Mining the productivity data of the garment industry

Abdullah Al Imran* and Md Shamsur Rahim

Department of Computer Science and Engineering,
American International University-Bangladesh,
408/1, Kuratoli, Dhaka 1229, Bangladesh
Email: abdalimran@gmail.com
Email: hemel2009@hotmail.com
*Corresponding author

Tanvir Ahmed

Department of Computer Science,
University of Central Florida,
4000 Central Florida Blvd, Orlando, FL 32816, USA
Email: tanvir.ahmed@ucf.edu

Abstract: The garment industry is one of the key examples of the industrial globalisation of this modern era. It is a highly labour-intensive industry with lots of manual processes. Satisfying the huge global demand for garment products is mostly dependent on the production and delivery performance of the employees in the garment manufacturing companies. So, it is highly desirable among the decision makers in the garments industry to track, analyse and predict the productivity performance of the working teams in their factories. This study explores the application of state-of-the-art data mining techniques for analysing industrial data, revealing meaningful insights and predicting the productivity performance of the working teams in a garment company. As part of our exploration, we have applied eight different data mining techniques with six evaluation metrics. Our experimental results show that the tree ensemble model and gradient boosted tree model are the best performing models in the application scenario.

Keywords: data mining; productivity prediction; pattern mining; classification; garment industry; industrial engineering.

Reference to this paper should be made as follows: Al Imran, A., Rahim, M.S. and Ahmed, T. (2021) 'Mining the productivity data of the garment industry', *Int. J. Business Intelligence and Data Mining*, Vol. 19, No. 3, pp.319–342.

Biographical notes: Abdullah Al Imran received his Bachelor's degree in Computer Science and Engineering from the American International University-Bangladesh. He is currently working as a Data Scientist in a reputed telecommunication company in Bangladesh. His research interest includes: data science, applied machine learning, deep learning, natural language processing and affective computing.

Md Shamsur Rahim completed his BSc in Computer Science and Software Engineering and MSc in Computer Science from the American International University-Bangladesh in 2014 and 2016. Currently, he is working as an

Copyright © 2021 Inderscience Enterprises Ltd.

Assistant Professor at the Computer Science Department in the same institute. His research interest includes: data mining, data science and software engineering.

Tanvir Ahmed is a Lecturer of Computer Science at the University of Central Florida (UCF). He completed his PhD in Computer Science and Engineering from the Centre for Data-Intensive System (Daisy), Aalborg University Denmark. He received his BSc and MSc from American International University-Bangladesh (AIUB). He has several years work experience in academic, industry and research organisation. He worked as a researcher at US FDA and Universite Libre De Brussels (ULB), Data Scientist at RadioAnalyzer and faculty member at AIUB. His research interests include data management, data science and data warehousing. He served as a reviewer and a technical program committee member for various conferences and workshops. He is a senior member of IEEE.

1 Introduction

The garments industry is recognised as one of the high value-added industries in the world. In the study of Lai and Christiani (2013), it is estimated that over 60 million people are employed in the garments industry worldwide. The huge amount of employment in the garment industry plays a crucial role in the economy of some developing countries such as Bangladesh, India, Pakistan and Vietnam. Most of the work in this industry is highly labour-intensive with lots of manual operations in various departments. The production of the garments industry is a sequential process which includes a series of operations such as designing, sample confirmation, sourcing and merchandising, lay planning, market planning, spreading and cutting, sewing, washing, finishing and packaging. The whole production target and the performance of this industry highly relies on the performance of the employees.

To keep the production performance up and achieve the targeted outcome, it is essential to systematically monitor and analyse the productivity level of the employees and also the overall working performance of the teams. Unfortunately, most of the garment companies are still performing this analysis manually because of the unawareness of data mining technology and computational techniques. However, nowadays some of the companies in this industry are realising the potential of computational data mining technology for solving such analytical problems.

Over the past few decades, data mining has been empowering many sectors in industrial engineering (IE) and manufacturing companies. The garments manufacturing companies generate huge amounts of data which has been collected, managed and analysed using the traditional and manual methodologies. However, data mining has been applied by many prominent researchers to mine the data generated by the companies and discover the hidden patterns and produce valuable insights to help the companies in their decision-making and production planning. For instance, Vazan et al. (2017) used data mining techniques to predict the manufacturing process behaviour using the production data. There are such plenty of scopes in this industry that can be filled up reliably by the empowerment of data mining in this industry.

The prediction of the productivity of the working teams is highly desirable among the decision makers in the garments industry as they are in a consistent fight to meet the deadline for product shipment. Furthermore, understanding the patterns behind the lower performance of the teams by mining the productivity data would enable the management to take the necessary steps to optimise production and costs. In this paper, we discuss a methodology for analysing and predicting the performance of the working teams in the garment industry. We have collected data from the working teams in different departments of a garment company which includes a set of attributes describing the overall activities of the teams.

This paper has several contributions from both technical and business perspective. Firstly, this paper applies a bunch of state-of-the-art data mining techniques to predict the productivity level or working performance class of the teams. Secondly, it rigorously performs multiple experiments from different perspectives and assesses the results with multiple comprehensive evaluation metrics. Thirdly, it reveals the underlying patterns and causes for productivity variation of the employees. Fourthly, it proposes a proper domain-specific workflow that is easily reproducible and interpretable for the industry experts which will help them to make better decisions where necessary.

To solve this problem, we have trained eight different data mining models over two types of datasets namely 'three-class dataset (3CD)' which includes three classes, such as normal, moderate and low and another one is the 'two-class dataset (2CD)' which includes two classes, such as low and not low. The original dataset contains three classes, but to make our experiment rigorous in finding a better model, we have additionally transformed the dataset into two classes and performed our experiment on both of the datasets. In both of the datasets, the class distribution is not balanced. Data imbalanceness is a critical issue in a data mining experiment. Because, when a model is trained over an imbalanced dataset, the model gets biased towards the majority class and the evaluation of the model become more complex. That is why each of the datasets in this study has been resampled using an oversampling technique named SMOTE to handle the class imbalance problem. Finally, all the experimental results are compared with their prediction performance. Our analysis has found that for predicting the three classes, the tree ensemble model trained on the data without oversampling resulted in the highest F-measure scores and overall accuracy (= 83.89%). For the case of predicting two classes, the gradient boosted tree (GBT) with the oversampled training data yielded the highest F-measure scores and overall accuracy (= 86.39%). These findings demonstrate that the oversampling technique does not perform well in predicting three classes, whereas for predicting two classes, the oversampling technique produces better results.

The rest of the paper is structured as follows: Section 2 discusses the existing similar works performed by the other researchers, Section 3 briefly discusses about the problem and the dataset background, Section 4 provides an outline of the solution steps, Section 5 describes the dataset preparation and description, Section 6 discusses the methodologies that have been used in this study, Section 7 discusses the results and analysis of the experiment and Section 8 draws the conclusion and discusses about the future works.

2 Related work

Due to the nature of the garment or apparel industry, a large amount of data is being produced during the manufacturing process. This huge amount of data can be a game changer for the decision makers. However, the application of data mining techniques in the garment industry has received less attention compared to other industries like telecommunication, health, banking and finance. The existing literature shows that data mining techniques have been applied to a wide spectrum of fields in the garment industry. These studies can be broadly categorised into two types:

1 predicting the productivity only

2 predicting, optimising the overall manufacturing process in the garment industry.

The literature of these two categories is critically reviewed below.

Predicting the productivity in the garment industry has been performed by Rahim et al. (2017), Imran et al. (2019) and Ersoz et al. (2017). Rahim et al. (2017) has proposed a data mining methodology for mining the industrial engineered manufacturing data that fulfils the requirements of the apparel industries. This methodology incorporated the analysis of the manufacturing unit and the analysis of different departments in the manufacturing unit to identify the correlation and dependencies among the departments. The authors have conducted an experiment using their proposed model and discovered useful knowledge from the data that the efficiency in productivity increases as the continuous working hour increases. The productivity of the workers is lowest in the first hour of the morning and increases every hour up to lunch time. However, this paper has not applied further data mining techniques. On the other hand, Imran et al. (2019) considered this problem as a regression problem and employed deep neural network approach to predict the productivity of garment employees. The authors designed a neural network architecture with two hidden layers where the first hidden layer contains 64 and the second hidden layer contains 32 neurons. They have achieved a mean absolute error of 0.086 which outperformed their baseline model. A similar kind of research was performed by Ersoz et al. (2017) where four different data mining techniques were used on textile industry data. In this study, the data of 198 personnel from a trouser producing firm were collected by the authors that contained seven attributes namely the number of employees, working hour, overtime, total working time, number of products produced in daily batch and per person production. The authors used CHAID, CART, regression and ANN to estimate the production per person and figured out the variables that affect the production. Their analysis also revealed a number of other valuable insights from the data. Their result showed that the production per person decreases with the increase in the daily working time. They also found a negative relationship between total daily working and production per person. They concluded that the accuracy of their estimation could be improved by using genetic algorithms and nonlinear regression methods. Both of these studies indicate the scope of further research and development to apply and develop data mining techniques for the garment industry. Though the number of literature is very low in this category, several studies have been observed under the other category such as optimising the overall manufacturing process in the garment industry.

Optimising the overall manufacturing process in garment industry includes a wide range of varieties like uniform sizing, body type classification, planning algorithm for automatic job allocation and improving overall quality. Hsu (2009) proposed a data mining framework on a large anthropometric data to classify the body types of adult females for developing an industry standard in the apparel industry. The framework was based on a two-stage cluster approach and blended with popular statistical and data mining methods to generate useful patterns and rules for standard size charts. They used the combination of Ward's minimum variance method and K-means algorithm to uncover patterns and rules in the anthropometric data. The author also conducted an empirical study in an apparel industry to support the manufacturing decision for production management for the industry and marketing with various customers' needs. Lin et al. (2007) applied a decision tree-based data mining technique to develop uniform sizing systems for army soldiers in Taiwan. Their system could correctly predict the requirements for different sizes of uniforms and produce realistic production planning. They applied the CART algorithm to the data and got a total coverage of 89% for the sizing systems, which were relatively high. Mok et al. (2013) proposed a planning algorithm for automatic job allocations based on group technology and genetic algorithms for the complex garment manufacturing companies. To improve the planning efficiency and optimise the job allocation problem, they used to single-run and two-run genetic algorithm approaches. To validate the proposed method, they used real production data from the garment manufacturing companies. Their proposed algorithms showed significant improvements in the planning quality and efficiency and were adopted by the apparel manufacturers in Hong Kong for their planning operations. Hou et al. (2003) performed research on intelligent remote monitoring of the manufacturing process and fault diagnosis. They used a backpropagation neural network technique to classify the faulty quality categories like wrinkles and uneven thickness. Apart from providing remote monitoring guidelines, a rough set was used to learn the causal relationship between parameters such as process temperature and output quality measures. The combination of neural networks and a rough set approach not only provided the information about what was expected to happen but also showed the reason for the occurrence. It also provided specific guidelines on how to recover from the abnormal conditions with process parameter setting. Kusiak (2002) applied data mining technique to support decision-making processes for the manufacturing systems. They used data mining algorithms to generate rules to meet the established decision-making criteria. A subset of these rules was selected to produce a decision signature of the manufacturing process. The decision signature is a set of parameter values that drive towards the expected result. These control signatures were updated using a framework produced by the learning classifier system. The accuracy of the decision signatures was evaluated prior to making predictions. The evaluation was done by group assessment and decision-making procedures developed by the human factors community. Though the goal of these studies was to improve the manufacturing process in the apparel industry, however, it cannot be achieved by overlooking the production phase.

From the related work study, we can conclude that the application of data mining in the garment industry requires further attention. As most of the related literature in this area is aimed at optimising the overall manufacturing process, so predicting the production efficiency in the apparel industry would be highly desired among the stakeholders. The goal of our study is to provide a solution for the identified research gap that is predicting the productivity performance using data mining techniques in the apparel industry and develop a data mining methodology to predict the productivity performance in the apparel industry.

3 Problem formulation

3.1 Dataset background

The goal of this section is to provide the necessary background regarding the dataset. The dataset has been originated from the IE department of a garments manufacturing unit. The role of IE department is vital for an organisation as this department is responsible for determining the most effective ways for utilising the basic factors of production: people, machines, materials, information and energy. According to Cooklin et al. (2006), this department works as a bridge to connect the management goals and operational performance. Furthermore, this department is also responsible for delivering different reports like daily individual efficiency report, daily line and factory efficiency report, daily line and factory lost time report, style costing report, MIS meeting report and incentive working report. Our collected dataset can be categorised as daily line and factory efficiency report.

The daily line and factory efficiency report represent the production efficiency of the lines and overall efficiency of the factory on a day to day basis using several attributes. The line represents the combination of machines, manpower and centre tables. The layout of the line is chosen based on the style requirements. This line efficiency report can be used for comparing the efficiency among multiple lines, calculating per minute labour cost or planning for capacity building. The formula for measuring the line efficiency is given below:

$$\text{Line efficiency(in \%)} = \frac{\text{(Total minutes produced by a line} \times 100}{\text{Total minutes worked by operators on that line}}$$

From a data mining perspective, this line efficiency data along with the other attributes like the number of style changes, total man, standard minute value (SMV), overtime and incentive can be useful to extract hidden patterns and translate them to the meaningful and valuable knowledge. Though the description of these attributes is discussed briefly in Section 5.2, however, this section provides an overview of the following selected attributes:

SMV is defined by the required time to accomplish a job satisfactorily, especially in the garment industry. It is expressed in minute value. For different garments, the SMV is different as this value depends on several factors like the type of garments and fabrics, machines and technologies, garments size and design. According to Sarkar's (2015) paper, SMV is calculated using the following formula:

$$\text{SMV} = \text{basic time} + \text{allowance}$$

where

$$\text{basic time} = \frac{\text{observed time} \times \text{speed of operation}}{100}$$

$$\begin{aligned}\text{allowance} = {} & \text{relaxation allowance} + \text{contingency allowance} \\ & + \text{machine delay allowance}\end{aligned}$$

Work in progress (WIP) is referred to the set of large unfinished items of products in the production process. Most of these items are either incomplete or waiting in a buffer storage and measured in pieces. WIP is applicable for cutting, sewing and finishing departments in a garment industry. Using the following equations from the study of Nayak and Padhye (2015), the WIP is calculated for the individual department:

$$\text{Cutting WIP} = \text{Total cut qty} - \text{Total qty sent to sewing}$$

$$\text{Sewing line WIP} = \text{Total pieces loaded to the line-pieces completed}$$

$$\text{Finishing room WIP} = \text{Total received from sewing} - \text{Total pieces packed}$$

Overtime is considered as the amount of time that someone works beyond the normal time schedule. In Bangladesh, daily eight hours of working is counted as the normal working hour and beyond that is called over time. In the garment industry, overtime is important to overcome the lower productivity and meet the deadline for product delivery. However, as the garment industry is a labour-intensive industry, so over time can play a negative impact on the overall productivity of the manufacturing lines.

In manufacturing industries like the garment, incentive plays a vital role as a factor for achieving higher productivity with the same resource available. Through the incentives, the organisations try to motivate their employees that results an increase in their productivity. According to Billikopf (1992), these incentives can be categorised into two types: structured incentives and casual incentives. Structured incentives help the employees directly through the form of higher pay for achieving the milestone or targeted performance. Shafiqul and Com (2013) mentioned in their study that a structured incentive depends on the performance, so it fluctuates with the change in performance level. On the other hand, casual incentives refer to informal incentives such as providing food, transport service, training. In this study, by 'incentive' we refer to structural incentive only, the information related to casual incentives is not considered due to unavailability.

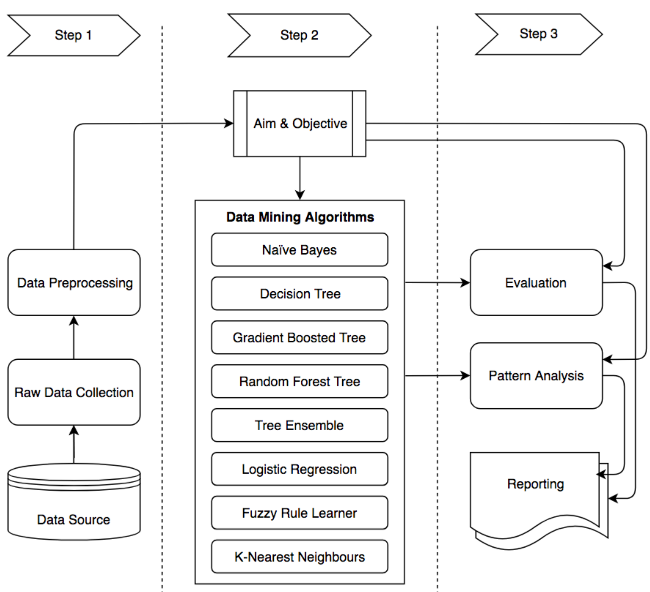
3.2 Problem statement

Line efficiency in the garment industry is a crucial measure for any organisation to maximise profit. It can be simply defined as the productivity of the working teams. Using this single measure alone, an organisation can push the productivity upwards, bring discipline in the organisation through an optimised and efficient production line and thus can control the cost of the production. But, most of the garment authorities still use a manual calculation to set the targeted productivity for each team. This manual approach is not efficient enough to meet the actual productivity of the teams. Thus, a targeted and actual productivity gap is created. This mismatch can cause a huge loss for the company and hampers the production to meet its deadline. This study identifies this critical issue and aims to solve the targeted and actual productivity gap by predicting the actual productivity of the working teams. In this study, our primary goal is to build predictive models to predict the actual productivity class of the working teams using various prominent data mining algorithms. This study also finds interesting patterns in the data that can answer important business questions for the authorities. Finally, this study finds the best predictive model by evaluating the performance of the models and try to provide a novel direction for mining the productivity data of garment industry using data mining techniques.

4 Solution steps

To obtain an effective solution for a problem, a study must follow a set of solution steps or methodology. Moreover, every applied data mining study includes multiple layers of operations which need to be properly organised and maintained to efficiently get a reliable result. In this study, we have designed a proper workflow that will lead our experiment to obtain an effective output. Furthermore, this workflow will make our study reproducible and provide a guide to the industry personnel for performing such experiments efficiently. The whole workflow is divided into a three-step process. The first step is data collection and pre-processing. In this step the data source and variables are determined and the raw data is collected from the source. After collecting the raw data, some pre-processing actions are performed to make the data clean and suitable for the data mining techniques. The second step is determining the aim and objective of the study and selecting proper algorithms for the experiment. We have provided a pool of eight modern data mining techniques that have been found to be effective for this type of classification problems. The final and third step is evaluation, pattern analysis and reporting. After the modelling phase in the second step, it is needed to evaluate the performance of the models. Some necessary evaluation metrics need to be selected based on the aim and objective of the study. We have used six different evaluation metrics in this study. After the evaluation process, the best model is identified and some meaningful patterns can be extracted from the best model. Finally, all the experimental results and observations are reported through an easy and interpretable way. Figure 1 represents the total workflow or solution steps discussed above.

Figure 1 Workflow of this study



5 Dataset preparation and description

5.1 Data collection

The first phase of any data mining project starts with collecting the data. In this study, in the first phase, we have collected the unstructured data of the IE department from a reputed garment manufacturing company, situated in Bangladesh. Initially, the data has been stored in three Excel files where each file contains the productivity data of a particular month and in each sheet of a file contains the production data of a particular day. Later, all the files and sheets were merged into a single excel file with more cleaning and popper formatting. The collected files contain the production data of the sewing and finishing department for three months from January 2015 to March 2015 of the company.

5.2 Description of the attributes

As we have mentioned above, our processed dataset includes 15 attributes finally. Table 1 presents the description of these attributes.

Table 1 Description of the attributes

Attribute	*Description*	*Example*
date	Date in MM-DD-YYYY	1/1/2015
day	Day of the week	Sunday, Monday
quarter	A portion of the month. A month was divided into four quarters	Quarter 1
department	Associated department with the instance	Sewing, finishing
team_no	Associated team number with the instance	Team 1, team 2
no_of_workers	Number of workers in each team	36
no_of_style_change	Number of changes in the style of a particular product	0, 1, 2
targeted_productivity	Targeted productivity set by the authority for each team for each day	0.80 (80%)
smv	Standard minute value, it is the allocated time for a task	4.15
wip	Work in progress. Includes the number of unfinished items for products	1,396
over_time	Represents the amount of overtime by each team in minutes	960
incentive	Represents the amount of financial incentive (in BDT) that enables or motivates a particular course of action	113
idle_time	The amount of time when the production was interrupted due to several reasons	4.5
idle_men	The number of workers who were idle due to production interruption	10
class_multi	Class of the actual productivity (three classes)	Normal, moderate, low
class_binary	Class of the actual productivity (two classes)	Low, not low

In Table 1, 'class_multi' and 'class_binary' are the output columns. 'class_multi' includes three output classes whereas, 'class_binary' includes two output classes. The purpose of defining two output columns are briefly discussed in Section 6.

5.3 Preprocessing the dataset

In the real world, data is often noisy, missing, incomplete and imbalanced. After collecting the raw data, we have transformed it into a semi-structured dataset and stored as a CSV file for our convenience of processing and manipulation. We have performed the transformation of the data manually and to ensure the quality, we cross-checked several instances randomly.

1 Attribute selection: As the ultimate output is measured using the daily targeted productivity for each team, we have only included the attributes in which we are interested, or which may have a direct impact on productivity performance. For the purpose of noise cleaning from raw data, we have used OpenRefine developed by Ham (2013), a powerful open source-based data cleaning tool. After this phase, we get 15 attributes with 1,197 instances.

2 Constructing the class column and versioning the dataset: the output column contains a corresponding productivity efficiency class for each of the instances based on the 15 attributes. To make our experiments reliable and comprehensive, we have made two versions of this dataset, namely '3CD' and '2CD'. The class labelling has been performed by measuring the daily targeted productivity, which is available in the collected raw data. We have followed the industry conventions and settings to determine the class labels. For the 3CD we have labelled the instances in three classes, namely 'normal', 'moderate' and 'low'. The class is assigned as 'normal' when the targeted achievement is 95% or above, 'moderate' when the targeted achievement is between 85% to 95% and 'low' when the targeted achievement was lower than 85%. For the 2CD we have labelled the instances in only two classes namely 'normal' and 'not normal'. The class was assigned as normal when the targeted achievement was 95% or above and 'not normal' when the targeted achievement was below 95%. Figure 2 represents the class distribution for the two versions of the dataset.

   In Figure 2, the left chart represents the class distribution of 3CD and the right chart represents the class distribution of 2CD dataset. The charts clearly show that both of the datasets contains 877 (= 73.3%) 'low' productivity class which dominates over the other classes. This huge gap between class distribution arises the class imbalance problem. However, most of the garments manufacturing authorities are sensitive to the 'low' productivity class. Because the overall production of a company gets affected if the productivity of the employees is lower. Experimenting on two versions of the dataset will show us a clear scenario of how well the models can classify the 'low' class.

3 Handling missing values: an attribute named WIP contains missing values. We observed that the values were missing only for the finishing department. We have imputed the missing values with 0 as in most of the cases the finishing department does not have any WIP.

Figure 2 Class distribution of the datasets (see online version for colours)

6 Methodology

The datasets discussed in Section 3, are imbalanced and building predictive models with them may not lead to satisfactory results as the majority class will be dominating the other classes. To deal with this issue, we have used a resampling technique developed by Chawla et al. (2002) named synthetic minority over-sampling technique (SMOTE) for balancing the training set of each version of the datasets. SMOTE is a popular oversampling technique that was proposed to improve random oversampling. SMOTE selects similar samples from the minority class with respect to some distance measure and adds perturbations on the selected attributes. Then it creates new minority samples within the clusters of the existing minority samples. As our dataset is relatively small, we choose not to use any under-sampling technique, but oversampling technique.

The entire experiment has been conducted in five phases such as:

Experiment 1 Training the eight data mining models on the 3CD without oversampling.

Experiment 2 Performing the same experiment as Experiment 1 with oversampling on the same dataset.

Experiment 3 Training the same eight models on the 2CD without oversampling.

Experiment 4 Performing the same experiment as Experiment 3 with oversampling on the same dataset.

Experiment 5 Evaluating the performances of the models in each experiment, performing a comparative analysis of the experimental results and finding out the best performing classifier among them.

The main activities that we have performed in this study – are collecting the data, cleaning and pre-processing the raw data, resampling and partitioning the dataset, classification, evaluation and a comparative analysis. All of the activities from dataset resampling to evaluation are described in the following subsections. All the implementations and experiments are performed using a free open-source data analytics and reporting tool called KNIME (Berthold et al., 2009). KNIME, the Konstanz

Information Miner, is a free and open-source data analytics, reporting and integration platform.

6.1 Partitioning and resampling the dataset

There are two prominent approaches for data partitioning: K-fold cross-validation and train-test split. In this study, we have decided to stick with the train-test approach because of two reasons:

1 we need to handle the class imbalance problem by using resampling methods with maximum authenticity

2 cross-validation requires a good amount of computational power which is a drawback in the real-world implementation scenario.

So, we partition both versions of the dataset into training and test sets with a ratio of 70:30%. The training set has been used for training the models and the test set has been used for evaluating the performance of the models. For both versions of the datasets, the training set includes 837 and the test set includes 360 instances which have been drawn at random. To maintain the maximum authenticity of the actual data, we have applied the SMOTE on the training sets only and kept the test sets unchanged as the real-world data will not be always balanced. When applying SMOTE, we chose oversample by minority classes with the nearest neighbour value of 5 and a static seed of 1,234. Previously many researchers such as Ahmed et al. (2015) used this technique on industrial data and found significant positive results. After rebalancing the training sets using SMOTE, the oversampled training set of the 3CD contained 1,815 instances and the oversampled training set of the 2CD contained 1,210 instances. Figure 3 shows the class distribution before and after oversampling.

Figure 3 Class distribution before and after oversampling (see online version for colours)

In Figure 3, the left bar chart represents the comparison of class distribution after oversampling on 3CD dataset and the right bar chat represents the comparison of class distribution after oversampling on 2CD dataset. From the left chart, we can see that the SMOTE oversampling technique generated additional 522 observations for the 'normal' and 456 observations for 'moderate' classes to equalise the number of observations (= 605) with the 'low' class. The right chart shows that the SMOTE oversampling technique generated more 373 observations for the ‘not low’ class to equalise the number of observations (= 605) with the ‘low’ class. This oversampled datasets will help us to tackle class imbalance problem and perform an unbiased evaluation of the predictive models.

6.2 Classification algorithms

In data mining, classification is the problem of identifying to which of a set of categories a new observation belongs. A classification model attempts to draw some conclusion on the basis of a training set of data containing observations whose category is known. In this study, we have chosen eight different data mining algorithms for classification purpose which are briefly described next:

1 Naive Bayes: naive Bayes (Murphy, 2006) is a popular probabilistic classifier which relies on Bayes’ rule. Naive Bayes calculates the prior probability for each class based on how frequently each label occurs in the training data. It considers all the features to contribute independently to the probability that an instance is a member of a certain class, whether or not they’re in fact related to each other or to the existence of the other features.

2 Decision tree: this is the traditional C4.5 decision tree classifier (Quinlan, 2014). In this classifier, the target variable must be categorical and the other attributes can be either categorical or numerical. Numeric splits are always in binary, whereas categorical splits can be either binary or as many outcomes as categorical values. C4.5 provides two quality measures for split calculation namely Gini index and gain ratio.

3 GBT: GBTs (Friedman, 2001), are ensembles of decision trees. GBT uses very shallow decision trees with a gradient boosted method to build an ensemble of trees. By default, a tree is built using binary splits for numeric and categorical variables, however, later it can be changed to multi-way splits. GBT supports built-in missing value handling mechanism which works to find the best direction for missing values to go to by examining each possible direction and selecting the one delivering the best result.

4 Random forest tree: random forest (Breiman, 2001), is a popular tree-based learning algorithm which is basically a collection or ensemble of decision trees. It randomly generates multiple decision trees by randomly selecting subsets of features. It outputs the class that is the mode of the output classes generated by individual trees.

5 Tree ensemble learner: tree ensemble learner is an ensemble of decision trees such as random forest variants. Each of the decision tree models is learned on a different set of rows and/or a different set of columns. The output model depicts an ensemble of decision trees and is applied in the corresponding predictor variable using the selected aggregation mode to aggregate the choices of the individual decision trees. It uses the divide-and-conquer approach to improve the model performance.

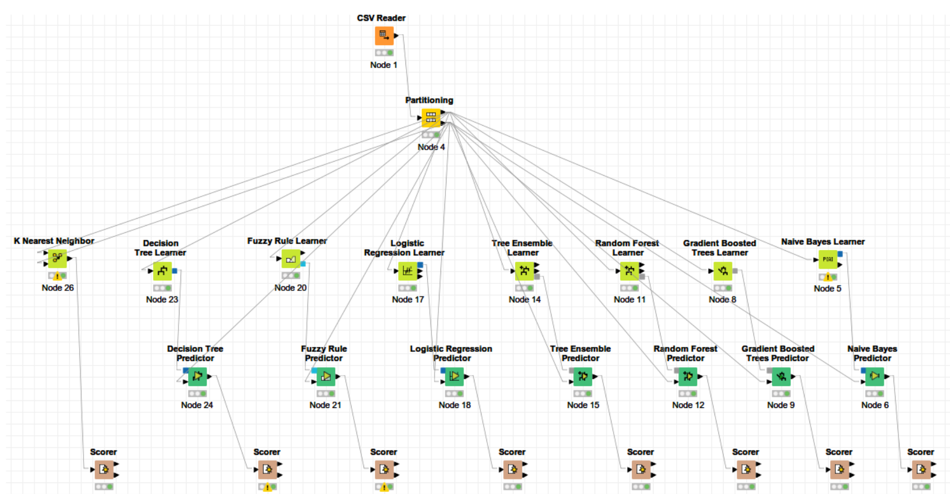
6 Logistic regression: logistic regression is a very popular statistical regression analysis algorithm that is usually used for binary classification problem. It is a nonlinear algorithm which represents the relationship between the input variables and the output variable.

7 Fuzzy rule learner: fuzzy rule learner (Berthold, 2003), is a system that uses the fuzzy rules as its main decision structure. This algorithm generates rules based on numeric data, which are fuzzy intervals in higher dimensional spaces. The selected numeric columns of the input data are used for training and additional columns are used as the classification target. It outputs the fuzzy rules after executing the model.

8 K-nearest neighbours (KNN): KNN (Larose and Larose, 2014) classifier is a similarity measure-based learning algorithm that uses a distance function (such as the Euclidean distance and the Manhattan distance) for pairs of observations. It finds the K closest data points in the training dataset and to identify the category of the input data point. The value of K we have used in our experiment is 5.

To ensure the best training performance of the classifiers, we have used grid search to fine-tuned the hyper-parameters of the classifiers. Figure 4 represents a screenshot of the KNIME workflow for the experiments.

Figure 4 A screenshot of the KNIME workflow (see online version for colours)



6.3 Evaluation

For all the experiments, we have used the confusion matrix to represent the classification results. We have also used four other different evaluation metrics namely accuracy, precision, recall and F-measure. For the 2CD, we have additionally considered the AUC score along with the other four metrics. All the evaluation metrics are described as follows:

1 Confusion matrix:

	Predicted (positive class)	*Predicted (negative class)*
Actual (positive class)	True positive (TP)	False negative (FN)
Actual (negative class)	False positive (FP)	True negative (TN)

TP true positive score describes the number of correctly classified 'positive' classes

TN true negative score describes the number of correctly classified 'negative' classes.

FP false positive score describes the number of misclassified 'negative' classes

FN false negative score describes the number of misclassified 'positive' classes.

2 $$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{FP} + \text{TN} + \text{FN}}$$

3 $$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}$$

4 $$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

5 $$\text{F-measure} = 2 \times \frac{(\text{recall precision})}{(\text{recall} + \text{precision})}$$

6 AUC score: the area under the receiver operating characteristic (ROC) curve is a single-valued score that is defined by computing the area under the ROC curve and summarising the curve information into one number. The AUC score value will always be between 0.0 and 1.0. A higher AUC score implies a better classification performance.

According to He and Garcia (2008), precision, recall, F-measure and AUC is considered as good metrics for evaluating the model performance that has been trained on imbalanced data.

7 Result and analysis

The performances of the classifiers are presented for all the four datasets. Firstly, for the 3CD without oversampling, secondly, for the 3CD with oversampling, thirdly, for the 2CD without oversampling and fourthly, for the 2CD with oversampling. In different situations, the authorities may require to predict a single class amongst the three classes. We have reported a comprehensive list of performance measures to evaluate the performance of the models from various perspectives. We have reported the confusion matrix with the actual number of occurrences of different classes classified by each model, precision, recall and F-measure for each class. These values can represent the performance of the model for different classes. Then we have also reported the accuracy and AUC to analyse the overall performance of the model. Table 2 shows the results for the 3CD without oversampling.

Table 2 Results for the 3CD without oversampling

		Normal	*Moderate*	*Low*	*Total (P)*	*Accuracy*	*Precision*	*Recall*	*F-measure*
Naïve Bayes	Normal	8	7	11	26	67.50	0.17	0.31	0.22
	Moderate	12	17	33	62		0.33	0.27	0.30
	Low	27	27	218	272		0.83	0.80	0.82
Gradient boosted tree	Normal	22	3	1	26	76.67	0.30	*0.85*	0.44
	Moderate	17	40	5	62		0.61	0.65	*0.63*
	Low	35	23	214	272		*0.97*	0.79	0.87
Random forest tree	Normal	14	4	8	26	83.61	0.88	0.54	0.67
	Moderate	1	26	35	62		0.65	0.42	0.51
	Low	1	10	261	272		0.86	0.96	*0.91*
Tree ensemble	Normal	14	2	10	26	*83.89*	*0.93*	0.54	*0.68*
	Moderate	1	20	41	62		0.77	0.32	0.45
	Low	0	4	268	272		0.84	*0.99*	*0.91*
Logistic regression	Normal	3	6	17	26	75.28	0.27	0.12	0.16
	Moderate	4	13	45	62		0.41	0.21	0.28
	Low	4	13	255	272		0.80	0.94	0.87
Fuzzy rule learner	Normal	12	10	4	26	80.74	0.55	0.46	0.50
	Moderate	2	41	16	59		0.52	*0.69*	0.59
	Low	8	28	232	268		0.92	0.87	0.89
Decision tree	Normal	11	1	14	26	80.00	0.69	0.42	0.52
	Moderate	0	12	50	62		*0.80*	0.19	0.31
	Low	5	2	265	272		0.81	0.97	0.88
KNN	Normal	9	3	14	26	76.99	0.45	0.35	0.39
	Moderate	2	20	40	62		0.53	0.32	0.40
	Low	9	15	248	272		0.82	0.91	0.86

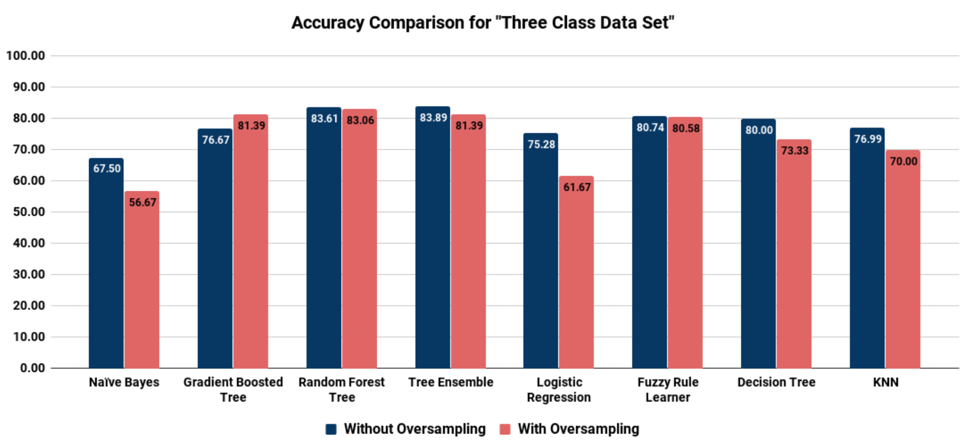
Table 3 Results for the 3CD with oversampling

		Normal	*Moderate*	*Low*	*Total (P)*	*Accuracy*	*Precision*	*Recall*	*F-measure*
Naïve Bayes	Normal	12	12	2	26	56.67	0.12	0.46	0.19
	Moderate	26	20	16	62		0.29	0.32	0.30
	Low	62	38	172	272		0.91	0.63	0.74
Gradient boosted tree	Normal	16	6	4	26	81.39	*0.53*	0.62	*0.57*
	Moderate	7	40	15	62		0.54	0.65	0.59
	Low	7	28	237	272		0.93	0.87	0.90
Random forest tree	Normal	19	3	4	26	*83.06*	0.39	*0.73*	0.51
	Moderate	14	37	11	62		*0.70*	0.60	*0.64*
	Low	16	13	243	272		*0.94*	*0.89*	*0.92*
Tree ensemble	Normal	19	3	4	26	81.39	0.38	*0.73*	0.50
	Moderate	13	39	10	62		0.64	0.63	0.63
	Low	18	19	235	272		*0.94*	0.86	0.90
Logistic regression	Normal	12	11	3	26	61.67	0.14	0.46	0.21
	Moderate	27	27	8	62		0.34	0.44	0.38
	Low	47	42	183	272		0.94	0.67	0.79
Fuzzy rule learner	Normal	11	10	4	25	80.58	0.52	0.44	0.48
	Moderate	2	41	13	56		0.51	*0.73*	0.60
	Low	8	30	226	264		0.93	0.86	0.89
Decision tree	Normal	15	5	6	26	73.33	0.23	0.58	0.33
	Moderate	21	29	12	62		0.51	0.47	0.49
	Low	29	23	220	272		0.92	0.81	0.86
KNN	Normal	10	5	11	26	70.00	0.24	0.38	0.29
	Moderate	13	31	18	62		0.40	0.50	0.44
	Low	19	42	211	272		0.88	0.78	0.82

From Table 2, we can see that in terms of accuracy, the tree ensemble model produces the highest accuracy score (= 83.89%) whereas, we can see some variation in the model performance in terms of precision and recall and F-measure score. When the 'normal' class is considered, the highest precision (= 0.93) and F-measure score (= 0.68) are produced by the tree ensemble model, whereas the highest 0.85 recall score is produced by the GBT. For the 'moderate' class, the highest precision score (= 0.80) is produced by the decision tree, the highest recall score (= 0.69) is produced by the fuzzy rule learner and the highest F-measure score (= 0.63) is produced by GBT. In terms of 'low' class, the highest precision score (= 0.97) is produced by the GBT, the highest recall score (= 0.99) is produced by the tree ensemble model and the highest F-measure score (= 0.91) is produced by both random forest and tree ensemble model. Table 3 reports the results for the 3CD with oversampling.

From Table 3, we can see that in terms of accuracy, the random forest produces the highest score (= 83.06%). Some variations can be seen in the score of precision and recall. When the 'normal' class is considered, the highest precision (= 0.53) and F-measure score (= 0.57) are produced by the GBT, whereas the highest recall score (= 0.73) is produced by both random forest and tree ensemble model. For the 'moderate' class, the highest precision (= 0.70) and F-measure score (= 0.64) are produced by the random forest and the highest recall score (= 0.73) is produced by the fuzzy rule learner. And in terms of 'low' class, the highest precision (= 0.94), recall (0.89), F-measure score (= 0.92) are produced by the random forest model. Tree ensemble also produced the highest precision score (= 0.94). Figure 5 represents a comparative visualisation of the accuracy score of the models for the 3CD.

Figure 5 Accuracy comparison of the models for 3CD (see online version for colours)



From Figure 5 analysis, we can see that for this kind of dataset oversampling does not work well in the improvement of the results, however, the dataset without oversampling works better. Considering the overall scenario, the tree ensemble classifier found to be the best performing classifier when applied on the dataset without oversampling. Whereas in some cases the random forest, fuzzy rule learner, GBT and decision tree models produce better precision and recall scores. The naïve Bayes classifier produces the lowest results for most of the evaluation metrics and thus it is found to be the worst classifiers among all. Table 4 represents the results for the 2CD without oversampling.

Table 4 Results for the 2CD without oversampling

		Low	*Not low*	*Total (P)*	*Accuracy*	*Precision*	*Recall*	*F-measure*	*AUC*
Naïve Bayes	Low	206	66	272	70.83	0.84	0.76	0.80	0.76
	Not low	39	49	88		0.43	0.56	0.48	
Gradient boosted tree	Low	240	32	272	85.83	*0.93*	0.88	0.90	0.89
	Not low	19	69	88		0.68	0.78	*0.73*	
Random forest tree	Low	250	22	272	*86.39*	0.90	0.92	*0.91*	*0.93*
	Not low	27	61	88		*0.73*	0.69	0.71	
Tree ensemble	Low	250	22	272	86.11	0.90	0.92	*0.91*	*0.93*
	Not low	28	60	88		*0.73*	0.68	0.71	
Logistic regression	Low	252	20	272	79.44	0.82	*0.93*	0.87	0.83
	Not low	54	34	88		0.63	0.39	0.48	
Fuzzy rule learner	Low	226	39	265	84.05	*0.93*	0.85	0.89	0.73
	Not low	17	69	86		0.64	*0.80*	0.71	
Decision tree	Low	240	32	272	82.50	0.89	0.88	0.88	0.85
	Not low	31	57	88		0.64	0.65	0.64	
KNN	Low	244	28	272	77.78	0.82	0.90	0.86	0.62
	Not low	52	36	88		0.56	0.41	0.47	

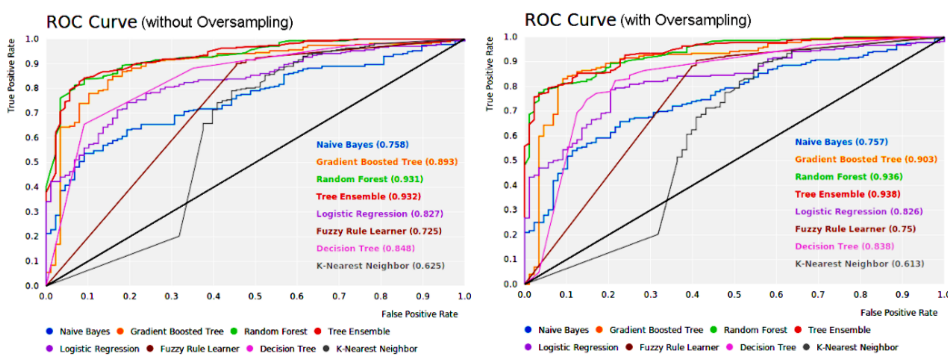
Table 5 Results for the 2CD with oversampling

		Low	*Not low*	*Total (P)*	*Accuracy*	*Precision*	*Recall*	*F-measure*	*AUC*
Naïve Bayes	Low	195	77	272	70.00	0.86	0.72	0.78	0.76
	Not low	31	57	88		0.43	0.65	0.51	
Gradient boosted tree	Low	240	32	272	*86.39*	0.93	*0.88*	*0.91*	0.90
	Not low	17	71	88		*0.69*	0.81	*0.74*	
Random forest tree	Low	233	39	272	84.72	*0.94*	0.86	0.89	*0.94*
	Not low	16	72	88		0.65	*0.82*	0.72	
Tree ensemble	Low	233	39	272	84.17	0.93	0.86	0.89	*0.94*
	Not low	18	70	88		0.64	0.80	0.71	
Logistic regression	Low	212	60	272	78.33	0.92	0.78	0.84	0.83
	Not low	18	70	88		0.54	0.80	0.64	
Fuzzy rule learner	Low	223	39	262	84.15	0.93	0.85	0.89	0.75
	Not low	16	69	85		0.64	0.81	0.72	
Decision tree	Low	230	42	272	82.22	0.91	0.85	0.88	0.84
	Not low	22	66	88		0.61	0.75	0.67	
KNN	Low	221	51	272	76.11	0.86	0.81	0.84	0.61
	Not low	35	53	88		0.51	0.60	0.55	

From Table 4, we can see that in terms of accuracy and AUC score the random forest produces the highest accuracy (= 83.39%) and AUC score (= 0.93) respectively. Some variations can be seen in the score of precision, recall and F-measure. When the 'low' class is considered, the highest precision score (= 0.93) is produced by the GBT and fuzzy rule learner, the highest recall score (= 0.93) is produced by the logistic regression model and the highest F-measure score (= 0.91) is produced by the random forest model. While considering the 'not low' class, the highest precision score (= 0.73) is produced by both random forest and tree ensemble model, the highest recall score (= 0.80) is produced by the fuzzy rule learner and the highest F-measure score (= 0.73) is produced by the GBT. Table 5 represents the results for the 2CD with oversampling.

Table 5 shows, in terms of accuracy, GBT produces the highest accuracy (= 86.39%) and AUC score (= 0.94) by both random forest and tree ensemble model. Variations can be seen in the score of precision, recall and F-measure. When the 'low' class is considered, the highest precision score (= 0.94) is produced by the random forest, the highest recall (= 0.88) and F-measure score (= 0.91) are produced by the GBT. While considering the 'not low' class, the highest precision score (= 0.69) and F-measure score (= 0.74) are produced by the GBT, whereas the highest recall score (= 0.82) is produced by the random forest model. Figure 6 represents a comparative visualisation of the ROC curves of the classifiers for the 2CD.

Figure 6 ROC curve comparison of the models for 2CD (see online version for colours)



From Figure 6 analysis, we can see that in the case of 2CD the results on both with and without oversampling are very close. However, from Figure 6 we can clearly determine that in terms of AUC or ROC, the GBT classifier with oversampling outperforms the other models. As we are interested in the model that maximises the prediction of 'low' class and lessen the false negative score, GBT is the best prediction model on the basis of ROC. The random forest and the tree ensemble model also produce promising results which are very close to the best one. The KNN classifier produces the lowest results for most of the evaluation metrics and thus it is found to be the worst classifiers of all.

Additionally, we have discovered some interesting generalised patterns in these datasets from the decision tree models which are discussed as follows:

Patterns in 3CD:

a among the 762 cases, when the 'incentive' is ≤ 69.5 BDT, the productivity level is low in 78.48% cases

b among the 50 cases, when the 'incentive' is ≤ 99 BDT and > 69.5 BDT, the productivity level is moderate in 85% cases

c among the 25 cases, when the 'incentive' is > 99 BDT, the productivity level is normal in 80.0% cases.

Patterns in 2CD:

a among 409 cases, when 'wip' > 3.5 and 'incentive' ≤ 69.5 BDT, the productivity level is low in 94.87% cases

b among 186 cases, when 'no_of_workers' ≤ 8.5 and 'wip' ≤ 3.5 and 'incentive' ≤ 69.5 BDT, the productivity level is low in 77.42% cases

c among 167 cases, when 'no_of_workers' > 8.5 and 'wip' ≤ 3.5 and 'incentive' ≤ 69.5 BDT, the productivity level is not low in 60.48% cases

d among 75 cases, when 'incentive' > 69.5 BDT, the productivity level is not low in 90.67% cases.

From the above patterns, we can say that the incentive is the primary causation behind the productivity level. Along with 'incentive', 'wip' and 'no_of_workers' are also found as important features for predicting the low cases. When the company needs good productivity from the employees, they should pay an incentive greater than 69.5, increase number of workers and lessen the WIP.

8 Conclusions and future work

In this study, our aim was to find out a reliable data mining-based solution for predicting the performance of the working teams in a garment company. To deal with this problem, we have applied eight different data mining techniques, namely naïve Bayes, GBT, random forest, tree ensemble, logistic regression, fuzzy rule learner, decision tree and K-nearest neighbours on a dataset that was collected from a renowned garment company in Bangladesh. The performances of the techniques were compared rigorously to figure out the best performing technique. The experiments took place for two types of datasets, one is the 3CD which included three classes namely 'normal', 'moderate' and 'low' and another one is the 2CD which included two classes namely 'low' and 'not low'. To handle the class imbalance problem, we have used an oversampling technique called SMOTE on the training data. Our experimental results showed that the for three-class classification, oversampling technique is not effective enough. However, for two-class classification, oversampling techniques produce better results. Our analysis found that, for predicting three classes, a tree ensemble classifier trained on the dataset without oversampling produces the highest accuracy (= 83.89%). In terms of predicting two classes, the GBT trained on the oversampled data produces the highest accuracy (= 86.39%) and AUC score (= 0.90). Furthermore, we extracted some useful patterns from the tree-based models which exposed that 'incentive' was the major reason behind the low productivity of the employees, 'wip' and 'no_of_workers' were the next most influential factors that were affecting the productivity.

In the future, we would like to solve this problem from a different data mining perspective such as a regression problem and also examine some advanced ensemble techniques to improve the prediction performance.

References

Ahmed, T., Calders, T. and Pedersen, T.B. (2015) 'Mining risk factors in RFID baggage tracking data', in *16th IEEE International Conference on Mobile Data Management*, IEEE, June, Vol. 1, pp.235–242.

Berthold, M.R. (2003) 'Mixed fuzzy rule formation', *International Journal of Approximate Reasoning*, Vol. 32, Nos. 2–3, pp.67–84.

Berthold, M.R., Cebron, N., Dill, F., Gabriel, T.R., Kötter, T., Meinl, T., Ohl, P., Thiel, K. and Wiswedel, B. (2009) 'KNIME-the Konstanz information miner: version 2.0 and beyond', *ACM SIGKDD Explorations Newsletter*, Vol. 11, No. 1, pp.26–31.

Billikopf, G.E. (1992) 'Establishing and troubleshooting incentive pay programs', in Van Horn, H.H. and Wilcox, C.J. (Eds.): *Large Dairy Herd Management*, pp.793–802, American Dairy Science Association, Champaign, IL, USA.

Breiman, L. (2001) 'Random forests', *Machine Learning*, Vol. 45, No. 1, pp.5–32.

Chawla, N.V., Bowyer, K.W., Hall, L.O. and Kegelmeyer, W.P. (2002) 'SMOTE: synthetic minority over-sampling technique', *Journal of Artificial Intelligence Research*, Vol. 16, pp.321–357.

Cooklin, G., Hayes, S.G., McLoughlin, J.J. and Hou, F. (2006) *Introduction to Clothing Manufacture*, Blackwell Science Limited, Wiley-Blackwell, Hoboken, New Jersey, USA.

Ersoz, F., Ersoz, T. and Guler, E. (2017) 'Knowledge discovery and data mining techniques in textile industry', *World Academy of Science, Engineering and Technology, International Journal of Computer, Electrical, Automation, Control and Information Engineering*, Vol. 11, No. 7, pp.906–910.

Friedman, J.H. (2001) 'Greedy function approximation: a gradient boosting machine', *Annals of Statistics*, October, Vol. 29, No. 5, pp.1189–1232.

Ham, K. (2013) 'OpenRefine (version 2.5) http://openrefine. org. Free, open-source tool for cleaning and transforming data', *Journal of the Medical Library Association: JMLA*, Vol. 101, No. 3, p.233.

He, H. and Garcia, E.A. (2008) 'Learning from imbalanced data', *IEEE Transactions on Knowledge and Data Engineering*, Vol. 21, No. 9, pp.1263–1284.

Hou, T.H.T., Liu, W.L. and Lin, L. (2003) 'Intelligent remote monitoring and diagnosis of manufacturing processes using an integrated approach of neural networks and rough sets', *Journal of Intelligent Manufacturing*, Vol. 14, No. 2, pp.239–253.

Hsu, C.H. (2009) 'Data mining to improve industrial standards and enhance production and marketing: an empirical study in apparel industry', *Expert Systems with Applications*, Vol. 36, No. 3, pp.4185–4191.

Imran, A.A., Amin, M.N., Rifat, M.R.I. and Mehreen, S. (2019) 'Deep neural network approach for predicting the productivity of garment employees', in *2019 6th International Conference on Control, Decision and Information Technologies* (*CoDIT*), IEEE, April, pp.1402–1407.

Kusiak, A. (2002) 'Data mining and decision making', in *Data Mining and Knowledge Discovery: Theory, Tools, and Technology IV*, March, Vol. 4730, pp.155–166, International Society for Optics and Photonics, AeroSense 2002, 2002, Orlando, FL, USA.

Lai, P.S. and Christiani, D.C. (2013) 'Long term respiratory health effects in textile workers', *Current Opinion in Pulmonary Medicine*, Vol. 19, No. 2, p.152.

Larose, D.T. and Larose, C.D. (2014) *Discovering Knowledge in Data: An Introduction to Data Mining*, John Wiley & Sons Hoboken, NJ, USA.

Lin, H.F., Hsu, C.H. and Wang, M.J.J. (2007) 'An application of data mining technique in engineering to facilitate production management of garments', in *Proceedings of the 11th WSEAS International Conference on Computers, Agios Nikolaos, Crete Island, Greece*, July, pp.56–61.

Mok, P.Y., Cheung, T.Y., Wong, W.K., Leung, S.Y.S. and Fan, J.T. (2013) 'Intelligent production planning for complex garment manufacturing', *Journal of Intelligent Manufacturing*, Vol. 24, No. 1, pp.133–145.

Murphy, K.P. (2006) *Naive Bayes Classifiers*, p.18, University of British Columbia, Canada.

Nayak, R. and Padhye, R. (Eds.) (2015) *Garment Manufacturing Technology*, Elsevier, Sawston, UK.

Quinlan, J.R. (2014) *C4.5: Programs for Machine Learning*, Elsevier, Burlington, Massachusetts, USA.

Rahim, M.S., Rahman, M. and Chowdhury, A.E. (2017) 'Mining industrial engineered data of apparel industry: a proposed methodology', *International Journal of Computer Applications*, Vol. 161, No. 7, pp.1–7.

Sarkar, P. (2015) *Garment Manufacturing: Processes, Practices and Technology*, 1st ed., Online Clothing Study, India.

Shafiqul, M.I. and Com, M. (2013) 'Labor incentive and performance of the industrial firm: a case study of Bangladeshi RMG industry', *IOSR Journal of Business and Management* (*IOSR-JBM*), Vol. 7, No. 3, pp.52–63.

Vazan, P., Janikova, D., Tanuska, P., Kebisek, M. and Cervenanska, Z. (2017) 'Using data mining methods for manufacturing process control', *IFAC-PapersOnLine*, Vol. 50, No. 1, pp.6178–6183.