Machine learning with Neo4J databases to identify at risk students and provide curriculum advice for first year CS students

Edwin Kassier
Department of Computer Science
University of Cape Town
Cape Town, Western Cape, South Africa
KSSRUB001@myuct.ac.za

Abstract

This literature review investigated the relationship between the academic performance and socio-demographic characteristic of the South African Computer Science student population in order to determine the variables that need to be considered when trying to identify or predict the probability of them extending their academic programme beyond the formally allocated number of study years, or terminating their studies altogether. It also provides an overview of how these at-risk students can be identified through use of Machine Learning (ML) systems integrated with a Neo4J database.

Factors that affect the academic performance of South African university students the most, are seemingly language proficiency, school background and socio-economic status. In addition to further investigation as to why these factors have such a significant impact on student academic performance, it is recommended that future datasets used for training of ML systems should include these variables as omission of them, could lead to ML systems providing ineffective predictions.

An overview of the current ML models that could be used as a system for effective risk analysis amongst these students included Artificial Neural Networks, Decision Trees, Support Vector Machines and Bayesian Networks. The review revealed that for the purpose of risk analysis, Support Vector Machines and Artificial Neural Networks are relatively unsuited for the task in a student analysis context, despite their prediction accuracy across empirical tests being higher than that of the tested Bayesian Network and Decision Tree systems. This is simply due to their inability to provide contextual output that could aid in further remedial recommendations of students experiencing academic difficulties alongside with the complexity of additional frameworks needed to mitigate these issues.

Apart from the paucity of data regarding the application of student academic risk factors in the integration of a ML system with a Neo4j database, Neo4j databases have also been shown to have increased levels of performance over their SQL counterparts that is directly related to an increase in the size of the data base.

Keywords

big data, graph databases, graph algorithms, machine learning

1. Introduction

Machine Learning (ML) has recently been hailed as a figure head of computing, with the public using the terms Artificial Intelligence (AI) and ML interchangeably as an umbrella term to describe the act of computers learning/predicting and sometimes acting on these predictions to miraculous effect. This is a misnomer; and to achieve a better understanding in this review the term "Machine Learning" must be formally defined.

ML is the process of a computer utilising a set of algorithms to develop some form of statistical model from a given dataset, learning the relationship between explanatory variables, then using this "learnt model" to make predictions on a given input into the system [20]. ML is therefore a subset of AI (being a broader system that seeks to simulate an intelligence that can perform tasks associated with human minds), where AI can partially describe the workings of ML, but ML doesn't describe AI [20].

This literature review aims to investigate the use of ML tools that enable tertiary education institutions to identify students at risk of not completing their degrees in accordance with the degree's specified time frames, if at all. The reasons why students would be at risk include projected performance issues, not completing the degree with the required credits due to insufficient course credits or course clashes.

Hence the scope and focus of this literature review will be to provide an overview of the multifactorial nature of why students drop out of university, how these trends can be analysed using ML, which ML method is optimal for this task and how ML algorithms can be integrated with graph databases.

2. Attrition rates in tertiary education

Attrition in this context can be described as the process of students dropping out of their academic programmes while at the tertiary education level. Attrition rates amongst university students have always been a source of concern for tertiary education institutions, as they represent not only a loss to the future workforce, but also a loss to the institutions themselves due to loss of future revenue, university rankings and reputation [17].

In this section, the quantifiable causes for high dropout rates amongst the South African student population is explored, allowing for the construction of a future dataset that will result in the most accurate predictions for risk analysis.

2.1 Reasons for attrition

The reasons why students extend the number of years to complete a degree beyond the formally allocated number of study years our drop out of it altogether, are often as varied as the students themselves. However, by reviewing past studies, commonalities can be found, thereby acting as potential data points for future data analysis.
In a study conducted in 2010 at Northwest University (NWU) [8], the first year cohort of degrees with a high attrition rate such as Information Technology, Mathematics and Computer Science were investigated. The purpose of the study was to identify criteria that can be used to identify students that would benefit from being enrolled in the extended degree programme of their respective degrees.

It was found that there were three primary risk factors, not associated with student marks, that predisposed students to performing sub-optimally at a tertiary level. These reasons were noted by the authors in their own literature review as well as being corroborated by their own research [8].

While other studies have been conducted in this area and identified additional factors that contribute to student attrition or extension of the number of years it took to complete the qualification beyond defined degree completion time frames, they were not relevant to the context of South African Computer Science (CS) students and were thus not included in the review.

2.1.1 Language proficiency

English is the language of instruction at most tertiary education institutions in South Africa. In addition, it is the primary language of almost all entry tests used to screen prospective candidates for suitability to enrol at tertiary institutions. However, it is only the fourth most spoken home language in South Africa (SA), according to the SA 2011 census data [18]. Due to the latter, higher education institutions seem to predispose certain students to failure as there has been strong evidence of a qualitative difference in the academic abilities between first and second language English speakers in an English learning environment [9]. This evidence has been corroborated by a Language of Instruction in Tanzania and South Africa (LOITASA) research project conducting a study that documented how a wider spread of test result scores when a cohort of African home language speaking students were forced to write their tests in English as opposed to their home languages [5].

2.1.2 Schooling background

A student's schooling background forms an important part of their tertiary education success, as their previous schooling not only provides an indication of their academic abilities but is also an indication of their exposure to advanced course material, high levels of quality instruction and the curricula under which they studied. All of which are important points for a university to consider when gauging new, or at risk, students [4].

An example of how schooling background has an impact on tertiary education success, is the seeming disconnect between levels of competency of those who matriculated from public versus private schools upon entry into university, with private school educated students in historically white and Indian schools outperforming their counterparts in historically black and coloured public schools [4]. This divide in ability is especially relevant to tertiary education performance in computer related degrees when considering the higher likelihood of students from previously advantaged schools acquiring computer training and exposure to existing and emerging technologies and skills over their disadvantaged counterparts, often referred to as the 'digital divide'[3].

While this phenomenon is in the process of changing due to new government initiatives and funding changes, the impact of a student's educational background remains a powerful predictor of their future academic performance and is therefore relevant as a variable to be considered in a risk analysis study.

2.1.3 Socio-economic status

Many prospective university students from lower socio-economic backgrounds often view higher education as a route to escape poverty. However, while this is the prevailing sentiment, it is a double-edged sword in that each new cohort embarking on a degree may contain many students who while achieving the necessary marks for admission, do not have the necessary background or inclination to achieve in the degree they have been admitted to. The degree is often viewed as a vehicle to escape poverty or their current financial status [14].

Subsequently, the lack of financial resources to self-fund their studies, often means that students are relying on NSFAS funding and student housing for the duration of their university career. It has been shown that there is a strong correlation between students on NSFAS funding and a high risk of dropping out from undergraduate studies

[14]. The cited reason for this phenomenon, is that students who hail from a lower socio-economic background, are often forced to take part-time jobs to help finance the resources they need to achieve in and hence complete their degrees. Areas that NSFAS funding is unable to cover, include computers, textbooks and additional tuition. In addition, socio-economically disadvantaged students are often forced to seek part-time employment to support themselves as well as their families. These additional stressors are likely to be the driving force behind NSFAS funding being viewed as a proxy risk factor for an extended study period or non-completion of degrees by NSFAS recipients.

3. Machine learning prediction models for risk classification

As noted previously, ML relies on building statistical models from large datasets that allows the learnt model to predict a given outcome given a set of training input [20]. This section of the literature review focuses on the current prediction models used in risk analysis and risk classification, how they function, how they differ, and which model/s is/are most reliable in attaining accurate predictions for risk analysis.

3.1 Artificial Neural Networks

Artificial Neural Networks (ANNs) describe a broad class of Neural networks, including: Multilayer Perceptron's (MLP), Convolutional Neural Networks (CNN) and Recurrent Neural Networks (RNN) to name but a few. They are a type of graphical statistical model that have recently received great attention from the academic and corporate communities for their varied uses. ANNs in laymen's terms are designed to function not unlike a neuron in the human brain, as they receive an input into what is called the input layer (with each node representing a type of input variable). Subsequently, the input is passed into what is referred to as the hidden layer (which is often multiple layers of interconnected nodes), where each layer extracts different pieces of information from the dataset and passes along its result, eventually leading to an output layer where the result may be interpreted by the algorithm. A major reason for its success, is the fact that in its graphical representation where the connections between nodes represent the relationships between variables, an ANN may discover the significance of these relationships through its learning process, thereby leading to high rates of prediction accuracy where especially complex relationships are present [2].

ANNs rely on two concepts, referred to as back propagation and gradient descent in its learning process, where labelled data (for example a student where it is known that they failed their degree) is used in a training data set letting the neural network "corrects" itself if the prediction made was incorrect. This triggers the neural net to change how the neurons in the hidden layer react to data. However, this is often computationally expensive. Meaning that it sometimes becomes impractical for especially difficult tasks that would require large datasets and long training times but does result in neural networks that are highly efficient at recognising patterns when given data sets provide enough data for the neural net to effectively learn from.

However, there is one significant caveat to ML with ANNs for risk analysis, as it functions like a black box. This means the researcher/user using it can only tweak the input they feed into the system and have no agency in the decision-making process. Nor do they have any insight into the relationships caused a specific outcome [2]. This is undesirable in cases where the causes for an outcome need to be known before remediation can take place (e.g. poor maths marks require extra maths lessons).

3.2 Decision Tree Learning

Decision tree learning, and more specifically Classification and Regression tree known together as Classification and Regression Tree (CART) learning, show promise as a risk classification technique. They function on the principle of Search Trees where an often binary (yes or no) decision dictates how one would traverse through the tree to the leaves which represent the decision of the tree given some set of inputs.

Classification trees as a ML technique have their roots in academic literature as early as 1969, where programs to perform the splitting of decisions to build decision trees were being investigated [25]. Recent studies, especially in the field of medicine, have shown strong evidence that classification trees can perform effective predictions for triage-like tasks of at-risk in-patients using a variety of numerical and non-numerical data as features for decision making [10].

However, while they have been shown to be effective in risk classification tasks, there is also evidence that there is a complex relationship between the types of algorithms used for tree construction and the subsequent prediction accuracy of the algorithm. The GUIDE algorithm results in the highest prediction accuracy and the RPART algorithm results in the lowest across a set of published empirical studies [16]. The reasons for these inaccuracies, are cited as the algorithms performing analysis on training data sets that lend themselves to overfitting due to noisy datasets or datasets with increased levels of entropy (or impure data) that do not have the correct amounts of data to satisfy reliable prediction in the test set [16].

3.3 Support Vector Machine

Support Vector Machines (SVMs) are a family of machine learning algorithms that use similar principles to ANNs with certain key differences that allow them to perform at higher levels of accuracy, even with smaller training sets than those required of its ANN contemporaries [26].
SVMs function on the mathematical principle of separating two types of data by an intervening "hyperplane", using this hyperplane as an established

boundary between the two data sets to classify input data into whichever side of the hyperplane it would lie on. This is called linearly separable data [13].

As an example, we might want to create an image classifier that tells the difference between dogs and cats using snout length and ear geometry as its features to classify the images with. Then, using the generated hyperplane, the machine will give a definitive answer depending on the snout length and ear geometry from the given input and what side of the hyperplane the example lies on. But what about cats that have been groomed to look like dogs or vice versa? A SVM will create margins around its hyperplane called the hyperplane margin that is optimised to deal with extreme cases that allow it effectively classify input data, especially those that might be difficult for other classifying algorithms. In cases where the addition of features is desirable to more effectively classify an input, using the cat and dog classifier as an example, the SVM will need to map the new feature to a higher dimension to ensure the data continues to be linearly separable. In this case, the 2d dataset will be mapped into a 3d one. This process is performed by kernel functions and is often computationally expensive, as the more factors used in the classification process, the more mappings will need to be performed. Thus increasing the complexity of the data set and the intervening hyperplane being calculated [26]. This becomes increasingly problematic for especially large datasets with large amounts of explanatory variables. However, there is evidence that clustering many SVMs together into a Clustered SVM (CSVM) can mitigate, but not solve, this problem [12].

While SVMs show promise as a risk classifier, their inherent flaw is that while they function effectively, they do not provide probabilities for their predictions, only the classification of the input. While this is desirable in most classification tasks, it is problematic when building a risk classifier where percentile thresholds are an important distinction when trying to decide on the level of action to be undertaken for the specific level of risk. However, it has been shown that this can be circumvented by the use of a hybrid model where sensitivity analysis can help identify the explanatory variables affecting the output most [26].

3.4 Bayesian networks

At the functional level, Bayesian Networks (BNs), unsurprisingly, classify input at the lowest level using the following theorem (Bayes theorem) as a classification algorithm [21].

$$P(A|B) = \frac{P(B|A)*P(A)}{P(B)}$$

What this theorem explains is the probability of an event A occurring given some event B.
A BN in its truest sense, functions as a Directed Acyclic Graph (DAG) where one node exists per random variable of the dataset, with each node containing a separate conditional probability table that catalogues the probabilities of its occurrence depending on all outcomes from its parent nodes. For example, if there are three variables in the DAG that all have the possibility of being true or false: Grass is wet, raining and the sprinkler system is on. “Raining” and “Sprinkler system is on” are parent nodes to Grass is wet as they have the potential to affect the probability of it being true or false, depending on their own values. This means that grass is wet will have a conditional probability table cataloguing its own probability of being wet, depending on these variable’s values. This classification of relationships between edges is only possible in a select number of simple problems where an expert could reliably provide the relationships between nodes. BNs then need some mechanism for, reliably, learning the relationships between large numbers of nodes to ensure reliable prediction for future questions in complex datasets. BNs perform learning by a process called inference, where, broadly speaking, the algorithm will infer the relationships between nodes to establish how the conditional probabilities between nodes should change depending on the training data [6].

There are many types of inference algorithms. In a study performed in 2006, some of the most effective inference methods of the time were measured against a newly invented Max-Min Hill-Climbing (MMHC) algorithm for BN learning. The study compared the performance of the algorithms with training sets of 500, 1000 and 5000 values for each algorithm, averaging results over five tests [22]. The following table summarises the results for Bayesian score tests for a set of inference algorithms.

Table 1. Summarised comparison of Bayesian network learning methods normalised relative to the MMHC method for Bayesian score results [22]

Algorithm	**SS=500**	**SS=1000**	**SS=5000**	**Mean**
MMHC	1.000	1.000	1.000	1.000
OR1 k = 5	1.016	1.019	1.021	1.019
OR1 k = 10	1.014	1.016	1.023	1.018
OR1 k = 20	1.016	1.022	1.022	1.020
OR2 k = 5	1.005	1.009	1.011	1.008
OR2 k = 10	1.000	1.009	1.003	1.004
OR2 k = 20	1.009	1.010	1.006	1.009
SC k = 5	0.999	0.996	1.016	1.004
SC k = 10	1.010	1.016	1.014	1.013
GS	0.976	0.983	0.991	0.984
PC	1.275	1.234	1.195	1.235
TPDA	1.406	1.367	1.206	1.326
GES	1.083	1.007	1.118	1.070

K=maximum allowed size for the candidate parents' sets

Inference algorithms tested

1. Sparse Candidate (SC), (Friedman, Nachman & Pe'er, 1999)
2. PC (Spirtes, Glymour & Scheines, 2000)
3. Three Phase Dependency Analysis (TPDA), (Cheng et al., 2002)
4. Optimal Reinsertion (OR) (Moore & Wong, 2003)
5. Greedy Hill-Climbing Search (GS, using the BDeu scoring)
6. Greedy Equivalent Search (GES), (Chickering, 2002)
7. Max-Min Hill-Climbing (MMHC, novel hybrid algorithm, (Brown, Tsamardinos & Aliferis, 2004)

The results of the Bayesian score tests, being a measure of prediction efficacy across multiple Bayesian networks, show that on average, the MMHC algorithm performed at the same, if not better, level as the Greedy Search, Optimal Reinsertions (types 1 and 2) and Sparse Candidate algorithms, with the other constraint-based algorithms performing worse. While these results are in favour of the MMHC algorithm, it must be noted that the authors of the study invented the algorithm and were thus likely biased in the types of tests performed to show the algorithm in a more positive light. While this research is relatively out of date, in terms of the rate of improvement in computer science, it is useful in analysing the various mechanisms used for inference and which types of mechanisms have been proven to be effective.

A popular counterpart of the BN is a naïve Bayesian classifier. A popular use of a naïve Bayesian classifier is its use in spam filters. The following example also provides a relatively simple example to facilitate an understanding of how a naïve classifier would works, and why it is considered naïve: There are 100 emails in the training set where the probability of an email being classified as spam is being determined by checking whether the email contains the words "buy" or "cheap" with five emails containing the word "buy" and ten containing the word "cheap", while none contained both, the implication thereof is that the probability of an email containing either word is 10% and 5% respectively for use in future predictions. However, when asking such a system what the probability of an email containing both the words "buy" and "cheap" would be, the system performs a simple operation calculating the value of 10% of 5%. The resultant outcome would be that the probability of an email containing both keywords is 0.5%. This illustrates why the classifier is considered naïve, as it assumes independence between its probability parameters when it could be that the word "buy" increases the probability of the word "cheap" being present within the same email. This is however beneficial when dealing with small training sets where no emails exist that contain both keywords, allowing the algorithm to relatively effectively fill in any gaps in the training data.

In a study conducted by Antonakis and Sfakianakis in 2009 to identify at-risk credit applicants in Greek banks using a naïve Bayesian classifier and a Convolutional Neural Network in head-to-head tests, it was found that in terms of performance, while naïve Bayesian classifiers are less complex and often on average perform worse than their Convolutional Neural Network counterparts in terms of prediction accuracy, the difference in performance was statistically insignificant [1]. This sentiment is validated by other studies suggesting that even with the simplicity of the method, the classifiers produced by naïve Bayesian systems are often more effective than their more complex counterparts in terms of the ratio between computational power expended and the resultant accuracy achieved [11]. However, these studies also noted that while naïve Bayesian classifiers can outperform ANNs, such as the CNN mentioned previously, the level of performance for the given CNN could be improved to levels far above and beyond those of the naïve Bayesian classifier by implementing an enhanced gradient decent method for the CNN, using a genetic algorithm or a quadratic descent formula that not only found the local minima for the CNNs learning graph, but its global maxima as well. This optimised CNN, while being a better option for prediction accuracy, is significantly more computationally expensive to train/optimise than its naïve Bayesian counterpart [1].

4. Efficacy comparison of current prediction models for risk classification

Apart from knowing what the different ML models for risk classification are, it is important to know their performance in empirical tests on real world examples. The following is a summary of the tests conducted by Vafeiadis et al. to perform risk analysis on the attrition probability of telecommunication company customers (called the "churn" of the company). The dataset used for training contained 5000 samples of anonymised customer data, with each sample containing data for 19 separate variables that included both numerical and binary data [23]. The following table summarises the prediction efficacy of the algorithms as well as the relative training costs of each algorithm, in terms of computational power expended.

Table 2. Summarised comparison of machine learning method efficiency in churn risk analysis [23]

Machine learning type	**Learning method**	**TC***	**PA****
ANN (Artificial Neural Network) -optimised	Backpropagation with gradient descent	High	94%
DTL (Decision tree learning) -optimised	Construction of optimised splits within the tree (GUIDE, RPART, etc.)	High	94%
SVM (Support Vector Machine) -optimised	Hyperplane construction w/kernel functions	High	93%
NBC (Naïve Bayes Classifier)	Bayes function on training datasets	Low	86%

* Training Cost
**Prediction accuracy

Despite being a useful comparison for the purposes of this review, it is important to note that the study design in the above study was flawed. It unevenly assigned computational power amongst its methods, meaning that if there was an existing framework that increased the learning efficacy of any of the techniques in its baseline tests, it was applied without consideration of the computational power it required and without any optimisations being performed on the decision-making processes underlying these techniques. Additionally, this table showcased the performance of a Naïve Bayes Classifier and not a full Bayesian network with the current optimised inference algorithms of the time. As yet, there is not sufficient literature to showcase direct performance results of a full Bayesian network against its ML peers.

5. Integration of machine learning models with a graph database (Neo4J)

First proposed in 1970 by Edgar Codd of IBM's research laboratory, relational databases have been a mainstay for computer storage for decades. They function as a large table where using the tables query language, known as Standard Query Language (SQL), allows information to be retrieved at the large or granular level [7]. Being the main method of data storage and retrieval in computers has made relational databases the main method of information storage and retrieval for ML methods as well. Graph databases on the other hand, and more specifically those of the graph database system Neo4J, follow in the footsteps of the relatively recent NoSQL trend. They store information in a series of Directed Acyclic Graphs (DAG) in an attempt to overcome the inefficiencies that large relational databases incur due to the high number of potential join operations within the database [15].

In an empirical study regarding the performance differences between 12 separate Neo4J and MySQL databases, it was shown that relational databases are best suited to storing and querying relatively small databases, with Neo4J performing only slightly slower in the small-scale tests [24]. However, as the size of the databases being used were scaled up, there was a dramatic change in the levels of performance between the two database types, with Neo4J performing queries ten times faster (in some cases) than its MySQL counterparts [24].

It would seem that the current body of evidence, does not mention any specific challenges related to a ML algorithm, let alone ML in general that could be experienced when integrating graph databases into the ML workflow. There is only mention of the increased efficacy a graph database brings to data mining and machine learning due to the reasons described above [19].

6. Discussion

What main aspects highlighted by this literature review includes the fact that when trying to predict whether Computer Science students are likely to complete their studies within the formally allocated time frame, if at all, does not entail a mere analysis of their academic performance in terms of marks obtained [4]. For ML systems to function effectively as a risk analysis tool in the context of South African students, the ML system must take under consideration the socio-demographic background of the user as well as their academic performance, as the socio-demographic characteristics of students represents a significant number of variables for consideration in risk analysis and prediction of future academic performance[17][8][4].

The ML algorithms presented are adequate for conducting risk analysis among students, given that there are no limitations on adding additional complexity to the systems [23]. However, at a basic level for assessing the needs of the risk analysis tool proposed, available evidence suggests that Decision Trees and Bayesian Networks provide a balance between usability and efficiency needed. However, due to the lack of academic material cataloguing further tests on pure Bayesian Networks against Decision Tree learning prediction efficacy, these two methods warrant further study into their efficacy in head to head tests using varying training set sizes across the various algorithms used in their learning process.

Graph Databases show promise as a database for ML as evidence shows a large increase in performance of Neo4j over its SQL counterpart in large scale databases likely to be used in the training of ML systems[24]. Especially due to limited evidence that their use causes any hinderance to the workflow of a ML system or integration between the system and its database.

7. Conclusion

Risk analysis then is a multiple faceted problem, with training data of the correct size and composition playing an essential role, alongside with the Machine Learning

method under consideration, in achieving high rates of effective prediction accuracy.

In relation to the investigatory questions posed in the introduction, it has been shown that there is a significant amount of evidence linking the socio-demographic background of a student and the effect these factors could have on their ability to complete their academic programmes in the recommended time, if at all [8][5][4]. This being a departure from the approach assuaging only academic performance being the basis for risk prediction. Consequently, it is recommended that any future systems attempting to perform risk analysis on student populations take into consideration the complex relationships that could arise due to a user's socio-demographic background when both collecting training data and passing it to the Machine Learning system.

In terms of analysis of such data it was seen that the use of a Bayesian Network or Decision Tree method would provide the desired prediction accuracy alongside with granular output that would give users the ability to not only receive their risk probability, but also understand what factors are contributing to their risk [10][21][23]. Additionally, these methods are prime candidates due their ability to provide the aforementioned benefits without the need for external frameworks being placed on top of them. The most efficient candidate between the two warrants further research. Due to a lack of academic material to the contrary alongside with other reinforcing data it is unlikely that the use of a Neo4j database in a Machine Learning system will have any caveats involved in either integrating or using it [15].

References

[1] Antonakis, A.C. and Sfakianakis, M.E. 2009. Assessing naïve Bayes as a method for screening credit applicants. *Journal of Applied Statistics*. 36, 5 (May 2009), 537–545. DOI:https://doi.org/10.1080/02664760802554263.

[2] Ayer, T. et al. 2010. Comparison of Logistic Regression and Artificial Neural Network Models in Breast Cancer Risk Estimation. *RadioGraphics*. 30, 1 (Jan. 2010), 13–22. DOI:https://doi.org/10.1148/rg.301095057.

[3] Barlow-Jones, G. and van der Westhuizen, D. 2011. Educational Studies Situating the student: factors contributing to success in an Information Technology course Situating the student: factors contributing to success in an Information Technology course. *Educational Studies*. 37, 3 (2011), 303–320. DOI:https://doi.org/10.1080/03055698.2010.506329.

[4] Van Der Berg, S. 2008. How effective are poor schools? Poverty and educational outcomes in South Africa §. *Van der Berg, S., 2008. How effective are poor schools? Poverty and educational outcomes in South Africa. Studies in Educational Evaluation*. 34, 3 (2008), 145–154. DOI:https://doi.org/10.1016/j.stueduc.2008.07.005.

[5] Brock-Utne, B. 2007. LANGUAGE OF INSTRUCTION AND STUDENT PERFORMANCE: NEW INSIGHTS FROM RESEARCH IN TANZANIA AND SOUTH AFRICA. *Die International Review of Education*. 53, (2007), 509–530. DOI:https://doi.org/10.1007/s11159-007-9065-9.

[6] Chen, S.H. and Pollino, C.A. 2012. Good practice in Bayesian network modelling. *Environmental Modelling & Software*. 37, (Nov. 2012), 134–145. DOI:https://doi.org/10.1016/J.ENVSOFT.2012.03.012.

[7] Codd, E.F. 1970. A relational model of data for large shared data banks. *Communications of the ACM*. ACM.

[8] Du, L. et al. 2012. Academic preparedness of students-an exploratory study. *Td The Journal for Transdisciplinary Research in Southern Africa*.

[9] Enright, M.K. et al. 2000. *Monograph Series TOEFL 2000 Reading Framework: A Working Paper*.

[10] Fonarow, G.C. et al. 2005. Risk stratification for in-hospital mortality in acutely decompensated heart failure: Classification and regression tree analysis. *Journal of the American Medical Association*. 293, 5 (Feb. 2005), 572–580. DOI:https://doi.org/10.1001/jama.293.5.572.

[11] Hand, D.J. and Yu, K. 2001. Idiot's Bayes?Not So Stupid After All? *International Statistical Review*. 69, 3 (Dec. 2001), 385–398. DOI:https://doi.org/10.1111/j.1751-5823.2001.tb00465.x.

[12] Harris, T. 2015. Credit scoring using the clustered support vector machine. *Expert Systems with Applications*. 42, 2 (Feb. 2015), 741–750. DOI:https://doi.org/10.1016/J.ESWA.2014.08.029.

[13] Joachims, T. 1998. Text categorization with Support Vector Machines: Learning with many relevant features. *European Conference on Machine Learning*. Springer, Berlin, Heidelberg. 137–142.

[14] Letseka, M. and Maile, S. 2008. *HSRC Policy Brief High university drop-out rates: a threat to South Africa's future*.

[15] Li, Y. and Manoharan, S. 2013. A performance comparison of SQL and NoSQL databases. *IEEE Pacific RIM Conference on Communications, Computers, and Signal Processing - Proceedings* (Aug. 2013), 15–19.

[16] Loh, W.-Y. 2011. Classification and regression trees CLASSIFICATION TREES. *Data Mining and Knowledge Discovery*. 1, (2011), 14–23.

DOI:https://doi.org/10.1002/widm.8.

[17] Moodley, P. and Singh, R.J. 2015. Addressing Student Dropout Rates at South African Universities. *Alternation Special Edition*. 17, (2015), 91–115.

[18] Ngyende, A. 2011. *Census 2011 Statistical release (Revised)*.

[19] Riesen, K. and Bunke, H. 2008. IAM Graph Database Repository for Graph Based Pattern Recognition and Machine Learning. Springer, Berlin, Heidelberg. 287–297.

[20] Sammut, C. and Webb, G.I. 2017. *Encyclopedia of machine learning and data mining*. Springer Publishing Company.

[21] Su, J. and Zhang, H. 2006. Full Bayesian network classifiers. *Proceedings of the 23rd international conference on Machine learning - ICML '06* (New York, New York, USA, 2006), 897–904.

[22] Tsamardinos, I. et al. 2006. *The max-min hill-climbing Bayesian network structure learning algorithm.*

[23] Vafeiadis, T. et al. 2015. A comparison of machine learning techniques for customer churn prediction. *Simulation Modelling Practice and Theory*. 55, (Jun. 2015), 1–9. DOI:https://doi.org/10.1016/J.SIMPAT.2015.03.003.

[24] Vicknair, C. et al. 2010. A comparison of a graph database and a relational database. *Proceedings of the 48th Annual Southeast Regional Conference on - ACM SE '10* (New York, New York, USA, 2010), 1.

[25] Winston, P. 1969. *A heuristic program that constructs decision trees.*

[26] Yu, R. and Abdel-Aty, M. 2013. Utilizing support vector machine in real-time crash risk evaluation. *Accident Analysis & Prevention*. 51, (Mar. 2013), 252–259. DOI:https://doi.org/10.1016/J.AAP.2012.11.027.