Hyperparameter Optimized Classification Pipeline for Handling Unbalanced Urban and Rural Energy Consumption Patterns

Deepak Kumar Panda[1], Saptarshi Das[1,2], and Stuart Townley[1,3]

1) Department of Mathematics, College of Engineering, Mathematics and Physical Sciences, University of Exeter, Penryn Campus, Cornwall TR10 9FE, United Kingdom. E-mail: dp457@exeter.ac.uk.

2) Institute for Data Science and Artificial Intelligence, University of Exeter, North Park Road, Exeter, Devon EX4 4QE, United Kingdom. E-mail: saptarshi.das@ieee.org, s.das3@exeter.ac.uk.

3) Environment and Sustainability Institute, University of Exeter, Penryn Campus, Cornwall TR10 9FE, United Kingdom. E-mail: s.b.townley@exeter.ac.uk.

***Abstract*—Energy consumer locations are required for framing effective energy policies. However, due to privacy concerns, it is becoming increasingly difficult to obtain the locational data of the consumers. Machine learning (ML) based classification strategies can be used to find the locational information of the consumers based on their historical energy consumption patterns. The ML methods in this paper are applied to the Residential Energy Consumption Survey 2009 dataset. In this dataset, the number of consumers in the urban area is higher than the rural area, thus making the classification problem unbalanced. The unbalanced classification problem has been solved in original and transformed or reduced feature space using Monte Carlo based under-sampling of the majority class datapoints. The hyperparameters for each classification algorithm family is represented as an optimized pipeline, obtained using the genetic programming (GP) optimizer. The classification performance metrics are then obtained for different algorithm families on the original and transformed feature spaces. Performance comparisons have been reported using univariate and bivariate distributions of the classification metrics viz. accuracy, geometric mean score (GMS), F_1 score, precision, area under the curve (AUC) of receiver operator characteristics (ROC). The energy policy aspects for the urban and rural residential consumers based on the classification results have also been discussed.**

***Keywords:* residential energy consumption, unbalanced data classification, ROC curve, genetic programming**

1. INTRODUCTION

Since the 2015 Paris Agreement (Shigetomi et al., 2018), there has been a growing interest in energy-related activities and sustainable transitions worldwide. Moreover, the United Nations (UN) Sustainable Development Goals (SDG) have highlighted the challenges of climate change and increasing energy consumption (Arora & Mishra, 2019). Amongst different energy-related activities, residential energy consumption (REC) plays a vital role as it is an essential driver in formulating national energy policies (De Lauretis, Ghersi, & Cayla, 2017; Mrówczynska, Skiba, Bazan-Krzywosza’nska, & Sztubecka, 2020). The dual nature of the urban and rural economy and regional economic development differences are essential factors (M. Li, Shan, Hernandez, Mallampalli, & Patiño-Echeverri, 2019) causing a disproportion in the REC usage (Krey et al., 2012), and analyzing them is essential in formulating energy-efficient policies.

As mentioned in (Krey et al., 2012), urbanization is helpful for economic development, which leads to relaxation on natural resources and overall improvement of well-being. Due to variability in economic structure and income levels in urban and rural areas, the behavioral pattern in energy usage differs along with the fuel usage. The primary reasons behind the differentiation between urban and rural energy are threefold. Firstly, new emission pathways can be explored due to the differences and the aggregate effects of urbanization. Secondly, it helps in representing key processes causing a differentiation in the usage pattern in integrated assessment or macroeconomic models. Thirdly, the disaggregated analysis of urban and rural households helps evaluate broader indicators like energy poverty and welfare and health impacts. The existing works in urban and rural energy consumption mainly focus on the energy sources and income, sociological and psychological parameters which causes variability in the consumption.

Besides climatic factors and energy efficiency, locational information is also utilized by the distribution system operators (DSO) to capture energy data to formulate schemes to enhance their business productivity. As per the survey and analysis conducted in (Grünewald & Reisch, 2020), due to privacy issues many consumers do not prefer to give out their locational information to the utility providers even though it leads to framing more efficient energy policies. The household energy usage and carbon emission is correlated with the income of the household members in urban or rural regions (Nie, Kemp, Xu, Vasseur, & Fan, 2018). The current energy policies generally favor the urban residents (Gill & Moeller, 2018), who are more engaged in energy-saving activities than the rural ones (Ding, Wang, Liu, & Long, 2017). However, urban residents emit higher greenhouse gas emissions as compared to their rural counterparts (C. Chen et al., 2019). Thus, the typical characteristics can be utilized by the energy portfolio marketing personnel (K. Li & Du, 2012; Lisovich, Mulligan, & Wicker, 2010) for the targeted advertisement for a specific set of consumers.

Moreover, these privacy concerns can also cause problems for the energy personnel to frame robust energy policies if the location of the consumers is not provided to them (Druckman & Jackson, 2008). Geomasking techniques (Seidl, Paulus, Jankowski, & Regenfelder, 2015) can be utilized where the consumer location data is blinded to the energy data analysts. However, the masked data can affect data accuracy, hence causing problems in framing robust energy policies. Hence, the locational information needs to be identified from energy consumption patterns using a classification technique. In the context of modern and future smart grids, such locational information agnostic classification method based on the residential energy consumption patterns are important component in the information layer of the unified smart grid architecture model (SGAM) so as to respect consumer's privacy, prevent possible unwanted targeted advertisements and cyber-breach etc. (Panda & Das, 2021).

1.1 Previous Works on Consumer Energy Consumption Data in Urban and Rural Areas

Several clustering and regression methods have been applied to the consumer energy consumption time series data in (Figueiredo, Rodrigues, Vale, & Gouveia, 2005; Tsekouras, Kotoulas, Tsirekis, Dialynas, & Hatziargyriou, 2008; Wijaya, Ganu, Chakraborty, Aberer, & Seetharam, 2014; T. Zhang, Zhang, Lu, Feng, & Yang, 2011), (Tso & Yau, 2007), respectively. The daily energy consumption (Tsekouras et al., 2008; Wijaya et al., 2014), context data and user surveys (Wijaya et al., 2014) have been utilized to identify the consumer classes using clustering techniques. Decision tree and neural networks have been used in (Tso & Yau, 2007) to predict the energy consumption data during summer and winter time. Classification schemes have also been used in (Bicego, Farinelli, Grosso, Paolini, & Ramchurn, 2018) to identify users from the electrical consumption data over a specific period.

The urban and rural energy for the economic dualism in less-developed countries has been discussed in (R. Pachauri, 1983) to provide subsidies for energy sources in the rural regions. The concept of energy poverty has been explored in (Khandker, Barnes, & Samad, 2012) which is caused due to income disparities between urban and rural regions. Furthermore, it has been emphasized that along with rural electrification, the deployment of cooking fuels like LPG and subsidies for cleaner energy sources should also be considered by the regional government. Hence, urban and rural consumption studies have been performed to analyse the energy consumptions and carbon emissions so that specific focus can be directed towards policies in respective regions.

Much of the contemporary literature focusing on urban and rural REC and the respective quantitative methods have been summarized in Table 1. The household-level historical data has been utilized to gain insights about the economic, social and technological drivers to the difference in the energy consumption in respective regions (S. Pachauri & Jiang, 2008). Static analysis of energy use in urban and rural use for emission inventories has been conducted in (Dodman, 2011). The impact of climate change on urban and rural energy consumption has been studied in (Fan, Zeng, Hu, Zhang, & Wang, 2020) using stochastic impacts by regression on population, affluence and technology (STIRPAT) model. It has been stressed that the energy governance should be according to the complex climate structure at different geographic regions. Due to higher income, people tend to use a more energy-intensive lifestyle, while technological advancement causes a decrease in energy consumption patterns (Galvin & Sunikka-Blank, 2018). Crucial factors influencing REC have been studied in (Wang, Sun, Zhao, & Wang, 2021), considering household energy consumption (HEC) per capita and respective regions. The "rebound" effect is observed when energy performance objectives are not fulfilled according to the design and technological progress. The rebound effect is more prevalent in rural regions than urban regions (Du et al., 2021), attributed to differences in income distribution and psychological and sociological behaviour. Multilinear regression models have been deployed in (G. Chen et al., 2019) to estimate the effect of the per capita heat consumption, income and incorporating respective cross-quadratic effects. Mental health has also been considered as another factor behind differences in energy consumption patterns. The decision tree classification model has been used to classify the REC in different regions and various categories. As shown in (Niu, Zhang, Zhao, & Niu, 2012), urban residents enjoy high-quality energy while rural residents utilize less effective heat. Driving forces between REC in urban and rural regions have been studied in (Nie et al., 2018), where the decomposition and logarithmic mean Divisia index (LMDI) model consider the climate change effect.

The adaptive weighting divisia with rolling base year index specification has been utilized in (Fan et al., 2013) to identify the quantitative effects of driving components and analyze end-use activities for energy conservation and mitigation of carbon emissions in urban and rural areas. Similarly, input-output structural decomposition analysis of the direct and indirect carbon emissions in (Y. Zhang, 2013) shows that indirect urban emissions are higher due to income disparities. The transition to high energy prices also plays a vital role in REC. Trends of energy use and carbon footprint have been studied in (C. Chen et al., 2019) using the consumer lifecycle approach and carbon footprint to deduce that the lifestyle in the urban region leads to higher emissions and consumption. Three-tier analysis along national, meso and macro-scale has been performed using decomposition method in (Dhakal, 2009) to analyze the energy usage and carbon footprints in urban and rural areas. The cooperation amongst urban and rural societies has been discussed in (Ren, Wu, Ren, & Gao, 2014) while considering natural, energy and economic aspects, thus facilitating a low carbon society with an optimal circulation of energy resources. Moreover, it has also been discussed that the cooperation via bottom-up optimization approach provides the best results as far as economic and environmental aspects are concerned.

Sociological aspects like energy-saving behavior and the other influencing factors have been studied using correlation analysis and multiple regressions in (Ding et al., 2017). The responsibility towards the environment and the knowledge of energy-saving methods plays an essential role in the difference of social behavior. The agent-based model has been used to simulate energy consumption in an urban and rural region in (Thorve et al., 2018) at hourly intervals. The simulation model considers householders demographic and behaviour, appliance ratings, physical characteristics of the home, weather conditions and the usage of fuels. A widely used method known as the non-intrusive load monitoring (NILM) and load disaggregation has also been applied to classify large archival datasets of residential electric appliances and energy usage behaviour of the consumers based on electrical measurements without using their geographical information along with diverse family of classifiers (Ghosh, Panda, Das, & Chatterjee, 2021) that we adopt in the present study.

Due to rapid urbanization, the number of consumers residing in urban areas is becoming skewed as compared to the ones in rural areas. Due to the high skewness of the features towards a particular class, training of the classification model is affected, yielding misleading performance metrics. Hence, special attention needs to be paid to the existing unbalanced classification methods in the following subsection.

Table 1: The study objectives and methods used for urban and rural energy consumptions and emissions in contemporary literature vs the present work

Reference	**Study Objective**	**Method Used**
(Krey et al., 2012)	Urban, rural energy use and CO_2 emissions	Integrated assessment models
(S. Pachauri & Jiang, 2008)	Urban and rural household energy consumption	Aggregate statistics
(Fan et al., 2020)	Urban and rural household energy consumption	Stochastic impacts by regression on population, affluence, and technology
(Wang et al., 2021)	Residential energy consumption	Kaya identity and logarithmic mean division
(Du et al., 2021)	Residential building consumption	Linear approximation of almost ideal demand system

(G. Chen et al., 2019)	Residential energy consumption	Multi-regional input output, tree-based models for classification
(Dhakal, 2009)	Urban energy uses and emissions	Decomposition methods
(Niu et al., 2012)	Urban and rural energy consumption	Multiple regression methods
(Ren et al., 2014)	Urban and rural energy sources	Bottom-up optimization model
(Y. Zhang, 2013)	Urban and rural carbon emissions	Decomposition methods
(Nie et al., 2018)	Residential energy consumption in urban and rural areas	Logarithmic mean divisia index
(Ding et al., 2017)	Urban and rural energy use	Correlation and multiple regression
(C. Chen et al., 2019)	Urban and rural residential energy	Consumer lifecycle approach
This Paper	Urban and rural residential energy	Imbalanced data classification with hyperparameter optimized pipeline

1.2 Previous Works on Unbalanced Classification Methods

The strategies for solving classification problems with unbalanced data can be divided broadly into two categories, as given in (Ganganwar, 2012; Lin et al., 2017). In the first case, the approach is based on modifying the existing classification algorithms like *k*-nearest neighbours (*k*-NN) (Mani & Zhang, 2003; X. Zhang et al., 2017) and naïve Bayes (Frank & Bouckaert, 2006), different variants of Gaussian processes and support vector machine (SVM) with evolutionary optimization methods (Bhowan, Johnston, Zhang, & Yao, 2012) and with adaptive weighted procedures for learning the features of the minority class (Qiao & Liu, 2009). In the second case, training the classifier with resampled features was carried out from majority or minority classes via under or over-sampling method, respectively. The effect of classification strategies on the unbalanced dataset has been reviewed in (Menardi & Torelli, 2014) using various sampling algorithms from the *Imbalanced Learn* toolbox in Python (Lemaitre, Nogueira, & Aridas, 2017). Under-sampling is a popular technique for handling the class imbalance problem, but it ignores the majority class features. As discussed in (Qiao & Liu, 2009), the oversampling can lead to model overfitting, and it will not significantly improve the recognition of the minority class data points. One major drawback of under-sampling the data from the majority class is that it potentially discards the information contained in the majority class data. Hence, several improved under-sampling algorithms like RHSBoost (Gong & Kim, 2017), weighted learning procedure for multi-label classification (Qiao & Liu, 2009), evolutionary algorithms (Garcia & Herrera, 2009; Krawczyk, Galar, Jelen, & Herrera, 2016), clustering of majority features using Fast-CBUS (Ofek, Rokach, Stern, & Shabtai, 2017) and exploratory under-sampling using *EasyEnsemble* and *BalanceCascade* have been described in (Liu, Wu, & Zhou, 2008). It disregards the majority class features without affecting the overall classification performance. The EUSBoost (Krawczyk et al., 2016) involves a boosting strategy where the minority class features are obtained from the test samples. The mixture of under-sampling and oversampling techniques have been described in (Qian, Liang, Li, Feng, & Shi, 2014; Seiffert, Khoshgoftaar, Van Hulse, & Napolitano, 2009). The bagging strategy is implemented for addressing class imbalance problems in (Qian et al., 2014) where random over-sampling is performed for minority class features using SMOTE (Chawla, Bowyer, Hall, & Kegelmeyer, 2002). A hybrid sampling and boosting strategy has been described, combining the boosting algorithm along with random sampling strategies. The metrics obtained after classification from the above strategies do not have any probabilistic interpretation since the classifier was trained on the single instance of the under-sampled data. However, in (Gong & Kim, 2017), random sampling was performed multiple times, and the classifier was updated based on the iteratively weighted training dataset. Hence, probabilistic classification metrics have been reported in (Gong & Kim, 2017).

1.3 Contributions of this Paper

In most of the contemporary research described in subsection 1.2, the classification of the energy consumption data is used to study user behaviors. Machine learning methods have not been explored much in the previous literature for urban and rural REC and emissions, as evident in Table 1. In (G. Chen et al., 2019), decision trees have been utilized to classify the REC into various domains of energy consumption levels. However, in this paper, relative classification of energy consumption sources between urban and rural areas have been performed, which helps in framing policies for the household energies which are correctly or incorrectly classified. Before classification, feature selection using the Fisher's score has been performed, which helps in identifying the differences and similarities in the urban and rural RECs and enable identifying critical features for classification. In addition, machine learning classification models can be utilized by the DSOs to store REC data without needing the locations, which can be used for targeted advertisement for specific regions. Furthermore, in earlier works, the energy policies focus on generic energy consumption. However, in this paper, policy implications have been discussed concerning the specific fuel consumption pattern in urban and rural areas. In addition, there is also a discussion on policy implications for the rural residents classified as urban, which is the main novelty of this paper.

The works described in subsection 1.2 do not involve any detailed analysis of the hyperparameter selection of the classification algorithms for handling unbalanced data. The earlier works are more focused on the performance or modifying ML algorithms for unbalanced data. However, no previous work has been done on the hyperparameter optimization for the ML families for the classification of unbalanced dataset. The optimal hyperparameter search has been conducted by forming a pipeline of the input features for a particular classification family using genetic programming to maximize the overall AUC score. A random under-sampling strategy has been used in designing the optimal pipelines to generate several probabilistic or uncertain classification performance metrics. Interpretation of classification metrics for the dataset is carried out using univariate/bivariate density plots. Similar data analysis is also performed on the classification of the transformed features in a lower-dimensional space viz. principal component analysis (PCA) and *t*-SNE (t-distributed Stochastic Neighbor Embedding).

Thus, we summarize the key contributions of this paper as follows:

- Obtaining the essential features responsible for the differences in urban and rural energy consumptions using the Fisher's score.
- Finding the optimal pipeline of hyperparameters for classification using genetic programming for different machine learning families. The optimal pipeline is used along with Monte Carlo based random under-sampling of data points in the urban class for classifying the REC into rural or urban regions. The analysis of the results is also performed in principal component analysis (PCA) and *t*-stochastic neighbor embedding (*t*-SNE) spaces.
- Policy implications are summarized based on the classification results like urban areas classified as rural and vice versa and the type of fuel used in urban and rural areas. Specific policies are also discussed for the rural residents classified as urban residents by hyper-parameter optimized machine learning models.

2. Feature Selection and Transformation

2.1 Dataset Description

The dataset chosen for classification is adopted from the Residential Energy Consumption Survey 2009 ("Residential Energy Consumption Survey"). United States energy information administration (EIA) conducts the Residential Energy Consumption Survey. Trained interviewers collect the data representing energy characteristics based on the housing unit, usage patterns, and household demographics. The data is obtained from 12083 households selected at random using a complex multistage area probability sample design. The selected samples represent 113.6 million US households, a statistical estimate for the occupied houses obtained from the American Community Survey consisting of over 900 variables (Administration, 2013). The samples represent the energy consumption and expenditures of the residents in different regions. Smaller samples, which are considered in the study, are weighted accordingly to the population. However, with the smaller samples representing the whole census, there are possibilities of sampling errors. The sampling errors represent the confidence interval of the policies to be implemented on a larger census. The REC survey used the method proposed by (Fay, 1989) for the balanced repeated replication (BPR) technique to calculate the sampling error. EIA has not considered those samples where the residual sampling error is higher than 50 or household count is 10. Data confidentiality is maintained where local geographic identifiers are eliminated, and various climatic regions are classified in the same category. Furthermore, the number of floors and apartment size is masked to ensure the privacy of the multi-family residential buildings.

The dataset considered in this paper has 12083 rows and 936 columns, which represents the information regarding the total household's fuel, end-use consumption information along with other household characteristics like location, region and weather. For the given classification task, the energy consumption data is considered. It consists of 39 major features, which includes:

- Total site electricity (in kWh),
- Liquified Petroleum Gas (LPG)/Propane gas (in gallons and BTU),
- Fuel oil (in gallons and BTU),
- Natural gas (in cubic feet),
- Kerosene (in gallons and BTU),
- Wood consumption (in Cords).

The dataset also summarizes the energy usage for different applications in space heating, refrigeration, air conditioning, water heating and other miscellaneous purposes along with the associated cost. Since the energy units are correlated with each other, it is essential to optimize the feature space by removing the highly correlated variables $\mathbf{x} \in \mathrm{IR}^n$ and $\mathbf{y} \in \mathrm{IR}^n$ using Pearson's correlation coefficient $\rho_{\mathrm{x,y}}$ given by:

$$\rho_{X,Y} = \sum_{i=1}^{n} (x_i - \bar{x})(y_i - \bar{y}) \bigg/ \sqrt{\sum_{i=1}^{n} (x_i - \bar{x})^2 \sum_{i=1}^{n} (y_i - \bar{y})^2} \, . \tag{1}$$

The features with a correlation coefficient higher than 0.95 are filtered out while only keeping the relevant features. The dataset has a typical characteristic that not all consumers use different energy sources. It is observed that very few urban residences prefer to use electricity for space and water heating, while rural households prefer to use natural gas for the same applications. It is also observed that fuel oil used for the heating application is rarely by urban and rural households as compared to other energy sources. Other similarities also include electricity usage for refrigeration purpose. The schematic for consumer classification based on their consumption patterns has been shown in Figure 1.

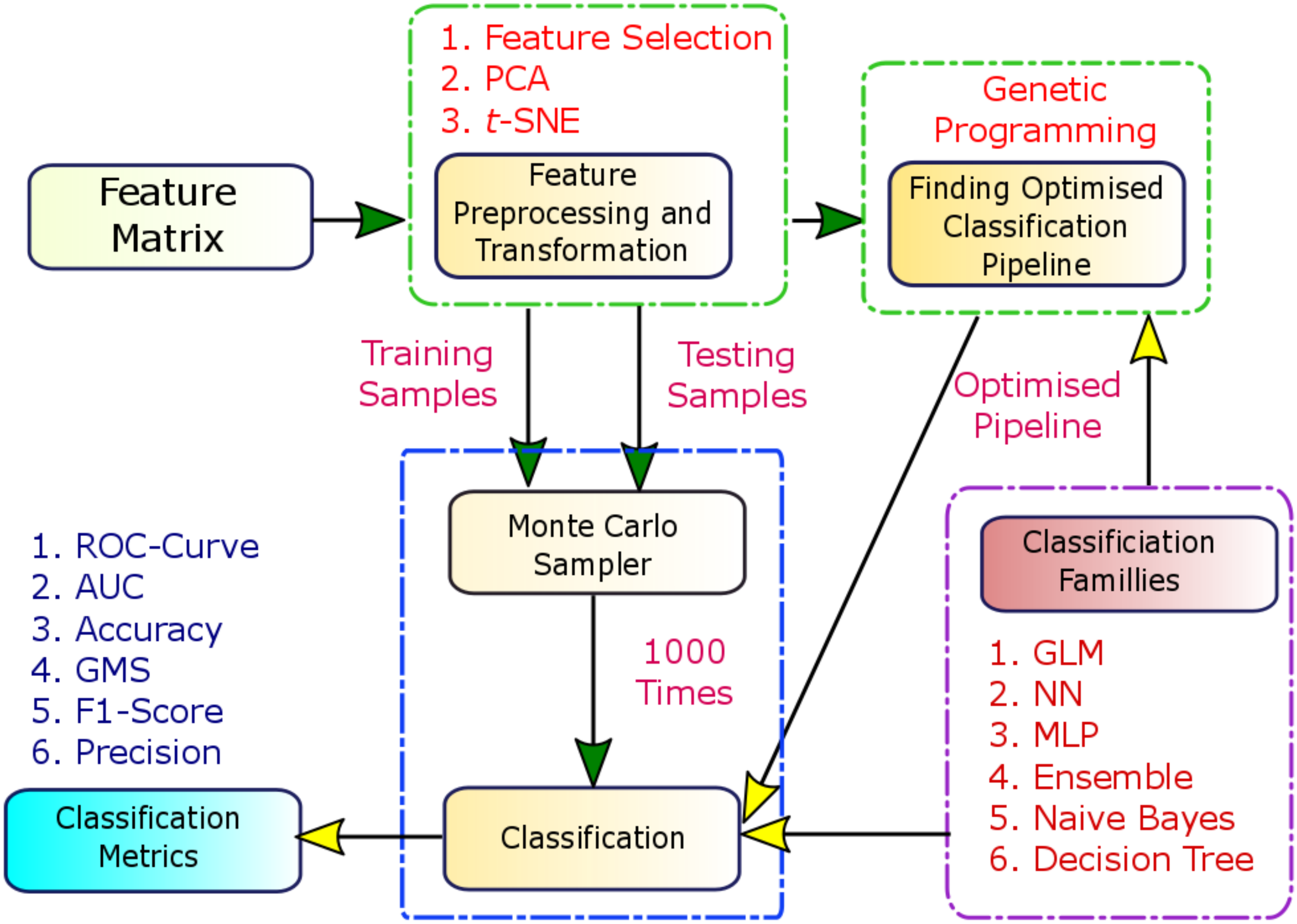

Figure 1. The overall classification scheme for the unbalanced data includes feature preprocessing and transformation and obtaining the optimal machine learning pipeline using genetic programming. The results are benchmarked along with six different machine learning families with different metrics.

Once the energy usage features are selected, and the correlated features are removed, Fisher score has been used to find out the essential features to carry out the classification. The features are transformed into lower dimensional space using the PCA and *t*-SNE algorithms. Then the optimized pipeline for classification using the selected features is found out by exploring the defined hyperparameter space using genetic programming which maximizes the AUC score. The optimized pipelines are obtained for different classification families and features in the original and transformed spaces. The optimized pipeline for each classification family and feature space is trained on under-sampled features and tested with a set of features held out for testing. The random under-sampling and training were performed 1000 times to generate probabilistic outcomes of the classification metrics. Thus, classification metrics are obtained in each sampling iteration for each family of classifiers after the hyper-parameter optimization and with the optimum pipeline. The next steps for the data analytics are feature selection and transformation, which are explained in following subsections.

2.2 Feature Selection

Feature selection technique is essential for classification as it prevents overfitting and improves the algorithm learning performance and computational efficiency. Here, a supervised feature selection technique is performed using the Fisher score (Duda, Hart, & Stork, 2012). Let us consider $\mathbf{F} = \{f_1, f_2, \ldots, f_n\}$ be the original features in the dataset. Fisher score is calculated by considering the similarity and dissimilarity of the features considering their labelled classes. It is represented as:

$$\text{Fisher Score}(f_i) = \sum_{j=1}^{c} n_j \left(\mu_{ij} - \mu_i\right)^2 \Big/ \sum_{j=1}^{c} n_j \sigma_{ij}^{\;2}, \tag{2}$$

where n_j, μ_i, μ_{ij} and $\sigma_{ij}^{\;2}$ represents the number of samples in class *j*, mean of the feature f_i, mean of the feature f_i for the samples in class *j*, variance of the feature f_i for the samples in class *j* respectively. In this problem, we have two classes hence *c* is 2. The feature selection performed on this dataset, and the Fisher score obtained for the features are shown in Figure 2.

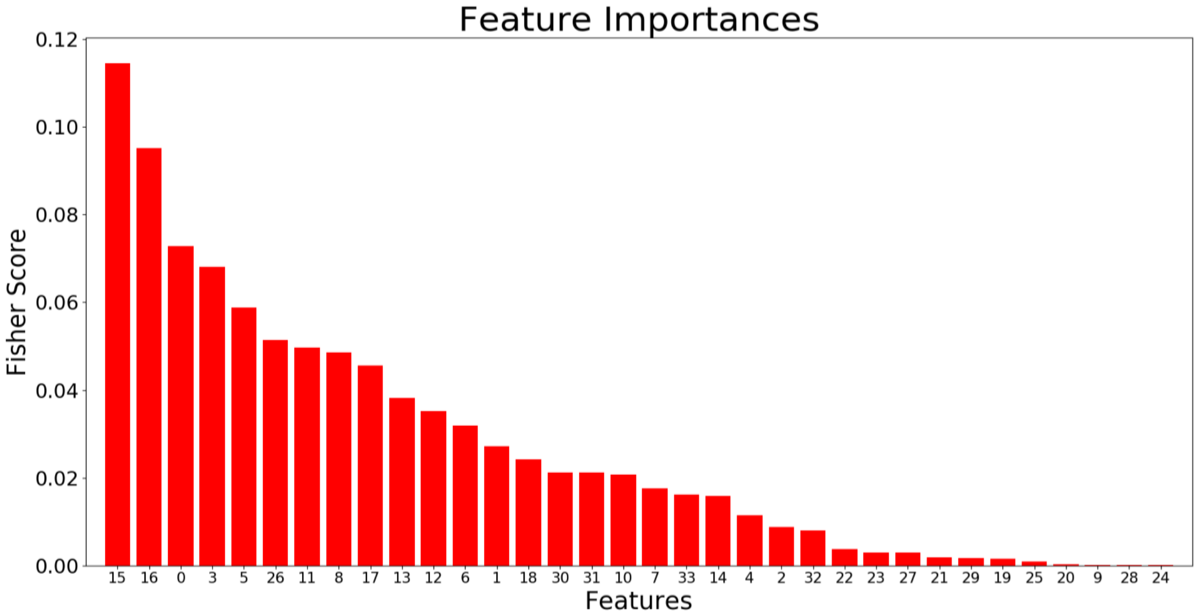

Figure 2. Feature importance analysis based on Fisher score, which highlights the differences and similarities in the type and the application of energy usage in urban and rural areas.

The top 5 features which influence the energy consumption patterns of the users are shown in Table 1. It was evident from the exploratory data analysis that most rural consumers do not use natural gas, and urban consumers do not use electricity for space heating. Hence, that feature became important as per the Fisher score. Even though all the consumers in urban and rural areas subscribe to electrical consumption, the overall usage of electricity does form an essential feature for classification, as shown in Table 1. However, features such as kerosene used for water heating, total thermal units used for space heating, electricity cost due to refrigeration use, fuel oil usage for water heating and kerosene for other appliance contributes less than 1% of the feature space. It states that the above energy use does not vary with the consumers residing in urban or rural areas. Hence, the top 95 percentile of the features was selected for classification problem.

Moreover, due to the non-subscription of various energy forms between different rural and urban consumers, the feature matrix is sparse. Hence it is worth checking the performance of the classification algorithms on the features obtained in the reduced dimensional space. In previous ML studies (Nasution, Sitompul, & Ramli, 2018; Vasan & Surendiran, 2016), it is shown that there was an improvement in accuracy when the classification was performed on PCA feature space as compared to the original high dimensional feature space. However, in (Martinez & Kak, 2001), it is shown that the performance of the classifier deteriorates in lower-dimensional spaces. In (Villalba & Cunningham, 2007), it was found that the performance of the classifier is dependent on the chosen family. Similarly, improvement in classification performance has been reported in (Miao et al., 2018), (Traven et al., 2017) on the input features transformed by *t*-SNE. The transformation techniques are explained in the following subsections.

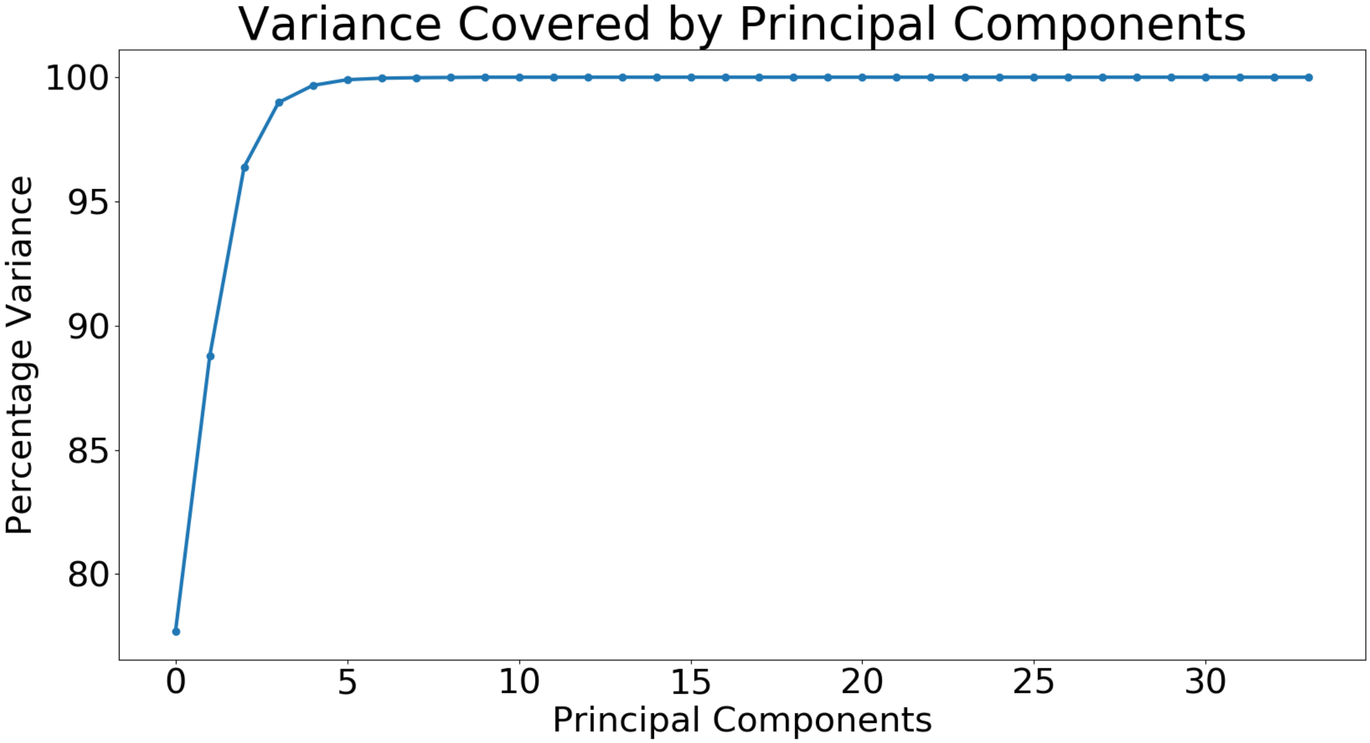

Figure 3. Percentage of variance explained by the principal components as explained in the scree plots to obtain the optimal number of features representing the variance of the original features.

Table 2: Description of the top five features used in the binary classification

Feature No.	Feature Name	Fisher’s Score
15	Total LPG/propane usage	0.1145
16	LPG/propane usage for space heating	0.0951
0	Total electricity consumed	0.0728
3	Electricity for water heating	0.0681
5	Electricity for miscellaneous purposes	0.0588

2.3 Principal Component Analysis (PCA)

PCA is defined as the orthogonal linear transformation of the input features into the new space. The projection is made such that the highest variance lies on the first component, the second highest variance lies on the second component, and it goes on for additional principal components. Let us consider the matrix $\mathbf{F_c}$ be the modified version of the feature matrix $\mathbf{F}$ centered around the mean. The transformation is carried out using weight matrix $\mathbf{W}_{(k)} = \left(w_1, \ldots, w_n\right)$ which maps each row vector $\mathbf{f}_{(i)}$ to a vector of principal component scores $\mathbf{t}_{(i)} = \left(t_1, \ldots t_n\right)_{(i)}$ given as:

$$t_{k(i)} = \mathbf{f}_{(i)} \cdot \mathbf{w}_{(k)}, \tag{3}$$

in such a way that $\mathbf{t}$ inherits the maximum possible variance from $\mathbf{F}$. The coefficient vector $\mathbf{w}$ is constrained to be a unit vector, thus forming an orthogonal basis for the feature space. Thus, in order to maximize the variance, the first weight vector $\mathbf{w}_{(1)}$ has to satisfy the following equation:

$$\mathbf{w}_{(1)} = \underset{\|\mathbf{w}\|=1}{\arg\max}\left\{\sum_i \left(t_1\right)_{(i)}^2\right\} = \underset{\|\mathbf{w}\|=1}{\arg\max}\left\{\sum_i \left(\mathbf{x}_{(i)} \cdot \mathbf{w}\right)^2\right\} = \arg\max\left\{\frac{\mathbf{w}^T\mathbf{X}^{\mathbf{T}}\mathbf{X}\mathbf{w}}{\mathbf{w}^T\mathbf{w}}\right\}. \tag{4}$$

The k^{th} component can be found by subtracting the first (k-1) principal components from $\mathbf{F}$ which is given as:

$$\hat{\mathbf{F}}_k = \mathbf{F} - \sum_{s=1}^{k-1}\mathbf{F}\mathbf{w}_{(s)}\mathbf{w}_{(s)}^{\mathrm{T}}, \quad \mathbf{w}_{(k)} = \arg\max\left\{\frac{\mathbf{w}^{\mathrm{T}}\hat{\mathbf{F}}_k^{\mathrm{T}}\hat{\mathbf{F}}_k\mathbf{w}}{\mathbf{w}^{\mathrm{T}}\mathbf{w}}\right\}. \tag{5}$$

The required number of principal components can be obtained using a scree plot, as shown in Figure 3. As evident from Figure 3, the top 5 principal components explain about 99% variance of the feature space. Hence, these components are used as new features in the transformed PCA space, which is visualized in Figure 4. It is observed that the principal components corresponding to two classes are mixed up against each other. Hence a classifier with complicated pipeline structure will be required for this classification task for the principal components.

2.4 *t-distributed Stochastic Neighbor Embedding (t-SNE)*

The *t*-SNE is a nonlinear dimensionality reduction technique well-suited for embedding high dimensional data in lower dimensional space (Bunte, Haase, Biehl, & Villmann, 2012; Maaten & Hinton, 2008). The *t*-SNE embeds the data in high dimensional features $\mathbf{F} = \{f_1, f_2, \ldots, f_n\}$ to lower dimensional space $\mathbf{Y} = \{y_1, \ldots, y_n\}$ using joint distribution represented by p_{ij} and q_{ij} respectively as:

$$q_{ij} = \left(1+\left\|y_i - y_j\right\|^2\right)^{-1} \Big/ \sum_{s\neq t}\left(1+\left\|y_s - y_t\right\|^2\right)^{-1}, \quad p_{ij} = \left(p_{i|j} + p_{j|i}\right)/2n, \tag{6}$$

where,

$$p_{i|j} = \exp\left(-\left\|f_i - f_j\right\|^2 / 2\sigma_j^2\right) \Big/ \sum_{s\neq j}\exp\left(-\left\|f_s - f_j\right\|^2 / 2\sigma_j^2\right). \tag{7}$$

The aim of *t*-SNE algorithm is to find $\{y_i\}$ that minimizes the Kullback-Leibler divergence from the low dimensional distribution *Q* to high dimensional distribution *P*, which is defined as follows:

$$\mathrm{KL}\left(P \,||\, Q\right) = \sum_{i\neq j} p_{ij}\log\left(p_{ij}/q_{ij}\right). \tag{8}$$

The parameter σ_j^2 in (7) influences the local scale around the feature f_j. Its value is obtained using the perplexity value defined as:

$$\mathrm{Perp}\left(p_j\right) = 2^{H\left(p_j\right)}, \tag{9}$$

where, $$H\left(p_j\right) = -\sum_j p_{i|j}\log_2 p_{i|j}. \tag{10}$$

A higher perplexity value leads to higher σ_j, thus higher $p_{i|j}$ is obtained for each datapoint. The *t*-SNE plots with four different perplexity parameters are shown in Figure 5. It is observed that the features corresponding to urban and rural is mixed up in the *t*-SNE domain except few small islands or clouds of datapoints for the rural consumers presumably due to the difference in heating consumption patterns, using electricity and natural gas. The feature spread becomes more compact as the perplexity of the algorithm increases. The *t*-SNE features with perplexities 25 and 100 are considered for the unbalanced classification task, as in the former case the transformed features are clustered together and in the latter case, small islands are formed in the corner with the remaining data points tightly clustered. The classification of the original features ranked and selected using the Fisher score and with the lower dimensional features of PCA and *t*-SNE space are performed using different classification families and the optimal pipelines are obtained. The details of the procedure have been explained in the next section.

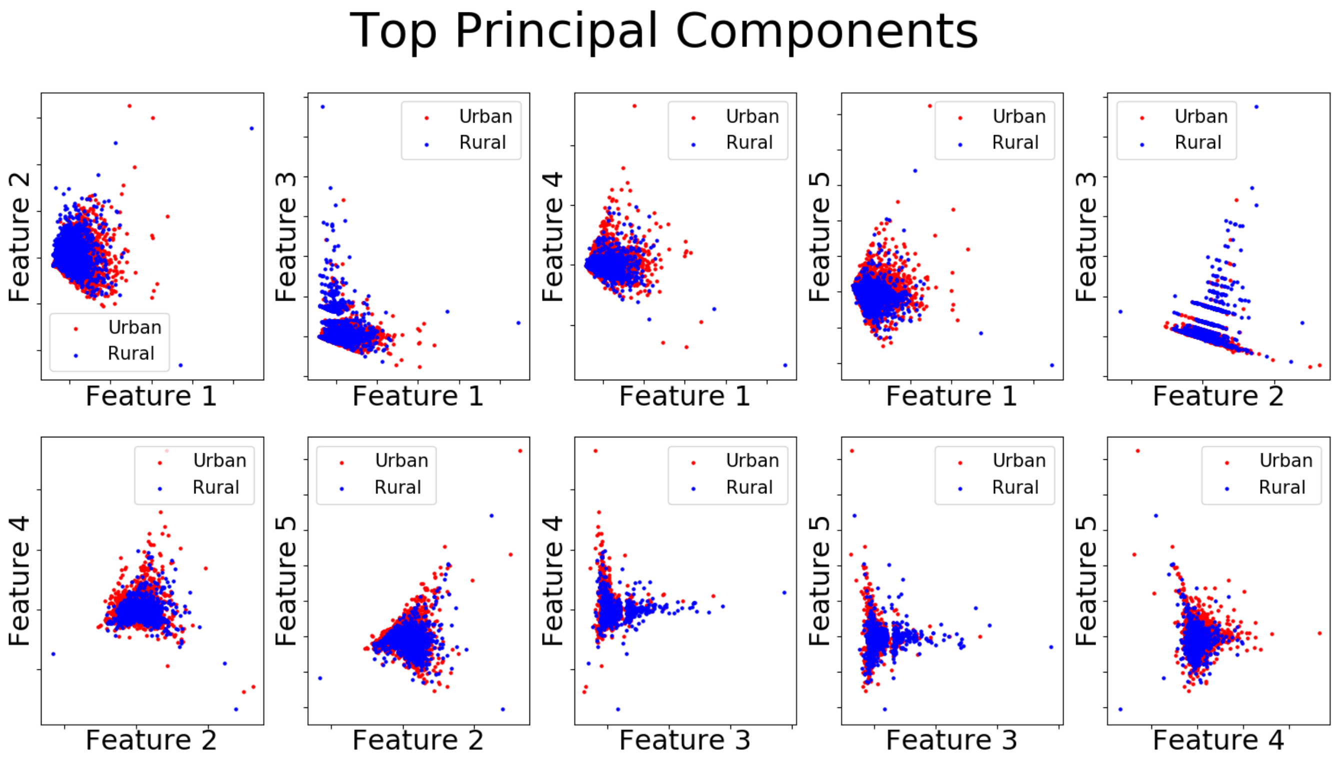

Figure 4. Scatter plots of the principal components as the new transformed features for the urban and rural areas, which covers variance similar to the original feature space.

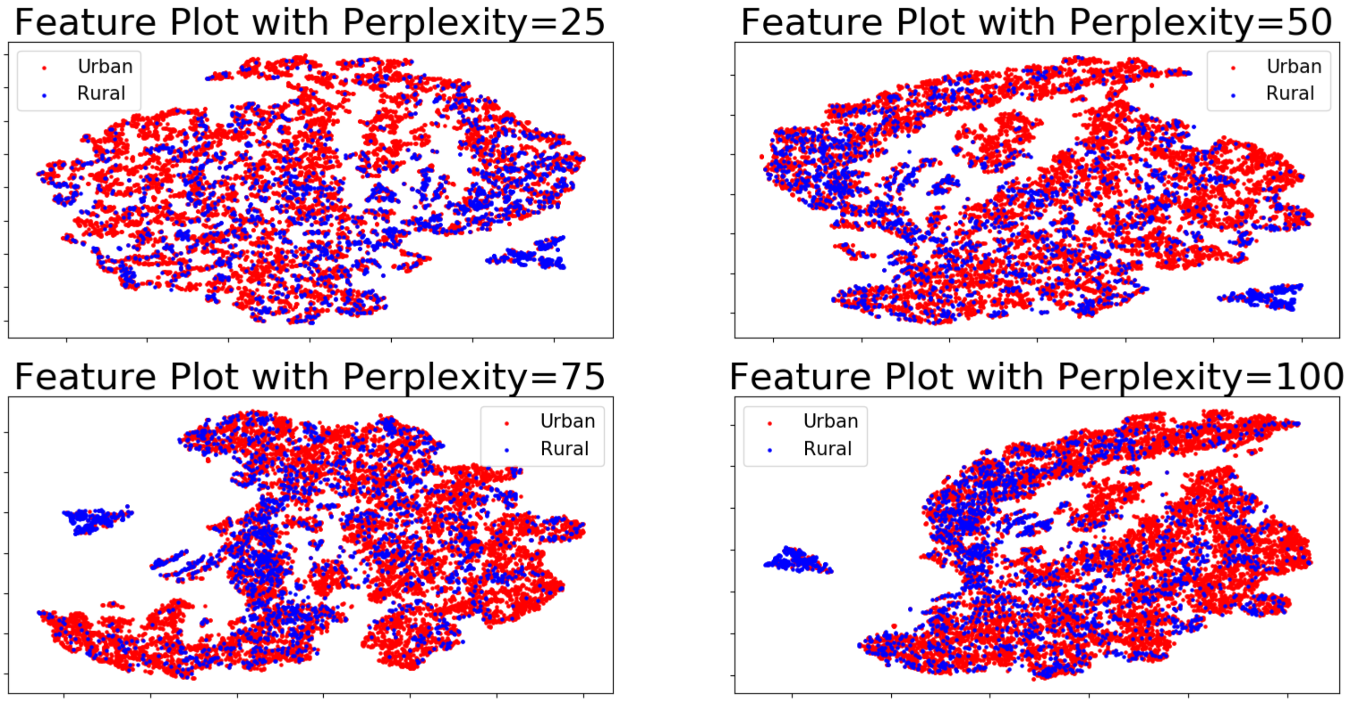

Figure 5. The scatterplot in reduced and transformed *t*-SNE space with different values of the perplexity parameter.

3. Description of Hyperparameters for the Classifiers and the Optimum Pipeline Selection

In this section, the hyperparameter spaces for the classification families are discussed along with the methodology for finding the optimal pipeline. It is observed in the previous section that there are some noticeable difference in energy consumption patterns for urban and rural consumers in the original feature space as per the feature scores in Figure 2 and the features transformed with principal components in Figure 4 and *t*-SNE in Figure 5. However, these differences are not very obvious when the features are transformed into a lower-dimensional space using PCA or *t*-SNE methods. Therefore, we here aim to obtain an optimal ML pipeline configurations for the classifier families in the original and transformed feature spaces using an evolutionary optimization algorithm. The hyperparameters for different classification families, as shown in Figure 1, are discussed in the next subsections.

3.1 Hyperparameters for Classification Families

Different classification families are used for classifying the features defined in the previous section. The classification families have their own hyperparameter space which involves several tunable parameters and solvers. The different algorithms and their hyperparameter space used for obtaining an optimal pipeline are defined next.

3.1.1 Generalized Linear Models (GLM)

The GLM has been defined to solve the following optimization problem:

$$\min_{\beta_0,\beta}\frac{1}{N}\sum_{i=1}^{N} w_i l\left(\mathrm{y}_i,\beta_0+\beta^T f_i\right)+\lambda\left[\left(1-\alpha\right)\left\|\beta\right\|_2^2\big/2+\alpha\left\|\beta\right\|_1\right], \tag{11}$$

where, w represents the weight that is to be used with the feature vector $\{f_i\}_{i=1}^n$ to predict the class label $\{\mathrm{y}_i\}_{i=1}^n$. The $l(.)$ constitutes the negative log-likelihood contribution for the observation *i*. The first term reduces the error between the class labels and the features by tuning the parameter β. The second term penalizes the complexity of the model with the term α. The penalty is called Lasso when $\alpha=1$ and it is Ridge when $\alpha=0$, and elastic-net when $0<\alpha<1$. The λ determines the strength of the penalty, which varies between 0 and 1. The logistic regression is a statistical model used for predicting the probability of the existence of specific events (Kleinbaum, Dietz, Gail, Klein, & Klein, 2002). The different penalty versions are considered to be the hyperparameter for this family. The $l(.)$ for logistic regression is a logarithmic function which maps the output into two levels. The solvers, described in (Pedregosa et al., 2011), used for updating the β have been used as a hyperparameter for the pipeline. *Passive-aggressive algorithms* are also used as another type of classifier with an exception that it requires a regularization parameter instead of a learning rate that projects iterated weights along the direction of minimizing the loss (Crammer, Dekel, Keshet, Shalev-Shwartz, & Singer, 2006). The loss functions are considered as hyperparameter for the classification, which is either hinged or square hinged. The *stochastic gradient classifier* has been used where it implements regularized linear models with stochastic gradient descent learning (Bottou, 2010). Here the gradient of the loss is estimated, and the model is updated to minimize the loss function defined in (11). This algorithm supports a lot of solvers and loss types as defined in (Pedregosa et al., 2011), which are considered in the hyperparameter space for the optimization problem. The *cross-validated Ridge classifier* has been used where the coefficients of the classifier are found by solving the minimization problem in (11), where the $l(.)$ is represented by an ordinary least square (OLS) form with cross-validation. The default cross-validation, i.e. leave one out cross-validation (LOOCV) is used in hyperparameter tuning. Different versions of GLM algorithms are used to find the optimal classifier for the family. The hyperparameters are defined using a dict file using `sklearn.linearmodel` within the optimizer.

3.1.2 k-Nearest Neighbor Classifier (k-NN)

Nearest neighbor classification is a type of instance-based learning established on the major votes of the nearest neighbors. The *k-nearest neighbor* and *radius neighbor classification* strategies are used where classification in the former case is performed based on the majority votes of *k*-nearest neighbors, while for the latter case, the classification is performed on datapoints within a certain radius (Coomans & Massart, 1982). Let us consider $\mathbf{f}_c$ be the features close to a test example $\mathbf{q}$. It is calculated on metric consisting of test feature $\mathbf{q}$ and the training vectors $\{f_i\}_{i=1}^n$ by taking the minimization of the distance metric $d(\mathbf{q},\mathbf{f}_i)$. The result of the test sample is obtained by the vote of the nearest neighbors defined by:

$$\mathrm{Vote}\left(y_i\right)=\sum_{c=1}^{k}\left(1\big/d\left(\mathbf{q},f_c\right)^n\right)\mathbf{1}\cdot\left(y_j,y_c\right). \tag{12}$$

Several algorithms can be used to calculate the distance between the neighbors, e.g. *brute force* which considers all neighbors and tree-based techniques like *k-dimensional* (KD) and *ball tree*. The algorithms to compute nearest neighbors are considered to be the hyperparameter for the pipeline for this classifier family. The calculation of the distances for the tree-based techniques is dependent upon the leaf size. The *leaf size, nearest neighbors (k)* and *radius* are considered from 1 to 1000, 0 to 50 and 0 to 50 in the hyperparameter space respectively. The hyperparameters are defined using a dict file using `sklearn.neighbors` within the optimizer.

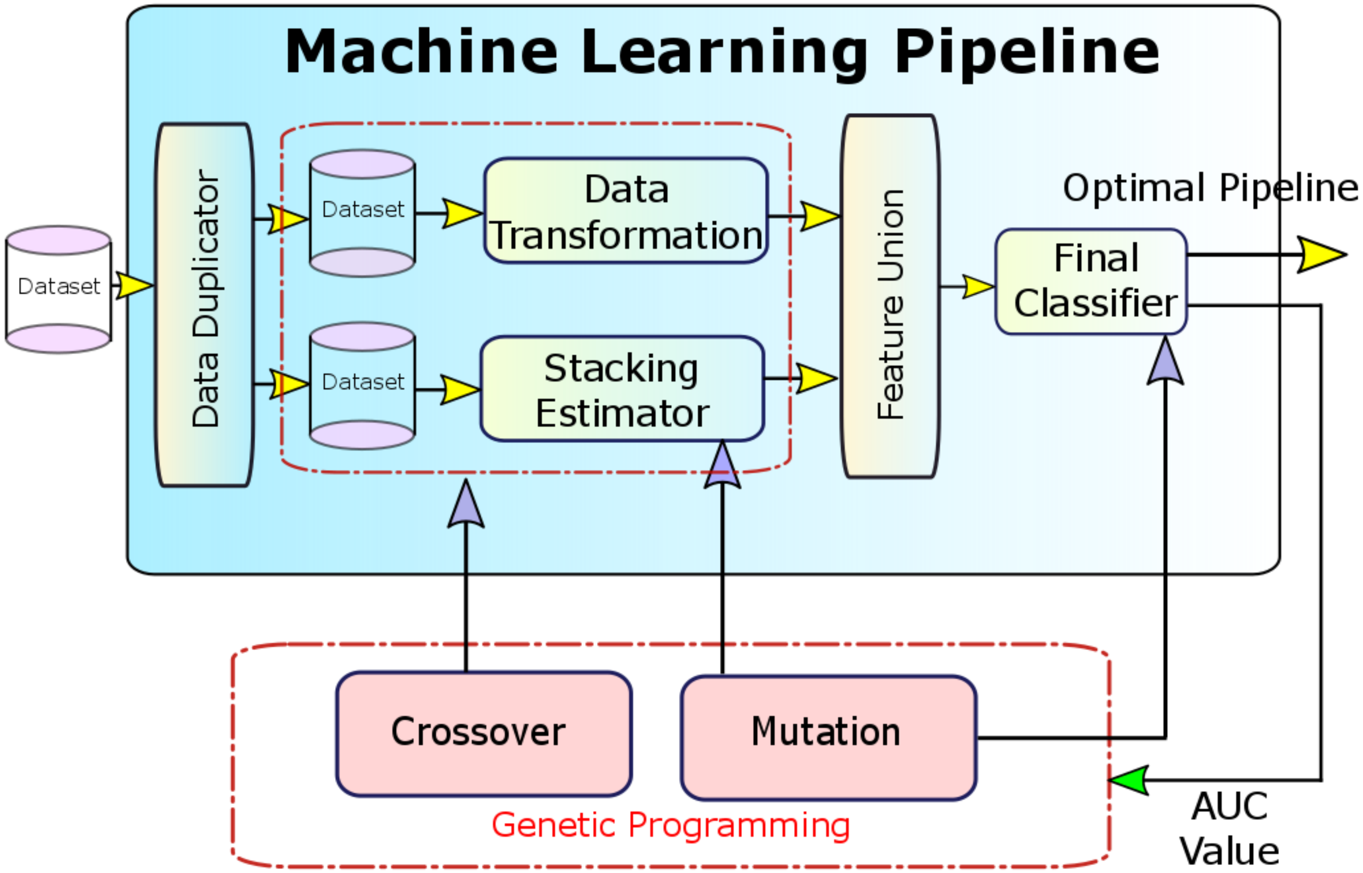

Figure 6. Optimal machine learning pipeline selection through genetic programming optimization of the AUC score.

3.1.3 Multi-Layer Perceptron (MLP)

The *multilayer perceptron classifier* is a class of feedforward neural networks which is realized using nonlinear computing elements. The nonlinear computing elements can be considered as a set of neurons linked amongst each other based on the size of the hidden layers. Considering $\hat{\mathbf{y}} = \{y_i\}_{i=1}^{n}$ the classification training labels, the classifier can be represented as follows (Bourlard & Kamp, 1988):

$$\hat{\mathbf{y}} = \phi(\mathbf{W} \cdot \mathbf{F} + w_0), \tag{13}$$

where, $\mathbf{W} \in \mathbb{R}^{p \times n}$ represents the weights for each neuron and p represents the number of layers for the neural network. These are considered to be the hyperparameters of this classifier which is varied from 1 to 1000. Here, $w_0 \in \mathbb{R}^n$ is considered as the bias vector and ϕ is another hyperparameter which involves all the activation functions mapping the response of the neurons to the class labels. The list of *activation functions* (ϕ) for hyperparameter space has been considered as:

$$\phi = \{\text{'identity', 'logistic', 'tanh', 'relu'}\}. \tag{14}$$

The iterative weight adjustment to minimize the error between the predicted and the training labels is performed using different *solvers*, that are also considered as hyperparameters for this classification family. The rate of minimization of the error depends on different *learning rates* (λ) per iteration. The hyperparameters for the learning rate has been considered as:

$$\lambda = \{\text{'Constant','Adaptive','invscaling'}\}, \tag{15}$$

which are considered for building the optimal pipelines through the GP optimizer. They can be constant, adaptive or inverse scaling. Since a shallow neural network is used, so it will be useful to find the optimal pipeline combination considering all the activation functions and learning rate. The hyperparameters are defined using a dict file using `sklearn.neural_network` within the optimizer.

3.1.4 Naïve Bayes (NB)

Naïve Bayes constitutes supervisory algorithms with the application to Bayes Theorem, assuming the independence of the features (Rish & others, 2001) as:

$$P(\mathrm{y}|\, f_1,\ldots,f_n) = P(\mathrm{y})P(f_1,\ldots,f_n|\mathrm{y})\big/P(f_1,\ldots,f_n)\,. \tag{16}$$

Due to the conditional independence of the features, $\{f_i\}_{i=1}^{n}$ we can write (16) as:

$$P(\mathrm{y}|\, f_1,\ldots,f_n) = P(\mathrm{y})\prod_{i=1}^{n}P(f_i|\mathrm{y})\Big/P(f_1,\ldots,f_n) \propto P(\mathrm{y})\prod_{i=1}^{n}P(f_i|\mathrm{y})\,. \tag{17}$$

Thus, we can write the predicted labels as:

$$\hat{\mathrm{y}} = \arg\max_{\mathrm{y}} P(\mathrm{y})\prod_{i=1}^{n}P(f_i|\mathrm{y})\,. \tag{18}$$

The hyperparameters for the training algorithm is the maximum *a-posteriori* (MAP) value to estimate $P(\mathrm{y})$ and $P(f_i|\mathrm{y})$. The optimal classifier is found by considering different probability distribution of $P(f_i|\mathrm{y})$. The distributions considered for finding the optimal pipeline are Gaussian, Bernoulli, Multinomial and complementary adaptive version of the Multinomial distribution, which is suited for imbalanced datasets. The hyperparameters are defined using a dict file using `sklearn.naive_bayes` within the optimizer.

3.1.5 Decision Tree (DT)

Decision Tree classifiers use nonparametric supervised learning method to create a model using simple decision rules from the given features to predict target labels (Safavian & Landgrebe, 1991). Let us consider a splitting parameter $\theta = (j, t_m)$ for the node *m* of the tree where j represents a feature and the t_m considered as threshold parameter. Hence, we can represent *Q* as left and right partition vectors as:

$$Q_{left}(\theta) = (f, \mathrm{y})\big|\, f_j \le t_m, \quad Q_{right}(\theta) = Q \setminus Q_{left}. \tag{19}$$

The aim of the decision tree is to find θ^* as: $\theta^* = \arg\min_{\theta} G(Q,\theta)$, (20)

where, $G(Q,\theta)$ is the impurity function defined as: $G(Q,\theta) = (n_{left}/n)H(Q_{left}(\theta)) + (n_{right}/n)H(Q_{right}(\theta))$, (21)

where, n_{left} and n_{right} represent the number of data points on the left and right-hand side of the partition. H is the *impurity function* which is considered in the hyperparameter tuning space. It can be either *Gini* or *Entropy* as defined in (Safavian & Landgrebe, 1991). *Extra Tree Classifier* forms an ensemble of decision trees by choosing the cut points randomly hence learning the samples to increase the depth of the trees (Geurts, Ernst, & Wehenkel, 2006). Similar impurity functions are used in the hyperparameter space for the classifier. The hyperparameters are defined using a dict file using `sklearn.tree` within the optimizer.

3.1.6 Ensemble Methods

The ensemble methods for classification aim to improve the robustness and the generalized nature of the single estimator by combining the prediction results of more than one estimator at a certain learning rate. The ensemble classifier is generally represented as:

$$F(f) = \sum_{m=1}^{M}\gamma_m h_m(f)\,, \tag{22}$$

where, $h_m(.)$ represents a basis variable representing a single estimator and γ_m is the weight of each basis estimator for the ensemble classifier. Here, *M* represents the number of *estimators* which is a hyperparameter in the optimization process. The ensemble methods for classification can be broadly classified into bagging or boosting classifiers (Dietterich, 2000). As described in (Dietterich, 2000; Geurts et al., 2006), the *bagging* and *random forest* *classifiers* tend to design an estimator that averages out the prediction results of the individual estimators $h_m(.)$ involved in the process.

In this case, $\gamma_m = 1/M$ where the results are averaged out. The *adaboost* and *gradient boosting classifiers* tend to grow the decision trees based on the information obtained from the previously grown trees. The γ_m for the boosting classifiers is represented as:

$$\gamma_m = \arg\min_{\gamma} \mathbf{L}\left[y_i, F_{m-1}(f_i) - \gamma \frac{\partial \mathbf{L}(y_i, F_{m-1}(f_i))}{\partial F_{m-1}(f_i)} \right], \tag{23}$$

where F_{m-1} represents the previous ensemble. The ensemble classifier at m^{th} node is defined as:

$$F_m(f) = F_{m-1}(f) + \gamma_m h_m(f). \tag{24}$$

The γ_m in (24) is obtained by minimizing the loss function defined in (23). The *loss functions* can be deviance or exponential which are considered as the hyperparameters for this classification family. The *voting classifier* aims to combine the results by either taking an average or majority votes of different ML classifiers to classify the class labels. The hyperparameter for each of these algorithms is the number of estimators for gradient boosting is varied from 1 to 1000. The voting strategy, whether hard or soft, is also used as hyperparameter for the voting classifier. The hyperparameters are defined using a dict file using `sklearn.ensemble` within the optimizer.

Using the above-described hyperparameters of the classification families, we now aim to find out the optimal pipeline with the right combination of them along with optimal hyperparameter selection using evolutionary optimization. The method of finding the optimal pipeline is described in the next subsection.

3.2 Optimal Pipeline Selection using Genetic Programming

Optimal pipelines are to be obtained for each classification families described in the previous section. Since the data here is an unbalanced one, it is essential to choose the right metric for finding an optimal pipeline. As described in (Fawcett, 2006), the AUC score is independent of the class distribution. Hence it is a robust metric used for finding optimal pipeline for classifying unbalanced datasets.

Automated machine learning (AutoML) is used here for optimizing the ML pipelines. Traditionally grid-search techniques have been used in (Hutter, Lücke, & Schmidt-Thieme, 2015) utilizing brute force methodology to find the parameters for the best model fit. However, as shown in (Bergstra & Bengio, 2012), a randomized search for hyperparameters yields the best model fit. The genetic programming (GP) (Zutty, Long, Adams, Bennett, & Baxter, 2015) based intelligent search is much efficient and performs much better than any other randomized search procedures for pipeline based hyperparameter tuning due to their tree-like evolving structure. Tree-based Pipeline optimization technique (TPOT) (Olson, Bartley, Urbanowicz, & Moore, 2016) has been used here, which uses genetic programming to find out the optimal pipeline. TPOT is used as a wrapper for the machine-learning package Scikit-Learn (Pedregosa et al., 2011), where the pipeline search is performed based on the hyperparameters of the classification families as defined in the previous section. The formulation of the pipeline using the GP optimizer is shown in Figure 6 for finding the optimal pipeline as described in the next subsections.

3.2.1 Pipeline Preprocessing

As shown in Figure 6, initially, the data is duplicated and transformed to create different features out of the dataset. Hence several operators are used to modify the features. One of these is the classification operators along with the hyperparameters which are defined in the previous sections. The predictions of the classification operators are used as features for final classification to be performed. Other forms of scaling the dataset involve standard scaling, robust scaling, minimum and maximum scaling, which are defined in (Olson et al., 2016). The final features are selected before passing through the classification algorithm family. After the scaling and new features are created out of the classification operators, they are passed through the classification algorithm. The amount of duplication and features to be added is dependant upon the tree-based algorithm of the genetic programming. The stacking estimator, as shown in Figure 6, generally stacks all the features obtained after preprocessing the dataset. The automatic generation and optimization of these tree-based pipelines are performed using genetic programming (Koza & others, 1994) as implemented in the distributed evolutionary algorithms in the Distributed Evoltionary Algorithms in Python (DEAP) package (Fortin, De Rainville, Gardner, Parizeau, & Gagné, 2012). The GP algorithm's working principle for optimal pipeline develeopment has been explained in the next subsection.

3.2.2 Genetic Programming (GP)

Genetic programming is an evolutionary algorithm to evolve new solutions which cannot be solved directly. The biological evolution and fundamental mechanisms inspire it. It involves the selection of the programs based on the fitness functions based on the proficiency of the task, which is performed generally in 3 steps as described in (Koza & others, 1994):

i) Selection: In the selection step, the individuals are randomly selected as the current generation sample. These individuals will act as parents for the next generation. It is dependent upon the number of generations. In TPOT, the number of generations means the number of iterations to run the pipeline optimization process.

ii) Crossover: The selected individuals are crossed over to produce new individuals as the next generation. In the given problem, crossover fraction explains how many pipelines are to be combined, which is kept at 0.1, by default.

iii) Mutation: The genetic properties of the previous generation individuals are modified to create new individuals for the next generation. In TPOT, the mutation fraction tells the GP to apply random changes in the defined hyperparameters and kept at 0.9.

The above processes for the GP algorithm are implemented using the function `TPOTClassifier()` where the above changes can be specified. The GP in TPOT generates a certain specified number of random tree-based pipelines. The balanced cross-validation metric is obtained for the dataset. For every generation of the GP algorithm, the top 20 pipelines in the population are selected according to the nondominated sorting genetic algorithm (NSGA-II) selection scheme (Deb, Pratap, Agarwal, & Meyarivan, 2002). The objective of the selection scheme is to maximize AUC score specified on the features while minimizing the number of preprocessing operators in the pipeline. Each of the selected pipelines generates a certain number of offsprings into the next generation's population. One-point crossover is used to different crossover offsprings. Mutation operator randomly changes the remaining offsprings. For every generation, the Pareto front of the non-dominated solutions (Deb et al., 2002) is updated. The algorithm repeats this evaluate-select-crossover-mutate process for a certain number of specified generations thus adding, tuning, and eliminating the pipeline operators to maximize the AUC score and the highest AUC pipeline is selected from the Pareto front as the "best" pipeline from the run. The optimization algorithm for searching the pipeline space is stochastic in nature. The significant advantage of the tree-based pipeline techniques over the traditional techniques is the exploration of pipeline configurations which is generally not possible to carry out manually and computationally infeasible by grid search methods. The hyperparameter space for each of the classification families is incorporated in the dictionary format within the specified function. The optimal pipeline is obtained using the function `make_pipeline()` which is obtained from the Python Scikit-Learn package (Pedregosa et al., 2011). The optimized pipeline obtained from the genetic programming is used for classification on the resampled features, as shown in Figure 1, and the classifier performance is analyzed using the metrics defined in the next subsection.

4. Classification Metrics and Hypothesis Testing

4.1 Classification Metrics for Unbalanced Dataset

Let us consider the testing dataset has **P** number of actual positive samples and **N** number of actual negative samples we define the following terms of the confusion matrix:

- True Positive (TP) - Positive labels classified as positive,
- True Negative (TN) - Negative labels classified as negative,
- False Positive (FP) – Positive labels classified as negative,
- False Negative (FN) – Negative labels classified as positive.

Thus, the accuracy of a classification algorithm is defined as:

$$\text{Accuracy} = \left(\text{TP+TN}\right)/\left(\text{P+N}\right). \tag{25}$$

The True Positive Rate (TPR) and False Positive Rate (FPR) are also defined in terms of the above four parts of the confusion matrix as:

$$\text{TPR=TP/P} = \text{TP}/\left(\text{TP+FN}\right), \quad \text{FPR=FP/N} = \text{FP}/\left(\text{FP+TN}\right). \tag{26}$$

TPR or recall can be conceptually represented as the probability of classifying a positive label out of all true positive labels, while FPR is represented as the probability of falsely classifying positive labels out of all the negative labels. The receiver operating characteristics (ROC) curve is obtained by plotting the FPR and TPR, by varying the threshold for classification, where the former is plotted in the *x*-axis and the latter in the *y*-axis. The area under the ROC curve (AUC) has been widely used as a single metric for comparing the performances of alternative classifiers. A higher value of AUC signifies better classification performance. As discussed in (Fawcett, 2006), AUC also signifies the probability that the classifier will rank a randomly chosen positive label over a randomly chosen negative label.

Let us define the input features as $\mathbf{F}$ and classification threshold as T. Let us consider $\mathbf{F}$ follows a probability density $p_1(x)$ under "positive" class and density of $p_0(x)$ otherwise. Thus, TPR and FPR are represented as:

$$\mathrm{TPR}(T)=\int_{T}^{\infty} p_1(x)\,dx, \quad \mathrm{FPR}(T)=\int_{T}^{\infty} p_0(x)\,dx. \tag{27}$$

From the above definition, AUC is defined as:

$$A=\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} I(T'>T)\,p_1(T')\,p_0(T)\,dT'\,dT=P(F_1>F_0). \tag{28}$$

The precision or positive predictive value (PPV) is defined as:

$$\mathrm{PPV{=}TP}\big/(\mathrm{TP{+}FP}). \tag{29}$$

The F$_1$-score can be calculated from the precision defined in (29) and the recall defined in (26), as:

$$F_1=2\cdot\big((\mathrm{precision}\cdot\mathrm{recall})\big/\mathrm{precision}+\mathrm{recall}\big). \tag{30}$$

Amongst other commonly used performance metrics for unbalanced classification, the geometric mean score or GMS is worth noting which calculates the geometric mean accuracy of the majority and the minority class and is given as:

$$\mathrm{GMS{=}}\sqrt{(\mathrm{TP/P})(\mathrm{TN/N})}. \tag{31}$$

As shown in Figure 1, the classifier is trained 1000 times by repeated resampling of the unbalanced dataset and tested accordingly, and we thus obtain a set of classification metrics per iteration. When the process is repeated 1000 times, we obtain many samples of these uncertain classification metrics showing the diversity of the unbalanced dataset. With the help of metrics obtained from Monte Carlo simulation, a comparison of the classification families can be performed, which is done using the nonparametric statistical tests as defined in the next subsection, since most of the classification metrics are found to not follow Gaussian distributions.

4.2 Kruskal Wallis Test

Let us consider the metric forms a vector $M=\{m_1,\ldots,m_n\}_k$ for the k^{th} classification algorithm. The Kruskal-Wallis test is a nonparametric method for comparing more than two samples. The test does not assume the underlying known probability distribution. Hence, the test statistic H is obtained based on the ranks of the samples $\{r_i\}_k$, which is defined as:

$$\mathrm{H}=(N-1)\left(\sum_{i=1}^{g} n_i(\bar{r}_{i\cdot}-\bar{r})^2\right)\Bigg/\left(\sum_{i=1}^{g}\sum_{j=1}^{n_i}(r_{ij}-\bar{r})^2\right), \tag{32}$$

where, n_i represents the number of observations in group *i*, r_{ij} is the rank of the observation *j* from group *i*. Here, $\bar{r}$ and $\bar{r}_{i\cdot}$ represents the average of all r_{ij} and the average rank of all the observations in *i*th observations, respectively. If the metrics are not similar, the *p*-value will be zero. Higher values of H represent more similarity in the medians of the samples and vice-versa.

5. Numerical Results of Optimal Imbalanced Classification

5.1 Optimized Pipeline Selection using Genetic Programming

It is shown in Figure 1 that the original features are first selected using the Fisher score and transformed to lower dimensional space using PCA and *t*-SNE algorithms. From these features, randomly selected 20% of the data is kept as held out for testing the classifier. The pipeline selection is performed on the remaining 80% data, which is split into half for training and testing purpose. The 10-fold cross-validation is adopted, and the AUC scores are attained on the dataset. The pipelines are obtained by maximizing the AUC score per iteration. A total of 100 generations and population size of 10 is used for the generation of the optimal pipelines. The genetic programming was run on a

Windows PC with Intel Xeon E5-2687W CPU, 3 GHz processor and 12 parallel cores. The Tree-based Pipeline Optimization Tool (TPOT) (Olson & Moore, 2016) is used to find the optimal pipeline utilizing the option of solving the problem on multi-threaded processors. The convergence characteristics and the wall clock time while running on 12 parallel processors are shown for genetic programming to find the optimal pipeline in Figure 7 and Figure 8, respectively. The computational complexity/burden increases with hybridization and the optimization process for searching the optimal pipelines. Due to the complexity of the pipelines involved, the training of the machine learning algorithm involves searching for the parameters in a higher-dimensional space. However, the performance of the algorithm is much more efficient on the held-out dataset as compared to any standard machine learning algorithm with predetermined hyperparameters as reported in many contemporary literatures.

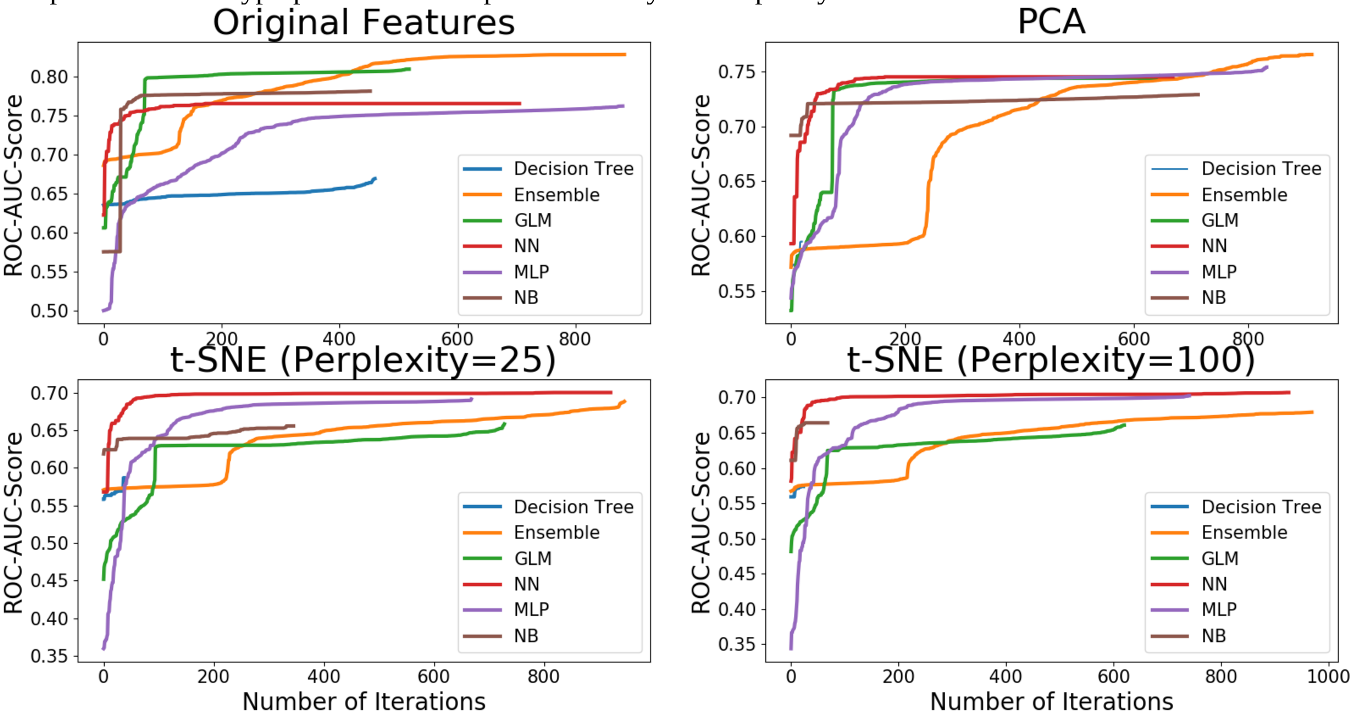

Figure 7. Convergence characteristics in different feature spaces (original, PCA, *t*-SNE with two different values of the perplexity parameter).

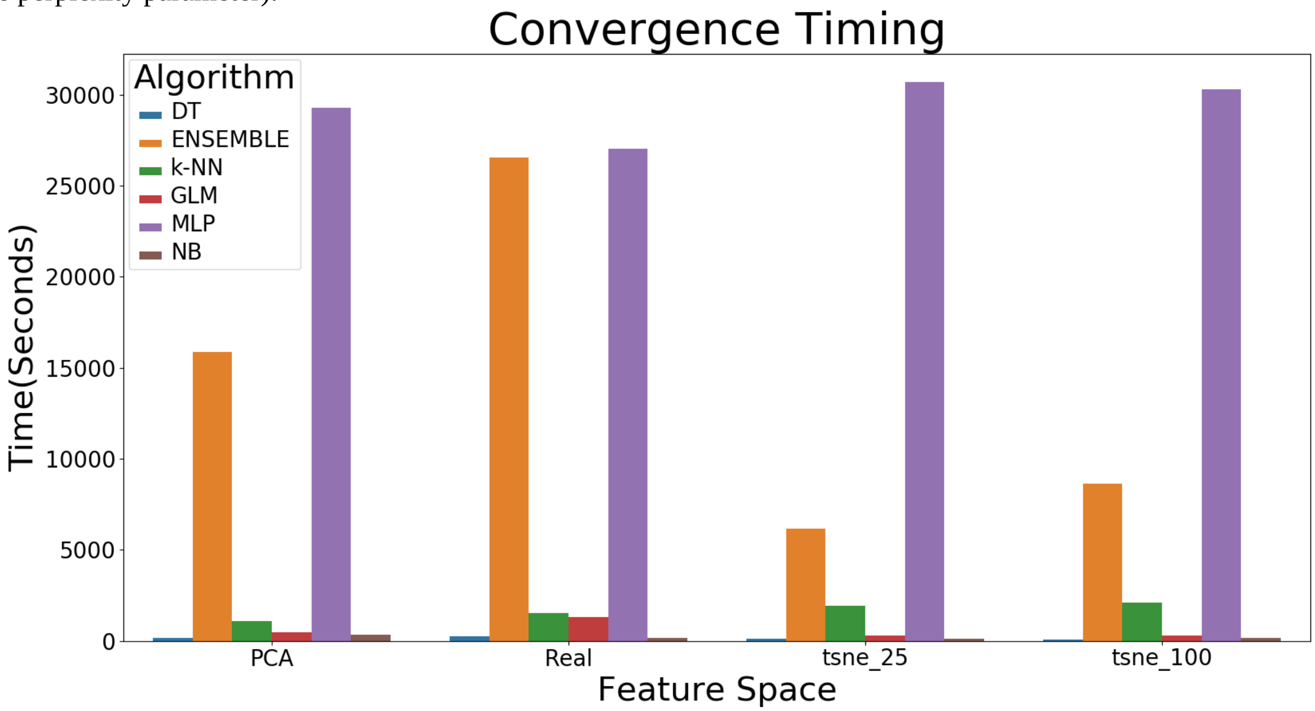

Figure 8. Convergence time of the GP algorithm for optimal classification pipeline design in different feature space.

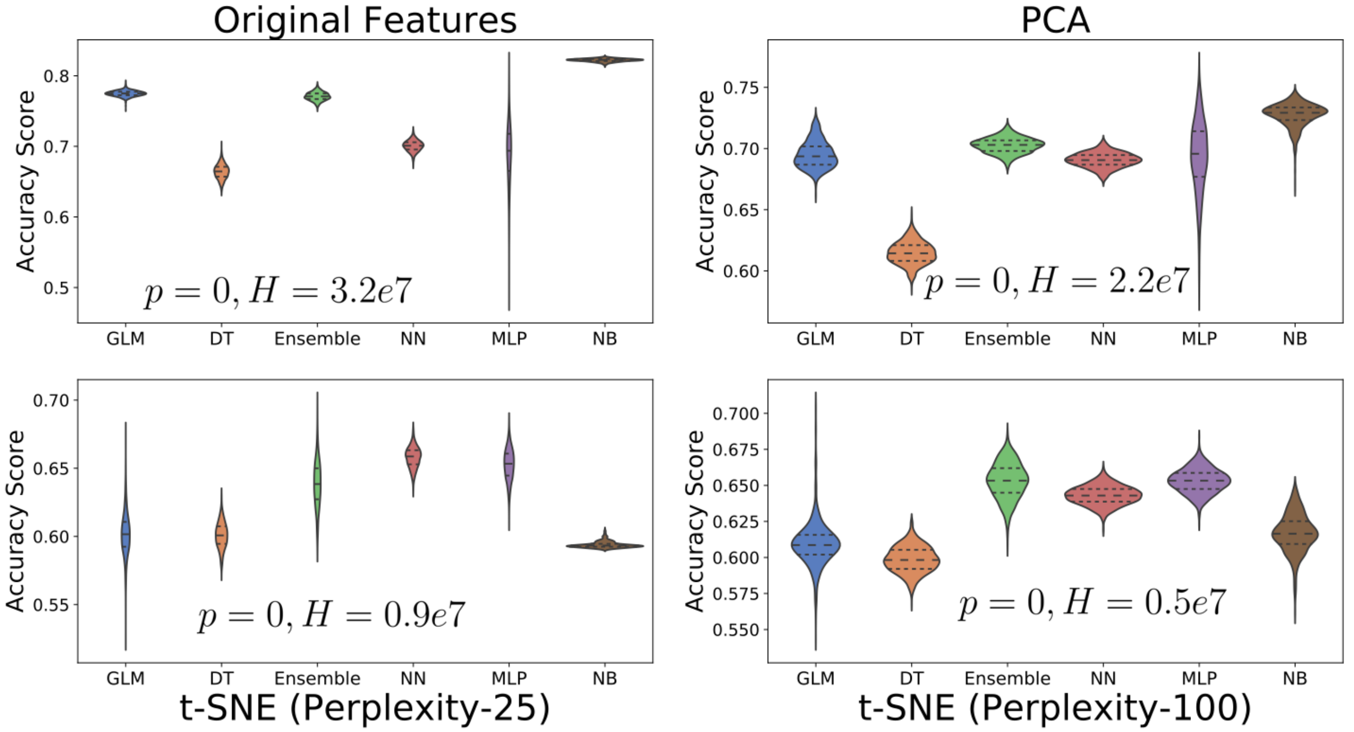

Figure 9. Univariate distribution of classification accuracy score on the testing data in the original and reduced space.

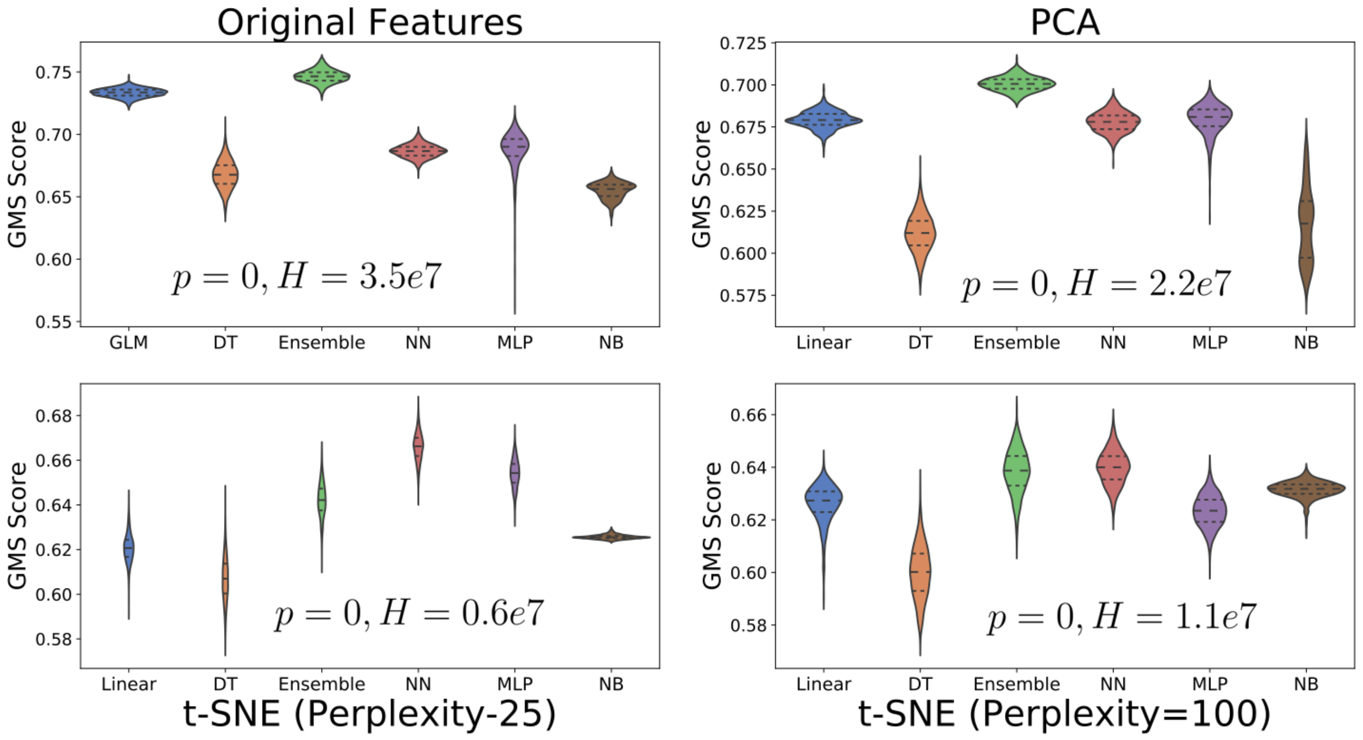

Figure 10. Univariate distribution of geometric mean score on testing data in the original/reduced space.

It is observed from Figure 7 that we obtain the highest AUC of >0.8 in the original feature space as compared to the transformed lower-dimensional spaces. The highest AUC-score is obtained from the ensemble method classifiers in the original and PCA feature spaces while in case of *t*-SNE space, the *k*-NN classifiers perform the best out of all the other families. The MLP classifiers take maximum time for convergence (~8 hours), followed by the ensemble method classifiers (~4 hours). Searching for the decision tree classifiers takes the least amount of time, even though the performance is not optimal as compared to other classification families in all the feature spaces as shown in Figure 8. The comparative performance of the GLM families on the training dataset reduces drastically in lower-dimensional transformed feature space. However, the performance of the *k*-NN improves in the reduced dimensional spaces. The structure of the optimal pipeline for different classification families is explained next.

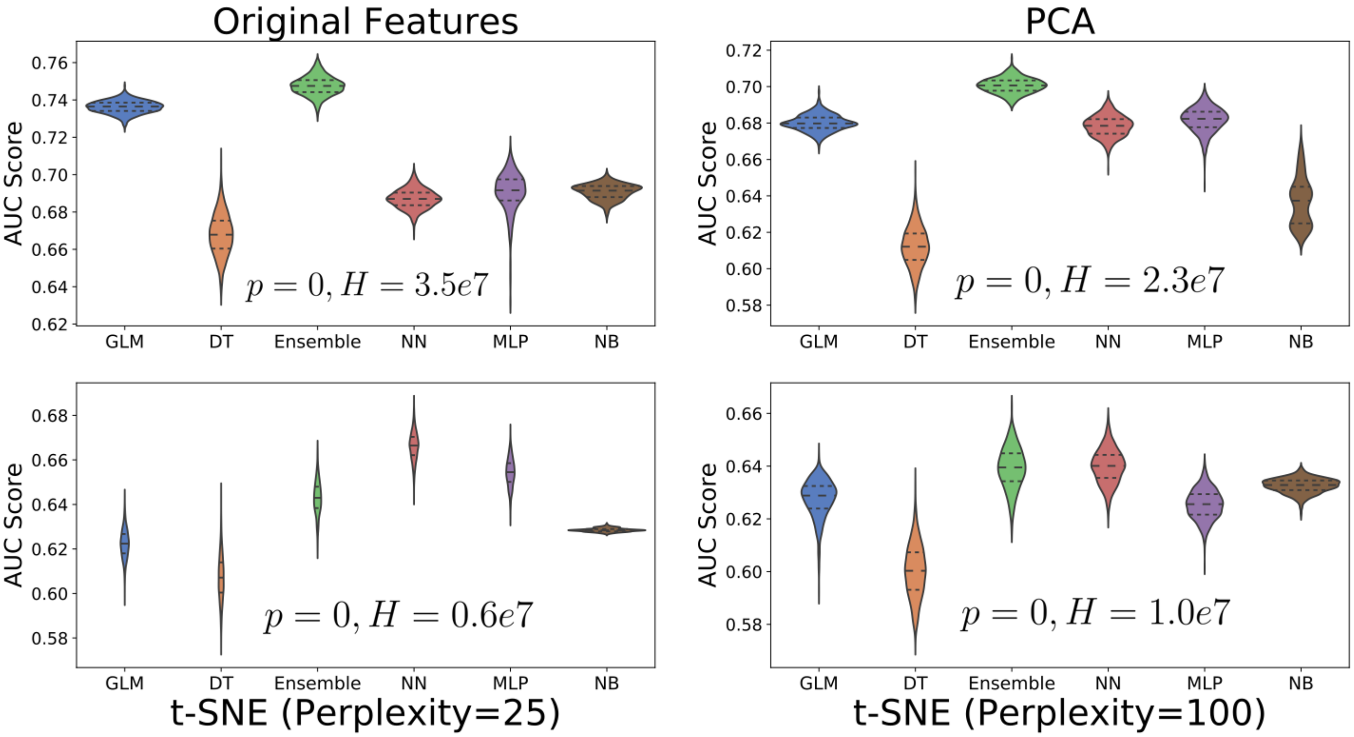

Figure 11. Univariate distribution of AUC score on testing data in the original and reduced space.

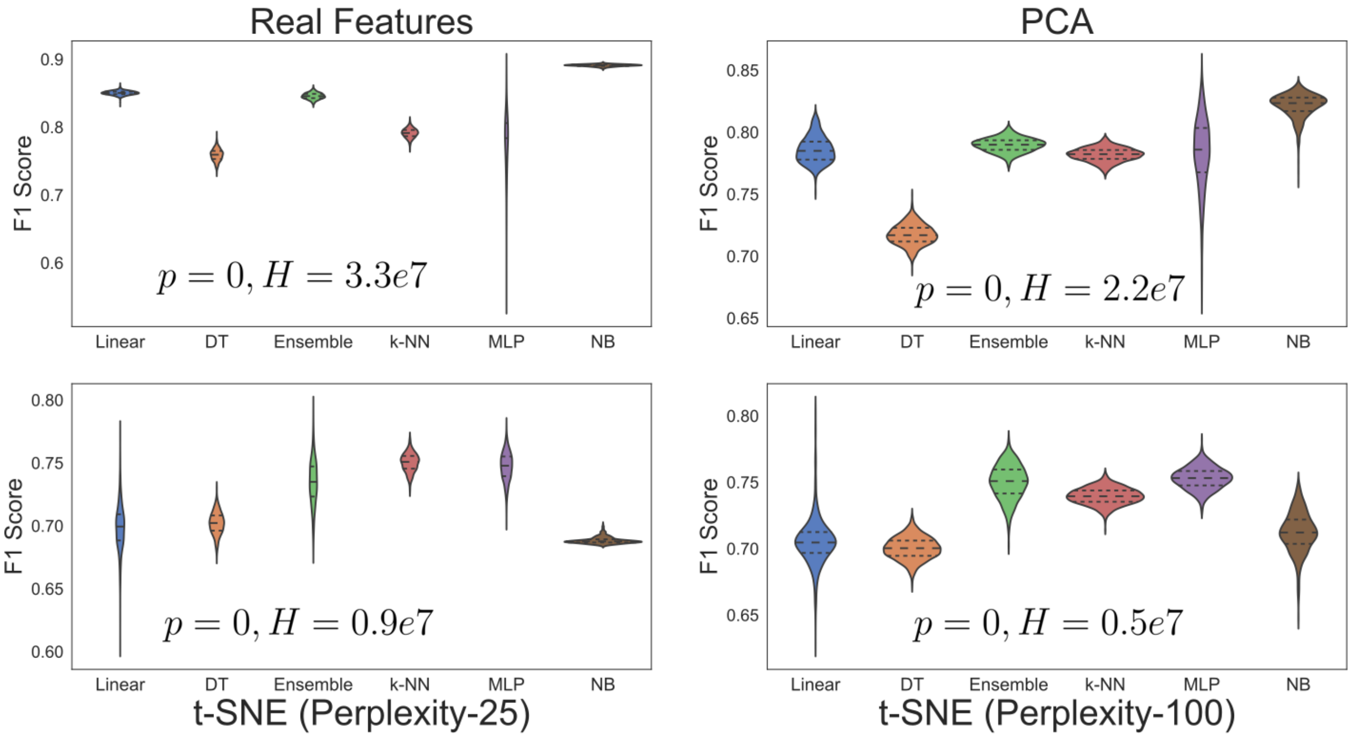

Figure 12. Univariate distribution of F_1 score on testing data in the original and reduced space.

5.1.1 Generalized Linear Model

Ridge classifier is the classifier obtained for all the feature spaces as a part of the optimal pipeline. A higher number of stacking estimators and function transformers are used in the pipelines for the lower-dimensional *t*-SNE space where the optimal AUC value is lower as compared to what is obtained with lesser number of stacking estimators in PCA and the original feature space. It takes less than 200 iterations to come within the convergence band for all the feature spaces. It is observed that the convergence time is highest for the features in the original feature space, even though the pipeline is small suggesting most of the computational time was spent on searching the hyperparameter space of the stacking estimators rather than the pipeline evolution.

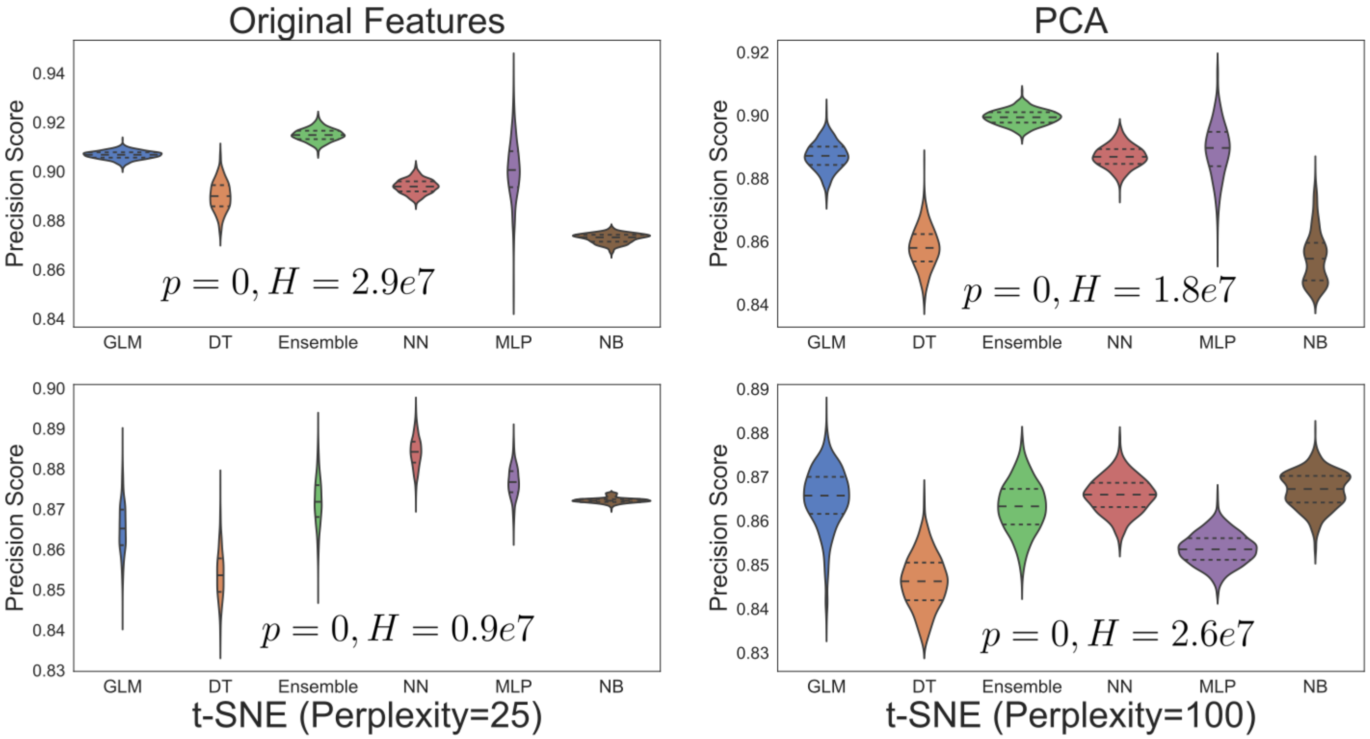

Figure 13. Univariate distribution of Precision Score on the testing data in original and reduced space.

5.1.2 k-Nearest Neighbors

The *k*-nearest neighbor classifier is the optimal classifier used for all the feature spaces. A higher number of stacking estimators are used for *t*-SNE feature spaces as opposed to the original and principal components feature spaces. As a result, the convergence timings are slightly more in the former case. The optimal pipeline search for all the feature spaces requires much less than 200 iterations to converge.

5.1.3 Multi-Layer Perceptron

The MLP classifier takes a considerable amount of time to find the optimal pipeline for all feature space with higher convergence time for *t*-SNE feature space as compared to the PCA and original features since the features in the former space requires a higher number of hierarchical stacking estimators than in other spaces to find out the optimal classification pipeline. Higher convergence time for the optimal pipeline is obtained as compared to other classification families due to exploration of a large number of hyperparameter spaces to select the hidden layer size for the input features, solver and activation function for minimizing the misclassification rate.

5.1.4 Naïve Bayes

Gaussian naïve Bayes is the optimal classifier used for classification for all the feature spaces. The convergence time for finding the optimal pipeline is higher in the principal component space as compared to other feature spaces. This is primarily due to the presence of high number of stack estimator and the function transformer. Function transformers are primarily used in the optimal pipelines. This is primarily due to the absence of the tuning parameters for the Naïve Bayes classification family. Thus, we see in Figure 7 that the number of iterations near the final convergence zone is high as each iteration evolves the pipeline by incorporating function transformers or stack estimators. However, we observe smaller convergence time in the *t*-SNE space with perplexity 100 as it does not have hierarchical pipelines as an optimal estimator.

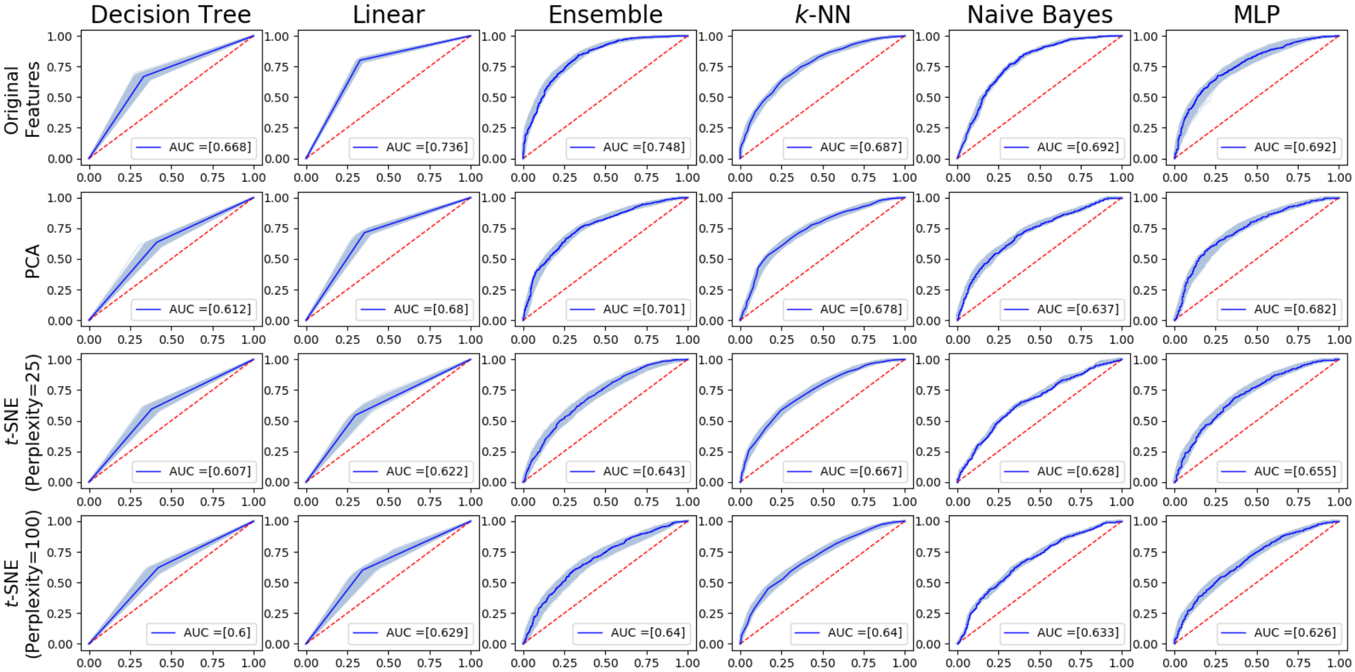

Figure 14. Median AUC value for different classification families in different feature spaces with Monte Carlo simulation with the median ROC curves along with the uncertainties generated by resampling.

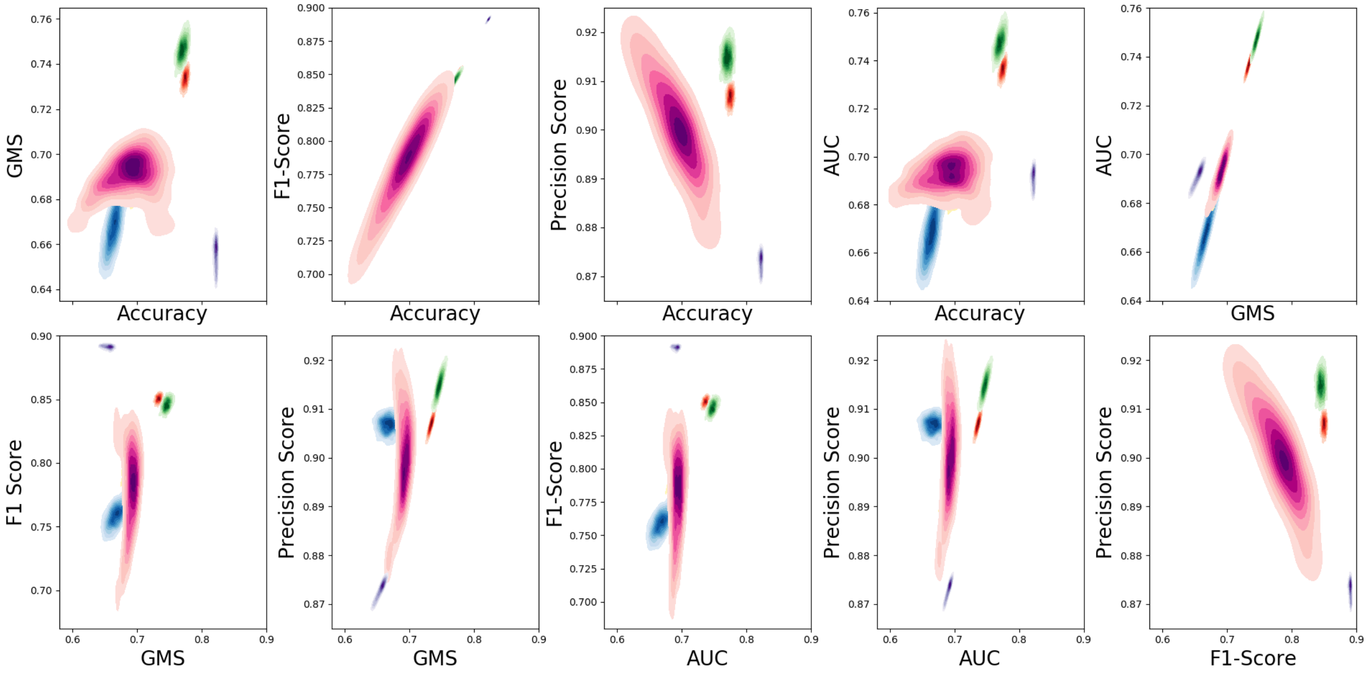

Figure 15. Bivariate distribution of the classification metrics in the original feature space, Red-GLM, Blue-DT, Green-Ensemble, Yellow-NN, Magenta-MLP, Purple-NB.

5.1.5 Decision Tree

The relative performance of the classifier with the decision tree is lower as compared to other classification families. The *ExtraTreeClassifier* is selected for the original and *t*-SNE space with perplexity 25, and decision tree classifier is selected for PCA and *t*-SNE space with perplexity parameter as 100. Decision tree provides higher AUC-score with complex pipelines consisting of function transformer and stacking estimators as compared to simpler optimal pipelines obtained in other feature spaces. Hence, we observe a slightly higher convergence time for finding the optimal pipeline in the original feature space as compared to the reduced feature spaces.

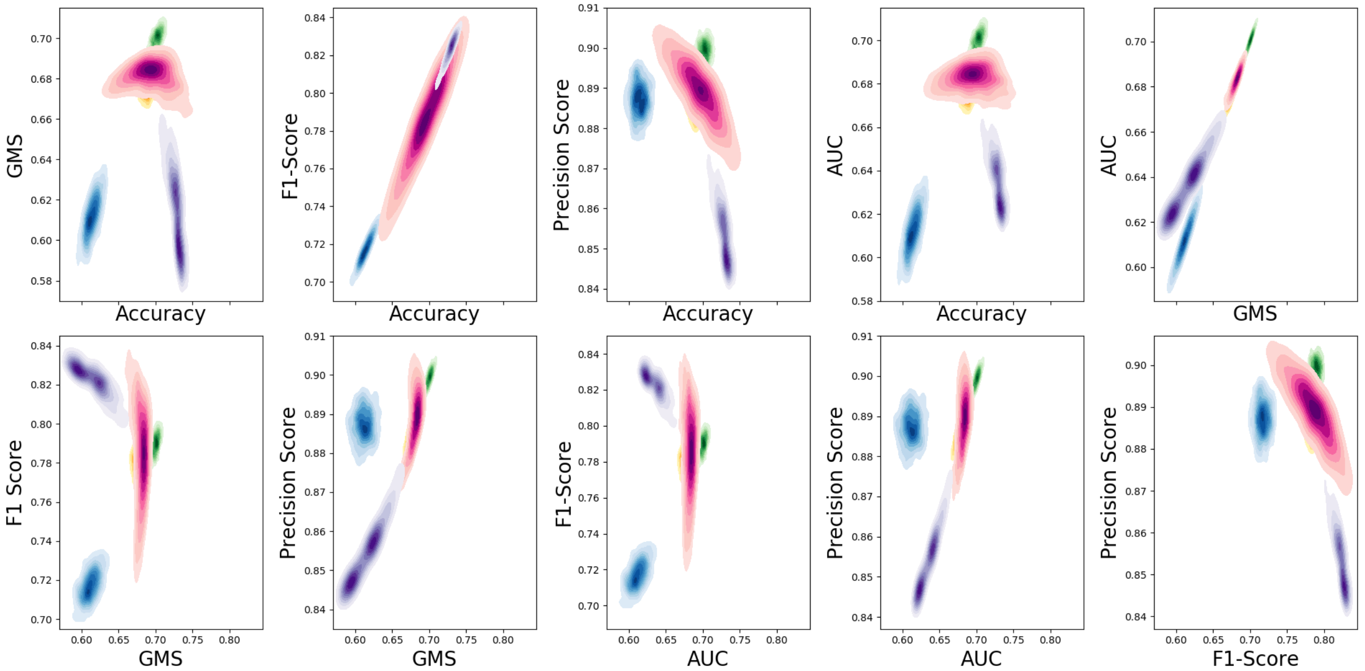

Figure 16. Bivariate distribution of the classification metrics in the principal component space, Red-GLM, Blue-DT, Green-Ensemble, Yellow-NN, Magenta-MLP, and Purple-NB.

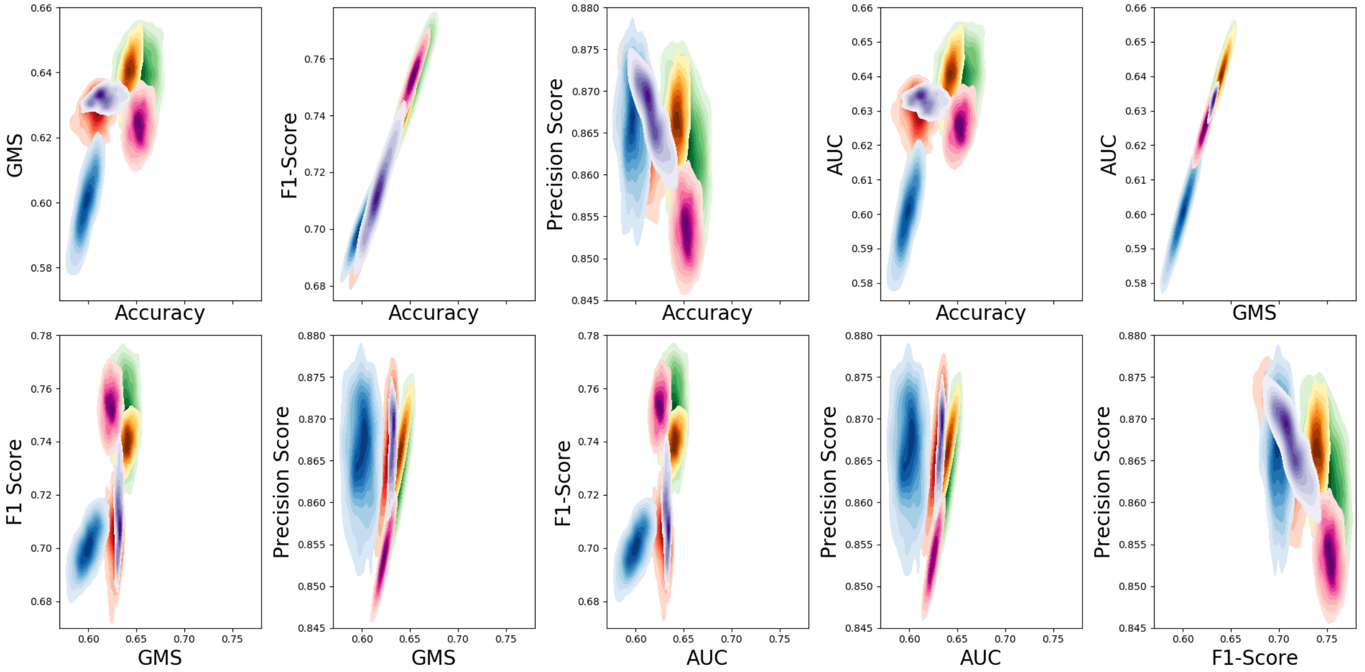

Figure 17. Bivariate distribution of the classification metrics in the *t*-SNE feature space with perplexity parameter as 25, Red-GLM, Blue-DT, Green-Ensemble, Yellow-NN, Magenta-MLP, and Purple-NB.

5.1.6 Ensemble Methods

Gradient boosting classifier is used as the optimal classifier for all the feature spaces. The optimal performances of the ensemble classifiers are better in the original and principal component feature spaces as compared to the *t*-SNE feature spaces. Ensemble methods take the highest number of iterations to converge as it involves the selection of the optimal estimator and the number of estimators as defined in (22). As shown in the optimal AUC score, it is evident that the ensemble methods work better with higher dimensional data and the relative performance to other classification families goes down with the reduction in the dimension of the data. The optimal pipelines took higher time for convergence for the reduced dimensional data as compared to the original higher dimensional one, thus making the pipeline significantly complicated in case of the former feature space. The performance evaluation of the optimal pipelines obtained is shown in the next subsections.

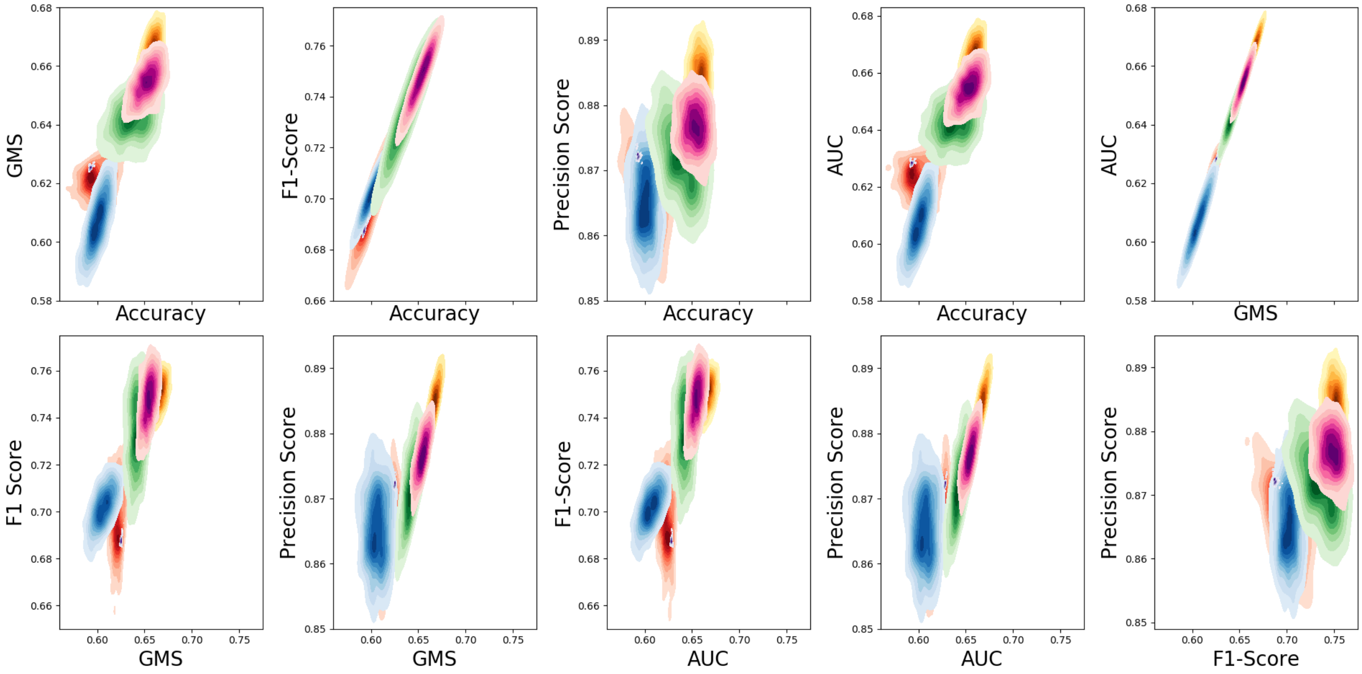

Figure 18. Bivariate distribution of the classification metrics in the *t*-SNE feature space with perplexity parameter as 100, Red-GLM, Blue-DT, Green-Ensemble, Yellow-NN, Magenta-MLP, and Purple-NB.

5.2 Comparison of Univariate and Bivariate Classification Performances

As explained in Figure 1, the 20% data which was held out is now used for testing the optimal pipeline obtained for the classifiers. The classifiers are trained on the data, which consists of the equal number of under-sampled majority class features as the minority class features as explained in Figure 1. The classifier is trained based on the optimal pipeline obtained using the GP optimizer.

The trained classifiers are then tested against the held-out testing data to obtain various classification metrics. The classifier with the optimal pipeline obtained via GP optimizer is trained with a sampled equal number datapoints from the majority and minority class features, and it is tested against the held-out data, thus obtaining a univariate distribution of the classification metrics viz. accuracy, GMS, AUC, F_1 score, precision, as shown in Figure 9-Figure 13 respectively. The ROC curves in Figure 14 suggests the plot between TPR and FPR as defined in (26), where classifying the urban and rural consumers are considered to be positive and negative labels respectively.

TPR indicates the probability that the classifier will classify the consumer as urban out of all urban consumers, while FPR indicates the probability that the rural consumer is classified as urban out of all the available rural consumers. Since the decision of the optimal classifier pipeline obtained for decision tree and GLM families are non-probabilistic, the ROC curve consists of two straight lines with a corner for all feature spaces. Classification measures like accuracy-F_1 score and AUC-GMS are correlated as evident from the bivariate distribution plots shown in Figure 15-Figure 18 where a 2D kernel density estimate (KDE) of the classification performance metrics are shown with high and low confidence regions out of the 1000 Monte Carlo under-sampling draws. From Figure 9, Figure 10 and Figure 11, we observe the accuracy, AUC and the geometric mean score, which is the normalized accuracy for both the classes. From this measure as defined in (25) and (31) we obtain the robust metrics of the classification algorithms against class imbalance which can be explained for both the original higher dimensional and reduced feature spaces.

5.3 Comparison of Classification in the Original vs Reduced Features Spaces

5.3.1 Classification Performance in the Original Feature Space

It is observed that the Naïve Bayes performs the best in terms of overall accuracy in real feature space. However, the GMS relatively drops down as evident in the bivariate plot in Figure 15 signifying that the independence of features in (17) will not be able to handle the class imbalance optimally and it is likely to label the rural energy household as urban. The accuracy and GMS of ensemble methods and the GLM families are similar which suggest that complex pipelines of GLM families in (11) can match the classification accuracy of the ensemble of estimators in (22) even with a small drop in the performance. The MLP classifier has the highest spread in the performance metrics. It is also observed in the univariate violin plots in Figure 9 and Figure 10 that samples of the GMS have lesser outliers, as compared to the samples of the accuracy metric. The AUC and GMS are similar positively correlated metrics with almost a straight line observed in the bivariate plots. The performance of MLP and NN are similar in the original feature space. However, the former has more spread than the latter. Class imbalance weighed accuracy score has higher kurtosis as compared to the accuracy scores. The GLM Ridge classifier performs better than the decision tree classifier as the classifier is robust in its performance due to producing a thinner uncertainty bound. Ensemble classifiers perform the best out of all the classifiers, but it is less robust as compared to the GLM Ridge classifiers due to the thicker uncertainty bound. So, it is probable that ensemble method classifiers will be more inconsistent with different Monte Carlo sampling iterations as compared to the linear classifiers suggesting that the hierarchical pipelines for GLM classifiers are more robust to sampling iterations than the ensemble classifiers. Even though the AUC score generally depends upon the precision value, as obtained from the almost linearly correlated bivariate plot in the case of GLM, ensemble and naïve Bayes classifier, these plots are not positively correlated in the case of MLP and DT classifiers. With naïve Bayes classifier, even though the performance is not as good as the ensemble or GLM classifiers, in general, it is more robust with different under-sampling runs.

5.3.2 Classification Performance in the Principal Components Feature Space

The performance of classifiers for principal component features is much inferior as compared to the real features. Similar to the original feature space, the Naïve Bayes is sensitive to the imbalance in the data in the principal component space in case of training samples, but the performance falls behind the ensemble classifiers. It suggests that the ensemble method pipelines are more robust in terms of various types of testing data. We observe that uncertainties in classifying the urban consumers are relatively high with the Monte Carlo under-sampling as compared to the features in the original high dimensional spaces. We observe a greater change in the performance for the GLM classifier even though a similar hierarchy of pipelines is used for the feature space. Even though decision tree classifiers perform the worst out of all the classifiers, but the optimal pipeline obtained performed better in the testing samples as compared to the training samples when the median AUC value is observed. The nearest neighbor classifier performs more distinctly as compared to the original feature space as seen from the bivariate metrics plot in Figure 16. The performance of the ensemble classifier is robust to the class imbalance as the performance curve does not change much with the accuracy and the GMS score. The performance of the classifier in PCA feature space is more similar as compared to the features in the original feature space as suggested from the lower values of H obtained from the hypothesis testing reported in the univariate violin plots in the previous subsections.

5.3.3 Classification Performance in the t-SNE Feature Space

We observe that the performance of classifiers is much inferior in the *t*-SNE feature space as compared to the principal component and original feature space. We do not observe a significant shift in the accuracy and GMS of the classifiers in the *t*-SNE space, which suggests that the class imbalance is not much affected in the lower dimensional space. However, the classifier performance in the *t*-SNE space is more similar as compared to other feature spaces due to the lower value of H obtained for the classification metrics samples. However, the distinctiveness of the classifier performance is observed, as shown in the bivariate plots in Figure 17 and Figure 18. We observe a higher kurtosis for the univariate classification performance values for the *t*-SNE feature space with the perplexity parameter as 100 as compared to 25. The bivariate spread in the decision tree is highest, unlike the MLP in higher dimensional spaces. The performance of the *k*-NN classifier is comparatively superior as compared to the other classifiers in the *t*-SNE feature space. In addition to inferior classification performance, we also observe an uncertainty in the classification performance with changing thresholds as compared to other feature spaces, as the uncertainty bound in the ROC curves in Figure 14 are thicker than the previous cases except in the case of Naïve Bayes classifier.

6. Energy Policy Implications of the Unbalanced Classification Problem

The primary objective of data driven investigation of urban and rural energy REC is to frame energy policies that ensure energy efficiency and reduce emissions, as summarized in Table 3. As described in (Krey et al., 2012), the emissions are not highly dependent on the urbanization rate. However, the urban residents have to take the initiative for awareness and actions via robust governance by the urban city administration regarding climate change (Dodman, 2011). Another way to enable rural residents to adopt healthier natural and energy lifestyles is by reducing fossil fuel energy subsidies (Fan et al., 2020). The influencing factor causing the differences in energy consumption can help implement robust governance (Wang et al., 2021). The energy policies can also be formulated based on the results obtained at different regional levels. However, too much reliance on technology is not beneficial as it can lead to rebound effects and shallow energy governance and policies (Du et al., 2021). In order to curb the rising REC, improve energy security and mitigate climate change, urban development and energy efficiency planning can be integrated as discussed in (Dhakal, 2009). Similar goals can be achieved in rural regions by investing in cleaner energy sources, improving indoor insulation and raising the lifestyle of the residents by increasing their incomes (Niu et al., 2012). Other aspects include optimizing the fuel mix to meet the demands, reducing taxes on energy-saving appliances, controlling extravagant lifestyles and providing incentives for saving household energy (Fan et al., 2013). Focus can also be directed towards establishing firm standards for energy conservation (Y. Zhang, 2013) and improving awareness by investing in advertisements for encouraging energy-saving behaviour (Ding et al., 2017).

Table 3: The policy implications from the recent studies regarding urban and rural energy consumption

References	**Energy Policy Implications**
(Krey et al., 2012)	• CO_2 emissions are not affected by the urbanization rate.
(Dodman, 2011)	• Improvement in awareness about energy efficiency and environmental concerns. • Active involvement of urban authorities in global governance and climate change.
(Fan et al., 2020)	• Removing subsidies on fossil fuel energy sources. • Improving rural consumer awareness about energy conservation and climatic factors.
(Wang et al., 2021)	• Policies on household energy consumption should be based on the regional level. • Influencing factors between the differences in REC should be considered while formulating policies.
(Du et al., 2021)	• Benefits are not solely obtained from technological progress due to a possible rebound effect. • The rebound effect is more in the rural regions; hence technologies have to be carefully deployed.
(G. Chen et al., 2019)	• Policies should also incorporate indirect energy consumption, which includes lifestyle aspects like smoking and alcohol. • Mental health in urban and rural residents also plays a crucial role in energy consumption.
(Dhakal, 2009)	• Integrated planning on urban development and energy efficiency. • Large cities should be targeted for improving energy security and climate action.
(Niu et al., 2012)	• Investment in cleaner energy sources in rural areas. • Rural heating insulation improvement. • Income-increase for the rural residents.
(Fan et al., 2013)	• Optimizing the fuel mix for energy generation. • The extravagant lifestyle of the urban residents should be considered for governance. • Tax reduction on energy-saving appliances.
(Ren et al., 2014)	• Systematic partnership between urban and rural regions for the optimal use of resources. • Usage of the local carbon price for economic and environmental benefits.
(Y. Zhang, 2013)	• Firm standards are to be set for energy conservation and circular economy. • Policies and subsidies for sustainable economic activities.
(Ding et al., 2017)	• Investment in the advertisement for the adoption of energy-saving behaviour. • Incentives for different regions inculcating a sense for the environment.
(C. Chen et al., 2019)	• Equitable allocation of resources in urban and rural areas.
This Paper	• Introducing incentives for the usage of cleaner electricity and demand response programs for the particular application of energy like space heating for urban residents. • Subsidies on LPG, propane, and natural gas usage for rural residents. Spreading awareness on sustainable behaviour. • Pilot projects for sustainable transition for rural residents who have energy consumption characteristics similar to urban residents.

Previous works have focused more on the general policies on improving energy efficiency. The classification results reported in this paper are based on energy usage and its purpose and aims are discussed in Section 2. As we saw from the feature selection, features containing overall LPG/propane usage and its application space heating and overall electricity and its usage for water heating form a difference between urban and rural energy consumption patterns, yielding higher Fisher score. It means that there is a difference in the consumption patterns for these energy sources and applications. However, we also found that the features like kerosene used for water heating, total thermal units used for space heating, electricity cost due to refrigeration use, fuel oil usage for water heating and kerosene yields low Fisher score. Additionally, the rural consumers do not use natural gas, while urban consumers do not use electricity for space heating. Hence based on these differences, we can frame the policies for the classification results as follows relating them with four elements of the confusion matrix of the unbalanced binary classification problem:

- ***Case 1: Urban residents classified as urban (TP):*** Urban residents have access to high-quality energy, and there is enough awareness amongst urban residents regarding energy conservation. Provision and incentives should be provided to use cleaner energy sources like promoting the use of solar power. Demand response strategies can be framed for specific applications like space or water heating, coupling with advanced technologies as the rebound effect is low in urban areas (Du et al., 2021). The tariff should be promoted to offset the rise in REC in urban areas. In addition, the taxes on energy-saving appliances should be relaxed.
- ***Case 2: Rural residents classified as rural (TN):*** Rural residents have access to lower quality of energy and subsidies should be provided mainly for energy sources which cause a difference in energy consumption compared to urban regions like LPG/propane and natural gas. There should be systematic disinvestment in energy sources like kerosene and fuel oil which causes environmental emissions. As the rebound effect is higher in rural regions, the focus has to be higher on technological progress. As electricity is prevalent in rural regions, investment in cleaner energy sources can be focused on.
- ***Case 3: Rural residents classified as urban (FP):*** The residents in this class live in rural regions with a lifestyle similar to urban residents, which is possible due to the migration of urban residents to rural regions to lead a quiet lifestyle (Kim, 1983). Hence these residents have access to higher quality energy as compared to others in rural areas. Provisions and incentives can be introduced to use the cleaner energy source and demand response techniques. Furthermore, pilot projects for sustainability transition can be introduced for these households. The subsidies for these residents on the usage of LPG/propane and natural gas can be waived off. This class of residents can be entrusted with the responsibility of spreading awareness regarding energy-saving behaviour.
- ***Case 4: Urban residents classified as rural (FN):*** Some residents might have moved from rural to urban areas for better lifestyle and financial opportunities who may not be aware of the importance of sustainable practices in energy consumption. These residents are rare, but one possible responsibility by the city authorities may be to spread awareness regarding cleaner energy sources. In addition, several incentives can be introduced for encouraging the transition to a sustainable lifestyle.

As we observe in Figure 13, that the precision score is almost 0.9. The precision score represents the proportion of the households classified as urban belong to urban areas. This suggests that almost 10% of rural residents have a similar lifestyle as urban areas. The AUC deals with the true positive rate and the false-positive rate with the change in threshold. This case signifies the true residents in urban regions and the rural residents classified as urban, respectively. We observe in Figure 15 that the GMS is high correlated or almost proportional to the AUC score. The maximum GMS value of 0.75 suggests that 75% of the residents are correctly classified, even though 10% of the rural residents have a similar lifestyle as urban areas.

7. Conclusions

We demonstrate the classification of urban vs rural consumers based on their energy usage using various classification families on an unbalanced dataset of energy consumption patterns. The classification is performed in the original higher-dimensional and two reduced feature spaces. However, due to the unbalanced distribution of positive and negative class labels, the classification problem becomes a highly unbalanced one. Firstly, the essential features are identified using feature selection technique using Fisher score. Then the original features are transformed into lower-dimensional spaces using dimensionality reduction methods like PCA and *t*-SNE. Optimal pipeline structure is obtained using genetic programming for each classification family to tune the optimal hyperparameters. The optimal pipeline for each classifier in each feature space is then used to classify the data, where each classifier is trained by Monte Carlo under-sampling of the majority class datapoints. The random under-sampling process is continued 1000 times, and the classifier is then tested against these drawn samples each time within a Monte Carlo simulation framework to generate several summary statistics of the classification performances. Thus, we obtain probabilistic classification metrics along with their univariate and bivariate distributions to compare across different classifier families and feature spaces. The main findings of the paper can be summarised as follows:

- The optimal classification performance is obtained in the original feature space, and the performance worsens with the decrease in the feature dimensionality. However, the performance uncertainty to class imbalance is better in the lower dimensional spaces after dimensionality reduction as compared to the higher dimensional ones.
- Ensemble family of classifiers perform the best in original and PCA feature space while nearest neighbour classifiers perform optimally in the *t*-SNE feature space.
- The uncertainty bound in the classifiers for varying threshold increases when classification is performed in the reduced dimensional spaces using PCA or *t*-SNE.
- Convergence algorithm for finding the optimal pipeline for multilayer perceptron classifier takes the maximum amount of computational runtime, and there is high variability in its classification performance, especially in the original and principal component feature spaces.
- Accuracy and F_1 Score, also the AUC and GMS are positively correlated with each other.
- Naïve Bayes classifier family is least robust to an imbalance dataset as it is most likely to identify the wrong labels in the case of high-class imbalance. Ensemble method classifiers are more robust to class imbalance.

The implications on energy policies obtained after the hyperparameter optimized classification are given as follows:

- Introducing incentives for the usage of cleaner electricity and demand response programs for the particular application of energy policy like space heating. Tax reforms on energy-saving appliances and encouraging cleaner sources of energy.
- Subsidies on LPG, propane and natural gas usage and spreading awareness on sustainable behaviour for rural residents.
- Pilot projects for sustainability transition can be introduced for rural households with similar energy consumption patterns as urban counterparts. The subsidies for these residents on the usage of LPG/propane and natural gas can be waived off.

This research can be extended further by incorporating more information about the geographical areas while performing a multi-class classification task. Different over-sampling and under-sampling strategies can be explored in Monte Carlo simulation framework to analyse different classification performance metrics.

ACKNOWLEDGMENT

SD was partially supported by the ERDF Deep Digital Cornwall project number: 05R18P02820.

AUTHOR CONTRIBUTIONS

Deepak Kumar Panda: Software, Formal analysis, Investigation, Data Curation, Writing - Original Draft, Visualization. **Saptarshi Das**: Conceptualization, Methodology, Validation, Resources, Writing - Review & Editing, Visualization, Supervision, Project administration, Funding acquisition. **Stuart Townley**: Formal analysis, Writing - Review & Editing, Resources, Supervision.

REFERENCES

Administration, U.S. Energy Information. (2013). *Residential Energy Consumption Survey (RECS), URL: https://www.eia.gov/consumption/residential/methodology/2009/pdf/techdoc-summary010413.pdf*.

Arora, N. K. & Mishra, I. (2019). United Nations Sustainable Development Goals 2030 and environmental sustainability: race against time. *Environmental Sustainability*, *2*(4), 339–342.

Bergstra, J. & Bengio, Y. (2012). Random search for hyper-parameter optimization. *Journal of Machine Learning Research*, *13*(Feb), 281–305.

Bhowan, U., Johnston, M., Zhang, M. & Yao, X. (2012). Evolving diverse ensembles using genetic programming for classification with unbalanced data. *IEEE Transactions on Evolutionary Computation*, *17*(3), 368–386.

Bicego, M., Farinelli, A., Grosso, E., Paolini, D. & Ramchurn, S. D. (2018). On the distinctiveness of the electricity load profile. *Pattern Recognition*, *74*, 317–325.

Bottou, L. (2010). Large-scale machine learning with stochastic gradient descent. In *Proceedings of COMPSTAT'2010* (pp. 177–186). Springer.

Bourlard, H. & Kamp, Y. (1988). Auto-association by multilayer perceptrons and singular value decomposition. *Biological Cybernetics*, *59*(4-5), 291–294.

Bunte, K., Haase, S., Biehl, M. & Villmann, T. (2012). Stochastic neighbor embedding (SNE) for dimension reduction and visualization using arbitrary divergences. *Neurocomputing*, *90*, 23–45.

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

Chen, C., Liu, G., Meng, F., Hao, Y., Zhang, Y. & Casazza, M. (2019). Energy consumption and carbon footprint accounting of urban and rural residents in Beijing through Consumer Lifestyle Approach. *Ecological Indicators*, *98*, 575–586.

Chen, G., Zhu, Y., Wiedmann, T., Yao, L., Xu, L. & Wang, Y. (2019). Urban-rural disparities of household energy requirements and influence factors in China: Classification tree models. *Applied Energy*, *250*, 1321–1335.

Coomans, D. & Massart, D. L. (1982). Alternative k-nearest neighbour rules in supervised pattern recognition: Part 1. k-Nearest neighbour classification by using alternative voting rules. *Analytica Chimica Acta*, *136*, 15–27.

Crammer, K., Dekel, O., Keshet, J., Shalev-Shwartz, S. & Singer, Y. (2006). Online passive-aggressive algorithms. *Journal of Machine Learning Research*, *7*(Mar), 551–585.

De Lauretis, S., Ghersi, F. & Cayla, J.-M. (2017). Energy consumption and activity patterns: an analysis extended to total time and energy use for French households. *Applied Energy*, *206*, 634–648.

Deb, K., Pratap, A., Agarwal, S. & Meyarivan, T. (2002). A fast and elitist multiobjective genetic algorithm: NSGA-II. *Evolutionary Computation, IEEE Transactions on*, *6*(2), 182–197.

Dhakal, S. (2009). Urban energy use and carbon emissions from cities in China and policy implications. *Energy Policy*, *37*(11), 4208–4219.

Dietterich, T. G. (2000). An experimental comparison of three methods for constructing ensembles of decision trees: Bagging, boosting, and randomization. *Machine Learning*, *40*(2), 139–157.

Ding, Z., Wang, G., Liu, Z. & Long, R. (2017). Research on differences in the factors influencing the energy-saving behavior of urban and rural residents in China-A case study of Jiangsu Province. *Energy Policy, 100*, 252–259.

Dodman, D. (2011). Forces driving urban greenhouse gas emissions. *Current Opinion in Environmental Sustainability, 3*(3), 121–125.

Druckman, A. & Jackson, T. (2008). Household energy consumption in the UK: A highly geographically and socio-economically disaggregated model. *Energy Policy*, *36*(8), 3177–3192.

Du, Q., Han, X., Li, Y., Li, Z., Xia, B. & Guo, X. (2021). The energy rebound effect of residential buildings: Evidence from urban and rural areas in China. *Energy Policy*, *153*, 112235.

Duda, R. O., Hart, P. E. & Stork, D. G. (2012). *Pattern Classification*. John Wiley & Sons.

Fan, J.-L., Liao, H., Liang, Q.-M., Tatano, H., Liu, C.-F. & Wei, Y.-M. (2013). Residential carbon emission evolutions in urban-rural divided China: an end-use and behavior analysis. *Applied Energy, 101*, 323–332.

Fan, J.-L., Zeng, B., Hu, J.-W., Zhang, X. & Wang, H. (2020). The impact of climate change on residential energy consumption in urban and rural divided southern and northern China. *Environmental Geochemistry and Health*, *42*(3), 969–985.

Fawcett, T. (2006). An introduction to ROC analysis. *Pattern Recognition Letters*, *27*(8), 861–874.

Fay, R. E. (1989). Theory and application of replicate weighting for variance calculations. In *Proceedings of the Survey Research Methods Section of the American Statistical Association* (pp. 212–7).

Figueiredo, V., Rodrigues, F., Vale, Z. & Gouveia, J. B. (2005). An electric energy consumer characterization framework based on data mining techniques. *IEEE Transactions on Power Systems*, *20*(2), 596–602.

Fortin, F.-A., De Rainville, F.-M., Gardner, M.-A. G., Parizeau, M. & Gagné, C. (2012). DEAP: Evolutionary algorithms made easy. *The Journal of Machine Learning Research*, *13*(1), 2171–2175.

Frank, E. & Bouckaert, R. R. (2006). Naive bayes for text classification with unbalanced classes. In *European Conference on Principles of Data Mining and Knowledge Discovery* (pp. 503–510).

Galvin, R. & Sunikka-Blank, M. (2018). Economic inequality and household energy consumption in high-income countries: a challenge for social science based energy research. *Ecological Economics*, *153*, 78–88.

Ganganwar, V. (2012). An overview of classification algorithms for imbalanced datasets. *International Journal of Emerging Technology and Advanced Engineering*, *2*(4), 42–47.

Garcia, S. & Herrera, F. (2009). Evolutionary undersampling for classification with imbalanced datasets: Proposals and taxonomy. *Evolutionary Computation*, *17*(3), 275–306.

Geurts, P., Ernst, D. & Wehenkel, L. (2006). Extremely randomized trees. *Machine Learning*, *63*(1), 3–42.

Ghosh, S., Panda, D. K., Das, S. & Chatterjee, D. (2021). Cross-correlation based classification of electrical appliances for non-intrusive load monitoring. In *2021 International Conference on Sustainable Energy and Future Electric Transportation (SEFET)* (pp. 1–6).

Gill, B. & Moeller, S. (2018). GHG emissions and the rural-urban divide. A carbon footprint analysis based on the German official income and expenditure survey. *Ecological Economics*, *145*, 160–169.

Gong, J. & Kim, H. (2017). RHSBoost: Improving classification performance in imbalance data. *Computational Statistics & Data Analysis*, *111*, 1–13.

Grünewald, P. & Reisch, T. (2020). The trust gap: Social perceptions of privacy data for energy services in the United Kingdom. *Energy Research & Social Science*, *68*, 101534.

Hutter, F., Lücke, J. & Schmidt-Thieme, L. (2015). Beyond manual tuning of hyperparameters. *KI-Künstliche Intelligenz*, *29*(4), 329–337.

Khandker, S. R., Barnes, D. F. & Samad, H. A. (2012). Are the energy poor also income poor? Evidence from India. *Energy Policy*, *47*, 1–12.

Kim, J. (1983). Factors affecting urban-to-rural migration (USA). In *Growth & Change* (pp. 38–43).

Kleinbaum, D. G., Dietz, K., Gail, M., Klein, M. & Klein, M. (2002). *Logistic Regression*. Springer.

Koza, J. R. & others. (1994). *Genetic programming II* (Vol. 17). MIT press Cambridge.

Krawczyk, B., Galar, M., Jelen, L. & Herrera, F. (2016). Evolutionary undersampling boosting for imbalanced classification of breast cancer malignancy. *Applied Soft Computing*, *38*, 714–726.

Krey, V., O'Neill, B. C., van Ruijven, B., Chaturvedi, V., Daioglou, V., Eom, J., … Ren, X. (2012). Urban and rural energy use and carbon dioxide emissions in Asia. *Energy Economics*, *34*, S272–S283.

Lemaitre, G., Nogueira, F. & Aridas, C. K. (2017). Imbalanced-learn: A python toolbox to tackle the curse of imbalanced datasets in machine learning. *The Journal of Machine Learning Research*, *18*(1), 559–563.

Li, K. & Du, T. C. (2012). Building a targeted mobile advertising system for location-based services. *Decision Support Systems*, *54*(1), 1–8.

Li, M., Shan, R., Hernandez, M., Mallampalli, V. & Patiño-Echeverri, D. (2019). Effects of population, urbanization, household size, and income on electric appliance adoption in the Chinese residential sector towards 2050. *Applied Energy*, *236*, 293–306.

Lin, C.-T., Hsieh, T.-Y., Liu, Y.-T., Lin, Y.-Y., Fang, C.-N., Wang, Y.-K., … Chuang, C.-H. (2017). Minority oversampling in kernel adaptive subspaces for class imbalanced datasets. *IEEE Transactions on Knowledge and Data Engineering*, *30*(5), 950–962.

Lisovich, M. A., Mulligan, D. K. & Wicker, S. B. (2010). Inferring personal information from demand-response systems. *IEEE Security & Privacy*, *8*(1), 11–20.

Liu, X.-Y., Wu, J. & Zhou, Z.-H. (2008). Exploratory undersampling for class-imbalance learning. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, *39*(2), 539–550.

Maaten, L. van der, & Hinton, G. (2008). Visualizing data using t-SNE. *Journal of Machine Learning Research*, *9*(Nov), 2579–2605.

Mani, I. & Zhang, I. (2003). kNN approach to unbalanced data distributions: a case study involving information extraction. In *Proceedings of Workshop on Learning from Imbalanced Datasets* (Vol. 126).

Martinez, A. M. & Kak, A. C. (2001). Pca versus lda. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *23*(2), 228–233.

Menardi, G. & Torelli, N. (2014). Training and assessing classification rules with imbalanced data. *Data Mining and Knowledge Discovery*, *28*(1), 92–122.

Miao, A., Zhuang, J., Tang, Y., He, Y., Chu, X. & Luo, S. (2018). Hyperspectral image-based variety classification of waxy maize seeds by the t-SNE model and procrustes analysis. *Sensors*, *18*(12), 4391.

Mrówczynska, M., Skiba, M., Bazan-Krzywosza'nska, A. & Sztubecka, M. (2020). Household standards and socio-economic aspects as a factor determining energy consumption in the city. *Applied Energy*, *264*, 114680.

Nasution, M., Sitompul, O. & Ramli, M. (2018). PCA based feature reduction to improve the accuracy of decision tree c4. 5 classification. In *Journal of Physics: Conference Series* (Vol. 978, p. 012058).

Nie, H., Kemp, R., Xu, J., Vasseur, V. & Fan, Y. (2018). Drivers of urban and rural residential energy consumption in China from the perspectives of climate and economic effects. *Journal of Cleaner Production*, *172*, 2954–2963.

Niu, S., Zhang, X., Zhao, C. & Niu, Y. (2012). Variations in energy consumption and survival status between rural and urban households: a case study of the Western Loess Plateau, China. *Energy Policy*, *49*, 515–527.

Ofek, N., Rokach, L., Stern, R. & Shabtai, A. (2017). Fast-CBUS: A fast clustering-based undersampling method for addressing the class imbalance problem. *Neurocomputing*, *243*, 88–102.

Olson, R. S., Bartley, N., Urbanowicz, R. J. & Moore, J. H. (2016). Evaluation of a tree-based pipeline optimization tool for automating data science. In *Proceedings of the Genetic and Evolutionary Computation Conference 2016* (pp. 485–492).

Olson, R. S. & Moore, J. H. (2016). TPOT: A tree-based pipeline optimization tool for automating machine learning. In *Workshop on Automatic Machine Learning* (pp. 66–74).

Pachauri, R. (1983). Third World energy policies The urban-rural divide. *Energy Policy*, *11*(3), 217–224.

Pachauri, S. & Jiang, L. (2008). The household energy transition in India and China. *Energy Policy*, *36*(11), 4022–4035.

Panda, D. K. & Das, S. (2021). Smart grid architecture model for control, optimization and data analytics of future power networks with more renewable energy. *Journal of Cleaner Production*, *301*, 126877.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., … others. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, *12*(Oct), 2825–2830.

Qian, Y., Liang, Y., Li, M., Feng, G. & Shi, X. (2014). A resampling ensemble algorithm for classification of imbalance problems. *Neurocomputing*, *143*, 57–67.

Qiao, X. & Liu, Y. (2009). Adaptive weighted learning for unbalanced multicategory classification. *Biometrics*, *65*(1), 159–168.

Ren, H., Wu, Q., Ren, J. & Gao, W. (2014). Cost-effectiveness analysis of local energy management based on urban-rural cooperation in China. *Applied Thermal Engineering*, *64*(1-2), 224–232.

Residential Energy Consumption Survey. Retrieved from https://www.kaggle.com/kingburrito666/residential-energy-consumption-survey

Rish, I. & others. (2001). An empirical study of the naive Bayes classifier. In *IJCAI 2001 workshop on Empirical Methods in Artificial Intelligence* (Vol. 3, pp. 41–46).

Safavian, S. R. & Landgrebe, D. (1991). A survey of decision tree classifier methodology. *IEEE Transactions on Systems, Man, and Cybernetics*, *21*(3), 660–674.

Seidl, D. E., Paulus, G., Jankowski, P. & Regenfelder, M. (2015). Spatial obfuscation methods for privacy protection of household-level data. *Applied Geography*, *63*, 253–263.

Seiffert, C., Khoshgoftaar, T. M., Van Hulse, J. & Napolitano, A. (2009). RUSBoost: A hybrid approach to alleviating class imbalance. *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, *40*(1), 185–197.

Shigetomi, Y., Matsumoto, K., Ogawa, Y., Shiraki, H., Yamamoto, Y., Ochi, Y. & Ehara, T. (2018). Driving forces underlying sub-national carbon dioxide emissions within the household sector and implications for the Paris Agreement targets in Japan. *Applied Energy*, *228*, 2321–2332.

Thorve, S., Swarup, S., Marathe, A., Chungbaek, Y., Nordberg, E. K. & Marathe, M. V. (2018). Simulating residential energy demand in urban and rural areas. In *2018 Winter Simulation Conference (WSC)* (pp. 548–559).

Traven, G., Matijevivc, G., Zwitter, T., Zerjal, M., Kos, J., Asplund, M., … others. (2017). The Galah survey: classification and diagnostics with t-SNE reduction of spectral information. *The Astrophysical Journal Supplement Series*, *228*(2), 24.

Tsekouras, G., Kotoulas, P., Tsirekis, C., Dialynas, E. & Hatziargyriou, N. (2008). A pattern recognition methodology for evaluation of load profiles and typical days of large electricity customers. *Electric Power Systems Research*, *78*(9), 1494–1510.

Tso, G. K. & Yau, K. K. (2007). Predicting electricity energy consumption: A comparison of regression analysis, decision tree and neural networks. *Energy*, *32*(9), 1761–1768.

Vasan, K. K. & Surendiran, B. (2016). Dimensionality reduction using Principal Component Analysis for network intrusion detection. *Perspectives in Science*, *8*, 510–512.

Villalba, S. D. & Cunningham, P. (2007). An evaluation of dimension reduction techniques for one-class classification. *Artificial Intelligence Review*, *27*(4), 273–294.

Wang, S., Sun, S., Zhao, E. & Wang, S. (2021). Urban and rural differences and regional assessment of household energy consumption in China. *Energy*, 121091.

Wijaya, T. K., Ganu, T., Chakraborty, D., Aberer, K. & Seetharam, D. P. (2014). Consumer segmentation and knowledge extraction from smart meter and survey data. In *Proceedings of the 2014 SIAM International Conference on Data Mining* (pp. 226–234).

Zhang, T., Zhang, G., Lu, J., Feng, X. & Yang, W. (2011). A new index and classification approach for load pattern analysis of large electricity customers. *IEEE Transactions on Power Systems*, *27*(1), 153–160.

Zhang, X., Li, Y., Kotagiri, R., Wu, L., Tari, Z. & Cheriet, M. (2017). KRNN: k Rare-class Nearest Neighbour classification. *Pattern Recognition*, *62*, 33–44.

Zhang, Y. (2013). Impact of urban and rural household consumption on carbon emissions in China. *Economic Systems Research*, *25*(3), 287–299.

Zutty, J., Long, D., Adams, H., Bennett, G. & Baxter, C. (2015). Multiple objective vector-based genetic programming using human-derived primitives. In *Proceedings of the 2015 Annual Conference on Genetic and Evolutionary Computation* (pp. 1127–1134).