Contents lists available at ScienceDirect

Machine Learning with Applications

journal homepage: www.elsevier.com/locate/mlwa

Designing a supervised feature selection technique for mixed attribute data analysis

Dong Hyun Jeong [a,*], Bong Keun Jeong [b], Nandi Leslie [c], Charles Kamhoua [d], Soo-Yeon Ji [e,*]

[a] *Department of Computer Science and Information Technology, University of the District of Columbia, DC, USA*
[b] *Department of Management and Decision Sciences, Coastal Carolina University, SC, USA*
[c] *Raytheon Technologies, MD, USA*
[d] *U.S. Army Research Laboratory (ARL), Adelphi, MD, USA*
[e] *Department of Computer Science, Bowie State University, MD, USA*

ARTICLE INFO

Keywords:
Mixed-attribute data analysis
Supervised feature selection
Machine learning
Visual analysis

ABSTRACT

Identifying optimal features is critical for increasing the overall performance of data classification. This paper introduces a supervised feature selection technique for analyzing mixed attribute data. It measures data classification performances of features with a user-defined performance criterion and determines optimal features to boost the overall data analysis performance. A performance evaluation is managed to highlight the usefulness of the technique with existing feature selection techniques such as analysis of variance test, chi-square test, principal component analysis, and mutual information. Visualization is also utilized to understand the differences in classifying instances with different features. From a comparative performance testing and evaluation, we found 5 ∼ 10% performance improvements with the proposed technique. Overall, evaluation results showed the usefulness of our proposed feature selection technique in mixed attribute data analysis.

1. Introduction

Analyzing data is considered a major research challenge due to the difficulty in handling high dimensional attributes (Chen, 2009) and mixed attribute data (Aggarwal, 2013). For handling high dimensional data, dimension reduction techniques (e.g., principal component analysis (PCA), multidimensional scaling (MDS), linear discriminant analysis (LDA), and among others) are used. With the techniques, the overall number of attributes can be reduced without sacrificing the nature of the data. Most data are comprised of mixed types (i.e., a mixture of numerical and categorical values), and existing numerical (or statistical) and categorical data analysis cannot be applied directly to the data. When analyzing categorical data attributes, a simple but broadly known solution transforms the attributes into binary attribute values, indicating each attribute's value as a binary attribute. Alternative approaches to handling the categorical data attributes include using different probabilistic models to determine the distributions of each data attribute.

For high-dimensional mixed attribute data analysis, either feature selection or extraction is often utilized. Although there is a slight difference between them, they are often used to determine variables (i.e., features) that are significant for understanding and analyzing the data. As a part of the feature selection process, selecting critical features is an essential task in data analysis because features can be used as predictors while maintaining the unique characteristics of data (Guyon & Elisseeff, 2003; Solorio-Fernández et al., 2020). Feature extraction builds a new set of features from the original feature set. Dimension reduction is a good example of feature extraction, which transforms data into a lower dimension by reducing dimensions (Fodor, 2002; Wang et al., 2014). When feature selection or extraction techniques are applied, understanding data is critical to achieving a better result. Otherwise, under-fitting (high bias) or over-fitting (high variance) problems might occur. These problems may degrade the overall data analysis performance.

In this paper, a supervised feature selection technique is proposed to support mixed attribute data analysis. It determines features that produce high data classification performance depending on machine learning algorithms and performance metrics by evaluating the significance of each feature. Three mixed-attribute financial datasets are utilized to determine the usefulness of our proposed technique. Performance testing with five performance metrics, such as precision, recall, F1-score, accuracy, and Area under the ROC Curve (AUC), is managed. We also utilize five classification algorithms (i.e., decision tree (DT), support vector machine (SVM), random forest (RF), logistic regression (LR), and k-nearest neighbors (KNN)) to measure classification

* Corresponding authors.
E-mail addresses: djeong@udc.edu (D.H. Jeong), bjeong@coastal.edu (B.K. Jeong), nandi.o.leslie@rtx.com (N. Leslie), charles.a.kamhoua.civ@mail.mil (C. Kamhoua), sji@bowiestate.edu (S.-Y. Ji).

https://doi.org/10.1016/j.mlwa.2022.100431
Received 2 March 2022; Received in revised form 27 October 2022; Accepted 27 October 2022
Available online 1 November 2022

performance differences. Lastly, we provide a visual representation of the feature selection techniques to determine the optimal features to examine differences in classifying instances.

Our primary research contribution is to propose an alternative feature selection technique, which improves the data analysis performances in analyzing mixed attribute data. More specifically, the contributions of this paper include:

- Introducing a supervised feature selection technique to analyze mixed attribute data with maximizing classification performance
- Conducting a performance evaluation with different classification algorithms to evaluate the effectiveness of the proposed technique
- Integrating a visualization approach to illustrate the differences of selected features among the feature selection techniques

This paper begins by discussing prior data analysis research in feature selection. After introducing a generalized data analysis process in Section 3, a detailed explanation of the proposed technique is explained in Section 4. In Section 5, descriptions of the used financial datasets and conducted evaluation studies are included. Then, conclusions and future work are included in Section 7.

2. Related work

Feature extraction and feature selection are broadly utilized when performing complex data analysis. Feature extraction builds a new set of features from the original feature set by transforming input values. Various statistical approaches are used for feature extraction. A commonly known feature extraction example is a dimension reduction technique. It focuses on identifying a minimum number of significant features while maintaining the original data's same characteristics. Because of this, it has been broadly applied in pattern recognition, data compression, and database management (Ding et al., 2012; Fodor, 2002). Popular linear dimensionality reduction techniques used in data analysis are PCA (Jolliffe, 2002) and LDA (Martinez & Kak, 2001). PCA is a multivariate statistical technique that computes eigenvalues as well as eigenvectors to determine correlated variables into principal components. Among the components, sorted components are determined to find highly dominant principal components. Therefore, the first principal component represents most of the variance in the data. Due to this reason, PCA has been applied in various application domains to determine lower-dimensional forms from high-dimensional data. When applying PCA, it is important to maintain comparable ranges of values in the data (i.e., applying feature normalization or scaling). Otherwise, principal components can easily be biased due to the high variance of the data because PCA tries to maximize the variance of each component when determining principal components. PCA is an unsupervised learning technique that determines principal information that represents internal data relationships. Thus, PCA is often used to classify clusters (Ding & He, 2004). Because of the effectiveness of understanding the data with having minimum requirements, PCA has been broadly applied in various scientific research (Jolliffe & Cadima, 2016). Linear regression (Rawlings et al., 1998) is considered a similar technique due to its characteristics of identifying internal data relationships that fit the data. It models the relationship between variables to determine a straight line that best fits the data. It is good for identifying the statistical relationship between two continuous variables (i.e., dependent and independent). However, unlike PCA, linear regression has not been used in feature extraction because of not producing any additional information from the analysis (e.g., principal components in PCA). LDA is also closely related to PCA because it also transforms features into a lower-dimensional space. It measures a linear combination of features to assess separated objects (events) classes in the data. LDA is considered a supervised learning technique because it predicts a categorical dependent variable. With one or more n independent variables, it separates the classes of dependent variables by determining k independent variables. Since it uses continuous or binary independent variables, it is suitable for analyzing data that includes categorical outcomes (Pohar et al., 2004). Because of its ability, it has been used broadly to used for dimensionality reduction and classification (Boulgouris et al., 2010; Li, Feng et al., 2018; Tharwat et al., 2017).

Feature selection is utilized to reduce the number of features to generate a data analysis model. Various feature selection techniques are often classified as supervised, unsupervised, and semi-supervised methods (Solorio-Fernández et al., 2020). Supervised methods select features by utilizing the information in the dataset, whereas unsupervised and semi-supervised methods use no (or limited) information when selecting features. There is a slight difference among them, but most feature selection techniques apply statistical approaches to score all features. Then, unnecessary (or insignificant) features are eliminated to produce low scores. Various statistical approaches are used to design feature selection techniques (Chandrashekar & Sahin, 2014; Dash & Liu, 1997; Guyon & Elisseeff, 2003). Well-known feature selection techniques utilize Pearson correlation coefficient (PCC), analysis of variance (ANOVA), and mutual information (MI). ANOVA is a statistical technique that assesses differences between two or more variables by analyzing their variations. ANOVA-based feature selection technique use ANOVA to measure statistical significances of all features (Elssied et al., 2014) based on the assumption that data is normally distributed and maintains equal data variances. It removes the features that do not satisfy the statistical significance. Due to the ability to evaluate the statical significance of all features precisely, ANOVA-based feature selection technique was used in research on microarray data analysis (Saeys et al., 2007). PCC measures the correlations of two quantitative variables. PCC-based feature selection technique determines all features' degrees of dependencies. Guyon and Elisseeff (2003) and Xie et al. (2006). It determines statistical relationships among variables for selecting features. But, it has a limitation of detecting only linear dependencies. MI is a measure of determining mutual dependence between two variables. MI-based feature selection technique uses MI to evaluate the contribution of each feature to find a feature set that has maximum dependencies to a target class (Gao et al., 2015; Peng et al., 2005; Ross, 2014). Since it quantifies the dependencies between feature(s) and the class, it has been broadly used to eliminate redundant or similar features when analyzing datasets.

In terms of evaluation criteria, feature selection can be categorized into three different models: filter models, wrapper models, and embedded (hybrid) models (Bolón-Canedo et al., 2013; Chandrashekar & Sahin, 2014; Kohavi & John, 1997; Saeys et al., 2007). The earliest method is the filter model, which relies on intrinsic characteristics of data. The selection process is performed independently of data mining algorithms, so it tends to neglect an interaction effect between the selected features and the performance of used ML algorithms. The wrapper model uses the predictive accuracy or error rate of predetermined learning algorithms to determine the quality of selected features. It achieves better performance and higher accuracy compared to the filter models, but they are computationally expensive and exposed to over-fitting issues (Jovic et al., 2015; Li, Cheng et al., 2018). Because of these limitations in each model, the embedded (hybrid) model has gained increasing attention (Solorio-Fernández et al., 2020; Zebari et al., 2020). By combining and integrating different methods, the embedded (hybrid) method can utilize the advantages of each model to achieve both accuracy and efficiency. The combination of filter and wrapper methods is the most common embedded method, while any combination can be used to develop hybrid models. Several interesting hybrid methodologies were recently proposed, such as PCA and ReliefF method for chronic disease classification (Jain & Singh, 2021), a combination of clustering and the modified binary ant system (BAS) for high-dimensional data (Manbari et al., 2019), multi-strategy feature selection and grouped feature extraction for dimension reduction (Li et al., 2020), and swarm intelligence (SI) algorithms (Brezočnik et al., 2018). Other recent developments in feature selection include

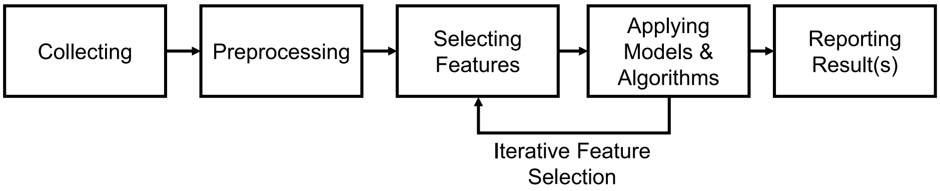


Fig. 1. Standard data analysis process with our consideration of iterative feature selection.

deep learning based feature selection (Rashid et al., 2020; Tian et al., 2020) and causality based feature selection (Bellizio et al., 2021; Sethi & Mittal, 2019; Yu et al., 2020). Feature selection techniques are suitable for analyzing and identifying significant features, but utilizing cleaned and sanitized data is essential to produce high data classification performance.

3. Data analysis

Data analysis has been performed broadly to analyze data, solve real-world or business problems, and detect anomalous events or activities. It is a process of applying various statistical or logical techniques to evaluate and discover useful information from data. Data analysis often consists of multiple analysis processes. Although it is defined differently depending on research domains, it simply considers utilizing basic statistical analysis procedures (i.e., mean and standard deviation) to understand data statistically. To produce high data analysis performances, fully understanding the data is crucial. However, it is not easy and often requires extensive data processing. To address this limitation, various advanced machine learning approaches of utilizing parallel processing models with multiple GPUs (Graphics Processing Units) have been proposed (Aida-Zade et al., 2017; Li et al., 2016).

Standard data analysis process consists of five phases: collecting, preprocessing, selecting features, applying models & algorithms, and reporting result(s). The collecting phase is an initial and important step in data analysis. Researchers often consider it as not part of the data analysis process. But, it is a critical step because collected data are used to design and validate data analysis models. Although the amount of data is not always a primary consideration in data analysis, collecting a reasonable amount of data with multiple attributes is vital to understanding the data fully by performing precise data analysis. If the data sample size is small, it may not represent the whole population of data groups or events. Thus, in such as case, the reliability of designed models and analyzed results cannot be guaranteed. The preprocessing phase removes redundancies and sanitizes unwanted features in collected data. This is a process of cleaning unformatted data and handling unwanted missing values. Depending on data types, it is critical to apply an appropriate data preprocessing method (Famili et al., 1997). Otherwise, unexpected or unreliable analysis results will be determined. The selecting features phase is an essential step in data analysis for improving data analysis performances. We believe that the feature selection process cannot be done as a one-time process. Instead, it needs to incorporate multiple feature selections. As shown in Fig. 1, the feature selection could be an iterative process (dF_o/dt) of continuously determining a list of optimal features (F_o) to boost performance on machine learning models. It handles the complexity of data by reducing the total number of features. With selected features, data analysis can be performed to run computational algorithms as the applying models & algorithms phase. When analyzing different types of data, applying appropriate models or algorithms is critical to achieving the best data analysis results. There is no specific logic that supports us in selecting data analysis techniques or algorithms. But different data analysis techniques are often considered depending on the type of data. For instance, support vector regression, polynomial regression, and linear regression are useful in analyzing continuous data. Instead, logistic regression, k-nearest neighbors, support vector machine, Naive Bayes, decision tree, and random forest are widely used to analyze categorical data or the dataset that does not follow a uniform data format (i.e., numerical or categorical). Alternatively, k-means, density-based clustering, mean-shift clustering, expectation–maximization (EM) clustering, and hierarchical agglomerative clustering (HAC) are used to determine clusters (Tan et al., 2005). Lastly, the reporting result(s) phase indicates the process of summarizing analysis results to highlight the significance of applied or newly proposed algorithms. To achieve a better performance result in data analysis, each phase needs to be managed independently. Among them, selecting features and applying models & algorithms phases are significant because they are closely related to producing reliable data analysis results.

4. Proposed feature selection technique

Feature selection process is critical for improving the overall data classification performance. It is important when performing large-scale mixed attribute data analysis because it reduces the overall data analysis time by determining optimal features. However, finding optimal features is not easy because high bias or variance can occur due to inadequately selected features. Our technique evaluates the performances of each feature with a user-defined performance metric (γ). Since a performance metric depends on the purpose of the measurement and the context of the problem being solved, evaluating the performance with the user-defined metric is critical to assess the importance of features. Various performance metrics are available such as Accuracy, Precision, Recall, F1-score, AUC, and others. However, when analyzing imbalanced mixed attribute data, accurate evaluation information is often not available because performance can be highly biased toward the majority of classes. The most commonly used metrics for imbalanced data analysis are receiver operation characteristics (ROC) analysis, the area under the ROC curve (AUC), Precision, Recall, and F-measure (He & Ma, 2013). To determine the best candidate features, it evaluates classification model performances of a finite set of selected features. Assuming that data (D) consists of n features (i.e., $F = \{f_i\}, i = \{1, 2, \dots, n\}$), it determines an optimal feature set ($F_o \subseteq F$) yielding increased performance score of $S(\cdot)$. For instance, if AUC is used as a performance metric (γ), $S_o(\gamma)$ for F_o is measured by calculating true-positive rate (TPR) and false-positive rate (FPR) with generating the corresponding ROC curve. With an initial $F_o = \{\}$, $F_o = F_o \cup \{f_i\}, f_i \in F$ is evaluated to find the next follow-up feature set. This process will be continued until no more feature influences to increase the performance score.

$$\arg\max S_c(C, F_c, \gamma, T_c) \quad \text{with} \quad F_c = \{x_i f_i\}, F_c \subseteq F \qquad x_i = 0 \text{ or } 1, \quad f_i \in F, i = \{1, \dots, n\} \tag{1}$$

Eq. (1) indicates the procedure of maximizing $S_c(\cdot)$ of classifier (C) with γ by evaluating candidate feature sets F_c, in where T_c indicates target class, f_i represents feature, and x_i denotes a vector of binary variables by numbering features from 1 to n. The datasets include T_c information denoting 1 (creditworthy application)/0 (non-creditworthy application) in the Australian dataset, 1 (good credits case)/2 (bad credits case) in the German dataset, and 1 (abnormal transaction)/0 (normal transaction) in the UCSD-FICO dataset. If the feature f_i is selected as a candidate feature, x_i sets to 1 as the following condition.

$$x_i = \begin{cases} 1, & \text{if } f_i \in F_c \\ 0, & \text{otherwise} \end{cases} \tag{2}$$

Algorithm 1: Proposed feature selection technique

```
Data: input feature set ($F = \{f_1, f_2, ..., f_n\}$), target ($T$), classifier ($C$),
      termination condition ($\lambda$)
      performance metric ($\gamma$), cross-validation ($k$), repetition check
      ($\rho$)
Result: determined optimal feature set ($F_o$) and its performance score
        ($S_o(\gamma)$)
$F_o \leftarrow \emptyset$ ;                          /* optimal feature set */
$S_o(\gamma) \leftarrow 0$ ;                    /* performance score for $F_o$ */
$\tau \leftarrow 1$ ;                                        /* terminator */
while $\tau$ OR $F_o \neq F$ do
    $S_\theta(\gamma) \leftarrow 0$ ;              /* tracking performance score */
    $F_\theta \leftarrow \emptyset$ ;              /* tracking feature set for $S_\theta$ */
    for $I \leftarrow 1$ to $|F| - |F_o|$ do
        Perform $k$-fold cross-validation with $C$, $T$, $\gamma$ for
          $F_c \leftarrow F_o \cup \{f_i\}, f_i \notin F_c$ ;
        Compute $S_{\bar{I}}(\gamma) = \frac{1}{k}(S_{I_1}(\gamma) + \cdots + S_{I_k}(\gamma))$ ;
        if $S_\theta(\gamma) < S_{\bar{I}}(\gamma)$ then
            $S_\theta(\gamma) \leftarrow S_{\bar{I}}(\gamma)$ ;
            $F_\theta \leftarrow F_c$ ;
        end
    end
    if $S_\theta(\gamma) - S_o(\gamma) \geq \lambda$ then
        if $S_\theta(\gamma)$ is repeated $> \rho$ then
            $\tau \leftarrow 0$ ;       /* terminate if $S_\theta(\gamma)$ appeared more
                                           than $\rho$ times */
        else
            $S_o(\gamma) \leftarrow S_\theta(\gamma)$ ;    /* update performance score */
            $F_o \leftarrow F_o \cup F_\theta$ ;  /* update optimal feature set */
        end
    else
        $\tau \leftarrow 0$
    end
end
```

Pseudocode for the proposed technique is presented in Algorithm 1. Six input parameters are needed as input feature set F, target class T, classifier C, user-defined performance metric γ and termination condition λ, k-fold cross-validation, and repetition check ρ. It evaluates combined feature sets to determine an optimal feature set through k-fold cross-validation with the metric γ. Instead of continuously evaluating all combined feature sets, it terminates the evaluation if no more feature is selected to yield maximizing the performance score. Specifically, it uses the termination condition (λ) to determine whether the measured performance score appears as a global maximum. Since our proposed technique analyzes mixed attribute data by generating one-hot encoded variables, a minor performance score fluctuation might happen (causing local maximum) because of the pattern similarity among the encoded variables. Local maxima peaks are often appeared when performing feature sub-set selection (Kohavi & John, 1997). Thus, finding the global maximum is important in analyzing mixed attribute data. ρ is also needed to determine if the same performance score appears multiple times.

5. Evaluation

5.1. Datasets

Three financial datasets, Statlog (Australian credit approval) dataset, German credit dataset, and UCSD-FICO data mining contest 2009 dataset, are used for performance evaluation (Table 1). In the rest of this paper, we will call them, for short, Australian dataset, German dataset, and UCSD-FICO dataset, respectively. The Australian dataset includes 690 credit card applications. All datasets are provided with anonymizing personal data to address the possibility of disclosing private information. Since they are imbalanced datasets, applying standard classification algorithms may not work well (He & Garcia, 2009; He & Ma, 2013; Stefanowski, 2016). The German dataset includes loan approval information by assessing money lending risk based on applicants' demographic and socio-economic profiles. It contains one thousand loan applicants with 1000 cases. It consists of twenty attributes (7: numerical and 13: categorical). In detail, the data includes financial standing (i.e., credit history), personal status and gender information, employment status, and more. The UCSD-FICO dataset includes real e-commerce transactions. It was designed as a data analysis competition dataset to detect fraudulent or anomalous activities. It includes 100,000 e-commerce transactions. It includes nineteen attributes. The German and the Australian datasets are from the UCI repository (Dua & Graff, 2017).

5.2. Data preprocessing

Most real-world data are incomplete or inconsistent, so applying data analysis methods directly to the dataset can be challenging and often causes poor results. Commonly used data preprocessing techniques are missing value treatment and categorical data transformation. If the data contains missing values, they must be handled by simply removing them or replacing them with highly related information. Various approaches have been proposed to replace the missing values (Rey-del Castillo & Cardeñosa, 2012; Zhu et al., 2011). Replacing with computed mean and median is commonly used. Using approximately determined values is also considered broadly. The Australian dataset contains missing values. But, they were already replaced with mode and mean values of the data. The other two datasets do not include any missing values. Thus, we did not apply any missing value treatments. For categorical data, it is not always necessary to apply categorical data handling methods because some existing data analysis techniques can handle the data. However, it is important to use data transformation to a more representative numerical format when analyzing mixed attribute data. When converting categorical values to numerical forms, a simple approach is replacing each categorical value with an integer value. However, it adds unnecessary ordering to the categorical values (for example, "saving account" as 1, "checking account" as 2, and "money market account" as 3). Since the three datasets that we used in our study include categorical values with no ranking or ordering information, one-hot encoding (Cerda et al., 2018; Szczepańska, 2011) was applied. It generates one-hot encoded variables to replace each categorical variable to represent each categorical value while preserving its own characteristics. For instance, assuming that the "account type" variable has three values as "saving account", "checking account", and "money market account", three new binary variables are created for each account type with a binary value of 1 or 0, indicating the existence of each account. For the German, the Australian, and the UCSD-FICO datasets, 52, 32, and 54 encoded variables are generated, respectively.

5.3. Performance testing

Performance evaluation was managed with five machine learning classifier algorithms: SVM, LR, DT, RF, and KNN. For our performance evaluation, SVM was used with Laplace Radial Basis Function (RBF) kernel ($K(x_i, x_j) = exp(-\frac{\|x_i - x_j\|^2}{2\sigma^2})$). LR was used to estimate the probability of an event occurring with L2 regularization. For running DT and RF, Gini impurity was measured to determine optimal splits in tree nodes. KNN ran with euclidean distance to determine five nearest neighbors. We compared the proposed technique with other feature selection techniques, including MI and ANOVA F-test. For running the proposed technique, λ and ρ are empirically determined as 0.01 and 4, respectively. PCA was also used to compare the classification performances with selecting k principal components as possible features (Guo et al., 2002). For our evaluation, 10-fold cross-validation was applied. Five evaluation metrics were utilized to understand the efficiency of the proposed technique. Specifically, precision, recall (i.e., sensitivity), F1-score, accuracy, and AUC were used to measure the performances

Table 1
Datasets used in our performance evaluation study.

	Normal	Abnormal	Attributes
Australian credit approval dataset	307 creditworthy applications	383 non-creditworthy applications	14 attributes (5: numerical and 9: categorical)
German credit dataset	700 good credits cases	300 bad credits cases	20 attributes (7: numerical and 13: categorical)
UCSD-FICO data mining contest 2009 dataset	97,350 normal transactions	2650 anomalous transactions	19 attributes (17: numerical and 2: categorical)

Table 2
Overall performance improvement (percentage) with the proposed technique compared to other feature selection techniques.

	Precision	Recall	F1-score	Accuracy	AUC
German	2.60% ± 1.28%	7.90% ± 3.61%	1.95% ± 1.21%	3.00% ± 1.45%	21.25% ± 4.43%
Australian	7.01% ± 2.36%	14.67% ± 14.02%	8.88% ± 8.62%	6.01% ± 4.89%	6.25% ± 6.15%
UCSD-FICO	6.60% ± 6.46%	4.97% ± 1.28%	21.25% ± 4.43%	0.12% ± 0.14%	1.88% ± 0.45%

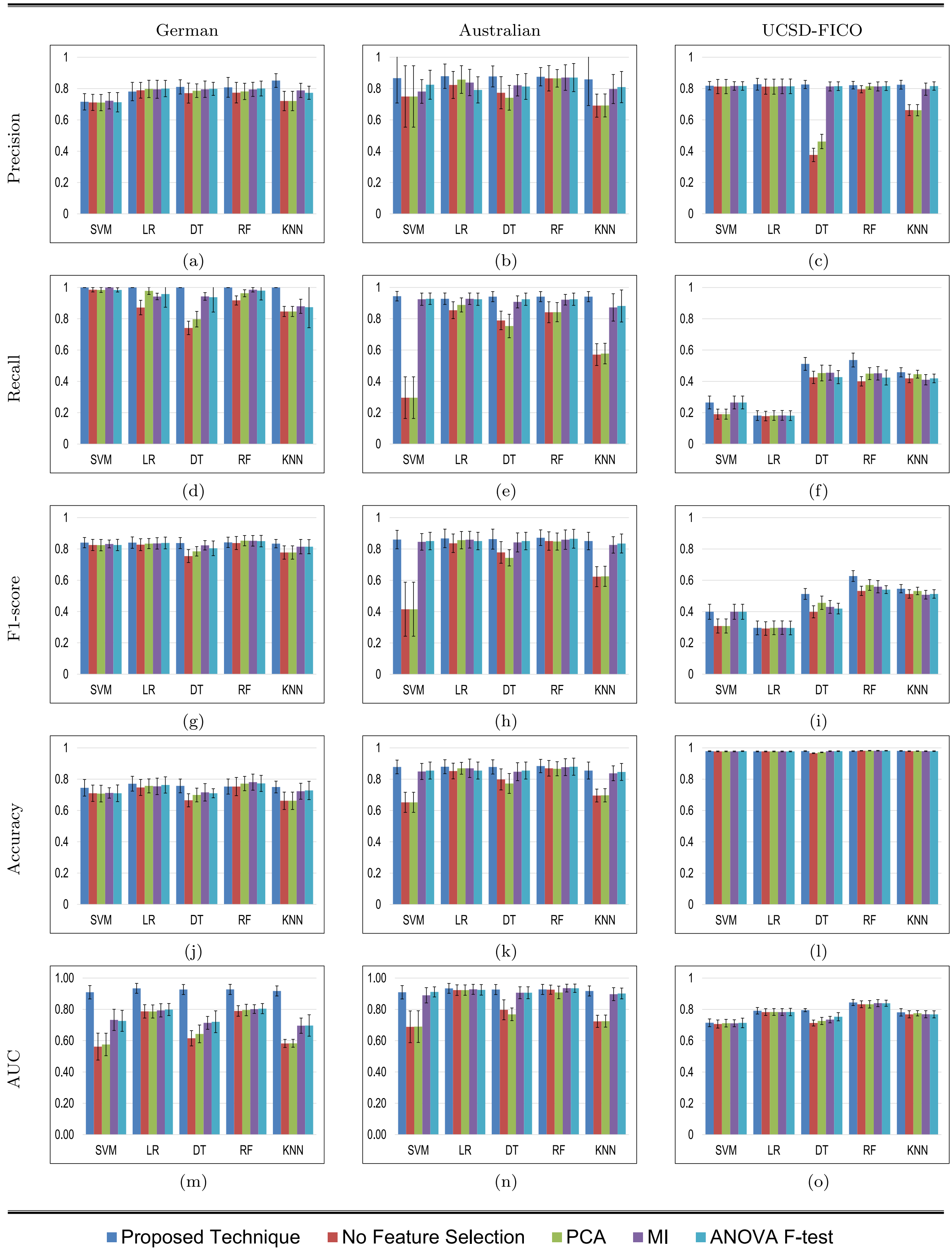
of the techniques. Among them, measuring AUC is important for analyzing imbalanced data because it summarizes the trade-offs between sensitivity and specificity measures through the receiver operating characteristic (ROC) curve. Fig. 2 presents the performance results. To understand the classification performance differences among the datasets, the mean and standard deviation of all classification algorithms were calculated. For the German dataset, we found average classification results as precision $= 0.77 \pm 0.05$, recall $= 0.93 \pm 0.03$, F1-score $= 0.82 \pm 0.03$, accuracy $= 0.73 \pm 0.04$, and AUC $= 0.73 \pm 0.04$. There were small performance variances depending on the applied machine learning algorithms with the different feature selection techniques. For instance, all raw and PCA features showed lower performances in most algorithms. But, a slightly better performance result was observed when using all raw and PCA features with RF. With different metrics, we found a performance improvement with our proposed technique compared to other feature selection techniques. For the Australian dataset, the average performance results were determined as precision $= 0.82 \pm 0.05$, recall $= 0.80 \pm 0.20$, F1-score $= 0.77 \pm 0.14$, accuracy $= 0.82 \pm 0.07$ and AUC $= 0.87 \pm 0.08$. We observed high variance performance results for the recall, F1-score, and AUC metrics when using all raw and PCA features with SVM, DT, and KNN because of high false negatives. We also observed much lower performances with the SVM, DT, and KNN algorithms when the raw and PCA features were used. While it is inconclusive, we suspect that the reason may be associated with high false-negative rates. When comparing the performance differences between the German and the Australian datasets, we found the overall performance result for the Australian dataset was slightly better, especially when using the accuracy and AUC metrics. We also identified that the performance result with the precision metric was high (on average) in the Australian dataset. But, the German dataset showed a better performance result for the recall metric. This would be because of relatively lower false positives in the Australian dataset and lower false negatives in the German dataset. Detailed performance results are added in Tables A.1–A.3 in Appendix.

For the UCSD-FICO dataset, we found much lower performance results (less than 50%) when evaluating the dataset with the recall and f1-score (see Fig. 2(f) and (i)). Identifying fraudulent activities in the UCSD-FICO dataset is difficult because very few instances indicate fraudulent activities (less than 3%). Because of the high similarity between normal and fraudulent activities, most machine learning algorithms cannot correctly classify them. In our study, we observed high accuracy (close to 100%) when analyzing the dataset. But, since the accuracy is computed based on evaluating the proportion of true positives and negatives in the dataset, it may not be a good indicator, especially when analyzing highly imbalanced datasets. This would be because the impact of least represented examples (i.e., fraudulent activities in the UCSD-FICO dataset) is relatively minor compared to the majority of instances when running machine learning algorithms (Branco et al., 2016). Based on the comparison of the AUC results, the Random Forest algorithm showed slightly better performance (see Fig. 2(o)). Overall, from the analysis, we found a better performance result with the proposed technique than with others. Among the different feature selection techniques, we also observed that the performance results of the feature selection techniques (using PCA and no feature selection) were generally low for all machine learning algorithms.

Although the datasets we used in this study are imbalanced, we found improved classification performance with feature selection techniques. Table 2 also shows average performance improvement with the proposed technique using different metrics. Although classification performance closely depends on applied classification algorithms, our proposed technique showed a slightly better performance. As explained above, there was a high variance in the performance results with the Australian dataset, especially when running SVM, DT, and KNN with PCA and without applying any feature selection techniques. Due to this reason, high standard deviations were observed when using recall, F-1 score, and AUC metrics in the Australian dataset. For the UCSD-FICO dataset, we also found a high standard deviation using DT with the precision metric for PCA and no feature selection technique. The lower performance results with the F1-score were caused by the performance difference of the precision or recall metric because the F1-score represents the weighted average of precision and recall. Overall, we noticed a performance improvement with our proposed technique in analyzing all three datasets.

Fig. 3 presents the total number of features determined by different feature selection techniques producing maximum performances appeared in Fig. 2. “No feature selection” in the figure indicates using all features for classifications. As mentioned above, MI and ANOVA F-test are used as they are broadly known feature selection techniques. PCA is a commonly used feature extraction technique because it extracts k principal components ranked from the input features by the importance of each variable contributing with varying degrees of components. In our study, PCA was used as an alternative feature selection approach for testing different principal components to improve the overall classification performances. When evaluating the German dataset, fewer features were selected, producing the highest performances with the precision and recall metrics for all machine learning algorithms. For evaluating the Australian dataset, our proposed technique used one or two more features to produce maximum performance. When analyzing the UCSD-FICO dataset, we discovered that fewer features were required to produce high classification performances with SVM with our proposed technique, specifically with the metrics of prediction, recall, F-1 score, and accuracy. With the AUC metric, we found a minor difference between our proposed technique (five features) and MI (four features), producing similar performances as 0.714 (our proposed technique) and 0.711 (MI) (see Fig. 2(n)). For analyzing the German dataset with KNN, our proposed feature selection technique required nine features, but other feature selection techniques used four to eight features.

When evaluating the classification performances, we found that several features were commonly selected in most classification algorithms. In detail, four to five features determined with the proposed technique were selected by other feature selection techniques for running SVM, LR, RF, and KNN. But, fewer features (about two to three) were selected with MI for the DT classification algorithm. DT measures the impurity of data to determine purity for generating a tree. Since



(a) (b) (c)

(d) (e) (f)

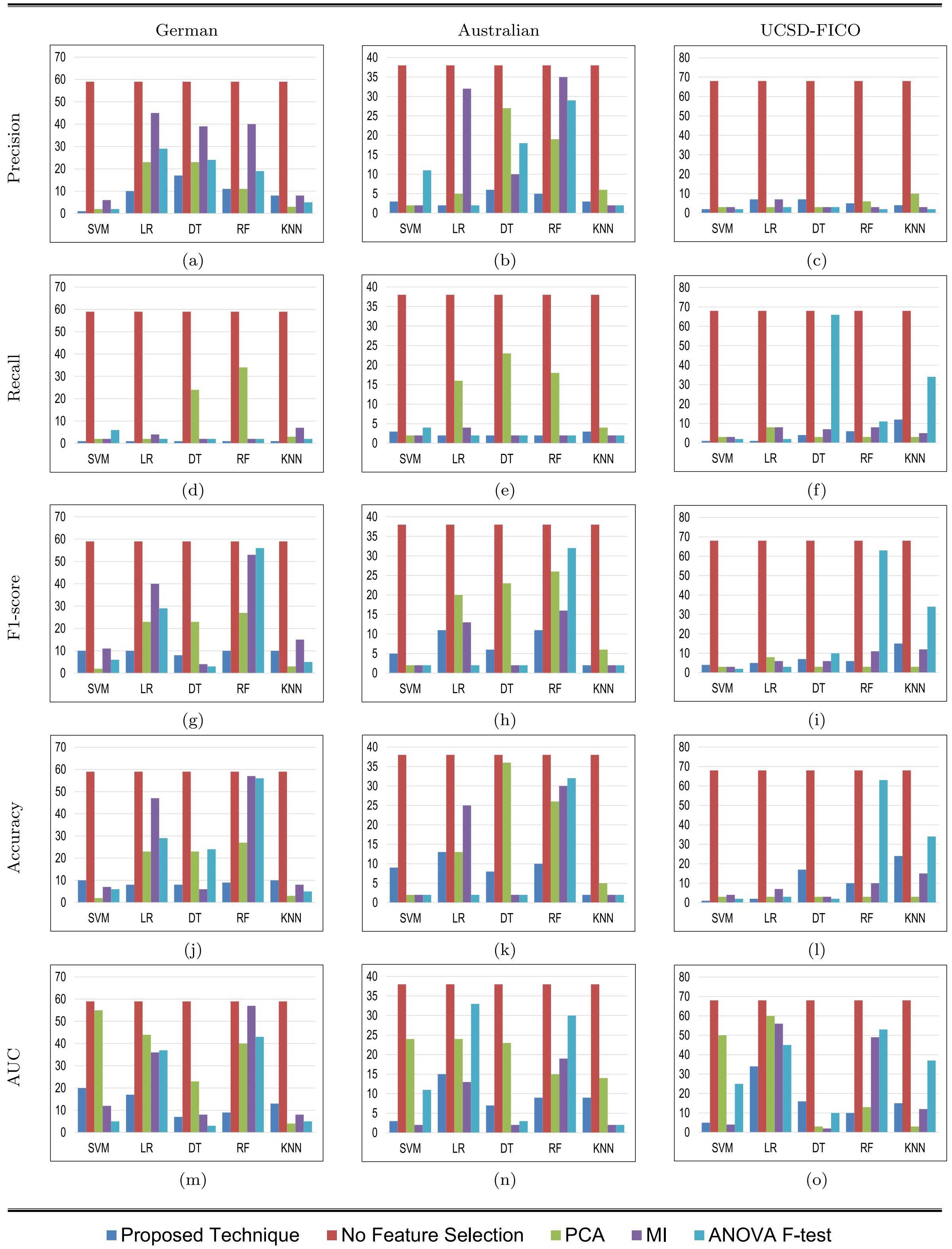
(g) (h) (i)

(j) (k) (l)

(m) (n) (o)

Fig. 2. Performance evaluation results with five metrics as precision, recall, F1-score, accuracy, and AUC with the five different feature selection techniques. No feature selection indicates the performance evaluation conducted without applying any feature selection techniques.

many variables develop more splits that result in a bigger depth of a generated tree, it often causes an overfitting problem with poor classification performance (Oates & Jensen, 1997). Therefore, in DT, maximum performance is often produced using fewer features. As discussed above, MI has been used broadly as a good feature selection technique. It measures information gain to determine dependencies between variables. Thus, MI has been used as an alternative method to evaluate the purity of nodes in DT (Nowozin, 2012). Due to the



(a) (b) (c)

(d) (e) (f)

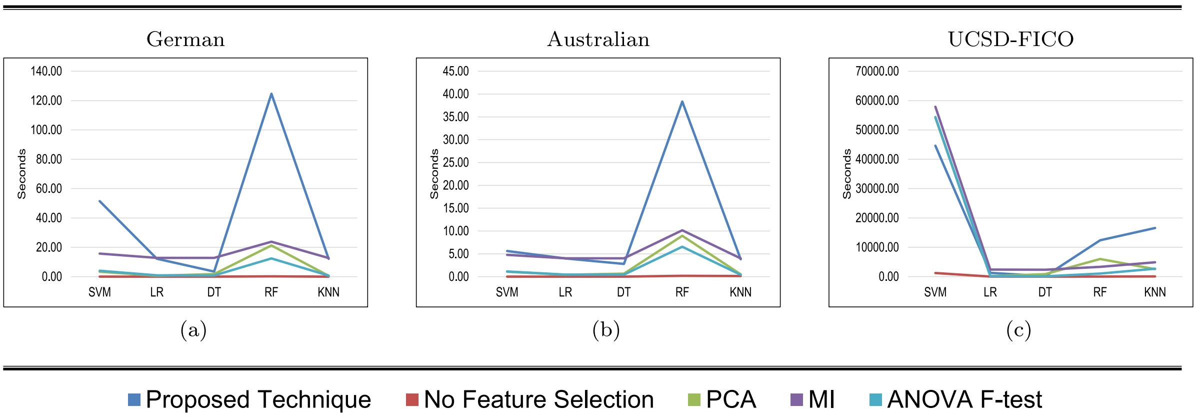
(g) (h) (i)

(j) (k) (l)

(m) (n) (o)

Fig. 3. Determined total number of features with different feature selection techniques, producing high classification performances as shown in Fig. 2. All raw features indicate the total number of input features.

similarity between MI and DT, performance degradation may exist by determining less number of features. But, further study is required to validate this relationship. When analyzing classification performances with different feature selection techniques by differentiating numerical and categorical variables, we observed our proposed technique determined about two ~ three variables among numerical variables to



(a) (b) (c)

Fig. 4. Average time computation of using different feature selection techniques with five classification algorithms. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

produce maximum performances in all classification algorithms. But, other feature selection techniques used mostly numerical variables for maximum performance in classification algorithms (except DT).

Overall, there was no correlation between performance results and the number of features because our analysis mainly focused on determining the total number of features to produce maximum classification performance. Instead, to understand the difference among the feature selection techniques, we utilized a visualization approach to understand the distributions of instances (see the following section).

5.4. *Complexity evaluation*

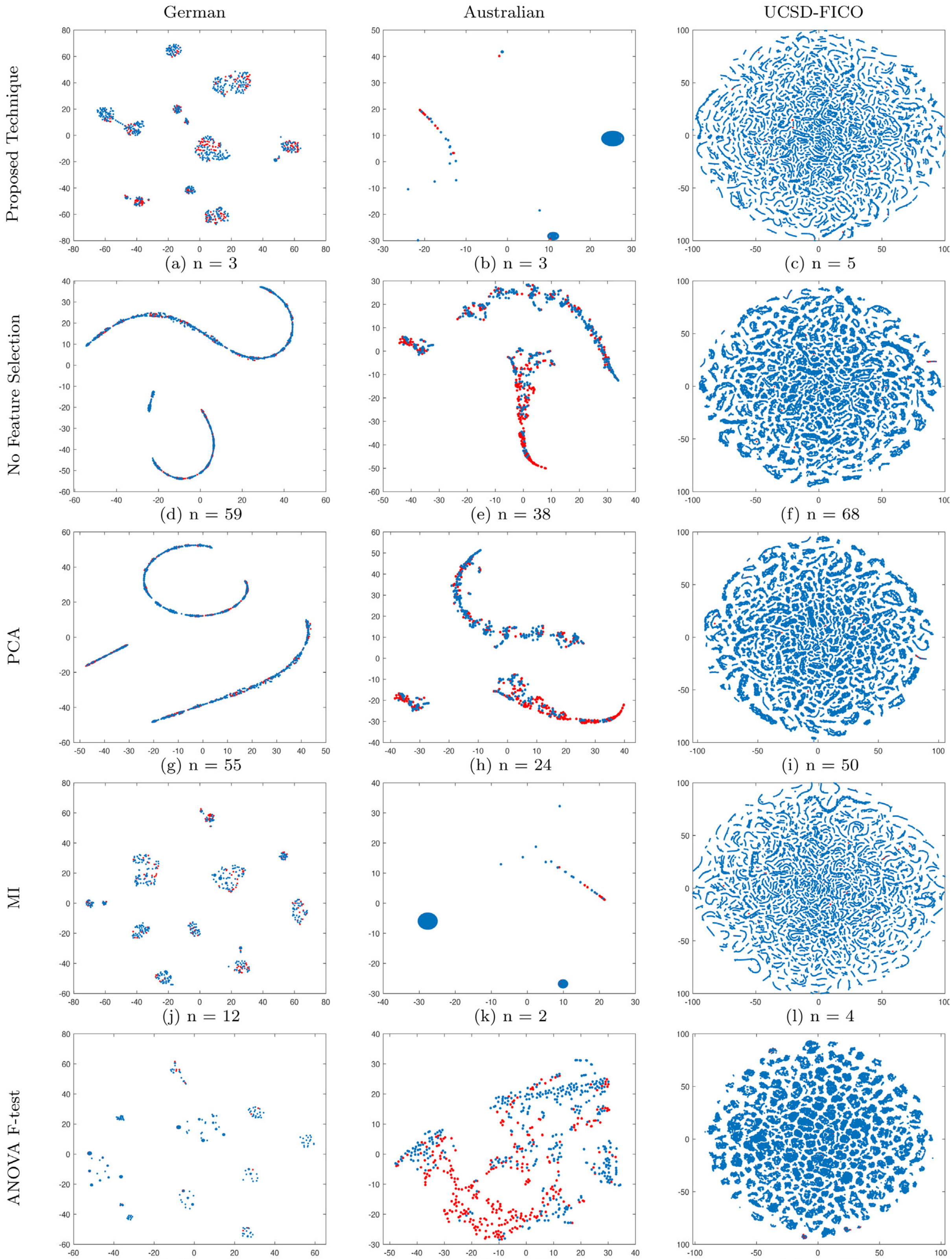
As discussed above, our proposed technique determines the best possible feature to produce maximum performances for mixed attribute data. Evaluating the performance of each feature takes more time depending on the size and number of features in the data. Since the technique also includes k-fold cross-validation when evaluating each feature, measuring the complexity of the technique is closely dependent on the defined cross-validation. Thus, with the consideration of using the cross-validation, the proposed technique requires the time complexity of $O(kn\log n)$. In the worst case, it takes $O(kn^2)$.

Fig. 4 shows the average computational time of our proposed technique compared with other techniques. For a fair comparison, all feature selection techniques were performed with 10-fold cross-validation to determine the optimal number of features. When no feature selection was applied, it required less classification time (see red color in the figure). As shown in Section 5.3, our proposed technique showed an improved classification performance. But, we found that it required more computational time when determining optimal features (see Fig. 4). In general, the classification training time complexity of SVM is $O(n^3)$. It is much higher than other methods (Tsang et al., 2005). Among the classification algorithms, LR, DT, RF, and KNN takes training time complexity of $O(nm)$, $O(nlog(n)m)$, $O(knlog(n)m)$, and $O(knm)$, respectively (Shalev-Shwartz & Ben-David, 2014; Witten et al., 2011). n indicates the number of instances, and m denotes the number of features. For KNN, it determines k nearest neighbors. For RF, k represents generated number of trees. Interestingly, in our study, RF took more computation time than others, especially when analyzing the German and Australian datasets. RF is an ensemble tree-based learning algorithm that builds many individual trees with bootstrapping (Schonlau & Zou, 2020). Thus, it took more computational time but produced more accurate results than DT. We also found that when the data size was small (i.e., the German and Australian datasets), RF required more computational time than SVM (see Fig. 4(a) and (b)). But, for the UCSD-FICO dataset, more computational time was required because the scale of the data was large (see Fig. 4(c)). We also found an interesting result as our proposed technique with KNN took more computational time than other classification algorithms. Although the primary cause of this was not explainable, it is worth having a further study. But, since it is not a major part of this study, we will leave it as future work.

6. Discussion

As shown above, different performance results were observed depending on applied machine learning algorithms. The differences may occur due to each algorithm's distinctive characteristics in classifying different imbalanced mixed datasets. To illustrate the differences among the feature selection techniques, we applied a visualization approach. Since the datasets include high dimensional attributes, a dimension reduction technique is considered to plot the data. In the visualization community, various dimensional reduction techniques, such as PCA, MDS, and LDA, are used to represent high-dimensional data into a lower-dimensional space (i.e., 2D space). While PCA is the most popular method for visually representing high-dimensional data, it must satisfy the linearity of the data requirement. Since our study analyzes mixed-attribute data, it may require a non-linear dimensionality reduction. Therefore, t-distributed stochastic neighbor embedding (t-SNE) is better suited to show the difference among the feature selection techniques. t-SNE is an unsupervised, non-linear technique that visualizes high-dimensional data into a lower-dimensional space by calculating a similarity measure between pairs of instances (van der Maaten & Hinton, 2008).

Fig. 5 shows visual representations with t-SNE. A simple color mapping was used to represent good credit (blue) and bad credit (red) for the German credit dataset, creditworthy (blue) and non-creditworthy (red) for the Australian credit approval dataset, and normal activity (blue) and fraudulent activity (red) for the UCSD-FICO dataset, respectively. t-SNE uses a stochastic neighbor embedding technique, and similar instances are placed in a nearby location with high probability. That is, related instances will be positioned close to each other based on highly relevant features determined by a feature selection technique. For the Australian dataset, we identified that creditworthy instances appeared on the right side of the display space (see Fig. 5(b)). Overall, with t-SNE, we found clear patterns and differences in the German and Australian datasets. However, we observed dense representations for the UCSD-FICO dataset because of large data instances (102,650). Due to the dense patterns, it was difficult to determine distinctive differences among different feature selection techniques. With regard to different feature selection techniques, raw features (i.e., no feature selection technique) and PCA features showed similar visual representations. For the German dataset, three curvy lines were generated. And, we identified that non-creditworthy instances mostly appeared in separated clusters for the Australian dataset. Although many non-creditworthy applicants in the Australian dataset appeared



(a) n = 3 (b) n = 3 (c) n = 5

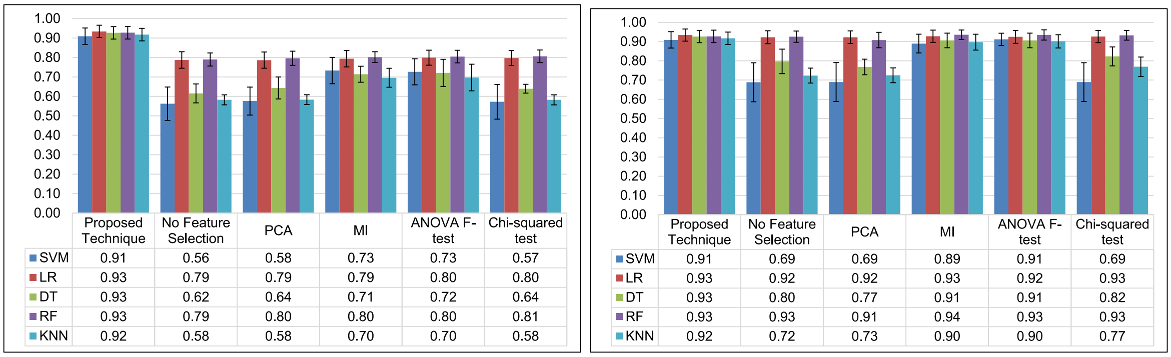
(d) n = 59 (e) n = 38 (f) n = 68

(g) n = 55 (h) n = 24 (i) n = 50

(j) n = 12 (k) n = 2 (l) n = 4

(m) n = 5 (n) n = 11 (o) n = 25

Fig. 5. Visual representations of the determined features (*n*) that produce the best AUC performance with SVM. t-SNE is used to represent data into a 2-dimensional scatterplot. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)



	Proposed Technique	No Feature Selection	PCA	MI	ANOVA F-test	Chi-squared test
SVM	0.91	0.56	0.58	0.73	0.73	0.57
LR	0.93	0.79	0.79	0.79	0.80	0.80
DT	0.93	0.62	0.64	0.71	0.72	0.64
RF	0.93	0.79	0.80	0.80	0.80	0.81
KNN	0.92	0.58	0.58	0.70	0.70	0.58

(a) German dataset

	Proposed Technique	No Feature Selection	PCA	MI	ANOVA F-test	Chi-squared test
SVM	0.91	0.69	0.69	0.89	0.91	0.69
LR	0.93	0.92	0.92	0.93	0.92	0.93
DT	0.93	0.80	0.77	0.91	0.91	0.82
RF	0.93	0.93	0.91	0.94	0.93	0.93
KNN	0.92	0.72	0.73	0.90	0.90	0.77

(b) Australian dataset

Fig. 6. A performance (AUC) comparison between the German and Australian datasets on different feature selection techniques including Chi-squared test.

Table A.1
Performance evaluation result for the German dataset.

		SVM	LR	DT	RF	KNN
Precision	Proposed technique	0.72 ± 0.05	0.78 ± 0.06	0.81 ± 0.05	0.81 ± 0.06	0.85 ± 0.04
	No feature selection	0.71 ± 0.05	0.79 ± 0.05	0.77 ± 0.06	0.77 ± 0.07	0.72 ± 0.06
	PCA	0.71 ± 0.05	0.80 ± 0.06	0.79 ± 0.04	0.78 ± 0.05	0.72 ± 0.06
	MI	0.72 ± 0.05	0.80 ± 0.06	0.80 ± 0.05	0.79 ± 0.05	0.79 ± 0.05
	ANOVA F-test	0.71 ± 0.06	0.80 ± 0.05	0.80 ± 0.04	0.80 ± 0.05	0.77 ± 0.04
Recall	Proposed technique	1.00 ± 0.00	1.00 ± 0.00	1.00 ± 0.00	1.00 ± 0.00	1.00 ± 0.00
	No feature selection	0.99 ± 0.01	0.87 ± 0.05	0.74 ± 0.04	0.92 ± 0.03	0.85 ± 0.03
	PCA	0.98 ± 0.02	0.98 ± 0.02	0.80 ± 0.05	0.96 ± 0.02	0.85 ± 0.03
	MI	1.00 ± 0.00	0.94 ± 0.02	0.94 ± 0.02	0.99 ± 0.01	0.88 ± 0.05
	ANOVA F-test	0.98 ± 0.01	0.96 ± 0.08	0.94 ± 0.09	0.98 ± 0.06	0.87 ± 0.13
F1-score	Proposed technique	0.84 ± 0.03	0.84 ± 0.04	0.84 ± 0.04	0.84 ± 0.03	0.83 ± 0.03
	No feature selection	0.83 ± 0.04	0.83 ± 0.04	0.75 ± 0.04	0.84 ± 0.04	0.78 ± 0.04
	PCA	0.82 ± 0.04	0.83 ± 0.03	0.78 ± 0.03	0.85 ± 0.03	0.78 ± 0.04
	MI	0.83 ± 0.03	0.84 ± 0.04	0.82 ± 0.03	0.85 ± 0.03	0.81 ± 0.05
	ANOVA F-test	0.83 ± 0.04	0.84 ± 0.04	0.80 ± 0.05	0.85 ± 0.04	0.81 ± 0.05
Accuracy	Proposed technique	0.75 ± 0.05	0.77 ± 0.05	0.76 ± 0.04	0.75 ± 0.05	0.75 ± 0.04
	No feature selection	0.71 ± 0.05	0.75 ± 0.05	0.67 ± 0.04	0.75 ± 0.06	0.66 ± 0.06
	PCA	0.71 ± 0.05	0.76 ± 0.04	0.70 ± 0.04	0.77 ± 0.05	0.66 ± 0.06
	MI	0.71 ± 0.03	0.75 ± 0.05	0.72 ± 0.06	0.78 ± 0.05	0.72 ± 0.05
	ANOVA F-test	0.71 ± 0.05	0.76 ± 0.05	0.71 ± 0.03	0.77 ± 0.05	0.73 ± 0.06
AUC	Proposed technique	0.76 ± 0.05	0.80 ± 0.05	0.75 ± 0.04	0.76 ± 0.05	0.74 ± 0.04
	No feature selection	0.56 ± 0.09	0.79 ± 0.04	0.62 ± 0.05	0.79 ± 0.03	0.58 ± 0.03
	PCA	0.58 ± 0.07	0.79 ± 0.04	0.64 ± 0.06	0.80 ± 0.04	0.58 ± 0.03
	MI	0.73 ± 0.07	0.79 ± 0.04	0.71 ± 0.04	0.80 ± 0.03	0.70 ± 0.05
	ANOVA F-test	0.73 ± 0.07	0.80 ± 0.04	0.72 ± 0.07	0.80 ± 0.03	0.70 ± 0.07

at the bottom (see red dots in Fig. 5(n)), the distinction was still not clear with the ANOVA feature selection. We identified similar results between our proposed and the MI feature selection techniques.

Chi-Squared χ^2 test (Alelyani et al., 2014) is a widely used supervised feature selection technique. It is often utilized to analyze categorical or mixed-attribute data. It measures the statistical dependencies of variables to determine the likelihood of correlation between attributes. We applied the Chi-Squared test to extract features in the German and Australian datasets. However, it works with non-negative features. Since the UCSD-FICO dataset includes negative values, it was not included in the comparison. Fig. 6 shows AUC performance results with different feature selection techniques. We expected similar performances between MI and Chi-squared test because they are related, often resulting in approximately equivalent rankings and yielding similar results (Fernández-García et al., 2019; Richter et al., 2019). However, we found interesting results between MI and Chi-squared test. In detail, similar performance results were observed when running LR and RF. But, MI showed better performance results with SVM, DT, and KNN. Based on the Pearson correlation coefficient measure, we also found a high similarity ($p < .001$) among the three feature selection techniques (i.e., no feature selection, PCA, and Chi-squared test).

7. Conclusion and future work

Feature selection is critical in data analysis because it can improve data classification performance. Due to this reason, various feature selection techniques have been proposed. Although they are well designed to select features, most feature selection techniques only handle the same data attribute (categorical or numerical). This paper presents a new feature selection technique to analyze mixed attribute data. It determines features with a user-defined computational algorithm and a specified performance criterion. Three imbalanced financial datasets were used to evaluate the effectiveness of the proposed technique. From the evaluation study, we found a performance improvement in classifying instances. Although our proposed technique is good at classifying financial datasets, an extensive evaluation should be performed to determine the effectiveness of the technique. Thus, we plan to extend our evaluation study with varying types of datasets and numerous feature selection techniques.

Since our proposed feature selection technique evaluates all features to determine a list of optimal features that boost classification performances, it often requires a high computational cost. Thus, for our future work, we plan to extend our study of addressing the limitation

Table A.2
Performance evaluation result for the Australian dataset.

		SVM	LR	DT	RF	KNN
Precision	Proposed technique	0.87 ± 0.16	0.88 ± 0.08	0.88 ± 0.07	0.88 ± 0.06	0.86 ± 0.17
	No feature selection	0.75 ± 0.20	0.82 ± 0.09	0.77 ± 0.10	0.87 ± 0.08	0.69 ± 0.07
	PCA	0.75 ± 0.20	0.86 ± 0.09	0.74 ± 0.08	0.87 ± 0.06	0.69 ± 0.07
	MI	0.78 ± 0.08	0.84 ± 0.08	0.82 ± 0.07	0.87 ± 0.08	0.80 ± 0.09
	ANOVA F-test	0.82 ± 0.09	0.79 ± 0.08	0.81 ± 0.08	0.87 ± 0.09	0.81 ± 0.10
Recall	Proposed technique	0.94 ± 0.03	0.93 ± 0.04	0.94 ± 0.03	0.94 ± 0.03	0.94 ± 0.03
	No feature selection	0.30 ± 0.13	0.85 ± 0.05	0.79 ± 0.06	0.84 ± 0.07	0.57 ± 0.07
	PCA	0.30 ± 0.13	0.89 ± 0.05	0.75 ± 0.08	0.84 ± 0.06	0.58 ± 0.07
	MI	0.92 ± 0.04	0.93 ± 0.04	0.91 ± 0.04	0.92 ± 0.03	0.87 ± 0.09
	ANOVA F-test	0.93 ± 0.04	0.92 ± 0.04	0.92 ± 0.04	0.92 ± 0.04	0.88 ± 0.10
F1-score	Proposed technique	0.86 ± 0.06	0.87 ± 0.06	0.86 ± 0.06	0.87 ± 0.05	0.85 ± 0.06
	No feature selection	0.42 ± 0.17	0.84 ± 0.06	0.78 ± 0.07	0.85 ± 0.06	0.62 ± 0.06
	PCA	0.42 ± 0.17	0.86 ± 0.06	0.74 ± 0.05	0.85 ± 0.05	0.63 ± 0.06
	MI	0.85 ± 0.05	0.86 ± 0.05	0.84 ± 0.06	0.86 ± 0.06	0.83 ± 0.05
	ANOVA F-test	0.85 ± 0.06	0.85 ± 0.06	0.85 ± 0.06	0.87 ± 0.06	0.83 ± 0.06
Accuracy	Proposed technique	0.88 ± 0.04	0.88 ± 0.04	0.88 ± 0.05	0.88 ± 0.04	0.86 ± 0.05
	No feature selection	0.65 ± 0.06	0.85 ± 0.05	0.80 ± 0.07	0.87 ± 0.05	0.70 ± 0.04
	PCA	0.65 ± 0.06	0.87 ± 0.04	0.77 ± 0.06	0.87 ± 0.04	0.70 ± 0.04
	MI	0.85 ± 0.05	0.87 ± 0.06	0.85 ± 0.06	0.88 ± 0.05	0.84 ± 0.05
	ANOVA F-test	0.86 ± 0.05	0.86 ± 0.05	0.86 ± 0.05	0.88 ± 0.05	0.85 ± 0.05
AUC	Proposed technique	0.91 ± 0.04	0.93 ± 0.03	0.93 ± 0.03	0.93 ± 0.03	0.92 ± 0.03
	No feature selection	0.69 ± 0.10	0.92 ± 0.03	0.80 ± 0.06	0.93 ± 0.03	0.72 ± 0.04
	PCA	0.69 ± 0.10	0.92 ± 0.03	0.77 ± 0.04	0.91 ± 0.04	0.73 ± 0.04
	MI	0.89 ± 0.05	0.93 ± 0.03	0.91 ± 0.04	0.94 ± 0.02	0.90 ± 0.04
	ANOVA F-test	0.91 ± 0.03	0.92 ± 0.03	0.91 ± 0.04	0.93 ± 0.03	0.90 ± 0.03

Table A.3
Performance evaluation result for the UCSD-FICO dataset.

		SVM	LR	DT	RF	KNN
Precision	Proposed technique	0.82 ± 0.03	0.83 ± 0.04	0.83 ± 0.03	0.82 ± 0.02	0.82 ± 0.03
	No feature selection	0.81 ± 0.05	0.81 ± 0.05	0.38 ± 0.04	0.80 ± 0.02	0.66 ± 0.03
	PCA	0.81 ± 0.05	0.81 ± 0.05	0.46 ± 0.05	0.81 ± 0.02	0.66 ± 0.04
	MI	0.82 ± 0.03	0.81 ± 0.05	0.81 ± 0.03	0.81 ± 0.03	0.80 ± 0.04
	ANOVA F-test	0.82 ± 0.03	0.81 ± 0.05	0.82 ± 0.03	0.82 ± 0.03	0.82 ± 0.03
Recall	Proposed technique	0.27 ± 0.04	0.18 ± 0.03	0.51 ± 0.04	0.54 ± 0.04	0.46 ± 0.03
	No feature selection	0.19 ± 0.03	0.18 ± 0.03	0.43 ± 0.04	0.40 ± 0.03	0.42 ± 0.03
	PCA	0.19 ± 0.03	0.18 ± 0.03	0.45 ± 0.05	0.45 ± 0.04	0.45 ± 0.03
	MI	0.27 ± 0.04	0.18 ± 0.03	0.46 ± 0.05	0.45 ± 0.04	0.41 ± 0.03
	ANOVA F-test	0.27 ± 0.04	0.18 ± 0.03	0.43 ± 0.04	0.42 ± 0.05	0.42 ± 0.03
F1-score	Proposed technique	0.40 ± 0.05	0.30 ± 0.04	0.51 ± 0.03	0.63 ± 0.03	0.55 ± 0.03
	No feature selection	0.31 ± 0.05	0.29 ± 0.04	0.40 ± 0.04	0.53 ± 0.03	0.51 ± 0.03
	PCA	0.31 ± 0.05	0.30 ± 0.04	0.46 ± 0.04	0.57 ± 0.03	0.53 ± 0.02
	MI	0.40 ± 0.05	0.30 ± 0.04	0.43 ± 0.04	0.56 ± 0.04	0.51 ± 0.03
	ANOVA F-test	0.40 ± 0.05	0.30 ± 0.04	0.42 ± 0.03	0.54 ± 0.03	0.51 ± 0.03
Accuracy	Proposed technique	0.98 ± 0.00	0.98 ± 0.00	0.98 ± 0.00	0.98 ± 0.00	0.98 ± 0.00
	No feature selection	0.98 ± 0.00	0.98 ± 0.00	0.97 ± 0.00	0.98 ± 0.00	0.98 ± 0.00
	PCA	0.98 ± 0.00	0.98 ± 0.00	0.97 ± 0.00	0.98 ± 0.00	0.98 ± 0.00
	MI	0.98 ± 0.00	0.98 ± 0.00	0.98 ± 0.00	0.98 ± 0.00	0.98 ± 0.00
	ANOVA F-test	0.98 ± 0.00	0.98 ± 0.00	0.98 ± 0.00	0.98 ± 0.00	0.98 ± 0.00
AUC	Proposed technique	0.71 ± 0.02	0.79 ± 0.02	0.80 ± 0.01	0.84 ± 0.02	0.78 ± 0.02
	No feature selection	0.71 ± 0.03	0.78 ± 0.02	0.71 ± 0.02	0.83 ± 0.02	0.77 ± 0.02
	PCA	0.71 ± 0.02	0.78 ± 0.02	0.73 ± 0.02	0.83 ± 0.02	0.78 ± 0.02
	MI	0.71 ± 0.02	0.78 ± 0.02	0.73 ± 0.02	0.84 ± 0.02	0.77 ± 0.02
	ANOVA F-test	0.71 ± 0.03	0.78 ± 0.02	0.75 ± 0.03	0.84 ± 0.02	0.77 ± 0.02

by utilizing high computing resources to speed up the evaluation of features. Specifically, the utilization of parallel distributed computing is considered to evaluate features in multiple computing nodes simultaneously. In detail, a known distributed processing system (called Apache Spark Zaharia et al., 2016) will be used to evaluate classification performances depending on features. Since it is designed using a MapReduce model to support in-memory cluster computing with providing distributed machine learning framework, it speeds up the data analysis process through cloud-based parallel operations by utilizing numerous computing nodes. In addition, due to the size of the two datasets is small. We plan to extend our study to applying the proposed technique into various large scale datasets.

CRediT authorship contribution statement

Dong Hyun Jeong: Conceptualization, Methodology, Formal analysis, Validation, Writing – original draft. **Bong Keun Jeong:** Validation, Writing – review & editing. **Nandi Leslie:** Writing – review & editing. **Charles Kamhoua:** Writing – review & editing. **Soo-Yeon Ji:** Formal analysis, Methodology, Investigation, Supervision, Writing – review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Acknowledgments

This material is based upon work supported by the National Science Foundation, USA under Grant No. (2107451 and 2219634).

Appendix. Detailed list of evaluation results

See Tables A.1–A.3.

References

Aggarwal, C. C. (2013). Outlier detection in categorical, text and mixed attribute data. In *Outlier analysis* (pp. 199–223). New York, NY: Springer New York, http://dx.doi.org/10.1007/978-1-4614-6396-2_7.

Aida-Zade, K., Mustafayev, E., & Rustamov, S. (2017). Comparison of deep learning in neural networks on CPU and GPU-based frameworks. In *2017 IEEE 11th international conference on application of information and communication technologies* AICT, (pp. 1–4). http://dx.doi.org/10.1109/ICAICT.2017.8687085.

Alelyani, S., Tang, J., & Liu, H. (2014). Feature selection for clustering: A review. In *Data clustering: Algorithms and applications* (pp. 29–60). http://dx.doi.org/10.1201/9781315373515-2.

Bellizio, F., Cremer, J. L., Sun, M., & Strbac, G. (2021). A causality based feature selection approach for data-driven dynamic security assessment. *Electric Power Systems Research*, *201*, Article 107537. http://dx.doi.org/10.1016/j.epsr.2021.107537.

Bolón-Canedo, V., Sánchez-Maroño, N., & Alonso-Betanzos, A. (2013). A review of feature selection methods on synthetic data. *Knowledge and Information Systems*, *34*(3), 483–519. http://dx.doi.org/10.1007/s10115-012-0487-8.

Boulgouris, N. V., Plataniotis, K. N., & Micheli-Tzanakou, E. (2010). Discriminant analysis for dimensionality reduction: An overview of recent developments. In *Biometrics: Theory, methods, and applications* (pp. 1–19). http://dx.doi.org/10.1002/9780470522356.ch1.

Branco, P., Torgo, L., & Ribeiro, R. P. (2016). A survey of predictive modeling on imbalanced domains. *ACM Computing Surveys*, *49*(2), http://dx.doi.org/10.1145/2907070.

Brezočnik, L., Fister, I., & Podgorelec, V. (2018). Swarm intelligence algorithms for feature selection: A review. *Applied Sciences*, *8*(9), http://dx.doi.org/10.3390/app8091521.

Rey-del Castillo, P., & Cardeñosa, J. (2012). Fuzzy min–max neural networks for categorical data: application to missing data imputation. *Neural Computing and Applications*, *21*(6), 1349–1362. http://dx.doi.org/10.1007/s00521-011-0574-x.

Cerda, P., Varoquaux, G., & Kégl, B. (2018). Similarity encoding for learning with dirty categorical variables. *Machine Learning*, *107*(8), 1477–1494. http://dx.doi.org/10.1007/s10994-018-5724-2.

Chandrashekar, G., & Sahin, F. (2014). A survey on feature selection methods. *Computers & Electrical Engineering*, *40*(1), 16–28. http://dx.doi.org/10.1016/j.compeleceng.2013.11.024, 40th-year commemorative issue.

Chen, L. (2009). Curse of dimensionality. In L. Liu, & M. T. Özsu (Eds.), *Encyclopedia of database systems* (pp. 545–546). Boston, MA: Springer US, http://dx.doi.org/10.1007/978-0-387-39940-9_133.

Dash, M., & Liu, H. (1997). Feature selection for classification. *Intelligent Data Analysis*, *1*(1), 131–156. http://dx.doi.org/10.1016/S1088-467X(97)00008-5.

Ding, C., & He, X. (2004). K-means clustering via principal component analysis. In *Proceedings of the twenty-first international conference on machine learning* ICML '04, (p. 29). New York, NY, USA: Association for Computing Machinery, http://dx.doi.org/10.1145/1015330.1015408.

Ding, S., Zhu, H., Jia, W., & Su, C. (2012). A survey on feature extraction for pattern recognition. *Artificial Intelligence Review*, *37*, 169–180. http://dx.doi.org/10.1007/s10462-011-9225-y.

Dua, D., & Graff, C. (2017). UCI machine learning repository. URL: http://archive.ics.uci.edu/ml.

Elssied, N., Ibrahim, A. P. D. O., & Hamza Osman, A. (2014). A novel feature selection based on one-way ANOVA F-test for E-mail spam classification. *Research Journal of Applied Sciences, Engineering and Technology*, *7*, 625–638. http://dx.doi.org/10.19026/rjaset.7.299.

Famili, A., Shen, W.-M., Weber, R., & Simoudis, E. (1997). Data preprocessing and intelligent data analysis. *Intelligent Data Analysis*, *1*(1), 3–23. http://dx.doi.org/10.1016/S1088-467X(98)00007-9.

Fernández-García, A. J., Iribarne, L., Corral, A., Criado, J., & Wang, J. Z. (2019). A recommender system for component-based applications using machine learning techniques. *Knowledge-Based Systems*, *164*, 68–84. http://dx.doi.org/10.1016/j.knosys.2018.10.019.

Fodor, I. K. (2002). *A survey of dimension reduction techniques*: *Technical Report*, CA (US): Lawrence Livermore National Lab., URL: https://www.osti.gov/biblio/15002155.

Gao, S., Steeg, G. V., & Galstyan, A. (2015). Efficient estimation of mutual information for strongly dependent variables. In *18th international conference on artificial intelligence and statistics (AISTATS)'15* (pp. 277–286). URL: http://arxiv.org/abs/1411.2003.

Guo, Q., Wu, W., Massart, D., Boucon, C., & de Jong, S. (2002). Feature selection in principal component analysis of analytical data. *Chemometrics and Intelligent Laboratory Systems*, *61*(1), 123–132. http://dx.doi.org/10.1016/S0169-7439(01)00203-9.

Guyon, I., & Elisseeff, A. (2003). An introduction to variable and feature selection. *Journal of Machine Learning Research*, *3*, 1157–1182, URL: http://jmlr.org/papers/v3/guyon03a.html.

He, H., & Garcia, E. A. (2009). Learning from imbalanced data. *IEEE Transactions on Knowledge and Data Engineering*, *21*(9), 1263–1284. http://dx.doi.org/10.1109/TKDE.2008.239.

He, H., & Ma, Y. (2013). *Imbalanced learning: Foundations, algorithms, and applications* (1st ed.). Wiley-IEEE Press, http://dx.doi.org/10.1002/9781118646106.

Jain, D., & Singh, V. (2021). A two-phase hybrid approach using feature selection and Adaptive SVM for chronic disease classification. *International Journal of Computers and Applications*, *43*(6), 524–536. http://dx.doi.org/10.1080/1206212X.2019.1577534.

Jolliffe, I. T. (2002). *Springer series in statistics*, *Principal component analysis*. New York: Springer-Verlag, http://dx.doi.org/10.1007/b98835.

Jolliffe, I. T., & Cadima, J. (2016). Principal component analysis: a review and recent developments. *Philosophical Transactions of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, *374*(2065), http://dx.doi.org/10.1098/rsta.2015.0202.

Jovic, A., Brkic, K., & Bogunovic, N. (2015). A review of feature selection methods with applications. In P. Biljanovic, Z. Butkovic, K. Skala, B. Mikac, M. Cicin-Sain, V. Sruk, S. Ribaric, S. Gros, B. Vrdoljak, M. Mauher, & A. Sokolic (Eds.), *38th international convention on information and communication technology, electronics and microelectronics, MIPRO 2015, Opatija, Croatia, May 25-29, 2015* (pp. 1200–1205). IEEE, http://dx.doi.org/10.1109/MIPRO.2015.7160458.

Kohavi, R., & John, G. H. (1997). Wrappers for feature subset selection. *Artificial Intelligence*, *97*(1), 273–324. http://dx.doi.org/10.1016/S0004-3702(97)00043-X.

Li, J., Cheng, K., Wang, S., Morstatter, F., Trevino, R. P., Tang, J., & Liu, H. (2018). Feature selection: A data perspective. *ACM Computing Surveys*, *50*(6), 94:1–94:45. http://dx.doi.org/10.1145/3136625.

Li, W., Feng, F., Li, H., & Du, Q. (2018). Discriminant analysis-based dimension reduction for hyperspectral image classification: A survey of the most recent advances and an experimental comparison of different techniques. *IEEE Geoscience and Remote Sensing Magazine*, *6*(1), 15–34. http://dx.doi.org/10.1109/MGRS.2018.2793873.

Li, M., Wang, H., Yang, L., Liang, Y., Shang, Z., & Wan, H. (2020). Fast hybrid dimensionality reduction method for classification based on feature selection and grouped feature extraction. *Expert Systems with Applications*, *150*, Article 113277. http://dx.doi.org/10.1016/j.eswa.2020.113277.

Li, X., Zhang, G., Huang, H. H., Wang, Z., & Zheng, W. (2016). Performance analysis of GPU-based convolutional neural networks. In *2016 45th international conference on parallel processing* ICPP, (pp. 67–76). http://dx.doi.org/10.1109/ICPP.2016.15.

van der Maaten, L., & Hinton, G. (2008). Visualizing data using t-SNE. *Journal of Machine Learning Research*, *9*(86), 2579–2605, URL: http://jmlr.org/papers/v9/vandermaaten08a.html.

Manbari, Z., Tab, F. A., & Salavati, C. (2019). Hybrid fast unsupervised feature selection for high-dimensional data. *Expert Systems with Applications*, *124*, 97–118. http://dx.doi.org/10.1016/j.eswa.2019.01.016.

Martinez, A., & Kak, A. (2001). PCA versus LDA. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *23*(2), 228–233. http://dx.doi.org/10.1109/34.908974.

Nowozin, S. (2012). Improved information gain estimates for decision tree induction. In *ICML 2012*. URL: https://www.microsoft.com/en-us/research/publication/improved-information-gain-estimates-for-decision-tree-induction/.

Oates, T., & Jensen, D. D. (1997). The effects of training set size on decision tree complexity. In D. H. Fisher (Ed.), *Proceedings of the fourteenth international conference on machine learning (ICML 1997), Nashville, Tennessee, USA, July 8-12, 1997* (pp. 254–262). Morgan Kaufmann, URL: https://dl.acm.org/doi/10.5555/645526.657136.

Peng, H., Long, F., & Ding, C. (2005). Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *27*(8), 1226–1238. http://dx.doi.org/10.1109/TPAMI.2005.159.

Pohar, M., Blas, M., & Turk, S. (2004). Comparison of logistic regression and linear discriminant analysis: A simulation study. *Metodoloski Zvezki*, *1*(1), 143–161. http://dx.doi.org/10.51936/ayrt6204.

Rashid, A., Siddique, M. J., & Ahmed, S. M. (2020). Machine and deep learning based comparative analysis using hybrid approaches for intrusion detection system. In *2020 3rd international conference on advancements in computational sciences* ICACS, (pp. 1–9). http://dx.doi.org/10.1109/ICACS47775.2020.9055946.

Rawlings, J. O., Pantula, S. G., & Dickey, D. A. (1998). *Springer texts in statistics*, *Applied regression analysis* (2nd ed.). New York, NY: Springer, URL: https://link.springer.com/book/10.1007/b98890.

Richter, B., Knichel, D., & Moradi, A. (2019). A comparison of χ^2-test and mutual information as distinguisher for side-channel analysis. In S. Belaïd, & T. Güneysu (Eds.), *Lecture notes in computer science*: *vol. 11833*, *Smart card research and advanced applications - 18th international conference, CARDIS 2019, Prague, Czech Republic, November 11-13, 2019, Revised selected papers* (pp. 237–251). Springer, http://dx.doi.org/10.1007/978-3-030-42068-0_14.

Ross, B. C. (2014). Mutual information between discrete and continuous data sets. *PLoS ONE*, *9*(2), Article e87357. http://dx.doi.org/10.1371/journal.pone.0087357.

Saeys, Y., Inza, I., & Larranaga, P. (2007). A review of feature selection techniques in bioinformatics. *Bioinformatics (Oxford, England)*, *23*, 2507–2517. http://dx.doi.org/10.1093/bioinformatics/btm344.

Schonlau, M., & Zou, R. Y. (2020). The random forest algorithm for statistical learning. *The Stata Journal*, *20*(1), 3–29. http://dx.doi.org/10.1177/1536867X20909688.

Sethi, J. K., & Mittal, M. (2019). A new feature selection method based on machine learning technique for air quality dataset. *Journal of Statistics and Management Systems*, *22*(4), 697–705. http://dx.doi.org/10.1080/09720510.2019.1609726.

Shalev-Shwartz, S., & Ben-David, S. (2014). *Understanding machine learning - from theory to algorithms* (pp. I–XVI, 1–397). Cambridge University Press, http://dx.doi.org/10.1017/CBO9781107298019.

Solorio-Fernández, S., Carrasco-Ochoa, J. A., & Martínez-Trinidad, J. F. (2020). A review of unsupervised feature selection methods. *Artificial Intelligence Review*, *53*(2), 907–948. http://dx.doi.org/10.1007/s10462-019-09682-y.

Stefanowski, J. (2016). Dealing with data difficulty factors while learning from imbalanced data. In S. Matwin, & J. Mielniczuk (Eds.), *Challenges in computational statistics and data mining* (pp. 333–363). Cham: Springer International Publishing, http://dx.doi.org/10.1007/978-3-319-18781-5_17.

Szczepańska, A. (2011). Research design and statistical analysis, third edition by Jerome L. Myers, Arnold D. Well, Robert F. Lorch, Jr. *International Statistical Review*, *79*(3), 491–492. http://dx.doi.org/10.1111/j.1751-5823.2011.00159_12.x.

Tan, P.-N., Steinbach, M., & Kumar, V. (2005). *Introduction to data mining* (1st ed.). Boston, MA, USA: Addison-Wesley Longman Publishing Co., Inc..

Tharwat, A., Gaber, T., Ibrahim, A., & Hassanien, A. E. (2017). Linear discriminant analysis: A detailed tutorial. *AI Communications*, *30*, 169–190. http://dx.doi.org/10.3233/AIC-170729, 2.

Tian, H., Chen, S.-C., & Shyu, M.-L. (2020). Evolutionary programming based deep learning feature selection and network construction for visual data classification. *Information Systems Frontiers*, *22*(5), 1053–1066. http://dx.doi.org/10.1007/s10796-020-10023-6.

Tsang, I. W., Kwok, J. T., & Cheung, P.-M. (2005). Core vector machines: Fast SVM training on very large data sets. *Journal of Machine Learning Research*, *6*, 363–392, URL: http://jmlr.org/papers/v6/tsang05a.html.

Wang, S. J., Yan, S., Yang, J., Zhou, C. G., & Fu, X. (2014). A general exponential framework for dimensionality reduction. *IEEE Transactions on Image Processing*, *23*(2), 920–930. http://dx.doi.org/10.1109/TIP.2013.2297020.

Witten, I. H., Frank, E., & Hall, M. A. (2011). *Data mining: Practical machine learning tools and techniques* (3rd ed.). San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., http://dx.doi.org/10.1016/C2009-0-19715-5.

Xie, Z., Quirino, T., Shyu, M.-L., Chen, S.-C., & Chang, L. (2006). A distributed agent-based approach to intrusion detection using the lightweight PCC anomaly detection classifier. In *IEEE international conference on sensor networks, ubiquitous, and trustworthy computing (SUTC'06), Vol. 1* (p. 8). http://dx.doi.org/10.1109/SUTC.2006.1636211.

Yu, K., Guo, X., Liu, L., Li, J., Wang, H., Ling, Z., & Wu, X. (2020). Causality-based feature selection: Methods and evaluations. *ACM Computing Surveys*, *53*(5), http://dx.doi.org/10.1145/3409382.

Zaharia, M., Xin, R. S., Wendell, P., Das, T., Armbrust, M., Dave, A., Meng, X., Rosen, J., Venkataraman, S., Franklin, M. J., Ghodsi, A., Gonzalez, J., Shenker, S., & Stoica, I. (2016). Apache spark: A unified engine for big data processing. *Communications of the ACM*, *59*(11), 56–65. http://dx.doi.org/10.1145/2934664.

Zebari, R., Abdulazeez, A., Zeebaree, D., Zebari, D., & Saeed, J. (2020). A comprehensive review of dimensionality reduction techniques for feature selection and feature extraction. *Journal of Applied Science and Technology Trends*, *1*(2), 56–70. http://dx.doi.org/10.38094/jastt1224.

Zhu, X., Zhang, S., Jin, Z., Zhang, Z., & Xu, Z. (2011). Missing value estimation for mixed-attribute data sets. *IEEE Transactions on Knowledge and Data Engineering*, *23*(1), 110–121. http://dx.doi.org/10.1109/TKDE.2010.99.