduate management admission test (GMAT) scores. The data was divided into training and test roles, where the test data contains only the last five observations from each group, with the training data taking into account the rest of the results. There were no reported missing values and it did not contain any personal known information, all in all, which followed the analysis coherently.

Variables

- GPA(X1): Undergraduate grade point average
- GMAT(X2): Graduate management admission test scores
- De(group type): Response variable index for admission rulings
    - Admit - 1
    - Do not admit - 2
    - Borderline - 3

Exploratory Data Analysis (EDA)

An exploratory data analysis (EDA) was performed to analyze GPA and GMAT scores across three different admission groups: admit, do not admit, and borderline on the training data set. The statistics from table 1 specifically side with group 1 (admit) with the highest GPA and GMAT scores, 693 and 3.78, respectively. Group 2 (do not admit) figured the lowest GPA, 2.13, while group 3 (borderline) resulted in the lowest GMAT score of 313. The most common (median) GPA resulted in a 3.0, with a range of 2.13 to 3.80. The most common (median) GMAT score resulted in 482, ranging from 313 to 693. Our data visualization plots concluded in providing specific relationships between GPA and GMAT scores for each group of the training data set. Figure 1. illustrates the relationship speaking relatively the same for our statistical data, concluding that group 1 has the highest GPA and GMAT scores, while group 2 and 3 have lower scores for their respective categories, GPA and GMAT in that order. On the contrary, figure 1.1 (GPA) and figure 1.2 (GMAT) indicated some overlap between the groups, hinting at GPA being a stronger predictor of admission because there is very less overlap as compared to GMAT, which showed more overlap in groups 2 and 3, being of medium importance..Higher curves in density plots represent where most data points are concentrated, helping to visualize how the GPA and GMAT distributions differentiate across groups. The wider curves in the plot indicate a higher standard deviation and higher variance, hinting at the scope of a broader admission criterion (looking at admitted students), meaning a wider range of scores can still secure admission. At the same time, higher variance doesn't mean admission is easier, but rather that low scores don't automatically disqualify applicants.

Table 1: Summary statistics of GPA and GMAT for Admit, Do not Admit, Borderline

Group	Type	Min	Q1	Median	Q3	Max	Mean/SD	N	Missing
Admit(1)	GPA	2.96	3.265	3.385	3.477	3.78	3.375/ 0.193	26	0
	GMAT	431	524	558	594.75	693	561.38/ 68.869	26	0
Do not Admit(2)	GPA	2.13	2.355	2.43	2.525	2.68	2.43/ 0.144	23	0
	GMAT	321	411.5	458	504.5	542	453.56/ 61.818	23	0
Borderline (3)	GPA	2.73	2.86	3	3.12	3.5	2.99/ 0.189	21	0
	GMAT	313	419	446	485	546	447.95/ 50.716	21	0

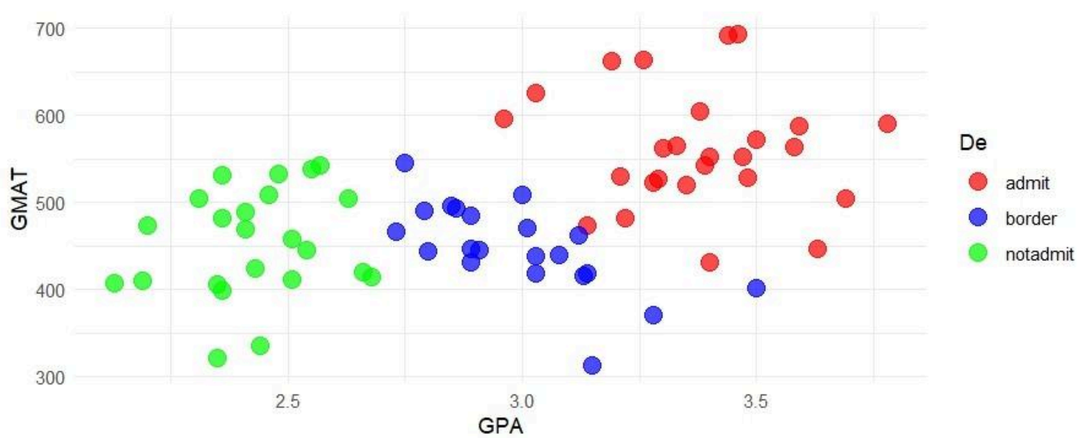


Figure 1. Relationship between GPA, GMAT scores, and distribution of three groups

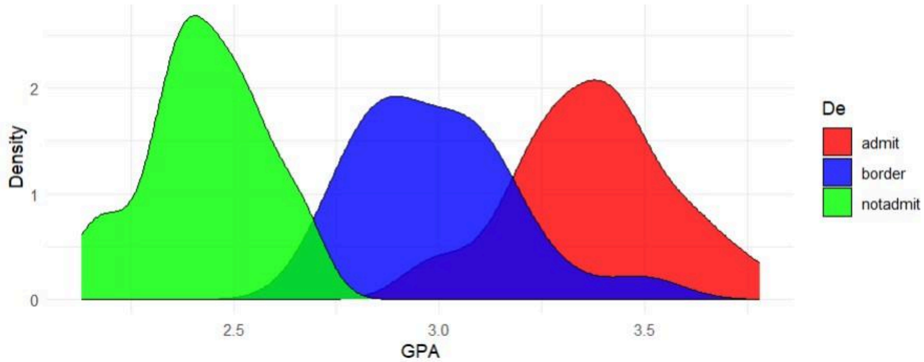


Figure 1.1 GPA values for three different groups

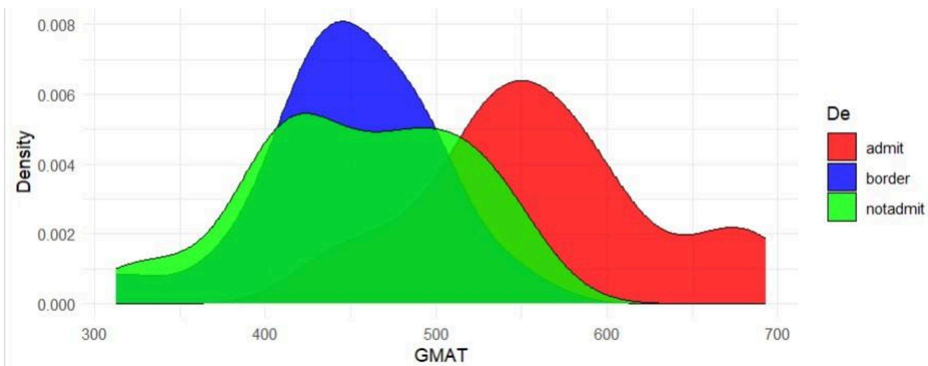


Figure 1.2 GMAT scores for three different groups

Methodologies

SVM Model Training and Evaluation: Implementing One-Versus-One Classification:

Training and Testing

To ensure the model's ability to generalize to unseen data, the dataset was divided into training and test sets:

- The last five observations from each admission category were reserved for testing.
- The remaining data was used for training the SVM model.
- The trained model was then applied to the test set, and its performance was evaluated using a confusion matrix and accuracy metrics.

Multi-class Classification

SVM and SVC are inherently designed for binary classification; however, they can be extended to multi-class classification using One-Versus-One (OvO) or One-Versus-All (OvA) strategies. (1)(2) In this project, the One-Versus-One classification was used, as it is the default strategy for the tune function in R. It is worth noting that One-Versus-One generally yields higher precision compared to One-Versus-All but at the cost of increased computational demand. (3) Since the dataset used in this study was relatively small, this computational overhead was negligible.

Support Vector Classifier

The Support Vector Classifier (SVC) with a linear kernel was selected for this analysis because it is effective when data is linearly separable. SVC aims to find the optimal decision boundary that maximizes the margin between different classes, improving classification performance. We applied 10-fold Cross-Validation (CV) to select the optimal hyperparameter C to ensure the best model performance and avoid overfitting.

Model Tuning and Selection

The tune function from the e1071 package was used to determine the optimal hyperparameters. The function was supplied with the model equation, kernel type, and a range of cost values, followed by specifying the cross-validation method.

The tuning process explored the following values:

- Cost parameter (C): {0.01, 0.1, 1, 10, 100}
- Kernel type: "linear"

After cross-validation, the optimal cost parameter was identified as C = 1, which provided the highest accuracy on the training set while maintaining generalization.

Support Vector Machines

Support Vector Machines (SVM) is a supervised learning algorithm used for classification tasks, particularly when data is not linearly separable. By transforming input data into a higher-dimensional space, non-linear data can be separated linearly using a hyperplane, which is a flat affine subspace of dimension $p-1$. (4)

Kernels play a crucial role in this process by mapping data into a higher-dimensional space, allowing for better class separation without the need for explicit transformation. Instead, the kernel function directly computes decision boundaries, significantly reducing computational costs. The precise mathematical formulation, however, is beyond the scope of this project. (5)(6)

Polynomial Kernel

The polynomial kernel function is given by:

$$K(x_1, x_2) = (x_1^T x_2 + c)^d$$

where:

- X1 and X2 are input vectors,
- C is the cost parameter,
- d is the degree of the polynomial,
- The output is a scalar value representing similarity between the two vectors.

A higher scalar value indicates a greater likelihood that the two data points belong to the same class. The cost parameter C in SVM controls the trade-off between maximizing the margin and minimizing misclassification errors, similar to its role in Support Vector Classifiers. While the degree influences the complexity and flexibility of the decision boundary. (7)

Model Tuning and Selection

To determine the optimal hyperparameters, the tune function from the e1071 package in R was used. The function was supplied with the model equation, kernel type, degree, and a range of cost and degree values, followed by specifying the cross-validation method.

A 10-fold cross-validation (CV) was applied to the training set to balance bias and variance while reducing the risk of overfitting. The input variables were GPA and GMAT as predictors, with De as the response variable.

The tuning process explored the following values:

- Cost parameter (C): {0.125, 0.25, 0.5, 1.0, 2.0, 4.0 8.0}
- Polynomial degree: {2, 3, 4, 5}

The optimal values identified were:

- Cost (C) = 4
- Polynomial degree d = 3

However, since the project required initial results for degree (d) = 2, these results are presented first. It is important to note that different ranges of C were explored during the tuning process; however, even with slightly different optimal values for C, the performance metrics did not improve.

Radial Kernel

A Support Vector Machine (SVM) with a Radial Basis Function (RBF) kernel was implemented to classify applicants into three admission groups. The RBF kernel was selected due to its ability to accommodate nonlinear decision boundaries, making it suitable when GPA and GMAT scores alone do not provide a clear linear separation between groups. The kernel function is defined as:

$$K(x_i, x_{i'}) = \exp(-\gamma \sum_{j=1}^{p} (x_{ij} - x_{i'j})^2).$$

This function measures the similarity between data points, where γ (gamma) controls the influence of individual training samples on the decision boundary. A higher gamma value results in a more complex decision boundary, whereas a lower gamma value produces a smoother boundary.

Hyperparameter Tuning

To make sure the model performed well, we used 10-fold cross-validation to fine-tune two key parameters: cost (C) and gamma (γ). The cost parameter (C) controls the balance between having a larger margin and minimizing classification errors, while gamma (γ) affects how much influence each training example has on the decision boundary. If gamma is too high, the model becomes too sensitive to individual points, leading to a complex boundary. On the other hand, a lower gamma value creates a smoother, more general boundary.

To find the best combination of these parameters, we ran a grid search with the following values:

- C values: {0.1, 1, 10, 100, 1000}
- Gamma values: {0.1, 1, 2, 3, 4}

We used the tune() function in R to test different combinations and pick the one that gave the highest classification accuracy. After testing all options, the best values were:

- Optimal Cost (C): 1
- Optimal Gamma (γ): 0.1

By tuning these parameters, we got our model, which let us check its performance using accuracy and a confusion matrix on the test data.

Sigmoid Kernel

As an extension of this project, the sigmoid kernel was selected for classification purposes. This kernel is based on the hyperbolic tangent function and attempts to separate data using a sigmoid-like curve. It enables the creation of nonlinear decision boundaries, though its complex structure makes it more challenging to interpret compared to other kernels. Due to its specificity and the difficulty in understanding how it generates decision boundaries, the sigmoid kernel is used less frequently than other kernels. (8)(9)

The sigmoid kernel function is defined as:

$$K(x_n, x_i) = \tanh(\gamma(x_n, x_i) + r)$$

where:

- Xn and Xi are input vectors,
- γ (gamma) is the scaling parameter,
- r (also known as Coef0) is the bias or offset term,
- C is the cost parameter.

This function measures the similarity between data points, where γ (gamma) controls the influence of individual training samples on the decision boundary, r shifts the data up or down, and C controls the trade-off between maximizing the margin and minimizing classification errors.

Model Tuning and Selection

To determine the optimal hyperparameters, the tune() function from the e1071 package in R was used. The function was supplied with the model equation, kernel type, and a range of values for gamma, Coef0 (r), and cost, followed by the specification of the cross-validation method.

A 10-fold cross-validation (CV) was applied to each parameter set using the training data. The input variables were GPA and GMAT as predictors, with De as the response variable.

The tuning process explored the following values:

- **Gamma parameter (γ):** {0.015625, 0.03125, 0.0625, 0.125, 0.25, 0.5, 1.0}
- **Coef parameter (r):** {-2.0, -1.5, -1.0, -0.5, 0.0, 0.5, 1.0, 1.5, 2.0}
- **Cost parameter (C):** {0.125, 0.25, 0.5, 1.0, 2.0, 4.0, 8.0}

The optimal values identified were:

- **Cost (C) =** 0.5
- **Gamma (γ) =** 0.25
- **Coef (r) =** -0.5

Data Analysis

Models

To get the results, comparisons were made between the model's predictions of the testing data and the actual values for the testing set.

The accuracy rate and confusion matrix were the primary performance metrics used for this project. Additionally, sensitivity and specificity were also used in specific cases. Please note that because there are 3 categories in the data the Confusion Matrix has 3 rows and 3 columns, where the diagonals reflect the number of correct predictions for that category.

Support Vector Classifier

Model Performance and Results

The trained SVC model was evaluated on the test data using a confusion matrix, which measures classification performance across different categories.

Accuracy

The SVC model's overall accuracy was 80%, meaning that it correctly classified 80% of all test cases.

Confusion matrix

Table 2: Confusion matrix for SVC

	Admit	Border	Not Admit
Admit	4	0	0
Border	1	5	2
Not Admit	0	0	3

From the matrix, we can observe:

- Admit: The model correctly classified 4 out of 4 applicants, accurately predicting accepted students.
- Borderline: The model correctly classified 5 out of 8 cases but misclassified 1 as Admit and 2 as Not Admit. This suggests that borderline applicants may not be as clearly separable.
- Not Admit: The model correctly classified 3 out of 3 applicants.

Sensitivity and Specificity

	Admit	Border	Not
Sensitivity	80%	100%	60%
Specificity	100%	70%	100%

Table: Statistics by class for SVC

Sensitivity measures the proportion of actual positives that were correctly identified, while specificity measures the proportion of actual negatives that were correctly identified.

- The Admit class had an 80% sensitivity, meaning that 80% of actual "Admit" cases were correctly classified as "Admit". Its specificity was 100%, indicating that all non-"Admit" cases were correctly identified as either "Border" or "Not Admit."
- The Border class had the highest sensitivity (100%), meaning all actual "Border" cases were correctly classified. However, its specificity was 70%, meaning that 30% of non-"Border" cases were misclassified as "Border".

- The Not Admit class had a 60% sensitivity, meaning that only 60% of actual "Not Admit" cases were correctly classified. Its specificity was 100%, meaning that all non-"Not Admit" cases were correctly identified as either "Admit" or "Border".

The patterns identified in the confusion matrix and the statistics-by-class table are further visualized in the plot below, providing a clearer representation of model performance across different classes.

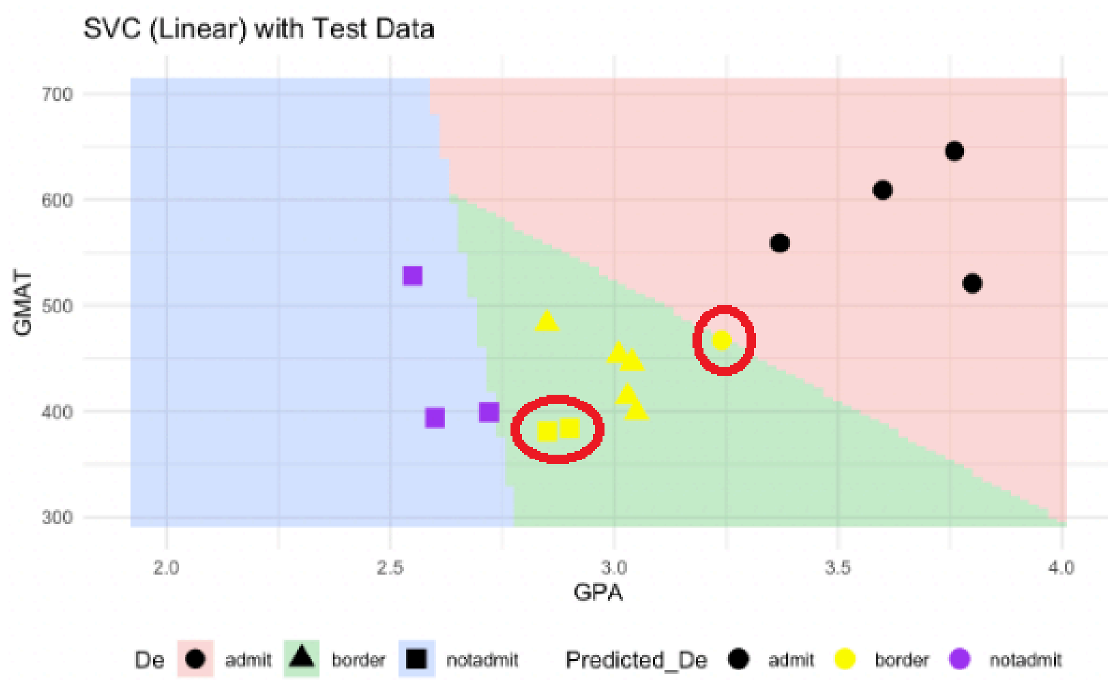


Figure 1.3 Relationship between GPA, GMAT scores using SVC

Note that the red circles indicate the instances where the model made classification errors for each class.

Support Vector Machines

Results for SVM with Polynomial Kernel

As mentioned previously the project required initial results for degree (d) = 2. However higher degrees were also analyzed in this section. The result as a follows:

Polynomial kernel (degree 2) Analysis

Accuracy

The overall accuracy rate of the model was 46.67%, meaning that the model correctly classified approximately 47% of the instances in the test set. While this performance is better than random guessing for some classes, it is not an accurate method for predicting classes.

Confusion Matrix

Table 3: Confusion matrix for SVM polynomial kernel (degree 2)

		Actual		
		Admit	Border	Not
Predicted	Admit	2	0	5
	Border	1	5	0
	Not	2	0	0

From the matrix, we can observe:

- The model correctly classified 2 instances of "Admit" but misclassified 5 instances of "Not Admit" as "Admit."
- The model performed well in predicting the "Border" class, correctly classifying 5 instances while misclassifying only 1 instance.
- The "Not Admit" class was poorly classified, with all instances being incorrectly predicted as either "Admit" or "Border."

Sensitivity and Specificity

Table 4: Statistics by class for SVM polynomial kernel (degree 2)

	Admit	Border	Not
Sensitivity	40%	100%	0%
Specificity	50%	90%	80%

- The Admit class had a 40% sensitivity, meaning that only 40% of actual positive "Admit" cases were correctly classified. Its specificity was 50%, indicating that half of the negative "Admit" cases were correctly identified.
- The Border class had the highest sensitivity (100%) and specificity (90%), showing that the model was highly effective at identifying "Border" cases.

- The Not Admit class had 0% sensitivity, meaning that none of the actual "Not Admit" cases were correctly classified, although its specificity was 80%, meaning the model was able to correctly recognize instances that did not belong to this category.

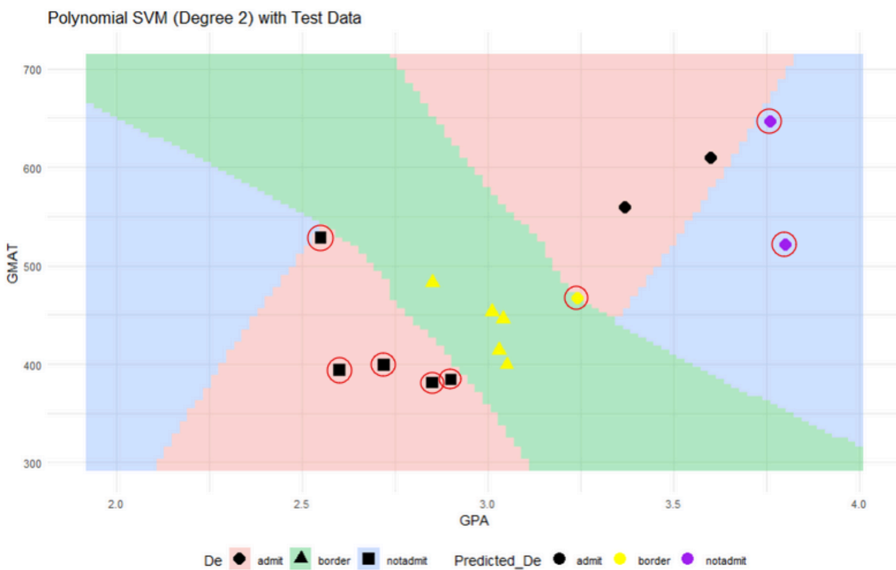


Figure 1.4 Relationship between GPA, GMAT scores using Polynomial (degree 2)

Polynomial Kernel (Degree k) Analysis

Accuracy

The overall accuracy rate of the model was 93.33%, meaning that the model correctly classified approximately 93% of the instances in the test set. This indicates a substantial improvement in classification performance compared to polynomial degree = 2 model.

Confusion Matrix

Table 5: Confusion matrix for SVM polynomial kernel (degree k)

		Actual		
		Admit	Border	Not
Predicted	Admit	4	0	0
	Border	1	5	0
	Not	0	0	5

From the matrix, we can observe:

- The model correctly classified 4 instances of the "Admit" class, with no misclassification.
- The "Border" class was well-predicted, with 5 correctly classified instances, though one instance was misclassified as "Admit."
- The "Not Admit" class was perfectly classified, with all 5 instances correctly predicted.

Sensitivity and Specificity

Table 6: Statistics by class for SVM polynomial kernel (degree k)

	Admit	Border	Not
Sensitivity	80%	100%	100%
Specificity	100%	90%	100%

Sensitivity measures the proportion of actual positives that were correctly identified, while specificity measures the proportion of actual negatives that were correctly identified.

- The Admit class had an 80% sensitivity, meaning that 80% of actual "Admit" cases were correctly classified. Its specificity was 100%, indicating no false positives for this class.
- The Border class had the highest sensitivity (100%) but a slightly lower specificity (90%), showing that while all actual "Border" cases were correctly classified, there were some false positives.
- The Not Admit class had 100% sensitivity and specificity, meaning that the model perfectly identified all instances belonging to this category.

The patterns identified in the confusion matrix and the statistics-by-class table are further visualized in the plot below, providing a clearer representation of model performance across different classes.

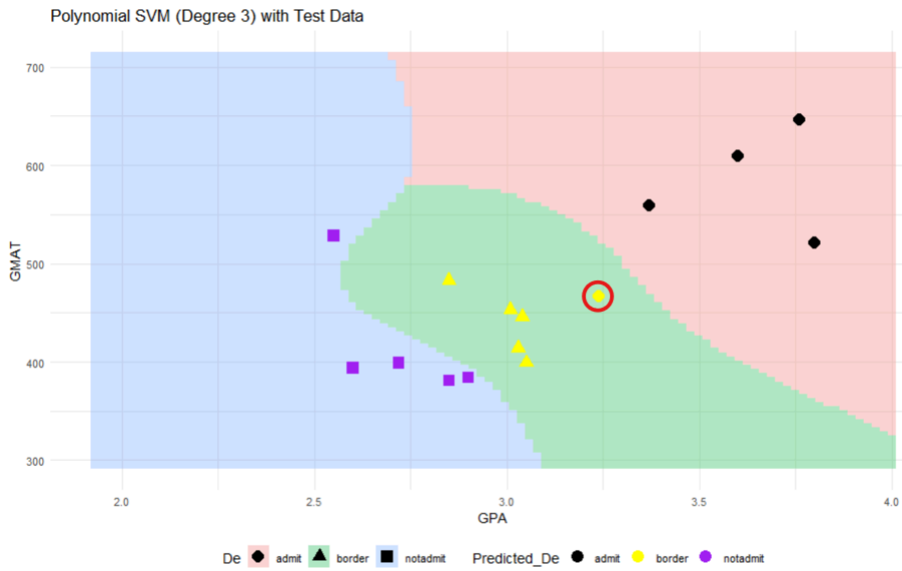


Figure 1.5 Relationship between GPA, GMAT scores using Polynomial (degree k)

Note that the red circles highlight the model's prediction errors for each class.

Support Vector Machine with Radial Kernel

Accuracy

The overall accuracy of the SVM model with the Radial Basis Function (RBF) kernel was 80%, correctly classifying 12 out of 15 test cases. This indicates that the model performed well in distinguishing between applicant categories based on GPA and GMAT scores.

Confusion Matrix

The confusion matrix for the RBF kernel SVM is presented below:

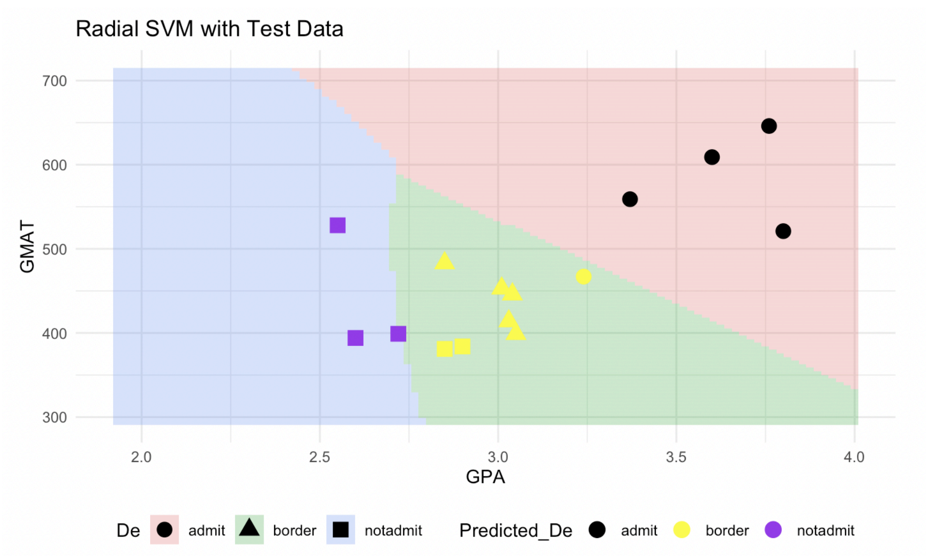
Table 7: Confusion matrix for RBF kernel for SVM

	Admit	Border	Not Admit
Admit	4	0	0
Border	1	5	2
Not Admit	0	0	3

Key Observations:

- Admit Class: The model accurately classified all 4 applicants in this category, demonstrating strong performance in identifying admitted students. However, there was one false positive in the Borderline class.
- Borderline Class: The model correctly identified 5 out of 8 cases. However, 1 instance was misclassified as Admit, and 2 instances were misclassified as Not Admit, suggesting some overlap in decision boundaries for borderline applicants.
- Not Admit Class: The model correctly classified all 3 cases in this category, demonstrating strong performance in identifying non-admitted applicants. However, it showed weaker performance in correctly distinguishing other classes from Not Admit.

A decision boundary plot was used to visualize how the model classified applicants based on GPA and GMAT scores. It highlights the regions assigned to each category and shows where test points landed, helping to identify patterns and misclassifications.



The RBF kernel SVM effectively classifies admitted and non-admitted applicants, with its decision boundary closely following the data distribution. However, the model struggles with borderline cases, leading to some misclassifications. This suggests that GPA and GMAT scores alone may not fully capture the distinctions necessary for precise classification in this category.

Results for SVM with Sigmoid Kernel

Accuracy

The overall accuracy of the model was 80%, meaning that the model correctly classified 80% of instances in the test set.

Confusion Matrix

Table 8: Confusion matrix for SVM sigmoid kernel

Predicted \ Actual	Admit	Border	Not
Admit	4	0	0
Border	1	5	2
Not	0	0	3

From the confusion matrix, we observe:

- The model correctly classified 4 instances of "Admit," but misclassified 1 instance of "Border" as "Admit."
- The "Border" class was the most accurately predicted, with 5 correctly classified instances, though 2 instances were misclassified as "Not Admit"
- The "Not Admit" class was also well classified, with 3 correctly predicted instances and no misclassifications.

Sensitivity and Specificity

Table 9: Statistics by class for SVM sigmoid kernel

	Admit	Border	Not
Sensitivity	80%	100%	60%
Specificity	100%	70%	100%

Sensitivity measures the proportion of actual positives that were correctly identified, while specificity measures the proportion of actual negatives that were correctly classified.

- The Admit class had an 80% sensitivity, meaning 80% of actual "Admit" cases were correctly identified, with 100% specificity, indicating no false positives for this class.
- The Border class had 100% sensitivity, meaning all actual "Border" cases were correctly classified, though its specificity was lower (70%), suggesting some false positives.

- The Not Admit class had a 60% sensitivity, indicating that 60% of actual "Not Admit" cases were correctly classified, but with 100% specificity, meaning there were no false positives for this class.

This performance suggests that while the sigmoid kernel effectively captured the patterns in the data, it had some difficulty distinguishing between "Border" and "Not Admit" classifications. The patterns identified in the confusion matrix and sensitivity-specificity statistics are further visualized in the plots below, providing a clearer representation of model performance across different classes.

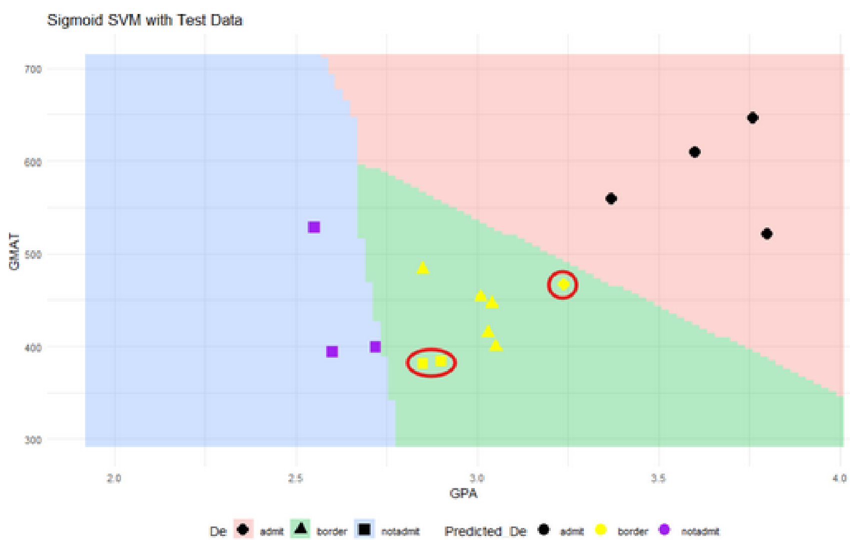


Figure 1.5 Relationship between GPA, GMAT scores using sigmoid kernel

Note that the red circles highlight the model's prediction errors for each class.

Conclusions and Recommendations

This study explored different Support Vector Machine (SVM) models to classify business school Applicants into Admit, Borderline, and Not Admit categories. The models testes included Support Vector classifier (SVC), SVM with a Polynomial kernel, and SVM with a Radial Basis Function (RBF) Kernel. The goal was to determine the most effective approach for predicting admission based on GPA and GMAT scores.

The results showed that SVC and RBF SVM both achieved 80% accuracy, meaning they performed equally well in classifying applications. However, SVM with a Polynomial Kernel (Degree 2) had the lowest accuracy at 46.67%. Showing us that a quadratic decision Boundary was not a good fit for this dataset. Since RBF SVM is used with more complex data, SVC is the preferred model as it provides the same level of accuracy with lower complexity.

We also tested two additional models to determine if they would improve applicant classification. The Sigmoid SVM achieved the same accuracy as the SVC and RBF (80%), but its decision boundary was less clear, causing misclassifications among borderline applicants. The Polynomial SVM (Degree 3) performed best overall with an accuracy of 93.33%, correctly classifying all Not Admit cases.

Given these results for what was outlined in our project, the SVC model is still the best choice because it gives good accuracy (80%), is easier to understand, and works efficiently. However, if we were to consider other kernels it seems that the Polynomial (Degree 3) model outperforms all other models. Meaning that the data is best separated with a cubic curve. This is because it has even higher accuracy and a high sensitivity and specificity across all classes.

References

1) (ISL pg. 385)
2) https://www.youtube.com/watch?v=ZvaELFv5IpM&t=15s
3) https://papers.ssrn.com/sol3/papers.cfm?abstract_id=5085740
4) (ISL pg. 368,390)
5) (ISL pg. 382)
6) https://www.youtube.com/watch?v=Toet3EiSFcM

7https://www.geeksforgeeks.org/how-to-adjust-the-degree-parameter-for-a-polynomial-kernel-in-svm)
8) https://scikit-learn.org/stable/auto_examples/svm/plot_svm_kernels.html
9)https://medium.com/analytics-vidhya/introduction-to-svm-and-kernel-trick-part-1-theory-d990e2872ace