SVM vs PCA

Parth Agrawal

October 2025

1 Introduction

This report provides a brief comparison between Support Vector Machines and Principal Component Analysis.

2 Support Vector Machines (SVM)

Support Vector Machines is a type of machine-learning algorithm used mainly for classification (but also regression) tasks. The key idea is to find a boundary (hyperplane) that best separates data into different groups by maximizing the margin between classes.

2.1 Mathematical Derivation

Let training set be (x_i, y_i) for $i = 1, ..., m$ where $x_i \in R^n$ is a feature vector and $y_i \in +1, -1$ the class label.

Let the hyperplane (margin be defined by -

$$w^T x + b = 0$$

with $w^T \in R^n, b \in R$.

The distance from hyperplane of a point is -

$$d_i = \frac{w^T x_i + b}{||w||}$$

for any point x_i. Here $||w||$ is the Euclidean norm of the vector.

To make a linear classification, we may ignore some outliers, Thus points are classified as

$$\hat{y} = \text{sign}(w^T x + b).$$

For **hard - margin** we enforce that $d_i \geq 1$ for all i.

Our objective function will be

$$\min_{w,b} \frac{1}{2} ||w||^2$$

since margin is equal to $1/2||w||^2$. So minimizing it will maximize the margin.

In the presence of non-separable data, we'll allow some misclassifications by including a slack variable $\zeta_i \geq 0$ and a regularization parameter (C).

Our optimization problem becomes -

$$\min_{w,b,\zeta} \frac{1}{2}|w|^2 + C\sum_i \zeta_i \quad \text{s.t. } y_i(w^T x_i + b) \geq 1 - \zeta_i, \zeta_i \geq 0.$$

This method enables us to apply the kernel trick, which maps the data into higher dimensions for classification.

2.2 Usage

- Classification is best in high-dimensional spaces when the number of features is larger.
- Due to margin maximization, they have good generalization properties (i.e., low over-fitting).

2.3 Advantages

- Effective with large input dimensions
- Uses a subset of training points (support vector), thus memory efficient

2.4 Limitations

- Training time can be high for huge datasets.
- Choice of kernel and parameters can have a huge effect on performance.

3 Principal Component Analysis (PCA

Principal Component Analysis is an unsupervised learning technique for dimensionality reduction. For any given high-dimensional data, PCA finds a representation in fewer dimensions while retaining the maximum possible original variance.

It computes the "principal components," which are linear combinations of the original variables, are uncorrelated (orthogonal), and are ordered in a way that the first principal component captures the most significant variance, the next lesser, and so on.

3.1 Mathematical Derivations

We'll start by standardizing the dataset, i.e, subtracting means from its values. $x_{\text{inew}} = x_i - x_{\text{mean}}$

Next, we'll calculate the correlation matrix C,

$$C = \frac{X \cdot X^T}{N-1}$$

where X is the **data matrix** and N is the number of elements.

To find the eigen vectors and eigen values, we'll then use the equation

$$|C - \lambda| = 0$$

where λ is the eigenvalue.

We'll then use $C \cdot X = \lambda \cdot X$ to find the eigenvectors.

Next, we'll arrange the eigenvalues in descending order. The eigenvector with the highest eigenvalue is the principal component.

Next, we'll form a feature vector V with columns of values $\lambda_1, \lambda_2, \ldots \lambda_n$.

We'll then apply this to the transformation of the original dataset using the equation

$$Z = XV$$

This will give us Z as the matrix containing the principal components.

To reconstruct the data, we'll use

$$X = Z * V^T$$

where $X =$ **Row Mean Data**

Row Original Dataset = Row Zero Mean Data + Original Mean

3.2 Usage

- It simplifies the data into a smaller number of unrelated components, allowing for better analysis.
- It is useful for visualizing higher-dimensional data into 2 or 3 dimensions.

3.3 Advantages

- It is unsupervised and doesn't need any labels.
- It is efficient to compute and well-understood mathematically.

3.4 Limitations

- It is a **linear** method and only finds a linear combination of variables. If the true data has a nonlinear structure, PCA may not capture it well.
- By focusing on only variance, PCA may omit features that are low variance but highly predictive of the target variable.

4 Comparison

- SVM is is a **supervised** learning algorithm whereas PCA is an **unsupervised** learning algorithm.
- SVM's goal is to find a maximal margin hyperplane to separate the classes, while PCA's goal is to find orthogonal vectors to capture maximum variance.
- Both are used as a part of a machine learning toolkit.

5 Conclusion

Both SVM and PCA are used as foundational methods in machine learning and statistics. SVMs are used in data classification, whereas PCA is used to identify the principal components that account for the most variance in the data. Both methods use mathematical models and are efficient and powerful. However, both methods have limitations and pitfalls when it comes to implementation, and we must be careful when choosing when to use them.

References

[1] GeeksforGeeks. (n.d.). *Mathematical Approach to Principal Component Analysis (PCA)*. `https://www.geeksforgeeks.org/machine-learning/mathematical-approach-to-pca/`

[2] GeeksforGeeks. (n.d.). *Support Vector Machine (SVM) Algorithm*. `https://www.geeksforgeeks.org/machine-learning/support-vector-machine-algorithm/`

[3] Scikit-learn Developers. (n.d.). *1.4. Support Vector Machines*. `https://scikit-learn.org/stable/modules/svm.html`