Exploratory Analysis and Preprocessing of Dataset for the Classification of Osteosarcoma Types

Amoakoh Gyasi-Agyei, Tahsien Al-Quraishi, Bhagwan Das, and Johnson I. Agbinya

School of Engineering and Information Technology, Melbourne Institute of Technology, Australia

{*agyasiagyei, tquraishi, bdas, jagbinya*}*@mit.edu.au*

***Abstract*—Osteosarcoma is a born-forming tumor which is more common with children and young adults than adults. Classification of its type is crucial to its proper treatment and possible survival. Machine learning models, trained on datasets of the disease, are are effective classification tool than hand-crafted features which are highly dependent on a pathologist's expertise. However, machine learning models are only useful if the dataset used to train them are representative, of good quality and well prepared. Thus, data preprocessing and statistical analysis of datasets used to train models are necessary precursors to model learning. Data preprocessing is the most demanding task in the model learning pipeline. Thus, availability of a preprocessed quality dataset for a given task is desirable for model learning tasks. Two things are needed to obtain good results in a machine learning project: good data preprocessing and good algorithms. This paper provides a thorough preprocessing and statistical analysis of a 1144-sample dataset of osteosarcoma patients, to render the dataset ready for model learning. The efficacy of the preprocessing methods is verified by training multiclass logistic regression in Python using datasets with 63 of the 69 variables, with PCA and feature selection to achieve the respective predictive accuracies of 19.27%, 65.14% and 80.28%.**

I. Introduction

Osteosarcoma is a rare but the most common malignant tumor in bones with about 4.4 cases per million children annually [1]. There are many ways to classify the disease, one of which is primary and secondary. Primary osteosarcoma, being 75% of the cases [1], usually occurs in children and young adults as abnormality in bone development, while secondary osteosarcoma is common with adults with mature bones as a result of another disease. Primary osteosarcoma can also occur in different forms, with the common types being intramedullary osteosarcoma, juxtacortical osteosarcoma and extraskeletal osteosarcoma [2]. Osteosarcoma can also be categorized into conventional central osteosarcoma (which also has the common types osteoblastic, chondroblastic and fibroblastic), telangiectatic, intraosseous and small osteosarcoma [3].

Osteosarcoma can affect any bone, but typically affects the knee, metaphysis, proximal tibia, distal femur or proximal humerus, and can spread to other body parts, such as lungs. Its diagnosis occur via medical imaging (i.e. X-rays, CT scans, and MRI), test for serum tumor markers, or core-needle biopsy [4]. Osteosarcoma can be treated by surgery, chemotherapy, radiation therapy or for benign cases, by observation and information [1]. Causes of osteosarcoma include genetic disorder (about 70% of cases) and epigenetic [4], and the survival rate has increased from 10-15% to 80-90% currently [4].

Exploratory data analysis (EDA) and data preprocessing [5], [6] are necessary precursors to any data analysis task, including machine learning. Data preprocessing validates data by improving its reliability, quality, accuracy and consistency, enabling machine learning algorithms to read, use and interpret the dataset, all of which in turn increases a model's performance and validity. For example, removing outliers or inconsistent samples and making up missing values (i.e. data imputation) increase data quality and reliability, which improve model's accuracy. Removing duplicate samples in a dataset makes the latter consistent. As data processing is necessary and resource demanding, the availability of preprocessed dataset speeds up the overall machine learning pipeline. For this reason, various data preprocessing methods have been proposed for model learning. A survey of some of the methods are found in, e.g., [7].

Some machine learning models, especially those based on similarity measures, are sensitive to the presence of noise in the dataset used to learn them [7], making data cleaning a crucial precursor to model learning. A dataset with too many variables or correlated variables should not be used for model learning as some of the variables are redundant and at best do not provide much benefit to the model [8]. We thus apply on the dataset dimensionality reduction tools, i.e. principal component analysis or PCA and feature selection.

Most of the datasets in real-world applications have missing values [7]. While some machine learning algorithms (e.g. naive Bayes classifier) are insensitive to missing data values, others (e.g. k-NN and artificial neural networks) do not do well amidst missing data values on the dataset used to train them. Thus, mining missing values in datasets and fixing them is an important task in data preparation. A variety of methods is used to handle missing values [9], such as deleting the variables with missing values, filling them with the variable mean, mode or median, or constructing a model and using it to predict the missing values [10], [11].

Application of machine learning in disease classification is increasing in popularity. For example, Mahore et al. used random forest algorithm to classify osteosarcoma dataset into viable, non-viable and non-tumor with accuracy, sensitivity and specificity of 92.4%, 85.44% and 93.38% and 0.95 area under the curve [12]. Non-tumor cell has the proper structure as it has no cancerous element. A viable tumor still has the cancerous element in a cell, while non-viable tumor shows signs of irreversible cell injury [12]. Segmentation and classification of histology tissue in H&E stained tumor images is a non-trivial task owing to noise, intra-class variations, inter-class similarity and crowded context. To circumvent this problem, [13] applied a five-layer convolutional neural network (CNNs) which performs automatic feature extraction to classify osteosarcoma tumor into tumor (viable tumor, necrosis) and non-tumor.

The rest of the paper is structured as follows. Section II discusses related works. We then study data understanding in Sec. III-A, summary statistics in Sec. III-B, data visualization in Sec. III-C, dataset cleaning in Sec. III-D, dataset normalization in Sec. III-E, and PCA and feature selection in Sec. III-F. Section IV validates the efficacy of the data preprocessing by comparing the performance of three logistic regression models using datasets with all original variables, PCA-transformed features and feature selection. Conclusions appear in Sec. V.

II. Related Works

This section reviews some of the recent studies that are important to the task of osteosarcoma cancer prediction and classification using machine learning. In [14], the authors employ a CNN based on the osteosarcoma Whole Slide Images (WSIs) dataset [15] to increase the effectiveness and precision of classifying osteosarcoma tumors into tumor classes (viable tumor, necrosis) and non-tumor. Owing to training restrictions, the original 1024 x 1024 images were cropped down to 128 x 128 patches. When CNN is used, the average classification accuracy for differentiating between tumor classes (VT and necrosis) and NT regions increases dramatically to 92%. The aim of the research work [16] was to improve the classification and prediction of osteosarcoma by utilizing four machine learning algorithms: namely, DT, SVM, KNN, and AdaBoost. The result of their study indicates that all techniques have successfully classified osteosarcoma into viable, necrotic, and non-tumor. The AdaBoost algorithm outperforms the others with an overall accuracy of 91.70%.

Anisuzzaman et al. [17] examines how CNNs can reliably predict osteosarcoma malignancy. This study exploits the power in transfer learning with updated VGG19 and Inception V3 models on a dataset of 40 whole slide pictures of osteosarcoma tumors in order to enhance the predictive accuracy by 2% over prior findings. The total accuracy for the VGG19 and InceptionV3 models, respectively, was 93.91% and 78.26% on multiclass classification.

In [18], the challenges of evaluating the treatment response of osteosarcoma is studied. It proposes a solution that employs digital image analysis to automate this process, enhancing accuracy and efficiency. The method combines pixel-based and object-based techniques to segment tumor and non-tumor regions in high-resolution WSIs of osteosarcoma. The approach involves tumor property analysis like nuclei clustering, density, and circularity to distinguish viable and non-viable regions. Initially, K-Means clustering with color normalization was used for tumor isolation. A Flood-Fill algorithm groups similar pixels into cellular objects, providing cluster data for further analysis. The results demonstrate around 90% accuracy.

Gawade et al. [19] introduced a model designed for the early-stage detection of bone cancer. Their work emphasized the urgency of timely cancer identification and management, highlighting the necessity of automation. They introduced an automatic approach rooted in Deep Learning, employing supervised techniques. The conceptual framework incorporates four algorithms: Visual Geometry Group 16 (VGG16), Visual Geometry Group 19 (VGG19), Dense Convolutional Network 201 (DenseNet201), and Residual Network 101 (ResNet101). Notably, the ResNet101exhibited exceptional performance, achieving an impressive accuracy of 90.36

Reference [20] addresses the methodology for segmenting the tumor and parosteal sarcoma tissues, which comprises the gathering of imaging data, format unification, selection of interested regions, selection of seed points, information fusion of multimodality MRI. The interesting tissues, which are dispersed throughout unconnected regions, are segmented using the vectorial fuzzy-connection approach. Additionally, they discussed how the algorithm has been improved to take less time to segment two objects at once. Finally, the study shows how this technology has been used in osteosarcoma segmentation and 3D reconstruction medical image analysis systems, which have been implemented in several hospitals.

Using a dynamic clustering algorithm known as DCHS, Mandava et al. [21] devised an automatic segmentation method for osteosarcoma in MRI images. The approach uses Fuzzy C-means (FCM) and Harmony Search (HS) to effectively segment the Osteosarcoma MRI images. New to HS, the ”empty operator” makes it easier to choose empty decision variables in the harmony memory vector. DCHS incorporates FCM to improve segmentation results. Haralick texture characteristics and pixel intensity values have been combined with multi-spectral data from STIR and T2-weighted MRI sequences for segmentation. The results indicated excellent results when they were compared to manually defined data for four patients, reaching an average Dice measurement of 0.72.

III. Data Preprocessing

The steps used in pre-processing our dataset are elaborated in this section.

A. Data Understanding

This paper is based on the dataset [15], which labels osteosarcoma cancer into the four groups: non-tumor, non-viable tumor, viable, and viable non-viable. This is a real-valued multivariate data with 1144 samples and 69 variables (aka features) shared across all the four classes. The basic features of the dataset is summarized into Table I with their encoding. We observe class imbalance issue in the dataset with ’non-tumor’ being the majority class, which may result in slow convergence of the weight updating process for the minority

TABLE I
STATISTICS OF THE ORIGINAL OSTEOSARCOMA DATASET.

Class	Number of Data Samples in Class	Code
Non-Tumor	536	0
Non-Viable-Tumor	263	1
Viable	292	2
Viable: non-viable	53	3

classes. However, there is only one minority class, which is the 'viable: non-viable' class. According to [22], this problem is not remarkable if there is only a single minority class. The class imbalance problem is studied in, e.g., [23], [24]

The dataset contains two types of features: statistical (e.g. variance, mean and correlation) and pathological features (e.g. nuclei count, texture and number of cells). The variables in a dataset can be grouped into three: those necessary to predict class labels and cannot be replaced, unnecessary ones which have no influence on the class prediction, and the redundant ones whose role can be assumed by other variables in the dataset [25]. Variables of a dataset should be correlated with the target variable but not be correlated among themselves. The first five samples in the dataset are shown in Table II. Six of the variables in the dataset, namely *Unnamed: 0*, *image.name*, `X.x`, `X.1`, `X.y` and `ImageNumber`, are clearly unnecessary for model building. So, we discard them as part of the data cleaning process, leaving 63 features.

Figure 1 summarizes the basic statistics of the variables in the dataset. We observe that 62 of the 63 attributes hold numeric data (i.e. either float or integer), while the remaining one is nominal as it holds the class label. Also, each of the attributes/features ”area” and ”circularity” has one NULL or missing value.

B. Statistical Summary of the Dataset

The statistics of the original dataset after deleting the six irrelevant variables is shown in Table III. We observe that each of the columns ”area” and ”circularity” contain a single ”NULL” or ”missing” value as the count for each is 1143 instead of 1144. A percentile gives the percentage of the data values smaller than it. For example, 25% of all values in the columns ”Blue.count” are smaller than 29569.5, while 50% (i.e. median value) are smaller than 50417.5.

C. Visualization of Dataset

The five-number summary of the dataset in Fig. 2 graphically shows the locality, spread and any skewness of the two variables with missing values through their quartiles. We can observe outliers in both variables. The data samples in the 'circularity' variable has a larger interquartile range and thus are more spread about the mean than those of the 'area' variable. The 'circularity' variable also shows some right-skewness, which is validated by the histograms in Fig. 3. The 'area' variable, however, is geared towards normal distribution, which is observable from both the boxplots and the histograms. Both histograms in Fig. 3 have multiple peaks. Thus, the two variables are multi-modal datasets.

Heatmap is used to study the correlation between the variables of a dataset. For example, the heatmap in Fig. 4 reveals a strong positive correlation (i.e. Pearson correlation coefficient $r \approx 0.96$ and p-value=0) between the variables 'Texture_Contrast_3_135' and 'Texture_DifferenceVariance_3_135', but near zero correlation (i.e. r=0.0996 and p-value ≈ 0.0007) between the variables 'Texture_Entropy_3_135', 'Texture_Variance_3_135'. A p-value less than 0.05 indicates the voidness of the null hypothesis, confirming a correlation. These dependencies between the four variables are verified by the scatterplots in Fig. 5. This means that we can keep only 'Texture_Contrast_3_135' or 'Texture_DifferenceVariance_3_135' to reduce the dimensionality of the dataset. This will be confirmed using principal component analysis (PCA) in Sect. III-F.

D. Dataset Cleaning

As part of the data preprocessing, we must correct erroneous values (or noise) arising from human error or malfunctioning data collection instrument. The noise includes duplicates, missing values, incorrect values (i.e. inliers and outliers) and mislabelled samples in the data. We found that the dataset contained no duplicates, NaN values and categorical features. However, we found in Sections III-A and III-B that the two variables 'circularity' and 'area' had missing values. How missing values in a dataset are handled depends on the absence or presence of outliers, as well as their quantity relative to the size of the dataset.

Outliers are data values that are dissimilar from other values in a given dataset [26]. Outliers can skew trends and seriously impair the accuracy of models, and lead misleading predictive models. The box and whiskers plots in Fig. 2 reveal outliers in both variables. The outliers in 'area' and 'circularity' amount to 34 (i.e. $< 3\%$) and 21 (i.e. $< 2\%$) outliers, respectively. Therefore, we would replace the missing values with the corresponding median values if we kept the outliers for the modeling. However, as less than 3% of the data samples are outliers, we deleted them from the dataset, allowing the replacement of the missing values with their corresponding mean values. We could have also dropped the samples with missing values as they are only two. We employed a distance-based, unsupervised outlier mining method to fish out the outliers in the dataset. Namely, a data value, say x_k, is considered to be an outlier if

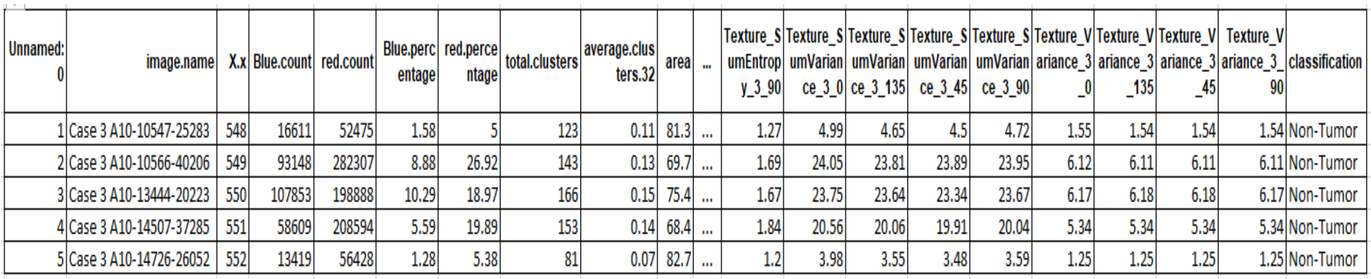
$$x_k < Q1 - 1.5 \times IQR \ \ \texttt{or} \ \ x_k > Q3 + 1.5 \times IQR \tag{1}$$

where the interquartile range (IQR) is the difference between the upper quartile (Q3) and the lower quartile (Q1), as shown in Fig. 2. Upon deleting the outliers in the dataset, we imputed the missing values in each variable by the mean of its available values.

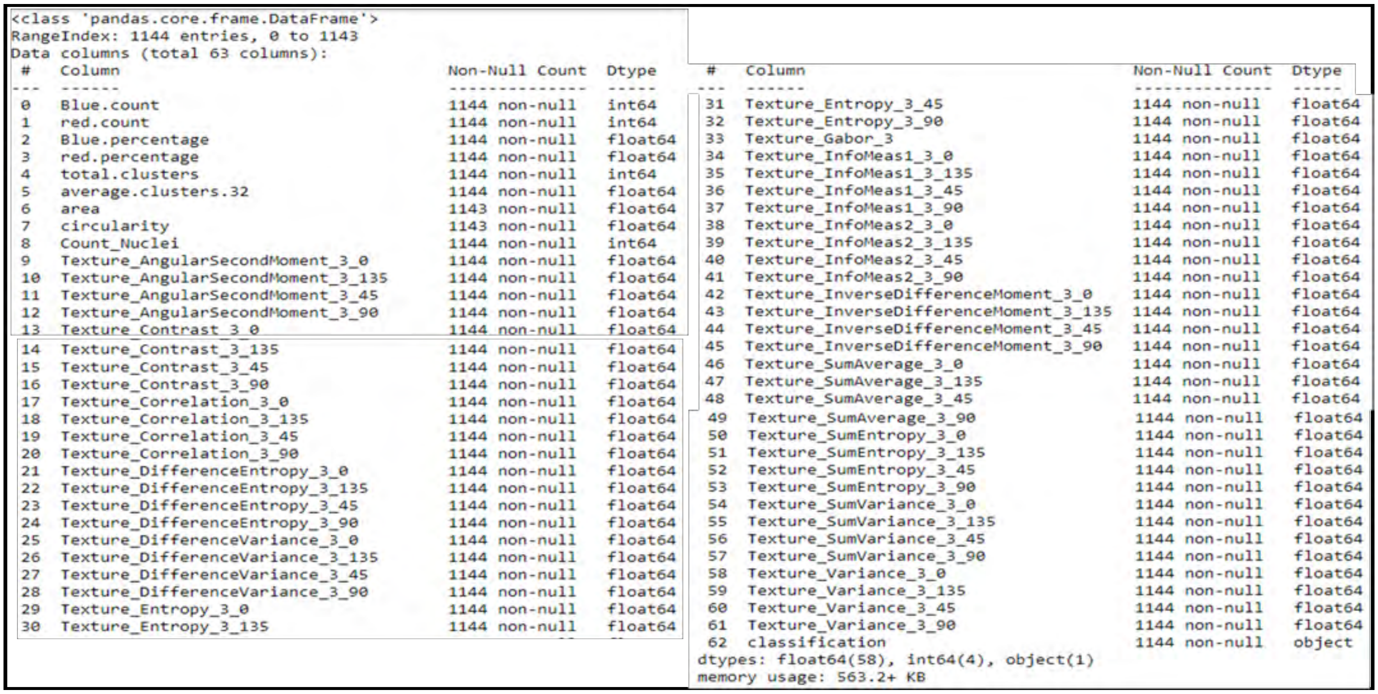
E. Dataset Normalization

We can observe from the summary statistics in Table III of the dataset under study that the values of the variables have very

TABLE II
THE FIRST FIVE SAMPLES IN THE ORIGINAL OSTEOSARCOMA DATASET.



Unnamed: 0	image.name	X.x	Blue.count	red.count	Blue.percentage	red.percentage	total.clusters	average.clusters.32	area	...	Texture_SumEntropy_3_90	Texture_SumVariance_3_0	Texture_SumVariance_3_135	Texture_SumVariance_3_45	Texture_SumVariance_3_90	Texture_Variance_3_0	Texture_Variance_3_135	Texture_Variance_3_45	Texture_Variance_3_90	classification
1	Case 3 A10-10547-25283	548	16611	52475	1.58	5	123	0.11	81.3	...	1.27	4.99	4.65	4.5	4.72	1.55	1.54	1.54	1.54	Non-Tumor
2	Case 3 A10-10566-40206	549	93148	282307	8.88	26.92	143	0.13	69.7	...	1.69	24.05	23.81	23.89	23.95	6.12	6.11	6.11	6.11	Non-Tumor
3	Case 3 A10-13444-20223	550	107853	198888	10.29	18.97	166	0.15	75.4	...	1.67	23.75	23.64	23.34	23.67	6.17	6.18	6.18	6.17	Non-Tumor
4	Case 3 A10-14507-37285	551	58609	208594	5.59	19.89	153	0.14	68.4	...	1.84	20.56	20.06	19.91	20.04	5.34	5.34	5.34	5.34	Non-Tumor
5	Case 3 A10-14726-26052	552	13419	56428	1.28	5.38	81	0.07	82.7	...	1.2	3.98	3.55	3.48	3.59	1.25	1.25	1.25	1.25	Non-Tumor



```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1144 entries, 0 to 1143
Data columns (total 63 columns):
 #   Column                              Non-Null Count  Dtype       #   Column                              Non-Null Count  Dtype
---  ------                              --------------  -----      ---  ------                              --------------  -----
 0   Blue.count                          1144 non-null   int64       31  Texture_Entropy_3_45                1144 non-null   float64
 1   red.count                           1144 non-null   int64       32  Texture_Entropy_3_90                1144 non-null   float64
 2   Blue.percentage                     1144 non-null   float64     33  Texture_Gabor_3                     1144 non-null   float64
 3   red.percentage                      1144 non-null   float64     34  Texture_InfoMeas1_3_0               1144 non-null   float64
 4   total.clusters                      1144 non-null   int64       35  Texture_InfoMeas1_3_135             1144 non-null   float64
 5   average.clusters.32                 1144 non-null   float64     36  Texture_InfoMeas1_3_45              1144 non-null   float64
 6   area                                1143 non-null   float64     37  Texture_InfoMeas1_3_90              1144 non-null   float64
 7   circularity                         1143 non-null   float64     38  Texture_InfoMeas2_3_0               1144 non-null   float64
 8   Count_Nuclei                        1144 non-null   int64       39  Texture_InfoMeas2_3_135             1144 non-null   float64
 9   Texture_AngularSecondMoment_3_0     1144 non-null   float64     40  Texture_InfoMeas2_3_45              1144 non-null   float64
 10  Texture_AngularSecondMoment_3_135   1144 non-null   float64     41  Texture_InfoMeas2_3_90              1144 non-null   float64
 11  Texture_AngularSecondMoment_3_45    1144 non-null   float64     42  Texture_InverseDifferenceMoment_3_0 1144 non-null   float64
 12  Texture_AngularSecondMoment_3_90    1144 non-null   float64     43  Texture_InverseDifferenceMoment_3_135 1144 non-null float64
 13  Texture_Contrast_3_0                1144 non-null   float64     44  Texture_InverseDifferenceMoment_3_45 1144 non-null  float64
 14  Texture_Contrast_3_135              1144 non-null   float64     45  Texture_InverseDifferenceMoment_3_90 1144 non-null  float64
 15  Texture_Contrast_3_45               1144 non-null   float64     46  Texture_SumAverage_3_0              1144 non-null   float64
 16  Texture_Contrast_3_90               1144 non-null   float64     47  Texture_SumAverage_3_135            1144 non-null   float64
 17  Texture_Correlation_3_0             1144 non-null   float64     48  Texture_SumAverage_3_45             1144 non-null   float64
 18  Texture_Correlation_3_135           1144 non-null   float64     49  Texture_SumAverage_3_90             1144 non-null   float64
 19  Texture_Correlation_3_45            1144 non-null   float64     50  Texture_SumEntropy_3_0              1144 non-null   float64
 20  Texture_Correlation_3_90            1144 non-null   float64     51  Texture_SumEntropy_3_135            1144 non-null   float64
 21  Texture_DifferenceEntropy_3_0       1144 non-null   float64     52  Texture_SumEntropy_3_45             1144 non-null   float64
 22  Texture_DifferenceEntropy_3_135     1144 non-null   float64     53  Texture_SumEntropy_3_90             1144 non-null   float64
 23  Texture_DifferenceEntropy_3_45      1144 non-null   float64     54  Texture_SumVariance_3_0             1144 non-null   float64
 24  Texture_DifferenceEntropy_3_90      1144 non-null   float64     55  Texture_SumVariance_3_135           1144 non-null   float64
 25  Texture_DifferenceVariance_3_0      1144 non-null   float64     56  Texture_SumVariance_3_45            1144 non-null   float64
 26  Texture_DifferenceVariance_3_135    1144 non-null   float64     57  Texture_SumVariance_3_90            1144 non-null   float64
 27  Texture_DifferenceVariance_3_45     1144 non-null   float64     58  Texture_Variance_3_0                1144 non-null   float64
 28  Texture_DifferenceVariance_3_90     1144 non-null   float64     59  Texture_Variance_3_135              1144 non-null   float64
 29  Texture_Entropy_3_0                 1144 non-null   float64     60  Texture_Variance_3_45               1144 non-null   float64
 30  Texture_Entropy_3_135               1144 non-null   float64     61  Texture_Variance_3_90               1144 non-null   float64
                                                                     62  classification                      1144 non-null   object
                                                                    dtypes: float64(58), int64(4), object(1)
                                                                    memory usage: 563.2+ KB
```

Fig. 1. Shapes and types of the osteosarcoma dataset upon deleting the two unnecessary features.

large dynamic ranges. For example, the range of values are 3007 to 322157 for Blue.count, 7398 to 513281 for red.count and 0.2766 to 38.1151 for Texture_SumVariance_3_135. Many machine learning algorithms, including $k-$NN, neural networks and Support Vector Machines (SVMs), exhibit poor performance if learned with dataset whose variables have such large dynamic range, requiring dataset normalization or scaling. Also, many machine learning functions, including RBF kernel of SVMs and the L1/L2 regularizers of linear models, presume that the underlying distribution of the dataset used to train them has zero mean and unit variance, and even normally distributed. Further, we observe from the histograms in Fig. 3 that at least some of the variables of the dataset being analyzed are approximately normally distributed. For these reasons, we applied feature-wise z-score normalization on the dataset, which transforms each variable in the dataset into zero-mean and unit variance variable. So, each data sample x^k_{train} in the training set was transformed into

$$\tilde{x}^k_{train} = (x^k_{train} - m_{train})/\sigma_{train} \tag{2}$$

where m_{train} and σ_{train} are the mean and standard deviation (stddev) of the training samples, respectively. After normalization, the train set has the mean of -8.64×10^{-19} and unit stddev. We then scaled the test and validation sets as

$$\tilde{x}^k_{test} = (x^k_{test} - m_{train})/\sigma_{train} \tag{3}$$

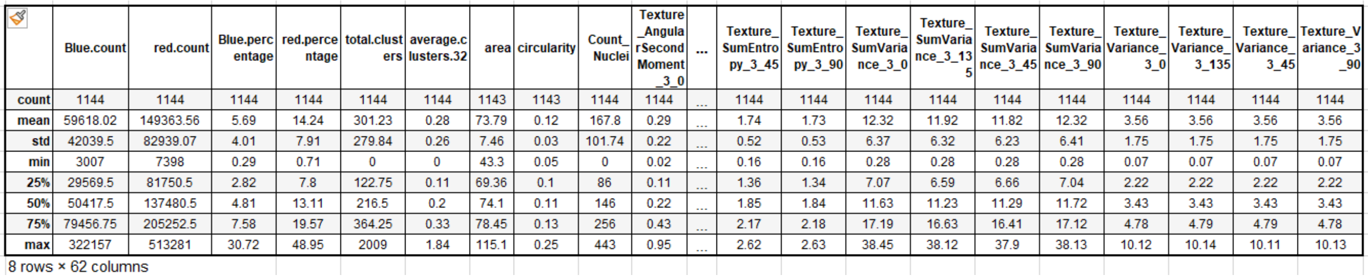
and

$$\tilde{x}^k_{val} = (x^k_{val} - m_{train})/\sigma_{train} \tag{4}$$

F. PCA and Feature Selection

Both principal component analysis (PCA) and feature selection are different techniques used to reduce the redundant and unnecessary variables in a dataset.

TABLE III
SUMMARY STATISTICS OF DATASET.



	Blue.count	red.count	Blue.percentage	red.percentage	total.clusters	average.clusters.32	area	circularity	Count_Nuclei	Texture_AngularSecondMoment_3_0	...	Texture_SumEntropy_3_45	Texture_SumEntropy_3_90	Texture_SumVariance_3_0	Texture_SumVariance_3_135	Texture_SumVariance_3_45	Texture_SumVariance_3_90	Texture_Variance_3_0	Texture_Variance_3_135	Texture_Variance_3_45	Texture_Variance_3_90
count	1144	1144	1144	1144	1144	1144	1143	1143	1144	1144	...	1144	1144	1144	1144	1144	1144	1144	1144	1144	1144
mean	59618.02	149363.56	5.69	14.24	301.23	0.28	73.79	0.12	167.8	0.29	...	1.74	1.73	12.32	11.92	11.82	12.32	3.56	3.56	3.56	3.56
std	42039.5	82939.07	4.01	7.91	279.84	0.26	7.46	0.03	101.74	0.22	...	0.52	0.53	6.37	6.32	6.23	6.41	1.75	1.75	1.75	1.75
min	3007	7398	0.29	0.71	0	0	43.3	0.05	0	0.02	...	0.16	0.16	0.28	0.28	0.28	0.28	0.07	0.07	0.07	0.07
25%	29569.5	81750.5	2.82	7.8	122.75	0.11	69.36	0.1	86	0.11	...	1.36	1.34	7.07	6.59	6.66	7.04	2.22	2.22	2.22	2.22
50%	50417.5	137480.5	4.81	13.11	216.5	0.2	74.1	0.11	146	0.22	...	1.85	1.84	11.63	11.23	11.29	11.72	3.43	3.43	3.43	3.43
75%	79456.75	205252.5	7.58	19.57	364.25	0.33	78.45	0.13	256	0.43	...	2.17	2.18	17.19	16.63	16.41	17.12	4.78	4.79	4.79	4.78
max	322157	513281	30.72	48.95	2009	1.84	115.1	0.25	443	0.95	...	2.62	2.63	38.45	38.12	37.9	38.13	10.12	10.14	10.11	10.13

8 rows × 62 columns

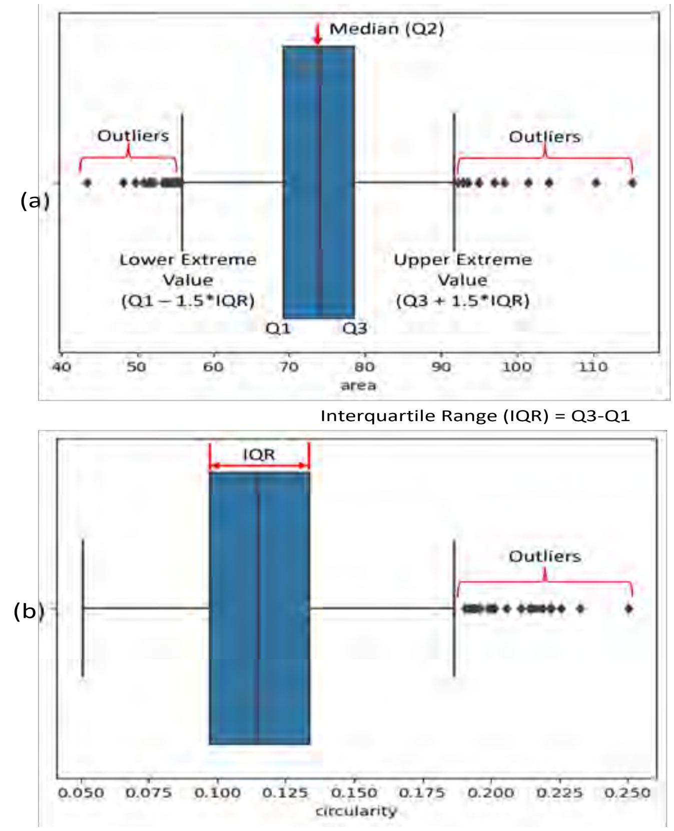


Fig. 2. Box and whisker plots of the two variables with missing values.

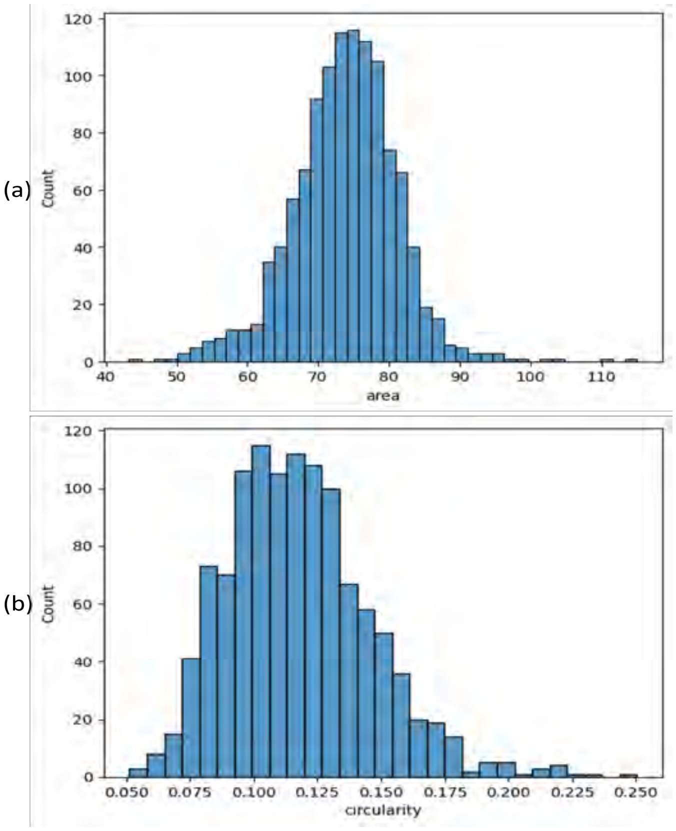


Fig. 3. Histograms of the two variables with missing values.

In PCA analysis, we replace the variables in a given dataset by a new set of variables called principal components (PCs), which are derived as linear combinations of the original variables. The main aim in PCA is to reduce the number of variables which adequately describe the dataset under study. Figure 6 uses bar chart to represent the variance explained by individual PCs, and a step plot to represent the cumulative variance explained by multiple PCs. In PCA, the number of PCs in a dataset equals its number of variables. However, it is clear from Fig. 6 that we need less than 10 PCs instead of the 69 variables in the original dataset to adequately describe the dataset. This is the reason why PCA delivers data dimensionality reduction, which in turn, reduces computational resources, such as data storage, memory requirements, model training time, and predictive time for instance-based learners. Large data also increases model complexity which reduces a model's explainability and interpretability of results. We did not perform instance selection [27] on the dataset as the latter has only 1144 examples.

PCA seeks to reduce the dimensionality in a dataset by exploring any linear dependencies between the variables of a dataset without considering the target variable, the reason why it is an unsupervised technique. Feature selection [28], unlike PCA, does not transform the variables and takes the target variable into consideration, by ranking the input variables in regard to how useful each is to predict the target value.

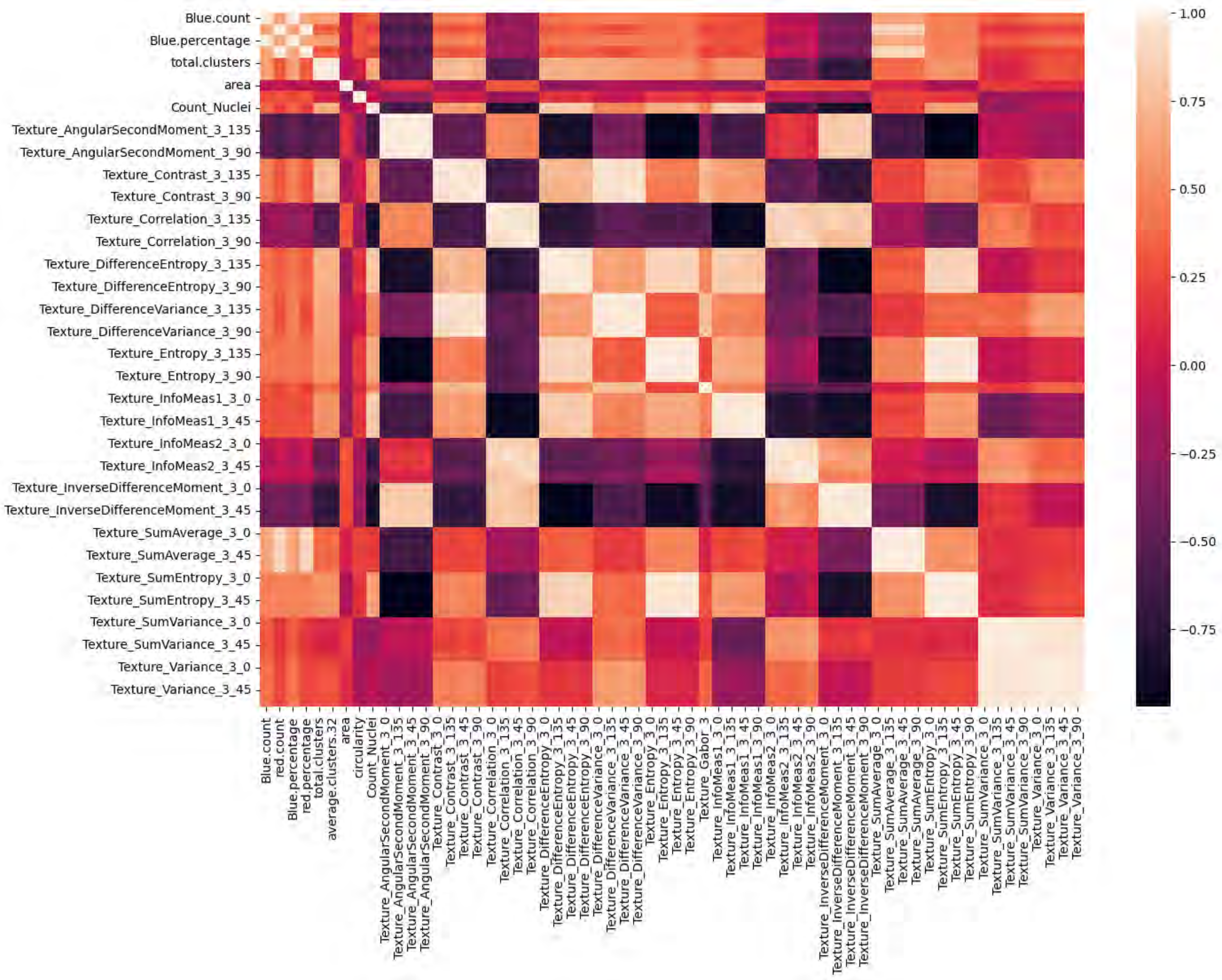


Fig. 4. Heatmap of the original dataset.

Irrelevant variables in a dataset degrade the performance of many machine learning algorithms (e.g. kNN) and neural networks and SVMs become inefficient or even impractical. Moreover, some algorithms (e.g. Bayesian classifiers) exhibit poor performance in the presence of redundant variables [6]. By reducing redundant variables we reduce the chance of model overfitting and high bias. Reducing overfitting, in turn, improves model generalization. Fewer variables also are easier to interpret, and reduces model training time.

IV. Data Preprocessing Efficacy

To validate the power in the data preprocessing techniques studied in this paper, we built multiclass logistic regression model using the three types of datasets: with raw 67 variables, PCA and feature selection with only 16 of the 69 variables. The model predictive performance in Fig. 7 reveals the differences. Model with all variables in the dataset delivers only 19.27%, while datasets with PCA and feature selection delivers 65.14% and 80.28% predictive accuracies, respectively.

V. Conclusion and Future Extension

Data preprocessing (incl. cleaning, transformation, and integration) is a very crucial and unreplaceable component in data analysis, and its benefits include:

- it improves accuracy and reliability: well pre-processed data is void of missing or inconsistent data values resulting from human or computer error, resulting in improved accuracy and quality of a dataset, making it more reliable.
- it makes data consistent: when collecting data, it is possible to have data duplicates, and discarding them during preprocessing can ensure the data values for analysis are consistent, which helps produce more accurate models.
- it increases the data's algorithm readability: preprocessing enhances the data's quality and makes it easier for

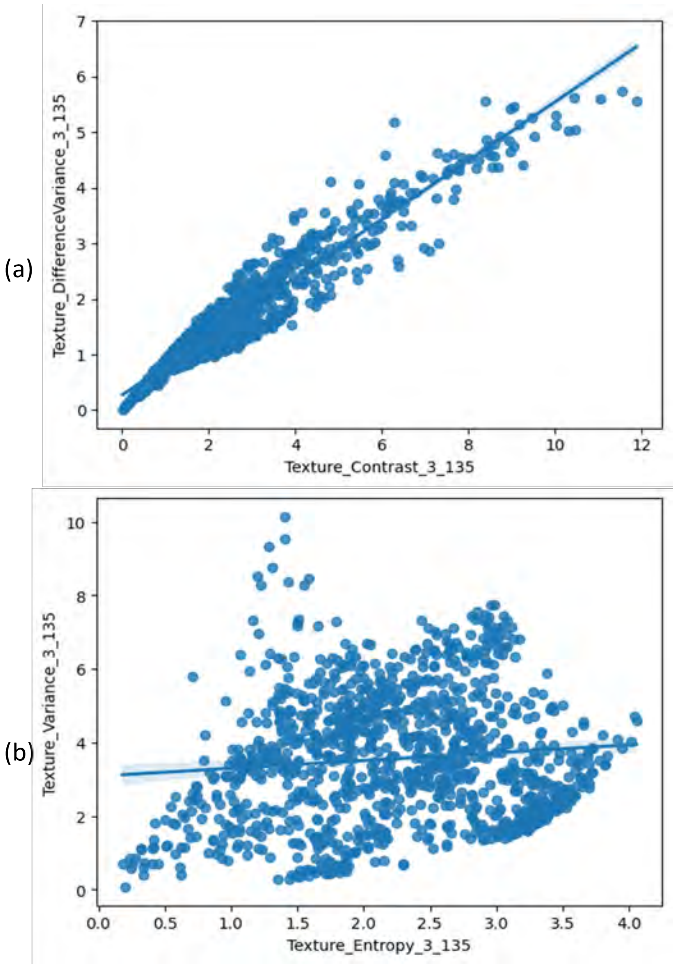
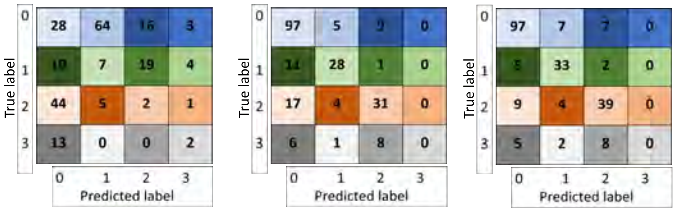


Fig. 5. Verifying correlations between variables in the heatmap.

Fig. 6. The variance in the dataset explained by the principal components.



(a) Logistic regression without PCA nor feature selection (accuracy = 17.9%)

(b) Logistic regression with PCA (accuracy = 71.6%)

(c) Logistic regression with feature selection (accuracy = 77.5%)

Fig. 7. Confusion matrices of logistic regression with datasets with raw variables, PCA and feature selection.

machine learning algorithms to read, use, and interpret it.

- available preprocessed dataset speeds up data analysis and model building for its adopters.

This paper has analyzed and prepared osteosarcoma dataset for 4-class model building. We have also demonstrated the achievable model performance via the appropriate data preprocessing, and that dataset with feature selection delivers the best results. We will use the prepared dataset to build powerful machine learning models which can classify osteosarchoma types of patients using histopathological dataset.

References

[1] B. R. Eaton, R. Schwarz, R. Vatner, B. Yeh, L. Claude, D. J. Indelicato, and N. Laack, "Osteosarcoma," *Pediatric Blood Cancer*, vol. 68, no. Suppl. 2, pp. 1–7, 2021. [Online]. Available: https://doi.org/10.1002/pbc.28352

[2] J. H. Schwab, C. R. Antonescu, E. A. Athanasian, P. J. Boland, J. H. Healey, and C. D. Morris, "A comparison of intramedullary and juxtacortical low-grade osteogenic sarcoma," *Clinical Orthopaedics and Related Research*, vol. 466, no. 6, pp. 1318–1322, 2008.

[3] K. Harper, P. Sathiadoss, A. Saifuddin, and A. Sheikh, "A review of imaging of surface sarcomas of bone," *Skeletal radiology*, vol. 50, no. 1, pp. 9–28, 2021.

[4] "Osteosarcoma," *Annals of Oncology*, vol. 21, pp. vii320–vii325, 2010.

[5] D. Pyle, *Data Preparation for Data Mining*. Burlington, MA, United States: Morgan Kaufmann, 1999.

[6] S. García, J. Luengo, and F. Herrera, *Data Preprocessing in Data Mining*. Berlin, Germany: Springer, 2015.

[7] S.-A. N. Alexandropoulos, S. B. Kotsiantis, and M. N. Vrahatis, "Data preprocessing in predictive data mining," *The Knowledge Engineering Review*, vol. 34, no. 1, pp. 1–33, 2019.

[8] I. Guyon and A. Elisseeff, "An introduction to variable and feature selection," *J. of Mach. Learn. Research*, vol. 3, no. 3, pp. 1157–1182, 2003.

[9] F. Cismondi, A. S. Fialho, S. M. Vieira, S. R. Reti, J. M. Sousa, and S. N. Finkelstein, "Missing data in medical databases: Impute, delete or classify?" *Artificial Intelligence in Medicine*, vol. 58, no. 1, pp. 63–72, 2013.

[10] F. Honghai, C. Guoshun, Y. Cheng, Y. Bingru, and C. Yumei, "A svm regression based approach to filling in missing values," in *Int. Conf. on Knowledge-Based and Intelligent Info. and Eng. Systems*. Berlin, Heidelberg: Springer-Verlag LNCS, 2005.

[11] Esther-Lydia Silva-Ramírez and Rafael Pino-Mejías and Manuel López-Coello and María-Dolores Cubiles-de-la-Vega, "Missing value imputation on missing completely at random data using multilayer perceptrons," *Neural Networks*, vol. 24, no. 1, pp. 121–129, 2011.

[12] M. Sanket, B. Kalyani, and R. Shashikant, "Machine learning approach to classify and predict different osteosarcoma types," in *IEEE 2021 8th Int. Conf. on Sig. Proc. and Integrated Networks (SPIN)*, 2021, pp. 641–645.

[13] M. Rashika, O. Daescu, L. Patrick, R. Dinesh, and S. Anita, "Histopathological diagnosis for viable and non-viable tumor prediction for osteosarcoma using convolutional neural network," in *Bioinformatics Research and Applications*. Cham: Springer Int. Publishing, 2017, pp. 12–23.

[14] R. Mishra, O. Daescu, P. Leavey, D. Rakheja, and A. Sengupta, "Convolutional neural network for histopathological analysis of osteosarcoma," *J. Comput Biol.*, vol. 25, p. 102931, Mar. 2018.

[15] P. Leavey, A. Sengupta, D. Rakheja, O. Daescu, H. B. Arunachalam, and R. Mishra. (2019) Osteosarcoma data from ut southwestern/ut dallas for viable and necrotic tumor assessment [data set]. Accessed: 2023-06-15. [Online]. Available: https://wiki.cancerimagingarchive.net/pages/viewpage.action?pageId=52756935

[16] S. Mahore, B. Kalyani, and R. Shashikant, "Comparative analysis of machine learning algorithm for classification of different osteosarcoma types," in *2021 12th International Conference on Computing Communication and Networking Technologies (ICCCNT)*, 2021, pp. 1–5.

[17] D. Anisuzzaman, H. Barzekar, L. Tong, J. Luo, and Z. Yu, "A deep learning study on osteosarcoma detection from histological images," *Biomedical Signal Processing and Control*, vol. 69, p. 102931, 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1746809421005280

[18] H. Arunachalam, R. M. R, B. Armaselu, O. Daescu, M. Martinez, P. Leavey, D. Rakheja, K. Cederberg, A. Sengupta, and M. Ni'suilleabhain, "Computer aided image segmentation and classification for viable and non-viable tumor identification in osteosarcoma," in *Pacific Symposium on Biocomputing*, 2017, p. 195–206.

[19] S. Gawade, A. Bhansali, K. Patil, and D. Shaikh, "Application of the convolutional neural networks and supervised deep-learning methods for osteosarcoma bone cancer detection," *Healthcare Analytics*, vol. 3, p. 100153, 2023. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2772442523000205

[20] M. Jing, L. Minglu, and Z. Yongqiang, "Segmentation of multimodality osteosarcoma mri with vectorial fuzzy-connectedness theory," in *Fuzzy Systems and Knowledge Discovery*, L. Wang and Y. Jin, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2005, pp. 1027–1030.

[21] M. Rajeswari, A. O. Moh'd, W. B. Chin, R. Dhanesh, A. M. Ezane, and S. I. Lutfi, "Osteosarcoma segmentation in mri using dynamic harmony search based clustering," in *2010 International Conference of Soft Computing and Pattern Recognition*, 2010, pp. 423–429.

[22] K. Ghosh, C. Bellinger, R. Corizzo, B. Krawczyk, and N. Japkowicz, "The class imbalance problem in deep learning," *Springer Mach. Learn.*, Dec. 2022. [Online]. Available: https://doi.org/10.1007/s10994-022-06268-8

[23] G. E. A. P. A. Batista, R. C. Prati, and M. C. Monard, "A study of the behavior of several methods for balancing machine learning training data," *ACM SIGKDD Explorations Newsletter*, vol. 6, no. 1, p. 20–29, Jun. 2004.

[24] A. Estabrooks, T. Jo, and N. Japkowicz, "A multiple resampling method for learning from imbalanced data sets," *Computational Intelligence*, vol. 20, no. 1, pp. 18–36, 2004.

[25] J. Hua, Z. Xiong, J. Lowey, E. Suh, and E. R. Dougherty, "Optimal number of features as a function of sample size for various classification rules," *Bioinformatics*, vol. 21, no. 8, pp. 1509–1515, 2005.

[26] H. Huang, J. Lin, C. Chen, and M. Fan, "Review of outlier detection," *Appl. Research of Computers*, vol. 8, pp. 2006–2008, 2006.

[27] Y. Fu, X. Zhu, and B. Li, "A survey on instance selection for active learning," *Knowledge and information systems*, vol. 35, no. 2, pp. 249–283, 2013.

[28] S. Piramuthu, "Evaluating feature selection methods for learning in data mining applications," *European Journal of Operational Research*, vol. 156, no. 2, pp. 483–494, 2004.