SVMs and PCAs

Idant Srivastava

October 2025

Contents

Chapter 1

Support Vector Machines

1.1 What are SVMs?

Support Vector Machines (SVMs) are supervised machine learning algorithms that are used primarily for classification tasks. They can also be used for regression tasks. They work by finding the optimal hyperplane to separate the data points into different classes. The support vectors are data points closest to the decision boundary, which defines the hyperplane.

1.2 Math behind SVMs

For a binary classification problem with two classes, SVM tries to find the equation of a hyperplane that separates the data points with maximum margin. The equation of a hyperplane is as follows:

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

- w: the normal vector to the plane
- b: bias term

Classification

$$\hat{y} = \begin{cases} 1 : w^T x + b \geq 0 \\ 0 : w^T x + b < 0 \end{cases}$$

Optimization problem for SVM

We need to find the hyperplane that maximizes the margin between the two classes. This leads to the following optimization problem:

$$\min \frac{1}{2} ||w||^2$$

Soft Margin in Linear SVM Classifier

In the presence of outliers or non-separable data, the SVM allows some misclassifications by introducing slack variables ζ_i. The optimization problem is modified as:

$$\min \frac{1}{2}||w||^2 + C\sum_{i=1}^{m}\zeta_i$$

- C is the regularization parameter controlling the trade-off between maximizing the margin and minimizing the classification errors.
- ζ_i are the slack variables that allow misclassifications.

1.3 Why use SVMs?

SVMs are very effective for high-dimensional data and cases where the number of dimensions exceeds the number of samples. They are memory-efficient, as they only use support vectors for making predictions. The kernel trick allows SVMs to classify data that are not linearly separable by mapping them to a higher-dimensional space without performing complex transformations.

1.4 Advantages of SVMs

SVMs are effective in high-dimensional spaces and are memory-efficient, since they rely only on support vectors instead of the entire dataset. The versatility of different kernel functions allows SVMs to adapt to various data structures, and they are robust against overfitting, especially in high-dimensional spaces. SVMs also provide a globally optimal solution, as the optimization function is convex in nature.

1.5 Pitfalls and Limitations

They do not directly provide probability estimates. The computational complexity becomes very high for larger datasets, and training the data could take significantly more time. Careful tuning of kernel functions and hyperparameters is required for performance. Finding these parameters could be very time-consuming. SVMs struggle with noisy data, and interpreting the model becomes difficult when using non-linear kernels.

Chapter 2

Principal Component Analysis

2.1 What is PCA?

Principal Component Analysis is an unsupervised learning technique for dimensionality reduction that transforms a large dataset with many features into a smaller set of new variables called principal components (PCs).

2.2 Math behind PCA

Initial Setup

We assume the data is centered: each column has zero mean. If not centered initially, we compute $X^{'} = X - \mu^T$ where μ is the mean vector.

Covariance Matrix

The covariance matrix shows us the relationship between the variables. The eigenvectors of this covariance matrix are the PCs, and the eigenvalues are the amount of variance in the data of each component. The formula of the covariance matrix is:

$$C = \frac{1}{n-1} X^{'T} X^{'}$$

- n: number of samples

After finding the covariance matrix, we find its eigenvectors and eigenvalues. These eigenvectors form the projection matrix W_k, where k is the number of dimensions.

Transform the data

To project the original data into lower dimensions, we compute:

$$Z = X^{'} W_k$$

2.3 Why use PCA?

PCA is used to reduce the dimensionality of datasets while preserving as much variance as possible. This helps in the visualization of data and improves computational efficiency.PCA can improve model performance, reduce overfitting, and speed up training.

2.4 Advantages of PCA

PCA helps in dimensionality reduction while retaining most of the dataset's variance. This makes dealing with large data sets more manageable. It removes correlated features and reduces multicollinearity, and can improve the performance of machine learning algorithms by reducing noise. PCA eases visualization of high-dimensional data in 2D or 3D graphs and is computationally efficient.

2.5 Pitfalls and Limitations

The principal components are linear combinations of the original features, making interpretation challenging. PCA assumes that directions of maximum variance are most important, which may not hold for all datasets. It is sensitive to outliers that can disproportionately affect PCs. Information loss is inevitable when dimensions are reduced, potentially discarding important but low-variance features.