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
Aradhya Goswami	220968034
Chaitra Narem	220968013

Department of Data Science & Computer Applications,
Manipal Institute of Technology,
Manipal
JAN -MAY 2024

Table of Contents

ABSTRACT

This mini project aims to compare the performance of three distinct machine learning models on an image dataset. Specifically, the Street View House Numbers (SVHN) dataset, a widely used benchmark dataset in the field of image recognition, 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 PCA using Random Forests. The SVHN dataset consists of real-world images of house numbers extracted from Google Street View Images. Each image contains a certain number of digits from 0-9, making it a multiclass classification problem.

Basic exploratory data analysis was conducted, and a few examples from the dataset have been displayed. The dataset is then preprocessed to normalize the pixel values, and feature scaling was applied as well. 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.

CHAPTER 1: INTRODUCTION

Image classification is a task in computer vision that involves categorizing images into predefined classes or categories based on their visual content. The goal is to develop algorithms or models that can assign labels to images, thereby enabling machines to understand and interpret visual data similar to how humans do. Such levels of automation helped humans achieve automation and efficiency, consistency and objectivity, detection of patterns and anomalies etc.

In this comparative analysis study, we have used Support Vector Machines, PCA with Random Forests and Random Forests. 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. 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 Random Forests, PCA can help improve the performance of the Random Forest algorithm by reducing overfitting and computational complexity. 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.

We have used the Street View House Numbers dataset which consists of approximately 73,257 images in the training set. These images were obtained from Google Street View and consist of digits numbered from 0-9. 10 classes, 1 for each digit. Digit '1' has label 1, '9' has label 9 and '0' has label 10. 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.

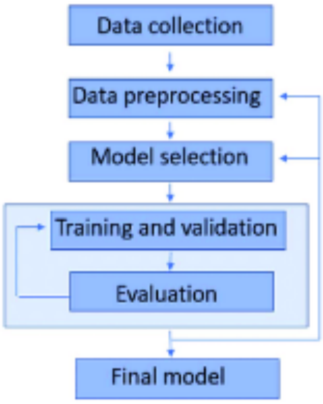


fig 3.1 Flow Chart

CHAPTER 2: METHODOLOGY

Data Collection:

The SVHN dataset was obtained using the TensorFlow Datasets (TFDS) library. TFDS provides a convenient interface for downloading and managing various datasets, including SVHN. A basic outline has been provided:

-We import the TensorFlow Datasets library as `tfds`.

- We use the `tfds.load()` function to load the SVHN dataset. The dataset is specified by its name `'svhn_cropped'`. We also specify the split `'train'` to obtain the training portion of the dataset.

- We set `as_supervised=True` to load the dataset in a tuple format `(image, label)`, where `image` represents the input images and `label` represents the corresponding class labels.

- We set `with_info=True` to retrieve additional information about the dataset, such as its size and number of classes.

Preprocessing:

Feature and label extraction was conducted. After which using the Standard Scaler library the pixel values were normalized.

Model Selection:

The rationale behind selecting Support Vector Machines (SVM), PCA with Random Forests, 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. Principal Component Analysis (PCA) with Random Forests:

- 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 Random Forests, another ensemble learning method, by mitigating overfitting and reducing computational burden.
- Random Forests are known for their robustness, scalability, and ability to handle high-dimensional data with ease, making them suitable for image classification tasks.

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.

In summary, SVMs, PCA with Random Forests, 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 offers unique advantages that can contribute to the overall performance and accuracy of the image classification system.

Model Training and Evaluation:

- 1. Dataset Splitting:

- The dataset is 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. We’ve set test-size as 0.3

2. Training Procedure:

- 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.
- PCA with Random Forests: PCA is applied to reduce the dimensionality of the feature space, followed by training a Random Forest classifier on the transformed data. The training procedure involves building multiple decision trees using bootstrapped samples of the training data and selecting random subsets of features at each split.
- 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.

- 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 Random Forests, parameters such as the number of trees, maximum depth of trees, and minimum number of samples per leaf may be tuned.

3. 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.

CHAPTER 3: EXPERIMENTAL SETUP

Environment:

- Python version: 3.8.5, 3.11.7

- Jupyter Notebook version: 6.1.4, 7.0.8

Libraries:

- matplotlib: 3.3.2, 3.8.0
- pandas: 1.1.3
- NumPy: 1.19.2, 1.26.4
- seaborn: 0.11.0, 0.12.0
- tensorflow_datasets : 4.9.4
- tensorflow 2.16.1
- scikit-learn (ensemble, svm as SVC, train_test_split): 0.23.2, 1.2.2
- StandardScaler: Included in scikit-learn

CHAPTER 4: DATASET

SVHN is a real-world image dataset for developing machine learning and object recognition algorithms with minimal requirement on data preprocessing and formatting. It can be seen as similar in flavor to MNIST (e.g., the images are of small cropped digits), but incorporates an order of magnitude more labeled data (over 600,000 digit images) and comes from a significantly harder, unsolved, real world problem (recognizing digits and numbers in natural scene images). SVHN is obtained from house numbers in Google Street View images.

- 10 classes, 1 for each digit. Digit `'1'` has label 1, `'9'` has label 9 and `'0'` has label 10.
- 73257 digits for training, 26032 digits for testing, and 531131 additional, somewhat less difficult samples, to use as extra training data
- Comes in two formats:
    1. Original images with character level bounding boxes.
    2. MNIST-like 32-by-32 images centered around a single character (many of the images do contain some distractors at the sides).

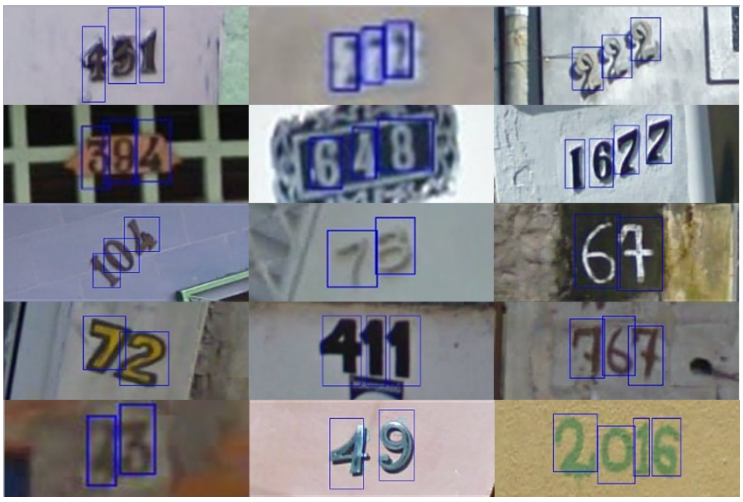


Fig 4.1 SVHN

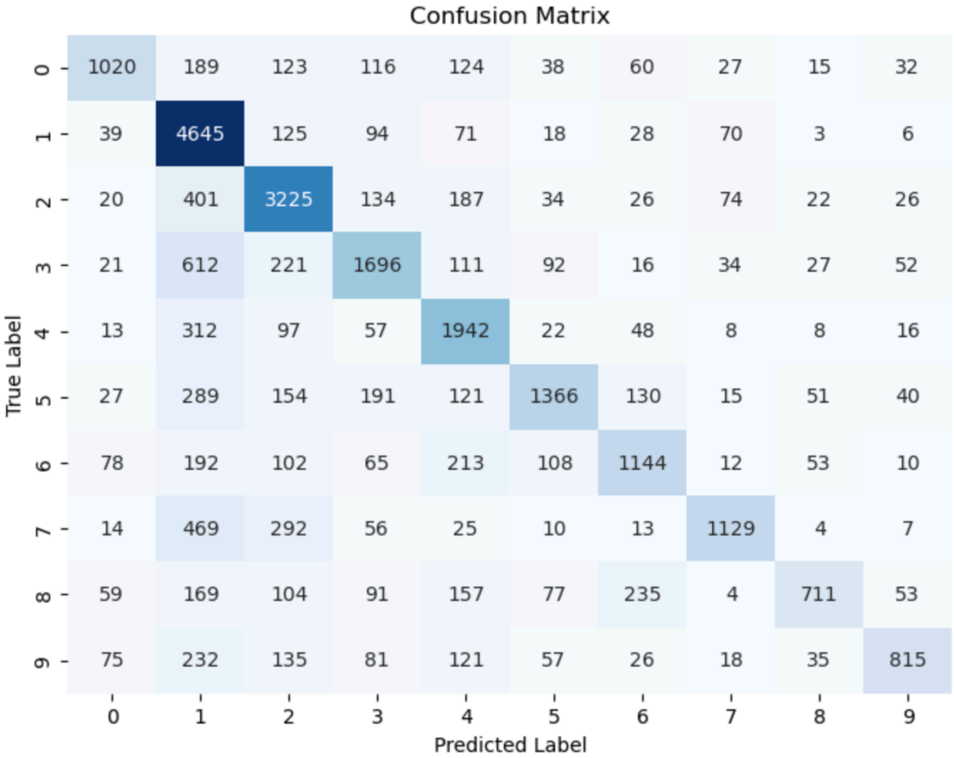
These are the original, variable-resolution, color house-number images with character level bounding boxes, as shown in the examples images above

CHAPTER 5: RESULT AND DISCUSSION

Support Vector Machines (SVM) is a powerful supervised learning algorithm commonly used for classification tasks. In an experiment conducted on the Street View House Numbers (SVHN) dataset, the SVM model exhibited moderate performance metrics. With an accuracy of 52%, the model correctly classified a substantial portion of the dataset. Additionally, the precision of 61% indicates that when the model predicted a positive class, it was correct approximately 61% of the time. A recall of 52% suggests that the model effectively captured around 52% of all instances of the positive class in the dataset. Furthermore, the F1 score, which balances precision and recall, yielded a value of 51%, indicating a reasonable overall performance of the SVM model on the SVHN dataset. Despite its limitations, such as handling large datasets efficiently, these results underscore SVM's utility in classification tasks, particularly in scenarios where interpretability and generalization are paramount.

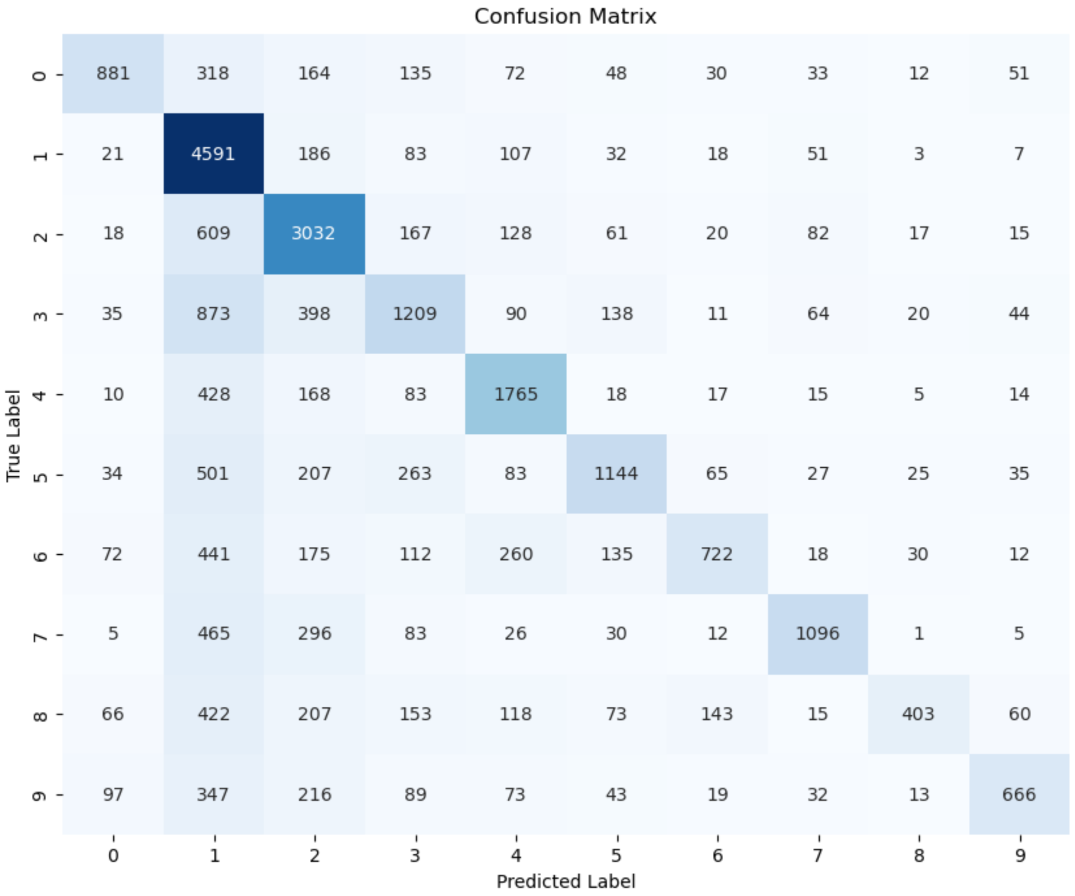
Principal Component Analysis (PCA) is a dimensionality reduction technique commonly used to preprocess data before applying machine learning algorithms. In an experiment conducted on the Street View House Numbers (SVHN) dataset, PCA was employed in conjunction with Random Forests, a popular ensemble learning method. With an accuracy of 59.58%, the PCA transformed the dataset into a lower-dimensional space, capturing the most important features while minimizing information loss. This pre-processed data was then fed into a Random Forest classifier, which leverages multiple decision trees to make predictions. 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 datasets like SVHN. 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 the SVHN dataset.

In the experiment conducted on the Street View House Numbers (SVHN) dataset, a Random Forest classifier was employed, resulting in an accuracy of 67.97%. Random Forest is an ensemble learning method that builds multiple decision trees during training and outputs the class that is the mode of the classes (classification) or mean prediction (regression) of the individual trees. This accuracy signifies the model's ability to correctly classify approximately 67.97% of the instances in the dataset. Random Forests are known for their robustness to overfitting and ability to handle high-dimensional data, making them suitable for tasks such as digit recognition in the SVHN dataset. Despite achieving a relatively high accuracy, further analysis of other performance metrics such as precision, recall, and F1 score could provide a more comprehensive evaluation of the model's effectiveness in classifying the SVHN dataset. Additionally, hyperparameter tuning and feature engineering techniques like PCA could potentially enhance the model's performance further.



	0	1	2	3	4	5	6	7	8	9
0	1020	189	123	116	124	38	60	27	15	32
1	39	4645	125	94	71	18	28	70	3	6
2	20	401	3225	134	187	34	26	74	22	26
3	21	612	221	1696	111	92	16	34	27	52
4	13	312	97	57	1942	22	48	8	8	16
5	27	289	154	191	121	1366	130	15	51	40
6	78	192	102	65	213	108	1144	12	53	10
7	14	469	292	56	25	10	13	1129	4	7
8	59	169	104	91	157	77	235	4	711	53
9	75	232	135	81	121	57	26	18	35	815

fig 5.1 Confusion Matrix Random Forest



True Label \ Predicted Label	0	1	2	3	4	5	6	7	8	9
0	881	318	164	135	72	48	30	33	12	51
1	21	4591	186	83	107	32	18	51	3	7
2	18	609	3032	167	128	61	20	82	17	15
3	35	873	398	1209	90	138	11	64	20	44
4	10	428	168	83	1765	18	17	15	5	14
5	34	501	207	263	83	1144	65	27	25	35
6	72	441	175	112	260	135	722	18	30	12
7	5	465	296	83	26	30	12	1096	1	5
8	66	422	207	153	118	73	143	15	403	60
9	97	347	216	89	73	43	19	32	13	666

fig 5.2 Confusion Matrix Random Forest

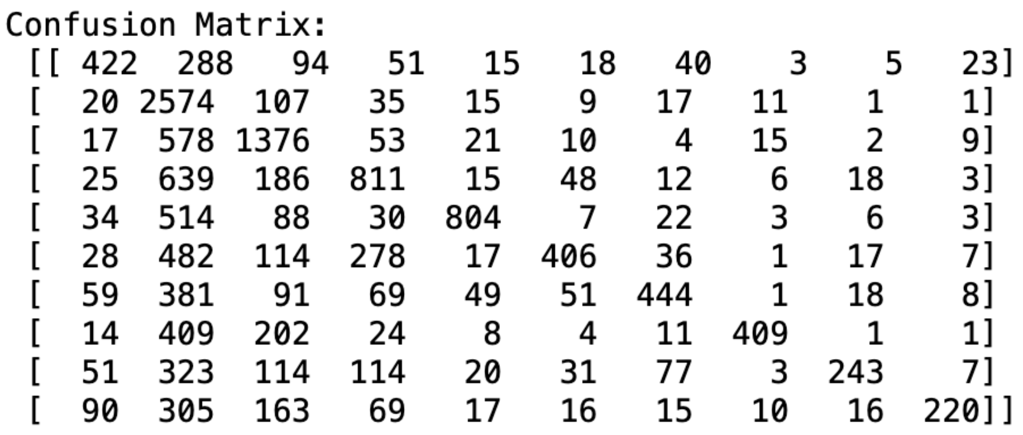

```
Confusion Matrix:
 [[ 422  288   94   51   15   18   40    3    5   23]
 [  20 2574  107   35   15    9   17   11    1    1]
 [  17  578 1376   53   21   10    4   15    2    9]
 [  25  639  186  811   15   48   12    6   18    3]
 [  34  514   88   30  804    7   22    3    6    3]
 [  28  482  114  278   17  406   36    1   17    7]
 [  59  381   91   69   49   51  444    1   18    8]
 [  14  409  202   24    8    4   11  409    1    1]
 [  51  323  114  114   20   31   77    3  243    7]
 [  90  305  163   69   17   16   15   10   16  220]]
```

fig 5.3 Confusion Matrix Random Forest

CHAPTER 6: CONCLUSION

In conclusion, this comparative analysis of machine learning models on the Street View House Numbers (SVHN) 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) and Random Forests on the Street View House Numbers (SVHN) dataset.

- SVM: Accuracy - 52%, Precision - 61%, Recall - 52%, F1 Score - 51%

- Random Forests: Accuracy - 67.97%

In this comparison, Random Forests outperformed SVM in terms of accuracy. Random Forests achieved an accuracy of 67.97%, whereas SVM achieved an accuracy of only 52%.

The higher accuracy of Random Forests suggests that it was able to generalize better to the SVHN 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.

While PCA may help in reducing dimensionality and possibly speeding up the computation, it still falls slightly short of Random Forests in terms of accuracy. However, it's important to note that PCA might provide computational benefits, especially with high-dimensional data like images.

On the other hand, SVM's lower accuracy might indicate that it struggled to capture the complex relationships within the SVHN dataset or that it required more fine-tuning of hyperparameters to achieve better performance.

In conclusion, based on the provided accuracy metrics, Random Forests performed better than SVM on the SVHN 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.