IBM-322- ANALYTICS FOR MANAGERIAL DECISION MAKING

FINAL PROJECT

Leukemia Classification using Machine Learning

Candidates:

Ishan Garg(22125012), Aakash Kumar Singh(22125001), Arahan Kalsoor(22125006)

Advisor:

Prof. Sumit Kumar Yadav

Abstract

This project proposes a method for the classification of leukemia based on images of white blood cells(WBC) using modern machine learning techniques. A dataset comprising of 3262 images is used to classify images into 4 categories benigin, early pre-B, pre-B and pro-B, the latter three being subtypes of malignant leukemia. The proposed technique consists of three steps feature extraction from the images using a pretrained model, feature analysis using correlation analysis, LDA etc, feature selection using PCA and SVD and classification on the selected features using traditional ML classifiers. All the different models are evaluated on the test dataset to find the best model for the classification purpose.

1 Introduction

1.1 Problem

When a cluster of cells undergoes unchecked growth in the body, it is termed as cancer or cancerous growth. The most common forms of cancer are skin, breast, lymphoma and leukemia. According to WHO lung cancer accounts for around 10 million deaths per year while skin and breast cancer account for around a million and 0.5 million fatalities per year respectively. From this it can be analyzed that cancer causes a high number of deaths per year around the globe. Leukemia, which is caused by abnormal growth of white blood cells and affects bone marrow, has high mortality rate. In India around 15000 cases of leukemia are found every year.. Thus, there is an urgent need to identify leukemia in its early forms using blood samples with a high accuracy to lower leukemia-caused mortality.

Hematologists examine, using microscopes, blood samples of a patient to diagnose acute lymphocytic leukemia (short form - ALL). The extent of correctness of these tests depends on how good the pathologist is and there are other compromising factors like the microscope, etc. This leads to a need of Ml diagnostic models which have proved to be efficient in other medical domains as well.

1.2 Dataset

The dataset has been taken from Kaggle(Dataset). It includes 3262 images of blood samples taken from bone marrow laboratory of Taleqani Hospital (Tehran, Iran) from 89 patients. The dataset consists of 2 parts in which the first part(25 images) are the healthy patients and second(64 images) consists of the rest which are suspected to have leukemia. The dataset has two classes - malignant and benign categories and further the malignant class is subdivided in three classes as early pre-B, pre-B and pro-B. Image size is 224x224x3.

Type	**Subtype**	**No. of samples**	**No. patients**
Benign	Hematogones	504	25
Malignant	Early pre-B ALL	985	20
	Pre-B ALL	963	21
	Pro-B ALL	804	23
Total		**3256**	**89**

Table 1: Summary of Samples and Patients by Subtype.

2 Methodology

Figure 1 shows our proposed pipeline for the classification task. The first step is feature extraction from the given dataset using a pretrained model. Then analysis of the features is done using PCA, LDA, t-SNE etc. This is followed by feature selection which is done using either Principal Component Analysis or Singular Value Decomposition (SVD). Then traditional ML classifiers like Logistic Regression, Support Vector Machines (SVM) and Random Forest Classifier are trained to classify the image in four classes. We will look into these steps in detail in the next sections.

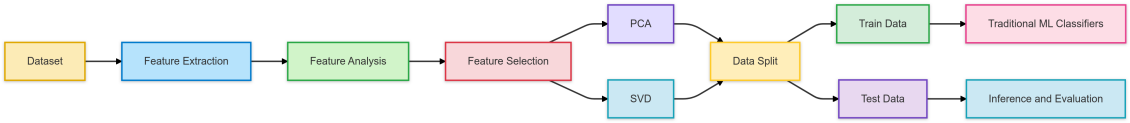


Figure 1: Proposed System Pipeline

2.1 Feature Extraction

In our approach we have used Microsoft's BiomedClip as our pretrained model for generating embeddings/features for the blood images. It converts a 224x224x3 image into a feature vector of size 512. This model was trained on an image-caption dataset taken from research articles on PubMed Central. The image encoder here has a vision transformer architecture. Since our dataset contains blood images this model becomes a good choice for extracting features out of the images. Other pretrained models can also be used to extract features from the images in place of BiomedClip.

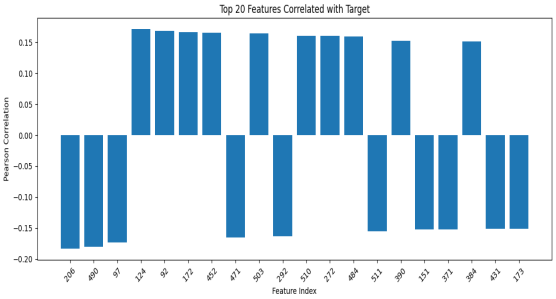
2.2 Feature Analysis

After extracting features out of our original dataset we perform some EDA or analysis on the feature embeddings to understand them better. The first technique is applied for analysis is t-SNE(t-Distributed Stochastic Neighbor Embedding) which is a data visualization technique for higher dimensional data in lower dimensions. In fig.2 (a) we have used t-SNE for visualization in 2-dimensions. The blue and pink clusters are nearly separated from the rest of the dataset. Overlapping of red and blue clusters could mean that they have some common features.

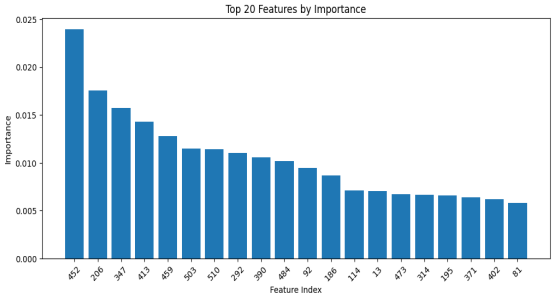
Fig.2(b) shows the correlation of top 20 features with the highest correlation with the output classes. In terms of absolute value they all are nearly equal indicating that a lot of features have moderate but meaningful relation with the class.

Fig.2(c) shows a feature importance graph generated using a Random Forest Classifier. The feature importance shows a decreasing trend indicating a hierarchy of features in terms of important with latter features being less important. Moreover it is visible that

(a) t-SNE Visualization



(b) Correlation Analysis



(c) Feature Importance

Figure 2: Feature Analysis

some features are common between feature importance and correlation graphs indicating that they have a meaningful relationship with our target.

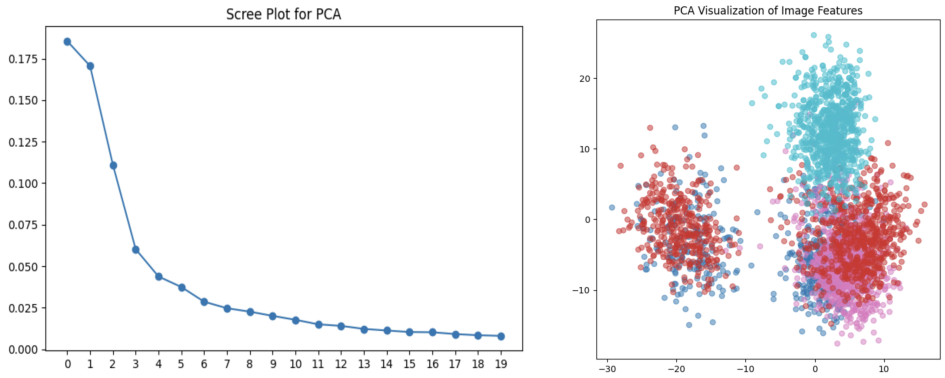
2.2.1 Principal Component Analysis

Principal Component Analysis (PCA) is a dimensionality reduction technique commonly used in data analysis and machine learning to identify the most significant vectors in a dataset in which the data can be transformed to retain the maximum variance. The vectors obtained are orthogonal to each other. It is used for data visualization as well as dimensionality reduction. The procedure of PCA includes calculating the covariance of the data matrix X which is the feature matrix of size 3256x512 here, then calculating eigenvectors of the covariance matrix and then transforming data using those vectors.

After applying PCA on our features we observe that around 10 principal components explain most of the variance. The visualization using PCA does not give good results compared to t-SNE as PCA is a linear technique.

2.2.2 Linear Discriminant Analysis

LDA is a supervised dimensionality reduction technique used in statistics as well as machine learning. Its main aim is to preserve maximum discriminative power among classes so that one class can be differentiated form another by maximizing separability. It is achieved by minimizing the inter class variance while maximizing the separability of



(a) Scree Plot (b) PCA visualization

Figure 3: PCA Analysis

the means of all the classes. The main objective function of LDA is:

$$J(w) = \frac{w^T S_B w}{w^T S_W w}$$

where w is the projection vector, S_B is the between-class scatter matrix and S_W is the inter class-scatter matrix. We have to maximize $J(w)$.

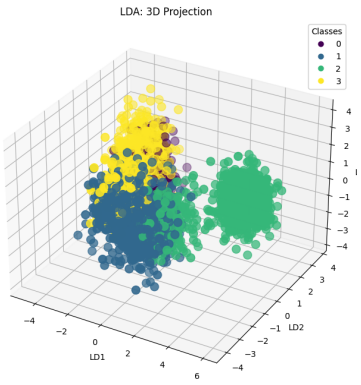


Figure 4: LDA Visualization (3-D)

Fig.4 shows the features after applying LDA. We can see one class is nearly separate from all other which was observed in t-SNE as well. The other classes are mixed showing that they have common features as observed previously. Thus from all the analysis we observe that two classes are easily separable while the some features in the embeddings have moderate to good relationship with the output. Next we on to feature selection and training of model.

2.3 Feature Analysis and Model Training

After feature extraction and feature analysis we use two methods for feature selection namely PCA and SVD. We train three traditional ML classifiers - Logistic Regression, Support Vector Machines and Random Forest Classifier.

2.3.1 Feature Selection Using PCA

From the analysis above we can see that most of the variance can be explained using 10 principal components. So we use PCA to convert our data from 512 dimensions to 10 dimensions. Then we split the data in train and test sets in ratio 8:2 and train our ML classifiers on the train data and evaluate their performance on the test dataset.

2.3.2 Feature Selection using SVD

Singular Value Decomposition is a matrix decomposition technique which has use cases in a multitude of domains. It decomposes a given matrix into three simpler matrices, revealing important properties of the original matrix, such as its rank, range, and null space. It can also be used as a compression and dimension reduction technique. In our approach we factorize our data matrix (features) using SVD. Using the resulting three matrices we calculate the important vectors that explain the data and convert our original data into lower dimension. SVD is given by :

$$A = U\Sigma V^T$$

where A is the original $m \times n$ matrix , U is an $m \times m$ orthogonal matrix, whose columns are the left singular vectors of A, Σ is an $m \times n$ diagonal matrix containing the singular values $\sigma_1, \sigma_2, \ldots, \sigma_r$, where $\sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_r \geq 0$ and V^T is the transpose of an $n \times n$ orthogonal matrix, whose columns are the right singular vectors of A. We use the singular values to calculate the relative importance. From Fig 5 we can see one singular is much

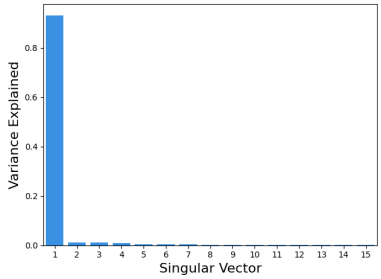


Figure 5: SVD Analysis

more responsible for variance than others. For an equal comparison between SVD and PCA we take 10 singular vectors in SVD to reduce our dimensions from 512 to 10. Then we split the data in the same ratio and apply the traditional classifiers on them and then

evaluate the models.

2.3.3 Logistic Regression

Logistic Regression is generally used for binary classification tasks using the sigmoid function. It uses cross entropy as a loss function. It predicts the probability of an outcome y belonging to one of two classes (e.g., 0 or 1) as:

$$P(y = 1|x) = \sigma(w^T x + b) = \frac{1}{1 + e^{-(w^T x+b)}} \tag{1}$$

In multiclass case like ours we extend it to use softmax function for calculating probabilities.

$$\sigma(z)_i = \frac{e^{z_i}}{\sum_{j=1}^{K} e^{z_j}}, \quad \text{for } i = 1, 2, \ldots, K \tag{2}$$

2.3.4 Support Vector Machines

SVM is a popular ML technique for classification purposes. Its main aim is to maximize margin decision boundary between classes.. SVM attempts to find the hyperplane that best separates the data into different sections by maximizing the margin between the classes. SVM can handle linearly as well as non-linearly separable data by using kernel functions known as the kernel trick to map the data into higher-dimensional spaces where separation becomes possible. The primal problem of SVM is stated as:

$$\text{minimize} \quad \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n} \epsilon_i \tag{3}$$

$$\text{subject to} \quad y_i(w^T x_i + b) \geq 1 - \epsilon_i, \quad \forall\, i \in n \tag{4}$$

2.3.5 Random Forest Classifier

Random Forest is an ensemble technique used for decision trees. It works by constructing multiple decision trees during training and gives the output using majority voting from output of all trees. Each tree in the Random Forest is built using a random subset of the training data called bagging and a random subset of the features, introducing randomness into the model and reducing the risk of overfitting. They can also be used for calculating feature importance as shown previously.100 decision trees were used for training.

3 Results

In this section we will analyze the results which we obtained after evaluating our ML classifiers. For SVD random forest performed the best while for PCA, SVM performed the best. 100 decision trees were used in Random Forest while SVM used the 'rbf' kernel. Overall it is clearly visible that PCA reduced features far outclassed the SVD reduced

features. One possible reason for this could be that PCA mean centers the data before applying transformation while SVD does no such thing. Mean centering could be the factor that led PCA to be much better than SVD.

	Logistic Regression	**SVM**	**Random Forest**
PCA	0.93	0.96	0.94
SVD	0.65	0.67	0.68

Table 2: Accuracy Scores

	Logistic Regression	**SVM**	**Random Forest**
PCA	0.93	0.96	0.94
SVD	0.65	0.63	0.67

Table 3: F1 Scores

Fig. 6 shows the confusion matrix and the ROC- AUC curve for the best model i.e. SVM using PCA features. The ROC curve has value 1 for nearly all the classes indicating that model is able to separate the data perfectly. Results for other models are present in the code files.

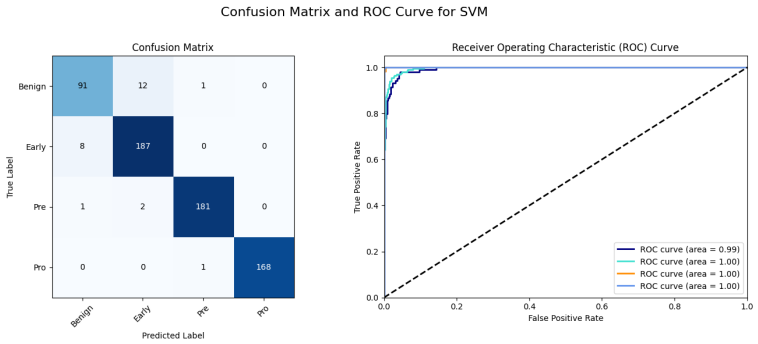


Figure 6: SVM for PCA Features

4 Conclusion

This project examined the application of a new approach for multiclass classification of leukemia. The first step in our approach was of feature extraction using a pre-trained model which converted the 224x224x3 image into 512 feature vector. Thereafter, feature analysis is done which showed us that two classes are different from others while others have common features and some of the features have a moderate relationship with the output class. Thereafter feature selection is done using PCA and SVD which select 10 features each reducing our dimension down to 10. After that multiple ML classifiers were applied to classify the data and found that PCA is much better than SVD possibly due to being mean centered with the best classifier being SVM with accuracy 0.96 and f1-score of 0.96.