MANIPAL INSTITUTE OF TECHNOLOGY

MANIPAL

(A constituent institution of MAHE, Manipal)

Project Report

On

Fundamentals of Machine Learning Lab

Subject Code: DSE 2242

Names	Registration No
Amit Anand	220968430
Mangalam Sontakke	220968428
Miriyala Amruni	220968402

Department of Data Science & Computer Applications,

Manipal Institute of Technology,

Manipal

JAN -MAY 2024

Table of Contents

ABSTRACT

CHAPTER 1
INTRODUCTION

CHAPTER 2
METHODOLOGY

CHAPTER 3
EXPERIMENTAL SETUP

CHAPTER 4
DATASET

CHAPTER 5
RESULT AND DISCUSSION

CHAPTER 6
CONCLUSION

ABSTRACT

This mini project aims to compare the performance of three distinct machine learning models on an image dataset by evaluating performance of each with and without Principal Component Analysis. A Bone Fracture Detection dataset has been utilized for this analysis.

The primary objective of this study is to evaluate the effectiveness of three machine learning models: Support Vector Machines, Random Forests, and KNN Classifier. The dataset consists of fractured and non-fractured X-ray images of several joints in the upper extremities.

The dataset is loaded and preprocessed to normalize the pixel values, feature scaling is applied. Subsequently, it is split into training and testing sets. Evaluation metrics such as accuracy, F1-score, precision, and recall are computed to assess the performance of each model.

This study aims to provide insights into the strengths and limitations of different machine learning models when applied to image classification tasks.

INTRODUCTION

Classification of images is a computer vision task which involves the classification of pictures according to their image content and grouping them in groups or categories. It aims to develop algorithms or models that will be able to label images, so computers can understand and interpret visual information in a similar way as humans. This level of automation has enabled humans to achieve efficiency and effectiveness, consistency and objectivity, detection of trends and anomalies etc.

For the purpose of our research, we have used the following algorithms:-

1. Random Forests
2. k-Nearest Neighbors
3. Support Vector Machines

SVM is a supervised learning algorithm used for classification and regression tasks. It works by finding the hyperplane that best separates different classes in the feature space, maximizing the margin between classes while minimizing classification errors.

Random Forests is an ensemble learning method used for classification and regression tasks. It works by constructing multiple decision trees during training and outputting the mode or mean prediction of the individual trees for classification or regression, respectively. Random Forests are known for their robustness, scalability, and ability to handle high-dimensional data with ease.

k-Nearest Neighbors is a non-parametric supervised learning algorithm which excels at classification. It stores all training data and employs a distance metric to find the k nearest neighbors for a new, unlabeled point. KNN is also called a lazy learner because it defers the actual learning

process until a new data point needs classification, rather than actively learning from the training data during a designated training phase.

Additionally, we have incorporated **Principal Component Analysis** (PCA), and have evaluated the performance of each of these models both with and without PCA. PCA is a dimensionality reduction technique used to reduce the number of features in a dataset while preserving most of its variance. When combined with the above algorithms, PCA can help improve performance by reducing overfitting and computational complexity.

We have used a Bone Fracture Detection dataset which comprises images of different joints in the upper extremities. This dataset consists of approximately 9,463 images in the training and testing set. We have iterated through this dataset to obtain 177 images. It consists of 2 classes. Fractured having a label of 0 and non-fractured having a label of 1.

After preprocessing and scaling the data our goal in this project is to delve deeper into the intricacies of the three models and compare the efficiency of each based on their evaluation metrics.

METHODOLOGY

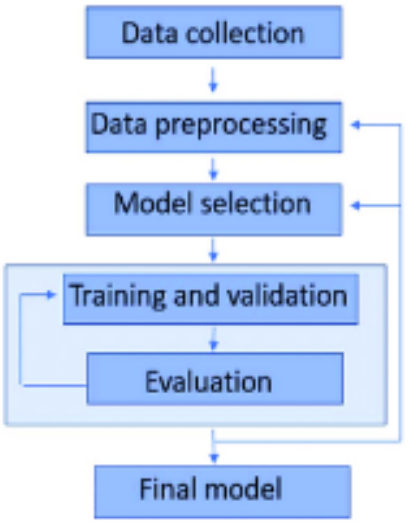

Fig 1: Flowchart of the model implementation on dataset

Data Collection & Preprocessing:

The dataset has been taken from Kaggle -

https://www.kaggle.com/datasets/vuppalaadithyasairam/bone-fracture-detection-using-xrays/data

The obtained dataset undergoes feature extraction using the Histogram of Oriented Gradients (HOG) method. This technique transforms the raw X-ray images into a representation that captures the distribution of gradients' orientations across localized regions. By employing HOG, the dataset ensures that essential structural information is extracted, enabling robust and effective training of machine learning models for bone fracture detection.

Datasets are typically split into two subsets: a training set and a testing set. The training set is used to train the models, while the testing set is used to evaluate their performance.

The splitting process ensures that the models are trained on one set of data and evaluated on a separate, unseen set, which helps assess their generalization ability.

It's common to reserve a certain percentage of the dataset (e.g., 20-30%) for testing, while the remaining data is used for training. Here, we have set the test-size as 0.2.

Model Selection:

The rationale behind selecting Support Vector Machines (SVM), k-Nearest Neighbors (KNN), and Random Forests for comparison in the image classification task lies in their unique characteristics and capabilities that make them suitable for handling image data:

1. Support Vector Machines (SVM):

- SVMs are well-suited for binary and multiclass classification tasks, making them a natural choice for image classification where the goal is to assign images to predefined classes or categories.
- SVMs excel in handling high-dimensional data, which is common in image datasets where each pixel serves as a feature. They are effective at finding the hyperplane that best separates different classes in the feature space.
- Additionally, SVMs have a regularization parameter (C) that helps control overfitting, making them robust models for image classification tasks with potentially complex decision boundaries.

2. k-Nearest Neighbours:

- KNN doesn't require complex model building during training. Instead, it stores all the training data points in its memory.

- When presented with a new, unlabeled data point, KNN identifies the k closest neighbors (data points) in the training set based on a distance metric (like Euclidean distance).
- KNN predicts the class label for the new data point by voting on the most frequent class among its k nearest neighbors. In essence, the new point is assigned the class it "resembles" the most in the training data.

3. Random Forests:

- Random Forests are ensemble learning methods that combine multiple decision trees during training and output the mode or mean prediction of the individual trees for classification.
- They are effective for image classification tasks due to their ability to handle high-dimensional data, nonlinear relationships, and complex decision boundaries.
- Random Forests are also known for their robustness to noise and outliers, making them suitable for real-world image datasets that may contain varying levels of noise and variability.

PCA is a dimensionality reduction technique that can be used to reduce the dimensionality of image data while preserving most of its variance. This is beneficial for image classification tasks where the original feature space may be high-dimensional, leading to computational complexity and potential overfitting.

By reducing the dimensionality of the feature space, PCA can help improve the performance of the above models, by mitigating overfitting and reducing computational burden.

In summary, SVM, KNN, and Random Forests were selected for comparison in the image classification task based on their ability to handle high-dimensional image data, robustness to noise and outliers, and effectiveness in capturing complex relationships and decision boundaries. Each

model, combined with PCA, offers unique advantages that can contribute to the overall performance and accuracy of the image classification system.

Model Training and Evaluation:

1. Training Procedure:

1. Support Vector Machines (SVM): SVMs are trained by finding the hyperplane that best separates different classes in the feature space. The training procedure involves optimizing the hyperplane parameters, including the margin and the regularization parameter (C), using techniques such as gradient descent or quadratic programming. This is first implemented without PCA, followed by reducing the dimensionality of the features using PCA, and then running the SVM model identically.
2. k-Nearest Neighbors: When a new image needs classifying, KNN finds the k closest data points in the training set based on distance. Then, it predicts the class label for the new image by voting on the most frequent class among its k nearest neighbors. In essence, KNN assigns a class to the new image based on the majority vote of its most similar neighbors in the training data. This algorithm is implemented both with and without PCA.
3. Random Forests: Random Forests are trained by constructing multiple decision trees during training. Each tree is trained on a bootstrapped sample of the training data, and at each split, a random subset of features is considered. The final prediction is obtained by averaging or taking the mode of the predictions of individual trees. Once again, this algorithm is implemented first without PCA, followed by with PCA.

Hyperparameter tuning may be performed for each model to optimize performance.

- For SVM, tuning parameters such as the kernel type (linear, polynomial, or radial basis function), C (regularization parameter), and gamma (kernel coefficient) may be optimized.
- For KNN, the parameter ‘k’, which specifies the number of nearest neighbors, can be varied.
- For Random Forests, parameters such as the number of trees, maximum depth of trees, and minimum number of samples per leaf may be tuned.

In all 3 models, PCA is applied to reduce the dimensionality of the feature space, followed by training on the respective model.

2. Evaluation Metrics:

- Accuracy: The proportion of correctly classified instances out of the total number of instances. It provides an overall measure of the model's correctness.
- Precision: The proportion of true positive predictions out of all positive predictions. It measures the model's ability to correctly identify positive instances.
- Recall: The proportion of true positive predictions out of all actual positive instances. It measures the model's ability to capture all positive instances.
- F1-score: The harmonic mean of precision and recall, providing a balance between the two metrics. It is useful when there is an imbalance between the classes in the dataset.

All of these have been displayed along with a confusion matrix.

EXPERIMENTAL SETUP

Environment:

- Python version: 3.8.5, 3.11.7

- Jupyter Notebook version: 6.1.4, 7.0.8

Libraries:

I. sklearn: version: 1.4.1.post1

1. model_selection:
   - GridSearchCV
   - train_test_split
2. metrics:
   - accuracy_score
   - classification_report
   - confusion_matrix
3. ensemble:
   - RandomForestClassifier
4. neighbors:
   - KNeighborsClassifier
5. preprocessing:
   - StandardScaler
   - MinMaxScaler
6. decomposition:
   - PCA
7. svm

II. skimage version: 0.20.0

1. transform:
   - resize
2. io:
   - imread
3. feature:
   - hog
4. exposure

DATASET

We have used a comprehensive dataset of fractured and non-fractured X-ray images of several joints in the upper extremities used in bone fracture detection. Our project uses this dataset to build an image classifier to detect fractures from a given X-ray image.

(a) (b) (c)

Fig 2: Representational images from the bone fracture dataset

(a) and (c) have fracture, while (b) has fracture on left hand only

The bone fracture detection datasets are a useful resource for researchers and developers who want to train machine learning models, specifically focusing on object detection algorithms, to automatically detect and classify bone fractures in X-ray images. These datasets; diversity of fracture classes enables the development of robust models capable of accurately identifying fractures in different regions of the upper extremities.

RESULT AND DISCUSSION

Support Vector Machines (SVM) is a powerful supervised learning algorithm commonly used for classification tasks. In an experiment conducted on the bone fracture dataset, the SVM model exhibited moderate performance metrics. Without PCA, its accuracy is 63.88%, implying the model correctly classified a substantial portion of the dataset. Additionally, the precision of 0.76 indicates that when the model predicted a fractured image, it was correct approximately 76% of the time. For non-fractured class, it was correct 53% of the time. With PCA however, its accuracy falls to 50%, and precision for fractured and non-fractured classes are 0.62 and 0.33 respectively. These results underscore SVM's utility in classification tasks, particularly in scenarios where interpretability and generalization are paramount. However, we see that when PCA is used in conjunction with SVM, it leads to a drop in overall performance of the model, which is further discussed below.

Next, k Nearest Neighbors was implemented on the data. Interestingly, we observe that the performance of the KNN model is exactly identical both with and without the inclusion of PCA. This may be explained by the fact that KNN is a lazy learner as well as an instance-based learning algorithm, which does not create the typical model structure of features and target variables, and instead learns from individual data points. Hence, a dimensionality reduction technique like PCA is redundant. The observed performance metrics are 61.11% accuracy, precision of 0.65 and 0.4 for fractured and non-fractured respectively.

The dataset was also fed to Random Forests, a popular ensemble learning method. Without the use of PCA, it yielded an accuracy of 91.66%, and precision for fractured and non-fractured classes as 0.95 and 0.86 respectively. When PCA was used to transform the dataset into a lower-dimensional space, capturing the most important features while minimizing information loss, there was a fall in performance metrics once again, at 83.33%, 0.9 and 0.76. Despite the modest accuracy, this approach demonstrates the effectiveness of combining dimensionality reduction techniques like PCA with

ensemble learning methods like Random Forests for handling complex tasks like bone fracture detection. The lower dimensionality achieved through PCA may enhance model performance and generalization while reducing computational costs and overfitting risks. However, further experimentation and parameter tuning may be required to improve the accuracy of the model on this dataset.

The performance metrics of the 3 models are tabulated below:-

Model		Accuracy	Precision for Fractured	Precision for Non-Fractured
Random Forest	*w/o PCA*	91.66%	0.95	0.86
	with PCA	83.33%	0.9	0.73
k-Nearest Neighbors	*w/o PCA*	61.11%	0.65	0.4
	with PCA	61.11%	0.65	0.4
Support Vector Machines	*w/o PCA*	63.88%	0.76	0.53
	with PCA	50%	0.62	0.33

Table 1: Performance metrics for the implemented models

In each of the 3 models above, we observe that implementation of PCA either leads to no change, or in fact leads to a drop in model performance. This can be attributed to the following drawbacks of PCA:-

1. **Loss of Information**: PCA involves compressing the original feature space into a lower-dimensional subspace by linear transformation. This compression inevitably leads to some loss of information, especially if the principal components chosen do not capture

enough variance in the data. Consequently, the reduced feature space may not fully represent the complexity of the original data, leading to a drop in accuracy.

2. **Suboptimal Component Selection**: PCA selects the principal components based on their ability to explain the variance in the data. However, these components may not always align well with the predictive patterns that our chosen model aims to capture. If the principal components do not adequately represent the underlying structure of the data relevant to classification, it can result in a decrease in accuracy.
3. **Loss of Interpretability**: PCA transforms the original features into a new space of uncorrelated components, making it more challenging to interpret the importance of individual features in the context of the classification task. Without this interpretability, the classification algorithm may struggle to capture complex relationships present in the original feature space, resulting in reduced accuracy.

CONCLUSION

In conclusion, this comparative analysis of machine learning models on a Bone Fracture Detection dataset provided valuable insights into their performance and effectiveness for image classification tasks.

Based on the provided information, we can compare the performances of Support Vector Machines (SVM) , Random Forests and k-Nearest Neighbour Classifier (KNN) with and without Principal Component Analysis(PCA) on the dataset.

In this comparison, **Random Forests without PCA** outperforms all the other models in terms of accuracy, with an accuracy of 91.66%. The higher accuracy of Random Forests without PCA model suggests that it was able to generalize better to the dataset and make more accurate predictions overall. This could be attributed to the inherent robustness of Random Forests to overfitting and their ability to handle high-dimensional data effectively.

In this comparison, **SVM with PCA** underperforms against all the other models in terms of accuracy. The lower accuracy of SVM with PCA may indicate that it struggled to capture the complex relationships within the dataset or that it required more fine-tuning of hyperparameters to achieve better performance or suboptimal dimensionality reduction.

The observed trend suggests that models without PCA outperform those with PCA. This discrepancy indicates that PCA may not efficiently capture relevant information or optimize feature representation for the given task. However, it's important to note that PCA might provide computational benefits, especially with high-dimensional data like images.

In conclusion, based on the provided accuracy metrics, Random Forests performs better than SVM and KNN on the dataset. This could be attributed to Random Forests' robustness and suitability for

handling high-dimensional data like images. Nonetheless, further analysis and experimentation may be required to fully understand the strengths and weaknesses of each model on this dataset.

The overarching objective of utilizing this dataset is to expedite the development of machine learning solutions aimed at automating fracture detection. By doing so, it facilitates progress in medical diagnostics, ultimately enhancing patient care and contributing to advancements in healthcare technology.