Universidad Politécnica
de Madrid

Escuela Técnica Superior de Ingenieros Informáticos

Master in Data Science

Master Thesis

Study of Dimensionality Reduction Techniques and Interpretation of their Coefficients, and Influence on the Learned Models.

Author: Miguel Ángel García-Gutiérrez Espina

Madrid, July, 2023

This Master Thesis has been deposited in ETSI Informáticos de la Universidad Politécnica de Madrid.

Master Thesis
Master in Data Science

Title: Study of Dimensionality Reduction Techniques and Interpretation of their Coefficients, and Influence on the Learned Models.
July, 2023

Author: Miguel Ángel García-Gutiérrez Espina
Supervisor: Esteban García-Cuesta
Artificial Intelligence Department
ETSI Informáticos
Universidad Politécnica de Madrid

Summary

Dimensionality reduction is the transformation of data from a high-dimensional space into a low-dimensional space so that the low-dimensional space retains the essential information of the data. It aims to overcome the curse of dimensionality, which refers to the challenges posed by high-dimensional data, such as increased computational complexity, the risk of overfitting, and, especially, the reduction of explainability.

The reduction of explainability is addressed by the field of Explainable Artificial Intelligence (XAI), which focuses on understanding machine learning models and explaining them in human and understandable terms. Combining XAI and dimensionality reduction, this thesis presents a method of explaining principal components based on their correlations to the input features.

First, the dimensionality of the data was reduced using state-of-the-art dimensionality reduction techniques such as SLMVP. These reduction techniques were combined with different machine learning classifiers to fine-tune their parameters. The objective was to identify the optimal configuration that achieves the highest accuracy with the given data. The accuracy obtained with only the first k components is measured for different values of k. A recommendation is then given as to the number of components that should be kept.

Second, the performance of the techniques in capturing and preserving the structure of the original dataset is analyzed by plotting their projections in 2 and 3-dimensional plots. We look into whether the data points are evenly distributed or not, this shows how effectively the technique has managed to capture the overall variance of the dataset, and whether the graph exhibits a clear separation of the different classes. This, paired with the accuracy obtained in the previous classification task, tells us about the goodness of the technique. Furthermore, we show that among the supervised dimensionality reduction techniques evaluated, SLMVP stands out as the sole method capable of effectively handling multilabel datasets.

Finally, the correlations between the original data and each one of the components obtained through dimensionality reduction are leveraged to extract meaningful qualitative information. This is based on the fact that the components are the directions of maximum variability of the data and it is fair to assume that the variables that have a high absolute correlation with a component are given a high significance by the dimensionality reduction technique. A recommendation is then given as to which features should be selected for a posterior machine learning task, based on their absolute correlation with the components.

In addition, the correlations are also leveraged to compare the similarity and dissimilarity of components realized by applying different techniques. This is done by

calculating the spearman correlation coefficient of the absolute correlation between two components, obtaining a similarity score. Observations are then made about the similarity of techniques and the techniques that stand out as unique.

The results indicate that SLMVP demonstrates clearer separation of classes in both single-label and multilabel datasets compared to other tested techniques. It achieved the highest accuracies in 3 out of the 4 datasets employed.

Abstract

Dimensionality reduction is the transformation of data from a high-dimensional space into a low-dimensional space so that the low-dimensional space retains the essential information of the data. It aims to overcome the curse of dimensionality, which refers to the challenges posed by high-dimensional data, such as increased computational complexity, the risk of overfitting, and, especially, the reduction of explainability. This last challenge is addressed by the field of Explainable Artificial Intelligence (XAI), which focuses on understanding machine learning models and explaining them in human and understandable terms. Combining XAI and dimensionality reduction, this thesis presents a method of explaining principal components based on their correlations to the input features. First, the dimensionality of the data was reduced using state-of-the-art dimensionality reduction techniques such as SLMVP. These reduction techniques were combined with different machine learning classifiers to fine-tune their parameters. Second, the performance of the techniques in capturing and preserving the structure of the original dataset is analyzed by plotting their projections in 2 and 3-dimensional plots. Furthermore, we show that among the supervised dimensionality reduction techniques evaluated, SLMVP stands out as the sole method capable of effectively handling multilabel datasets. Finally, the correlations between the original data and each one of the components obtained through dimensionality reduction are leveraged to extract meaningful qualitative information about the reduced-dimensional space. The correlations are also leveraged to compare the similarity and dissimilarity of components realized by applying different techniques, by calculating the spearman correlation coefficient of the absolute correlation between two components, obtaining a similarity score. The results indicate that SLMVP demonstrates clearer separation of classes in both single-label and multilabel datasets compared to other tested techniques. It achieved the highest accuracies in 3 out of the 4 datasets employed.

Contents

Chapter 1

Introduction

As the availability of data continues to grow exponentially, the need for efficient and effective methods to reduce dimensionality and extract relevant information becomes more important. Dimensionality reduction involves transforming data from a high-dimensional space to a low-dimensional space, while preserving meaningful properties close to its intrinsic dimension. It plays a crucial role in the preprocessing phase of a machine learning pipeline. It is typically applied after the data has been collected and before the training phase begins.

In the domain of machine learning, two commonly employed techniques for dimensionality reduction are feature selection and feature extraction. These techniques aim at identifying the most relevant features from the original set of input variables. Feature selection tries to identify the subset that contains the most relevant and informative features while excluding redundant ones. Feature extraction transforms the original set of features into a new set that is created by combining the original variables, thereby projecting the data onto a subspace of lower dimensionality. This process involves capturing the underlying structure and patterns of the data while reducing its dimensionality.

There are two main binary classes into which feature extraction techniques can be classified: supervised or unsupervised, and local or not local. Supervised learning consists on training a model on labeled data, where the input samples have associated target or output labels, whereas unsupervised learning consists on training a model on unlabeled data, where the input samples do not have associated target or output labels. A lot of research in dimensionality reduction has been devoted to unsupervised techniques, including PCA, KPCA, LPP and LLE. This approach results in dimensions that have no statistical guarantee of being close to the best ones for classification. The recent supervised techniques LOL and SLMVP make it worth shifting the focus, since they outperform their unsupervised counterparts in several metrics.

Local dimensionality reduction techniques focus on preserving the local structure and relationships among data points in the reduced feature space. They aim to maintain the similarities and dissimilarities among neighboring data points. LLE, LPP and SLMVP have this property, however SLMVP is the only dimensionality reduction technique that is both local and supervised. In the Related Work section of this thesis, we explore various feature extraction techniques including Principal Component Analysis [16] (PCA), Kernel-PCA [18] (KPCA), Linear Optimal Low Rank

[22] (LOL), Locality Preserving Projections [11] (LPP), Locally Linear Embedding [17] (LLE) and Supervised Local Maximum Variance Preserving [8] (SLMVP).

Explainable Artificial Intelligence [10] (XAI) aims at understanding machine learning model predictions and explain them in human and understandable terms to build trust with stakeholders. As artificial intelligence becomes more advanced and complex, they often operate as black boxes, making complex computations and generating results without transparently revealing the underlying processes. The objective of XAI methods is to reveal the internal mechanisms and decision-making of the models or the data. There exists machine learning models that provide inherent interpretability, such as decision trees or rule-based models. In addition, XAI methods such as Local Interpretable Model-Agnostic Explanations [15] (LIME) or SHapley Additive exPlanations [14] (SHAP) seek to explain the prediction of black-box models.

Having a good grasp of how the dependent variables explain the independent variable is key to select the features to use in a machine learning pipeline, as well as the number of components to take after a dimensionality reduction technique has been applied. I. T. Jolliffe [13] leverages the explainability that PCA provides, to exclude features from the modeling process using different methods. In this thesis, a recommendation will be given as to which features should remain and how many components should be used for the machine learning task.

The primary objectives of this master's thesis are the following.

1. Study the foundations of SLMVP.
2. Apply SLMVP and other models and find the configuration of their parameters that is best for a classification task.
3. Draw meaningful conclusions from the coefficients of the components that result from applying each of the dimensionality reduction techniques.
4. Compare SLMVP against other state-of-the-art techniques on basis of the interpretation of their coefficients.
5. Apply these techniques to real-world datasets.

In accordance with the goals stated above, this document is structured into five primary sections.

1. Introduction.
2. Related Work: explores existing literature and research in the fields of dimensionality reduction and explainable artificial intelligence.
3. Methodology: explains the experimental setup, interpretation of components, and the process of selecting features and components.
4. Results: shows the results of the evaluation of the dimensionality reduction techniques based on their performance in a classifying task, and explains the components of the discovered subspaces by the best-performing techniques.
5. Conclusion.

Chapter 2

Related Work

This section explores existing literature and research in the fields of dimensionality reduction and explainable artificial intelligence. The most relevant dimensionality reduction techniques are explained, as well as several methods for discarding features that leverage the techniques. Subsequently, explainable artificial intelligence is introduced.

2.1 Dimensionality Reduction

Due to the massive technological advancements in processing power and data storage that have happened in the last few decades, data has grown exponentially in all areas of society. This abundance of data has introduced many challenges, including the curse of dimensionality. The curse of dimensionality refers to the many challenges that arise when working with high dimensional data. [21] Some of those are: increased computational complexity, reduced interpretability, and the risk of overfitting.

Dimensionality reduction techniques have emerged as essential tools to address these issues by transforming high-dimensional data into a lower-dimensional space while preserving variability. Moreover, they facilitate the interpretation of high-dimensional datasets by identifying the key features that contribute most to the variability in the data. This thesis provides an in-depth exploration of many of the most relevant dimensionality reduction techniques (PCA, KPCA, LOL, LPP, LLE and SLMVP).

2.1.1 Principal Component Analysis (PCA)

Principal Component Analysis (PCA) [16] is arguably the most popular and widely used dimensionality reduction technique. It aims at extracting the components that maximize the variance of the data in a lower dimensional space. Introduced by Karl Pearson in 1901, PCA has since been extensively applied in various fields, including signal processing and image processing. It has also been often used in the medical field, where patients datasets usually have a small sample size and a great number of features.

PCA operates by identifying the principal components, which are orthogonal linear combinations of the original variables. These components capture the maximum

variance in the data and are ordered in terms of their significance. The first principal component accounts for the largest proportion of the variance, followed by the subsequent components in descending order.

Mathematically, let X be an $n \times m$ matrix, where each row represents a sample and each column corresponds to a feature. The first step in PCA is to calculate the covariance matrix C, which is computed as $C = \frac{X^T X}{N}$, capturing the relationships between different features.

Next, the eigenvectors and eigenvalues of the covariance matrix are calculated. Denoting the eigenvectors as V and the eigenvalues as λ, we have $CV = \lambda V$. The eigenvectors V represent the directions in the feature space along which the data exhibits the most significant variability, while the corresponding eigenvalues λ indicate the amount of variance explained by each eigenvector.

To reduce the dimensionality of the data, a subset of the eigenvectors corresponding to the largest eigenvalues is selected. These eigenvectors form a new basis for the transformed data. The original dataset is then projected onto this reduced basis, resulting in a lower-dimensional representation, $P = V'X$, where V' contains the selected eigenvectors.

Despite its widespread use, PCA has certain limitations. It assumes a linear relationship between variables, and therefore struggles with cases in which the data clusters cannot be linearly separated. Moreover, it performs a global analysis on the whole dataset and does not preserve the local structure. Extensions of PCA, such as kernel PCA, have been proposed to handle nonlinear relationships in the data. Other methods based on local structure preservation, such as ISOMAP, LLP, Laplacian Eigenmaps and Locally Linear Embeddings have been proposed to overcome the global characteristics of PCA.

2.1.2 Kernel Principal Component Analysis (KPCA)

Kernel Principal Component Analysis (KPCA) [18] is an extension of Principal Component Analysis (PCA) that addresses the limitations of linear methods by introducing a nonlinear mapping of the data into a higher-dimensional feature space. KPCA, proposed by Bernhard Schölkopf, Alexander Smola, and Klaus-Robert Müller in 1998, has since gained significant popularity in various domains, including computer vision, bioinformatics, and signal processing.

KPCA aims to find a low-dimensional representation of the data while capturing the nonlinear structures and relationships present in the dataset. It achieves this by taking advantage of the kernel trick. The kernel trick involves defining a kernel function $K(x_i, x_j)$ that measures the similarity between two data points x_i and x_j. Commonly used kernel functions include the Gaussian (RBF) kernel, polynomial kernel, and sigmoid kernel. By employing the kernel function, KPCA implicitly maps the data into a higher-dimensional feature space, which has the disadvantage that it increases the computational cost.

Let $\phi : \mathbb{R}^N \to F$ be a (nonlinear) map. We refer to F as the feature space. The covariance matrix is then

$$C = \frac{\phi(X)^T \phi(X)}{N}, \tag{2.1}$$

where N is the number of observations.

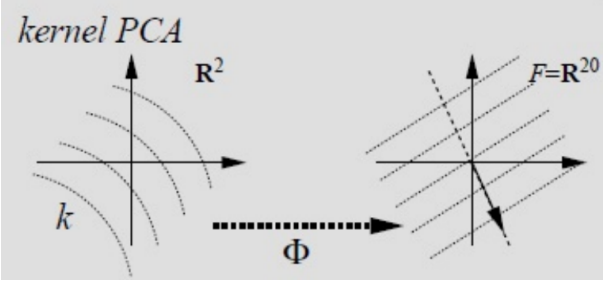


Figure 2.1: Representation of the mapping function used in KPCA. [18]

Similar to PCA, the eigenvectors and eigenvalues of the covariance matrix are calculated. Denoting the eigenvectors as V and the eigenvalues as λ, we have $CV = \lambda V$. The eigenvectors V.

To obtain the reduced-dimensional representation, a subset of the eigenvectors corresponding to the largest eigenvalues is selected. These eigenvectors, referred to as kernel principal components, define the transformed space in which the data is projected. The original dataset can then be mapped onto this space to obtain the lower-dimensional representation $P = V'\phi(X)$.

KPCA offers several advantages over linear PCA. It can effectively capture nonlinear patterns and structures in the data, enabling a more accurate representation of complex relationships. However, KPCA also comes with certain considerations. The choice of the kernel function and its associated parameters significantly impact the performance. Moreover, the computation of the kernel matrix can be computationally demanding because of the mapping to a higher-dimensional feature space.

2.1.3 Linear Optimal Low-Rank (LOL)

Linear Optimal Low-Rank (LOL) [22] is a dimensionality reduction technique that extends PCA by incorporating class-conditional means. This approach outperforms existing dimensionality reduction techniques and has demonstrated effectiveness with imaging and genetics data. LOL and SLMVP represent few of the supervised techniques that have been developed.

LOL was developed with the objective of enhancing the performance and accuracy of LDA, especially when the input data has a very large number of features (e.g. hundreds of millions), and a small sample size. This is known as the "large p, small n" problem, and it negatively affects the performance of classifiers, often making them overfit.

Mathematically, let $A \in \mathbb{R}^{d\times p}$ be a projection matrix i.e., the matrix that projects p-dimensional data into a d-dimensional subspace. We want to find the best projection matrix to pre-process the data before applying LDA. LOL suggests a matrix A_{LOL} that is built by considering the eigenvectors of the class-conditionally centered covariance.

The fist step is to compute the sample mean of each class. They are ordered from highest to lowest, and then the differences of the means are calculated. For two classes: $\delta = \mu_0 - \mu_1$. For more than two classes: $\delta_i = \mu_i - \mu_1$ where $i \in 2, ..., C$ and C is the number of classes. The class-centered covariance matrix is then calculated and the eigenvectors computed.

LOL offers several advantages over other dimensionality reduction techniques, its focus on improving classification accuracy when paired with a machine learning algorithm make it one of the best performing in that area.

2.1.4 Locality Preserving Projection (LPP)

Locality Preserving Projection (LPP) [11] is a dimensionality reduction technique that aims to preserve the local structure and relationships among data points in a lower-dimensional space. LPP. In was introduced by Xiaofei He and Partha Niyogi in 2003.

It builds a graph incorporating neighborhood information of the data set, that we will refer to as similarity matrix W . Using the notion of the Laplacian of the graph, a transformation matrix which maps the data points to a subspace is computed. This linear transformation optimally preserves local neighborhood information.

Mathematically, let $\mathbf{X}$ be an $n \times m$ data matrix, where each row corresponds to a sample and each column represents a feature. LPP involves two key steps: constructing an similarity matrix W and solving a generalized eigenvalue problem.

The first step is to construct a similarity matrix W that encodes the pairwise similarities between data points. Commonly used similarity measures include the Gaussian kernel, k-nearest neighbors, or graph-based techniques. The matrix W captures the local relationships and can be interpreted as a weighted adjacency matrix of a graph, where each data point is connected to its neighbors.

Given W, the goal is to find a projection matrix $\mathbf{P}$ that maps the data points into a lower-dimensional space. This is achieved by solving the generalized eigenvalue problem:

$$X\mathbf{L}X^T a = \lambda X D X^T a, \tag{2.2}$$

where $\mathbf{L} = D - W$ is the Laplacian matrix. The matrix $\mathbf{P}$ consists of the eigenvectors corresponding to the smallest eigenvalues of equation 2.2, representing the lower-dimensional representation of the data.

LPP offers several advantages over traditional dimensionality reduction techniques. It yields a map which is simple, linear, and defined everywhere. Furthermore, the algorithm can be easily kernelized and be made nonlinear.

However, LPP also has some considerations. The choice of the parameters when computing the similarity matrix (e.g. the number of neighbors k if k-neares neighbors is used) can significantly change the result of applying LPP.

2.1.5 Locally Linear Embedding (LLE)

Locally Linear Embedding (LLE) [17] is a powerful nonlinear dimensionality reduction technique that aims to preserve the local structure of the data in a lower-dimensional space. LLE, introduced by Sam T. Roweis and Lawrence K. Saul in 2000, has gained significant attention in various fields, including computer vision, data visualization, and manifold learning.

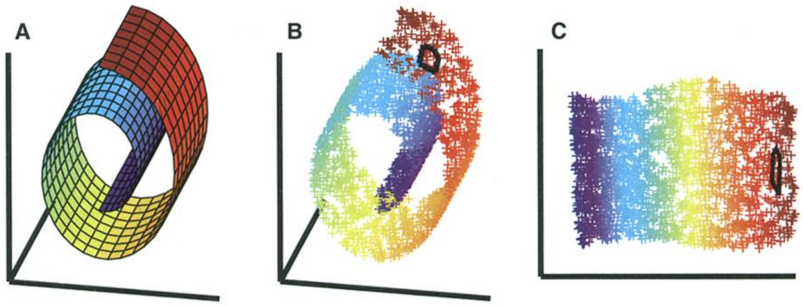


Figure 2.2: Illustration of the problem of nonlinear dimensionality reduction.

LLE addresses the limitations of linear projection methods by assuming that the data lies on a low-dimensional manifold that can be "unfolded". It seeks to find a representation of the data in a lower-dimensional space while preserving the local relationships between neighboring data points. By doing so, LLE is capable of capturing the underlying nonlinear structure of the data.

Mathematically, let X be an $n \times m$ data matrix, where each row corresponds to a sample and each column represents a feature. LLE involves three key steps: local reconstruction, weight determination, and global embedding.

The first step is local reconstruction, where for each data point x_i, the neighboring data points are identified. The goal is to reconstruct x_i as a linear combination of its neighbors. This can be achieved by using an algorithm, such as the K-nearest neighbors.

The weight determination step involves solving the optimization problem to find the optimal weights. This can be done by minimizing the residual sum of squares.

$$RSS(W) = \sum_{i=1}^{n} |X_i - \sum_{j \neq i} w_{ij} X_j|^2 \tag{2.3}$$

It holds $\sum_j W_{ij} = 1$, and $W_{ij} = 0$ if X_j is not part of the neighborhood of X_i.

Finally, the global embedding step aims to find the low-dimensional representation of the data. This is achieved by, once again, minimizing the residual sum of squares but using the new weights to find the low-dimensional representation of the data points Y.

$$\Phi(Y) = \sum_{i=1}^{n} |y_i - \sum_{j \neq i} w_{ij} Y_j|^2, \tag{2.4}$$

where $\sum_i Y_{ij} = 0$, so that the data points are centered on the origin and subject to $YY^T = I$

The previous equation can me rewritten in its matrix multiplication form as

$$\begin{aligned}\sum_{i=1}^{n}|y_i-\sum_{j\neq i}w_{ij}Y_j|^2 &= \sum_i y_i^2 - y_i(\sum_j w_{ij}y_j) - (\sum_j w_{ij}y_j)y_i + (\sum_j w_{ij}y_j)^2 \\ &= Y^TT - Y^T(WY) - (WY)^TY + (WY)^T(WY) \\ &= Y^T(I-W)^T(I-W)Y \\ &= Y^TMY,\end{aligned} \tag{2.5}$$

and after applying Lagrange multipliers, the minimization obtained by solving the following eigenvalue problem.

$$MY = \alpha Y \tag{2.6}$$

The projection matrix **P** consists of the eigenvectors corresponding to the smallest eigenvalues of equation 2.6, representing the lower-dimensional representation of the data.

LLE offers several advantages over linear projection methods. By preserving the local relationships, LLE can effectively capture the nonlinear structure of the data. It allows for the reduction of dimensionality while retaining the intrinsic properties and relationships of the data, enabling improved analysis and visualization.

However, LLE also has some considerations. The choice of the number of neighbors and the neighborhood size significantly impact the performance of LLE.

2.1.6 Supervised Local Variance Maximum Preserving (SLMVP)

Supervised Local Variance Maximum Preserving (SLMVP) [8][9], proposed by García-Cuesta is a dimensionality reduction technique that preserves the maximum local variance, and considers the distribution of the data by the response variable. SLMVP solves the "large p, small m" problem.

Mathematically, let $x_1, x_2, ..., x_m \in \Re^p$ be a set of inputs and $y_1, y_2, ..., y_m \in \Re^l$, where m denotes the sample size, and p and l the number of input and output features respectively. SLMVP involves two key steps: constructing a similarity graph, and the unsupervised dimensionality reduction problem.

The first step, consists in the application of the similarity function $\mathscr{S}(X) : X \in \Re^{m\times p}$ to the inputs, and $\mathscr{S}(Y) : Y \in \Re^{m\times l}$ to the outputs. The application of these similarity functions defines an input weighted graph $\{H, U\}$ and an output weighted graph $\{I, V\}$, where H and I are nodes, and U and V vertices. The weights of the links represent the similarity between two data points. This allows the dimensionality reduction technique with the capability of being local.

The second step consist in solving the unsupervised dimensionality reduction problem, which aims to choose the mapping $y'_i = A^Tx_i : y'_i \in \Re^k$ that minimizes the distance between neighbors in a multidimensional space. This can be expressed by the following cost function:

$$J_{ns} = \frac{1}{2}\sum_{ij}||y'_i - y'_j||w_{ij}, \tag{2.7}$$

where w_{ij} are the elements of matrix $W \in \Re^{m \times m}$, which is the similarity matrix $\mathscr{S}(X)$. SLMVP solves the unsupervised dimensionality reduction problem by choosing the mapping that minimizes the distance between neighbors, preserving only those that are shared in the input and output spaces. The cost function 2.7 is then extended to be

$$J_{ns} = \frac{1}{2} \sum_{ij} ||y'_i - y'_j|| z_{ij}, \tag{2.8}$$

where z_{ij} are the elements of matrix $Z \in \Re^{m \times m}$, which represents the joint similarity matrix between input and output similarity matrices, being $z_{ij} = \sum_{k=1}^{m} u_{ik} v_{kj}$. This minimization can also be expressed in its kernelization form as the following maximization problem:

$$\max tr(Y^T K_X K_Y Y) \tag{2.9}$$

where $K_X = \mathscr{S}_X(X)$ and $K_Y \mathscr{S}_Y(Y)$ are the input and output similarity graphs represented as kernel functions. The maximization of the trace can be solved as the following eigenvector problem:

$$X K_X K_Y X^T B = \lambda B \tag{2.10}$$

where B is the new transformed space.

2.2 Discarding Features

Rejecting variables in machine learning involves carefully evaluating and deciding which features to exclude from the modeling process. I.T. Jolliffe [13] examined some of the possible methods for deciding which variables to reject using principal component analysis. He introduced various methods, among which B1, B2, and B4 stand out as particularly relevant and significant.

- **B1**. First, principal component analysis is performed on the original K features. Then, the p_1 worse eigenvectors with eigenvalues less than a threshold λ_0 are considered. The features that correspond to the highest coefficient of these eigenvectors are then rejected, remaining $K - p_1$ features. This process is then repeated and the feature set is further reduced until all the computed eigenvalues are less than the threshold λ_0. At the end, $K - p_1 - p_2 - ...p_l = p$ variables will remain. The number of rejected features will depend on the choice of λ_0. B1 is very slow, as PCA has to be performed at every iteration.
- **B2**. B2 is the same as B1 except that only one iteration is done.
- **B4**. B4 can be thought of as a backward version of B1. Here, the $K - p_1$ best eigenvectors with the eigenvalues greater than the threshold λ_0 are considered, and the features that correspond to the highest coefficient of these eigenvectors remain, rejecting the rest.

I.T. Jolliffe tests the methods on artificially-generated data with redundant features. He shows that several of the rejection methods discard precisely the variables known to be redundant, for all but a few sets of data.

2.3 Explainable Artificial Intelligence

As machine learning models, such as neural networks, become more sophisticated, they often operate as black boxes, making complex computations and generating results without transparently revealing the underlying processes to human users. This lack of explainability poses significant concerns in several domains. For instance, in healthcare, where AI is increasingly utilized for medical diagnoses or treatment recommendations, doctors and patients need to understand the rationale behind AI-driven decisions to ensure trust and accountability. To address this issue, Explainable Artificial Intelligence [10] (XAI) focuses on understanding machine learning model predictions and explaining them in human and understandable terms to build trust with stakeholders.

In artificial intelligence, interpretability and explainability are two related but distinct concepts that focus on understanding and providing insights into the decision-making process of models. Interpretability focuses on model understanding techniques, while explainability focuses more broadly on model explanation and the interface for translating these explanations in human, understandable terms.

XAI overcomes several challenges that increasingly complex models face, such as debugging, understanding, and controlling them effectively. XAI seeks to overcome these issues by providing transparency, trustworthiness, and facilitating error detection. Furthermore, XAI is very relevant in the detection and mitigation of biases, and the compliance with regulators and ethical standards.

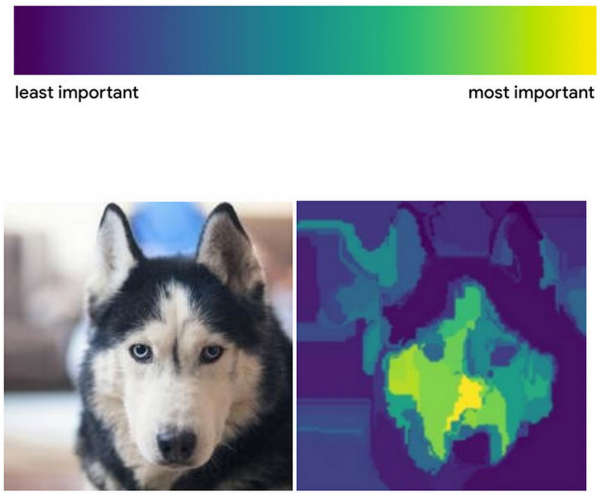


Figure 2.3: Example of explainable artificial intelligence in image recognition. It shows which regions contributed to the model's prediction of this image as a husky. [4]

The field of Explainable Artificial Intelligence (XAI) is continuously evolving, and several state-of-the-art techniques have been developed to enhance the explainability of AI models. Here are a few notable advancements:

- **Local Interpretable Model-Agnostic Explanations [15] (LIME)**: LIME is a model-agnostic method that explains the predictions of any black-box model by creating a surrogate interpretable model. It modifies the input data and observes the effect on the model's output, generating explanations for specific instances.
- **SHapley Additive exPlanations [14] (SHAP)**: SHAP values are based on cooperative game theory and aim to allocate the "credit" or contribution of each feature in a prediction. SHAP provides a unified framework for interpreting the output of a wide range of models, including deep neural networks.
- **Integrated Gradients [19]**: This method assigns importance scores to input features by integrating gradients along a path from a baseline input to the input of interest. It quantifies how each feature contributes to the prediction and has been applied to explain deep learning models.

It is important to note that the effectiveness of these techniques may vary depending on the application domain, the specific AI model used, and the requirements of the end-users. The field of XAI is still rapidly evolving, and researchers are continuously working on developing more sophisticated and reliable methods to enhance the transparency and interpretability of AI systems.

Chapter 3

Methodology

This section explains the experimental setup, interpretation of components, and the process of selecting features and components.

3.1 Interpretation of Components

We will provide an explanation of principal components in two dimensions. First, we will determine their relationship with the original features of the data. Second, we will examine how the components obtained through one technique relate to those obtained through another technique. The methods behind how these questions are addressed will be explained in detail to provide a comprehensive understanding.

3.1.1 Explaining Components

Before diving into the step-by-step process, it is essential to define some variables that are present in dimensionality reduction tasks: let the input data X be an $n \times p$ matrix, where each row represents a sample and each column, denoted as $X_1, X_2, ..., X_p$, corresponds to a feature. Furthermore, let $V \in \mathbb{R}^{d \times p}$ be the projection matrix, i.e., the matrix that projects p-dimensional data into a d-dimensional subspace and let V_i represent the i-th column of V sorted in descending order by their significance (e.g., by the magnitude of their eigenvalues or the variability they capture). Finally let $P \in \mathbb{R}^{n \times d}$ be the the projection of the data into the d-dimensional space, so that $P = XV$ holds, with columns $P_1, P_2, ..., P_d$.

The first step is to calculate the correlation between the input data represented in its original features $X_1, X_2, ..., X_p$ and the data projected onto the reduced d-dimensions $P_1, P_2, ..., P_d$. Let r_i be the vector that contains the correlations between the input data X and its projection onto the i-th dimension P_i.

$$r_i = \begin{bmatrix} corr(P_i, X_1) \\ corr(P_i, X_2) \\ ... \\ corr(P_i, X_p) \end{bmatrix}$$

This vector of correlations will be utilized to explain a component based on the original features. We could now make statements such as: "component 1 is positively cor-

related with feature X_1, negatively correlated with X_3 and does not show any other strong correlations", and draw insights from how the data is distributed along this the axis that this component represents.

3.1.2 Comparing Techniques

We can also leverage the correlations between the dimensions generated by the technique and the original features to compare the similarity and dissimilarity of the components obtained applying different techniques. Correlation can be negative or positive. However, it is the absolute value what determines the degree of significance of the correlation. Therefore, we take the absolute values $|corr(P_i, X_j)|$. Moreover, the difference between the value of correlations is not uniform among the different dimensionality reduction techniques and it prevents us from doing a fair comparison. To address that, we take the position of the absolute correlation in a list that is sorted in ascending order. Let $index(j)$ be the index such that $\#\{i : x_i \leq x_j\} = \lfloor j \rfloor$ for $i, j \in \{1, 2, ..., p\}$.

$$s_i = \begin{bmatrix} index(|corr(P_i, X_1)|) \\ index(|corr(P_i, X_2)|) \\ ... \\ index(|corr(P_i, X_p)|) \end{bmatrix}$$

For example, suppose we have some input data with 3 features X_1, X_2 and X_3 and we would like to explain the first reduced dimension P_1. Suppose also the correlations $corr(P_1, X_1)$, $corr(P_1, X_2)$ and $corr(P_1, X_3)$ are $-0.9, 0.3$ and 0.7 respectively. Vector r_1 would written as

$$r_1 = \begin{bmatrix} -0.9 \\ 0.3 \\ 0.7 \end{bmatrix},$$

and vector s_1 as

$$s_1 = \begin{bmatrix} 3 \\ 1 \\ 2 \end{bmatrix}.$$

Vector s_i contains the information about the significance that the i-th component assigns to each of the original features. The position in the vector of the element of s_i with the highest number indicates the feature that s_i gives more importance to. In the example above, X_1 is the feature that s_1 gives more importance to because the first position of s_1 has the highest number, 3. Subsequently, X_3 would be the second most important, and X_2 the least important. This allows us to compare and determine the similarity or dissimilarity of two components using the *Spearman's rank correlation coefficient* of the absolute correlations. We do so by simply taking the correlation of vectors s_i and s_j, there is no need for calculating the ranks, as that was done previously.

Following the previous example, suppose s_1 corresponds to the first projection of the reduced space discovered by PCA. We will refer to s_1 as $s_{PCA,1}$ from now on. Furthermore, let $s_{SLMVP,1} = [2, 1, 3]$ be the vector that corresponds to the first projection of the

reduced space discovered by $SLMVP$. We compute the *Spearman's rank correlation coefficient* by calculating $corr(s_{PCA,1}, s_{SLMVP,1})$ in the following manner.

$$corr(s_{PCA,1}, s_{SLMVP,1}) = corr\left(\begin{bmatrix}3\\1\\2\end{bmatrix}, \begin{bmatrix}2\\1\\3\end{bmatrix}\right) = 0.5$$

This way we can effectively calculate the similarities and dissimilarities of the first component calculated with the most prominent dimensionality reduction techniques and conclude that, for the dataset used, they are similar or dissimilar.

This can be further extended so that the we can compare all the components calculated with one dimensionality reduction technique with all the components calculated with another dimensionality reduction technique. In order to do that, we calculate a vector s_1^* which corresponds to the average of $s_1, ..., s_p$ weighted by the variability that each component captures. This variability can be thought of as the eigenvalues corresponding the eigenvectors that form the projection matrix V. As I do not have access to the eigenvalues for some of the dimensionality reduction techniques, this is calculated proportionally to the variation of each component. E.g., $Var(s_1)/\sum_i Var(s_i)$ would be the weight corresponding to s_1.

3.2 Choosing Features and Components

There are two big open questions when applying dimensionality reduction in a machine learning pipeline: which features should one retain or discard?, and how many components should one utilize for subsequent modeling? The writer of this thesis gives a recommendation based on the explainability of components. As to the first question, only the features that show an absolute correlation that is higher than a threshold λ should be chosen. Therefore the amount of features that are retained will depend on the choice of this threshold. The reader should be advised that this would work best with dimensionality reduction techniques that are supervised and not only try to capture the variability of the data, but also incorporate the relationship between dependent and independent variables.

The recommendation concerning the amount of components is to make that choice based on a heuristic visual analysis of a line plot of the eigenvalues of principal components, or of the cumulative explained variance, and look for the point where the curve starts to level off, this is sometimes known as the "elbow method".

3.3 Experimental Setup

This section details the selection and preparation of datasets, describes the dimensionality reduction techniques and machine learning algorithms utilized.

3.3.1 Techniques and Models

We want to explain dimensionality reduction techniques with the parameters that fit the data well, and we measure that by the accuracy that they yield when used in a machine learning pipeline. For that purpose, the algorithms are run in different

settings and then machine learning models use the components for the classification task. These machine learning models are, in turn, also tested with different parameters. Table 3.2 shows the parameters tested for the dimensionality reduction techniques.

Model	Parameters
PCA [16]	Default parameters.
SLMVP [8]	Kernel: linear, polynomial of order 5, radial rbf with gamma values 0.1 and 0.01 for both the dependent and the independent variables.
LOL [22]	Default parameters.
KPCA [18]	linear, polynomial of order 5, radial rbf with gamma value 0.1 and $\frac{1}{\text{\# rows}}$.
LPP [11]	Number of neighbors to consider on the adjacency graph: $\lfloor\sqrt{min(\text{\# rows, \# columns})}\rfloor$.
LLE [17]	Number of neighbors to generate the local linear approximation: $\lfloor\sqrt{min(\text{\# rows, \# columns})}\rfloor$. Regularization parameter: 0.001.

Table 3.1: Parameters tested for each of the dimensionality reduction techniques.

Similarly, the machine learning classifiers have been configured on the grid search with the the parameters in Table 3.2.

Model	Parameters
XGBoost [3]	Number of estimators: 5, 10, 20, 50, and 100.
KNN [6]	Number of neighbors: 3, 5, 10, 20, 50, and 100.
SVM [5]	Regularization parameter C: 0.1, 1, and 10. Kernel: linear, radial rbf and polynomial.
Decision Tree	Maximum depth: None, 5, 10, 20, and 50.
Naive Bayes	Default parameters.
Random Forest [12]	Number of estimators: 50, 100, and 200. Maximum depth: None, 5, 10, and 20
LDA [1]	Solver: svd, lsqr, and eigen.
AdaBoost [7]	Number of estimators: 50, 100, and 200. Maximum depth: None, 5, 10, and 20.

Table 3.2: Parameters tested for each of the machine learning models.

3.3.2 Datasets

This section introduces the datasets utilized in this master's thesis. The aim is to provide a comprehensive overview of the datasets, including their sources, characteristics, and any preprocessing steps applied.

Model	Library
XGBoost	xgboost.XGBClassifier
KNN	sklearn.neighbors.KNeighborsClassifier
SVM	sklearn.svm.SVC
Decision Tree	sklearn.tree.DecisionTreeClassifier
Naive Bayes	sklearn.naive_bayes.GaussianNB
Random Forest	sklearn.ensemble.RandomForestClassifier
LDA	sklearn.discriminant_analysis.LinearDiscriminantAnalysis
AdaBoost	sklearn.ensemble.AdaBoostClassifier

Table 3.3: Python libraries containing the implementation of the machine learning algorithms utilized.

Model	Library
PCA	sklearn.decomposition.PCA
KPCA	sklearn.decomposition.KernelPCA
LOL	lol.LOL
LPP	lpproj.LocalityPreservingProjection
LLE	sklearn.manifold.LocallyLinearEmbedding

Table 3.4: Python libraries containing the implementation of the dimensionality reduction techniques utilized.

3.3.2.1 Artificial Datasets

Sklearn's make_blobs was utilized to create a single-label dataset. It created a multiclass datasets by allocating each class one or more normally-distributed clusters of points. The dataset was of sample size 1000, 300 features and 20 classes, with a cluster standard deviation of 2. The high number of features and low sample size is perfect for the "large p, small n" problem that dimensionality reduction tries to address. Sklearn's make_multilabel_classification was utilized to create a multilabel dataset.

3.3.2.2 FIFA

The FIFA dataset encompasses data from various editions of the popular FIFA video game series, which serves as a reliable source of player information. The dataset contains detailed attributes for each player, including playing positions, skill ratings (such as dribbling, shooting, and passing), and performance statistics (such as goals, assists, and appearances).

Furthermore, it includes the estimated market value of each player, which has often been used as the dependent variable in a regression or classification task.

3.3.2.3 Our Database of Faces (ORL) Dataset

This dataset [2] was created at the AT&T Laboratories in Cambridge, UK, in the context of a face recognition project the laboratory was doing with the Speech, Vision and Robotics Group of the Cambridge University Engineering Department. It con-

Figure 3.1: Our Database of Faces (ORL) Dataset

tains face images taken between April 1992 and April 1994, 10 images of 40 different subjects, a total of 400 images. It consists of grayscale images of 40 individuals, with 10 images per person. The dataset was collected under controlled conditions, with variations in facial expression, lighting conditions, and facial details.

Each image in the ORL dataset has a resolution of 92 pixels by 112 pixels. The images were captured under different poses, including variations in head rotation and tilt, providing a diverse set of facial orientations. The ORL dataset offers a realistic representation of face images encountered in real-world scenarios, making it a suitable choice for evaluating the performance of face recognition algorithms. The dataset has been extensively used for training and testing various dimensionality reduction techniques.

3.3.2.4 COIL2000

This dataset used in the CoIL 2000 Challenge contains information on customers of an insurance company.[20] The data consists of 86 variables and includes product usage data and socio-demographic data. The data was supplied by the Dutch data mining company Sentient Machine Research and is based on a real world business problem. The training set contains over 5000 descriptions of customers, including the information of whether or not they have a caravan insurance policy. A test set contains 4000 customers of whom only the organizers know if they have a caravan insurance policy.

The dataset had class imbalance, as a large majority of customers have zero claims, only 348 customers had a claim. Sklearn's *RandomOverSampler* was used to balance it.

Chapter 4

Results and Discussion

Dimensionality reduction techniques prove particularly valuable when confronted with datasets with a high number of features. However, it is very challenging to explain and interpret components that are based on a multitude of features. Here, as in many areas of machine learning, there exists a trade-off between complexity and explainability. We try to navigate this trade-off by grouping features into categories when the datasets permits, as it happens in Section 4.2.

This sections shows the results of the evaluation of the dimensionality reduction techniques based on their performance in a classifying task, and proceeds to visually show and explain the components of the discovered subspaces by the best-performing techniques.

4.1 Artificial Dataset

4.1.1 Classification Results

In this section, the performance of various dimensionality reduction techniques is evaluated and compared on an artificially generated dataset. The aim is to identify the techniques that provide the best results in reducing the dimensionality of the dataset. Table 4.1 presents the results obtained from evaluating different dimensionality reduction techniques. Each technique is assessed based on accuracy. The techniques are ranked in descending order of overall performance.

For 2, 3, 4, and 5 dimensions, SLMVP, PCA and KPCA demonstrate the best performance among the tested dimensionality reduction techniques. They achieve an accuracy of 96% in 2 dimensions and are able to correctly classify all the samples with 3 dimensions or more. Both SLMVP and KPCA achieve these results in their linear configurations, paired with the LDA classifier or K-Nearest Neighbors. However, LLE emerges as the winner when only 1 dimension is allowed in the classification task. It achieves a 90% accuracy paired with the XGBoost classifier, much higher than the second-best performing, which is KPCA with a 63% accuracy.

Overall, SLMVP, KPCA and PCA demonstrate to be the most promising dimensionality reduction technique, especially when the classification task is not restricted to just 1 dimension. However, it is interesting to see how the results would change when applying the dimensionality reduction techniques to a real-world dataset.

Dimensions	Dim. Technique	Dim. Params	Model	Best Score
1	LLE	k=30-reg=0.001	XGBoost	0.90
	KPCA	Linear	XGBoost	0.63
	PCA		KNN	0.57
2	SLMVP	Linear	LDA	0.96
	PCA		Random Forest	0.96
	KPCA	Linear	KNN	0.96
3	SLMVP	Linear	LDA	1.00
	LLE	k=30-reg=0.001	LDA	1.00
	PCA		LDA	1.00
4	SLMVP	Polynomial-Order=5	LDA	1.00
	LLE	k=30-reg=0.001	KNN	1.00
	LPP	k=17	LDA	1.00
5	SLMVP	Linear	LDA	1.00
	LLE	k=30-reg=0.001	KNN	1.00
	LPP	k=17	Random Forest	1.00

Table 4.1: Performance Comparison of Dimensionality Reduction Techniques on an artificially generated dataset. This dataset contains 20 centers, 300 features and sample size 1000. (See Annex Table 1) for full results.

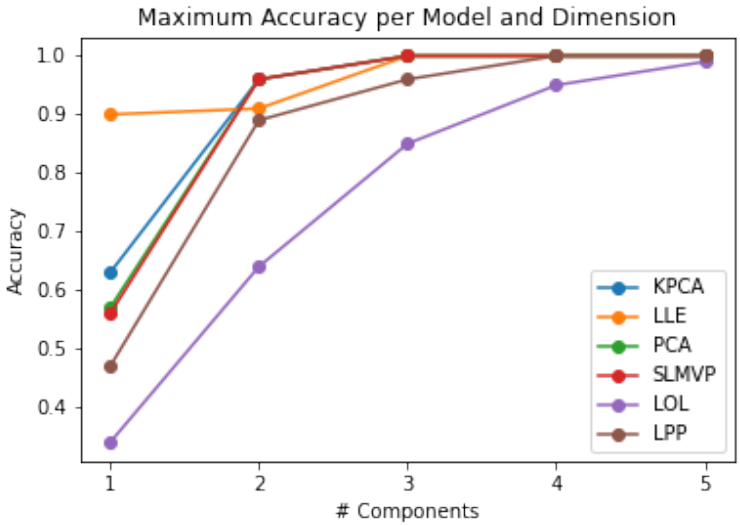


Figure 4.1: This figure shows how the accuracy of the best-performing model for each dimensionality reduction technique, evolves as the number of dimensions is increased.

4.1.2 Explanation

In this section, the results of applying various dimensionality reduction techniques are presented and compared using visual plots. The objective is to analyze and contrast the performance of these techniques in reducing the dimensionality of the dataset.

Figures 4.2 and 4.3 display a scatter plot of the dataset after applying the techniques. The plot shows the reduced-dimensional representation of the data, where the axes represent the principal components. By examining the scatter plot, we can observe the clustering and distribution of the data points in the reduced space.

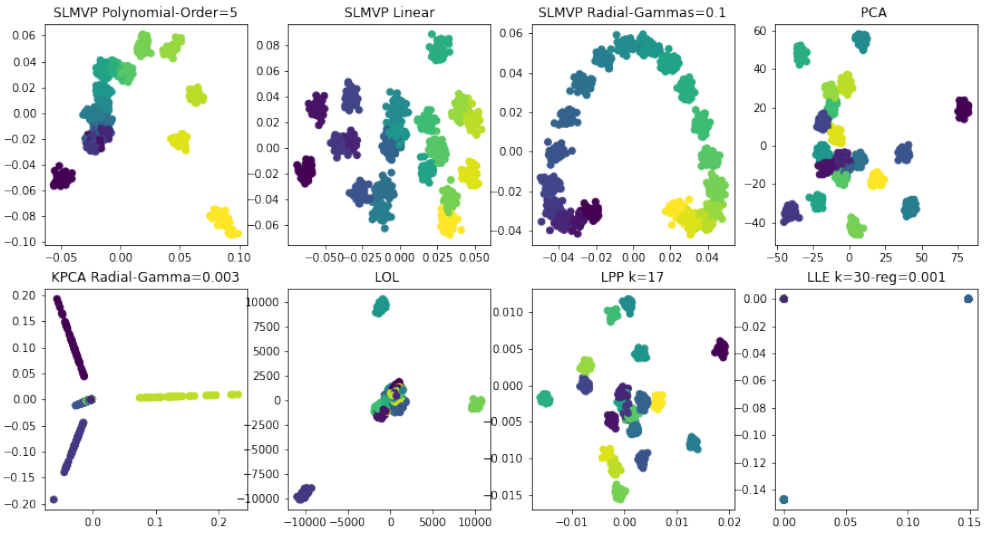


Figure 4.2: Visualization of the clusters projected into 2 dimensions that resulted from the application of the different dimensionality reduction techniques on the artificially-generated data.

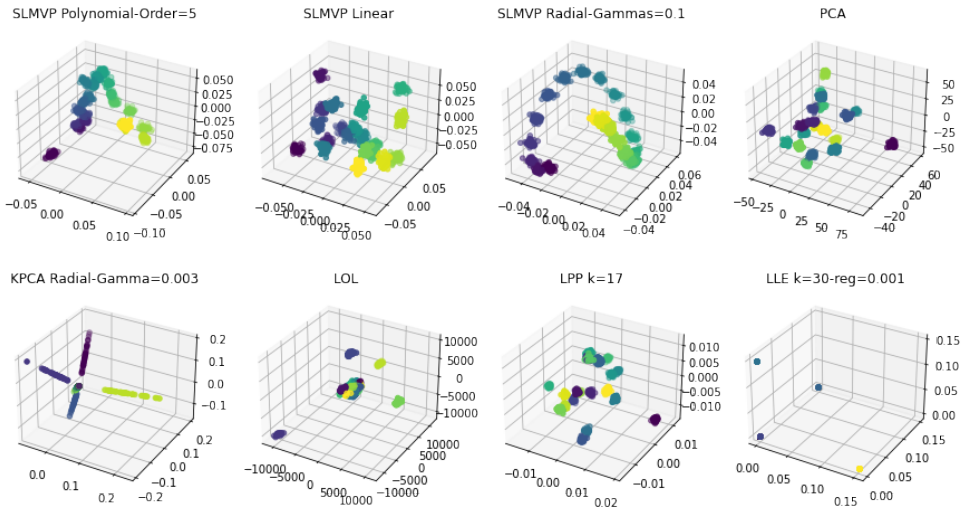


Figure 4.3: Visualization of the clusters projected into 3 dimensions that resulted from the application of the different dimensionality reduction techniques on the artificially-generated data.

By examining the scatter plots obtained from the different dimensionality reduction techniques, we can compare their performance in capturing and preserving the structure of the original dataset. The following observations can be made:

- SLMVP with a linear kernel, PCA and LPP demonstrate a relatively even distribution of data points in the reduced space, indicating that it effectively captures the overall variance of the dataset. However, it may not be able to capture complex nonlinear relationships.
- SLMVP and LOL exhibit a clear separation between different classes in the reduced space, even though this is not clearly visible in the LOL plot, since a few clusters lay very far away from where most of them are located. These are clearly separated although they appear to be overlapping. The notable class separation in the plots corresponding to SLMVP and LOL emphasizes the discriminative power between classes, making it useful for classification tasks.
- SLMVP with a radial kernel captures the classes in a curve, which can be clearly seen in its 3-dimensional representation (Figure 4.3).

4.1.2.1 Comparing Techniques

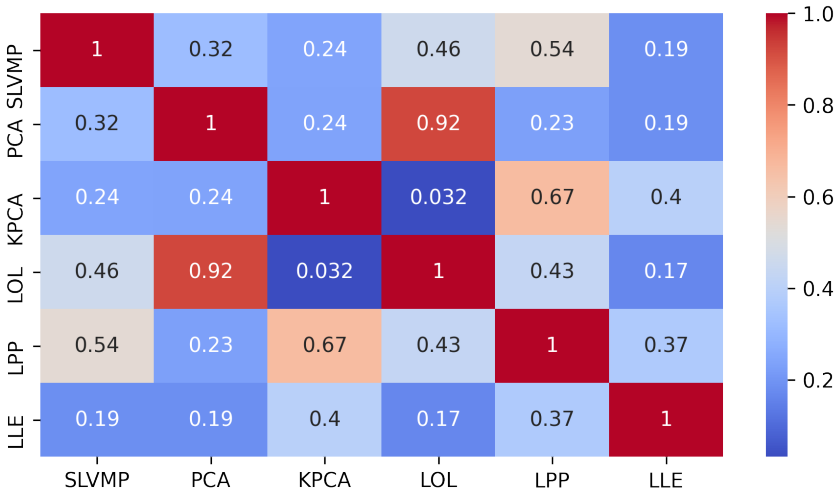


Figure 4.4: Heatmap of Spearman Rank Correlation Coefficients

The heatmap in Figure 4.4 visually represents the similarity and dissimilarity between different dimensionality reduction techniques based on their Spearman rank correlation coefficients. The correlation coefficients are calculated by measuring the absolute correlations between the data in the reduced space from each technique and the original data. Higher correlation coefficients indicate a stronger resemblance between the reduced-dimensional representations obtained using 2 dimensionality reduction techniques. The color and the color intensity in the heatmap represent the magnitude of the correlation coefficients. Darker red shades indicate higher correlation coefficients, indicating a stronger resemblance between the reduced-dimensional representations. Darker blue shades, on the other hand, suggest weaker correlations. We can make the following observations.

- SLMVP is most strongly-correlated with LPP and LOL, although this correlation is only moderate, below 60%. The correlation with LPP could be due to the

fact that both SLMVP and LPP are local and therefore incorporate neighborhood information of the dataset. On the other hand, the correlation with LOL could be due to another property that both SLMVP and LOL posses, the fact that both are supervised and therefore incorporate information about the dependent variable.

- The strongest correlation over all is between PCA and LOL with 92%, and the lowest between LLE and LOL with 10%.

4.1.2.2 Multilabel

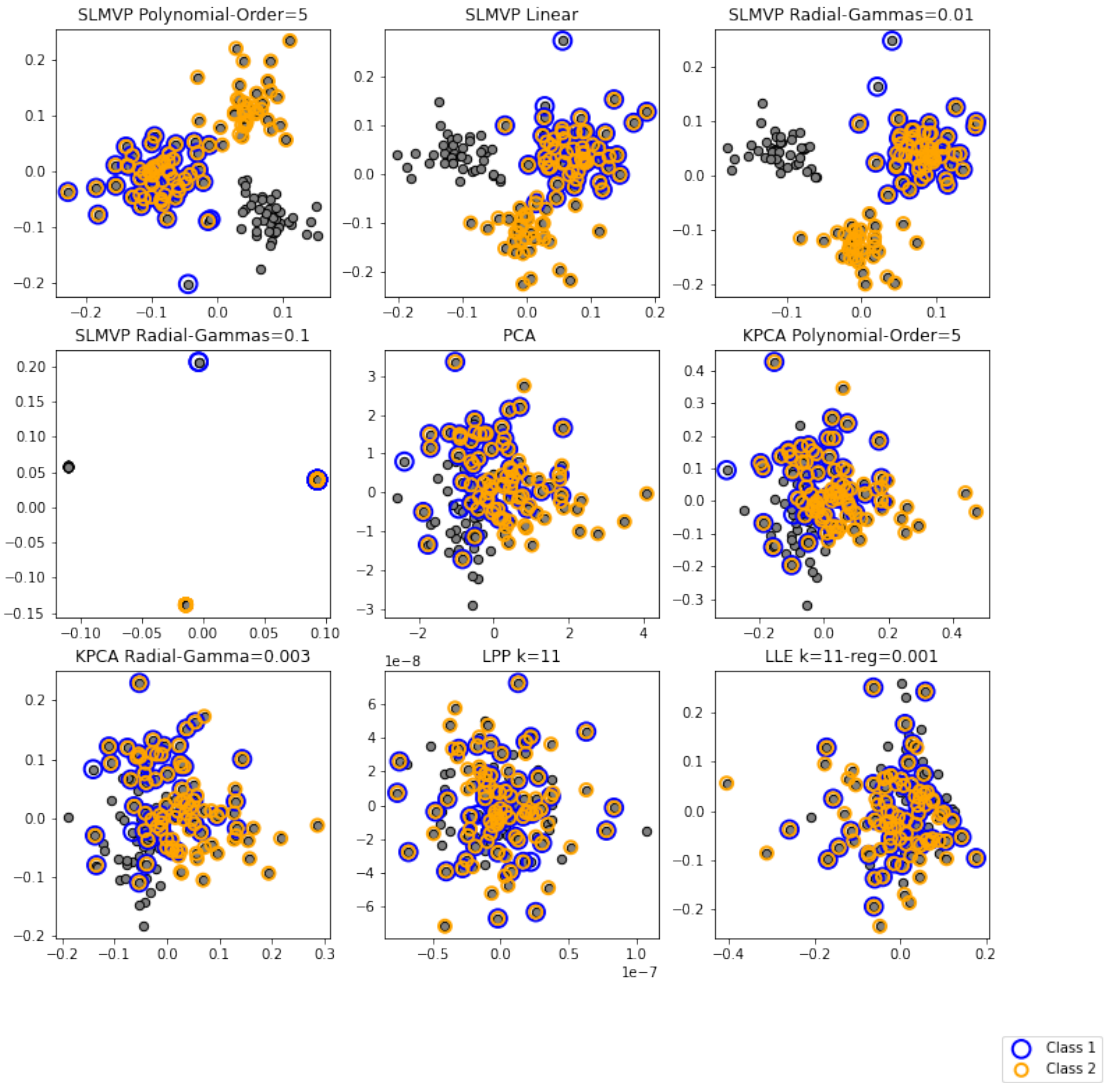


Figure 4.5: Visualization of the clusters projected into 2 dimensions that resulted from the application of the different dimensionality reduction techniques on artificially-generated multilabel data.

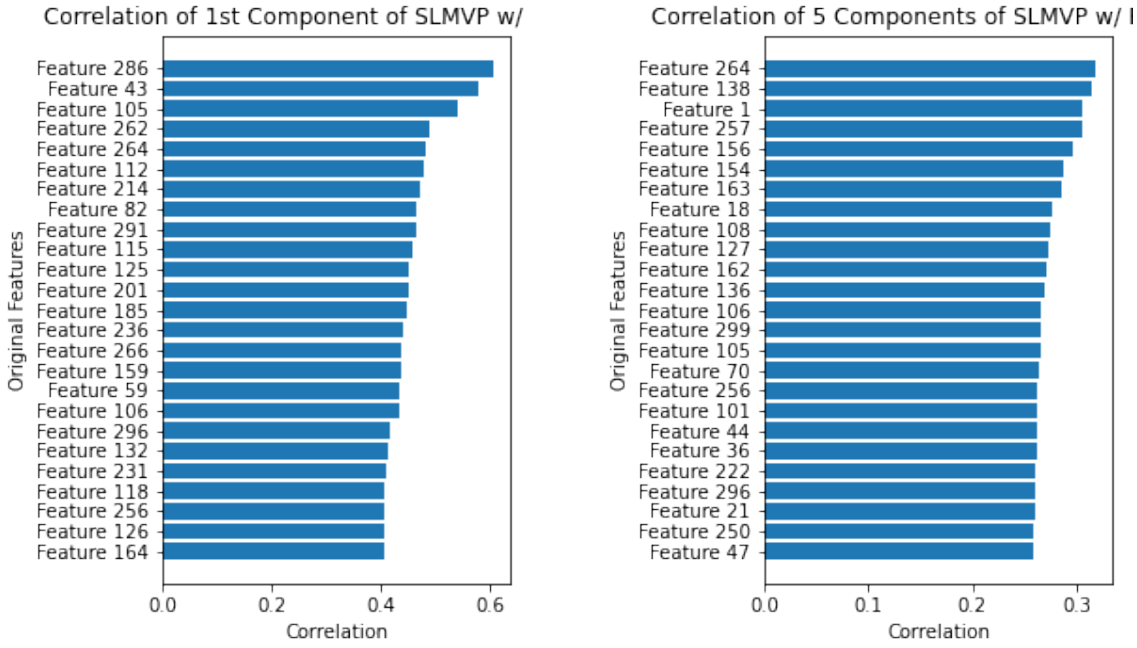
Among the supervised dimensionality reduction techniques evaluated, SLMVP stands out as the sole method capable of effectively handling multilabel datasets. This char-

acteristic holds great significance due to the important role that information about the relationships between multiple classes and variables plays in preserving the intrinsic structure of the data during dimensionality reduction. By leveraging SLMVP's unique ability to capture and retain these intricate connections, it ensures a more accurate and comprehensive representation of the data, thereby enhancing the effectiveness of dimensionality reduction tasks.

Figure 4.5 shows the reduced-dimensional representation of a 2-label dataset. It is worth noting that LOL is not capable of handling multilabel data and is therefore not on the plot. The figure shows that only SLMVP is capable of clearly separating the datapoints into their different four combinations of classes.

4.1.2.3 Choosing Features and Components

We can leverage the calculated absolute correlations to select the features that should be kept for a posterior machine learning task. Figure 4.6 shows the 20 features with the highest absolute correlations with the first component, in the case of the first subplot, and the average of the first 5 components, in the case of the second subplot. For the 1 component, the first three features stand out with correlations higher than 50%. For this artificially-generated data, the correlations follow a smooth curve and it is difficult to determine how many features to reject. Thresholds of 0.2 and 0.15 for the first and second subplots respectively are reasonable.



(a) 1 Component

(b) 5 Components

Figure 4.6: Original features of the artificially-generated dataset ordered by their correlation to the components obtained with SLMVP with radial kernel and gamma values 0.1.

4.2 FIFA Dataset

4.2.1 Classification Results

We assess and compare different techniques for reducing the dimensionality of the FIFA dataset introduced in section 3.3.2.2. We present the evaluation results in Table 4.2, which ranks the techniques based on their accuracy. SLMVP is the clear winner for this dataset. It achieves a higher accuracy than the other techniques for all the numbers of dimensions tested in its radial kernel configuration.

Dimensions	Dim. Technique	Dim. Params	Model	Accuracy
1	SLMVP	Linear	KNN	0.86
	LOL		Naive Bayes	0.85
	KPCA	Linear	LDA	0.83
2	SLMVP	Radial-Gammas=0.1	XGBoost	0.96
	LPP	k=5	Random Forest	0.90
	LOL		KNN	0.89
3	SLMVP	Radial-Gammas=0.1	XGBoost	0.96
	KPCA	Linear	SVM	0.93
	LPP	k=5	SVM	0.93
4	SLMVP	Radial-Gammas=0.01	Random Forest	0.96
	LPP	k=5	XGBoost	0.94
	KPCA	Linear	SVM	0.90
5	SLMVP	Radial-Gammas=0.1	SVM	0.95
	LLE	k=30-reg=0.001	SVM	0.94
	LOL		XGBoost	0.91

Table 4.2: Performance Comparison of Dimensionality Reduction Techniques on the FIFA dataset. (See Annex Table 2) for full results.

4.2.2 Explanation

By looking at the scatter plots created by the different techniques in figure 4.8 and 4.9, we can compare how well they perform in capturing and preserving the structure of the original dataset. Here are the key observations we can make:

- The results from all the techniques except KPCA with a radial kernel, demonstrate a relatively even distribution of data points in the reduced space, indicating that it effectively captures the overall variance of the dataset.
- SLMVP with linear and polynomial kernels, PCA and LPP manage to separate the three different classes (corresponding to attacker, defender and midfielder) in a single component. In the case of SLMVP with a linear kernel and PCA, it is the first component the one that captures the separation, whereas the second component captures the variability of the data points along other features that do not seem to have such a great impact on the dependent variable.
- As will be shown later, the axes that capture the separation of the classes show a strong correlation with features related to attacking and defending features.

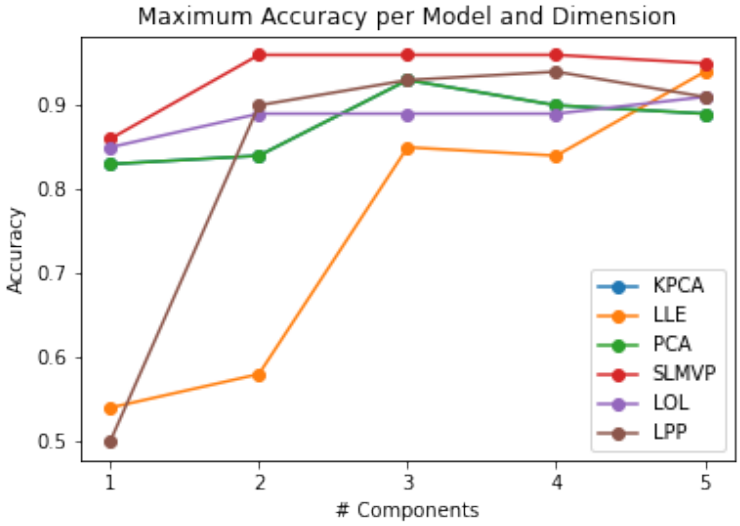


Figure 4.7: This figure shows how the accuracy of the best-performing model for each dimensionality reduction technique, evolves as the number of dimensions is increased.

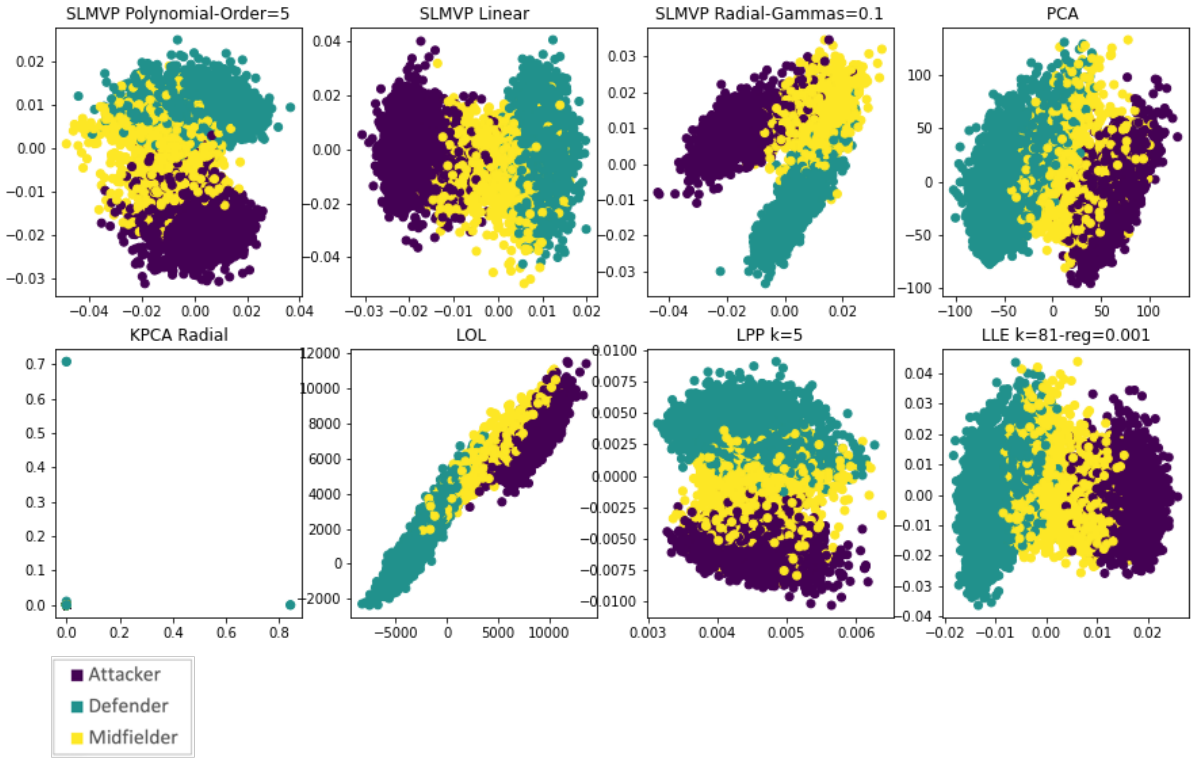


Figure 4.8: Visualization of the clusters projected into 2 dimensions that resulted from the application of the different dimensionality reduction techniques on the FIFA dataset.

Either a positive correlation with attacking and a negative correlation with defending, or a negative correlation with attacking and a positive correlation with defending.

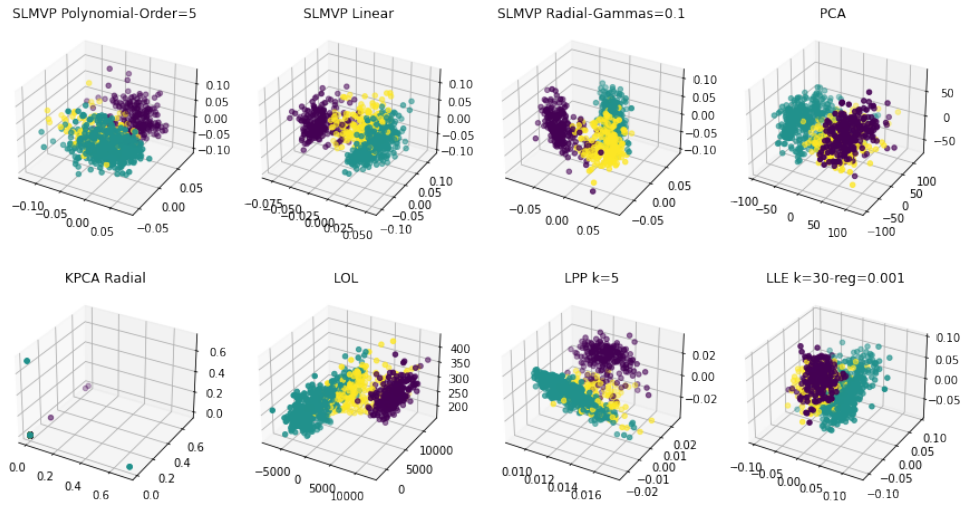


Figure 4.9: Visualization of the clusters projected into 3 dimensions that resulted from the application of the different dimensionality reduction techniques on the FIFA dataset.

- The dimensionality reduction techniques do a great job clearly separating the classes *attacker* and *defender*. However the class *midfielder* lies between the two other classes, often overlapping with them.

4.2.2.1 Comparing Techniques

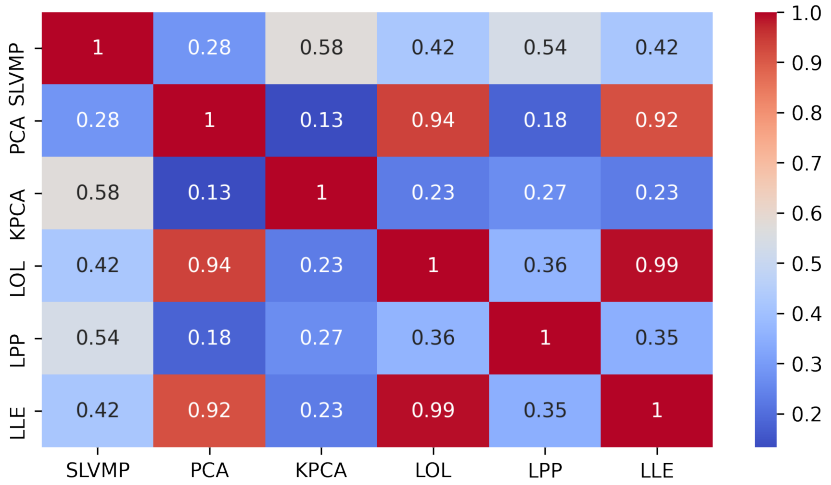


Figure 4.10: Heatmap of Spearman Rank Correlation Coefficients

The heatmap in Figure 4.10 visually represents the similarity and dissimilarity between different dimensionality reduction techniques based on their Spearman rank correlation coefficients. The correlation coefficients are calculated by measuring the absolute correlations between the data in the reduced space from each technique

and the original data. Higher correlation coefficients indicate a stronger resemblance between the reduced-dimensional representations obtained using 2 dimensionality reduction techniques. We can make the following observations:

- SLMVP is most strongly-correlated with LPP, KPCA and LOL, although this correlation is only moderate, below 60%. The correlation with LPP could be due to the fact that both SLMVP and LPP are local and therefore incorporate neighborhood information of the dataset. On the other hand, the correlation with LOL could be due to another property that both SLMVP and LOL posses, the fact that both are supervised and therefore incorporate information about the dependent variable. Finally, SLMVP and KPCA both use radial kernels.
- The strongest correlation over all is between PCA and LOL with 94%, and the lowest between KPCA and PCA with 13%.

4.2.2.2 Explaining Components

As mentioned earlier in this section, dimensionality reduction techniques are the most effective when applied to datasets with a large number of features. However, explaining a dataset with many features is challenging, due to the difficulty of showing high-dimensional data in graphs that humans can read. The FIFA dataset helps us address that challenge. The features provide insights about a soccer player's ability and they can be grouped into 6 categories: defending, attacking, skill, power, movement and mentality. For example, the features *attacking_crossing*, *attacking_finishing*, *attacking_heading_accuracy*, *attacking_short_passing*, and *attacking_volleys* are grouped as attacking.

As explained in section 3.1.1, the correlations between the input data represented in its original features and the data projected onto the first reduced dimensions is calculated. For the sake of a good visualization, the correlations concerning the features of each of the aforementioned categories are grouped and averaged. For example, *attacking* in Figure 4.11 shows the average of the correlations of *attacking_crossing*, *attacking_finishing*, etc, with the first or second component. This is a way of bringing a muti-dimensional problem to a number of variables that we are more comfortable drawing insights from.

By looking at the spider plots created by the different techniques in Figure 4.11, we can make some key observations and explain the components:

- Every dimensionality reduction technique, with the exception of LLE, shows high correlation coefficients for *defending*. This feature is key to separate the classes attackers and defenders, because the first shows a high negative correlation with *defending* and the latter shows a high positive correlation. Intuitively this makes sense, as defenders are better at defending than attackers.
- LOL's first two components are very similar in their correlations to the original features.
- PCA's and LOL's spider plot concerning the first component (marked in blue in the graph) are very similar. This matches what was shown in the heat map of Figure 4.10, where they shared a 96% Spearman Rank Correlation Coefficient.

Now we select the results of *SLMVP Radial-Gammas=0.01* in order to give detailed explanations of their components. Figure 4.12 shows side-by-side a scatterplot with

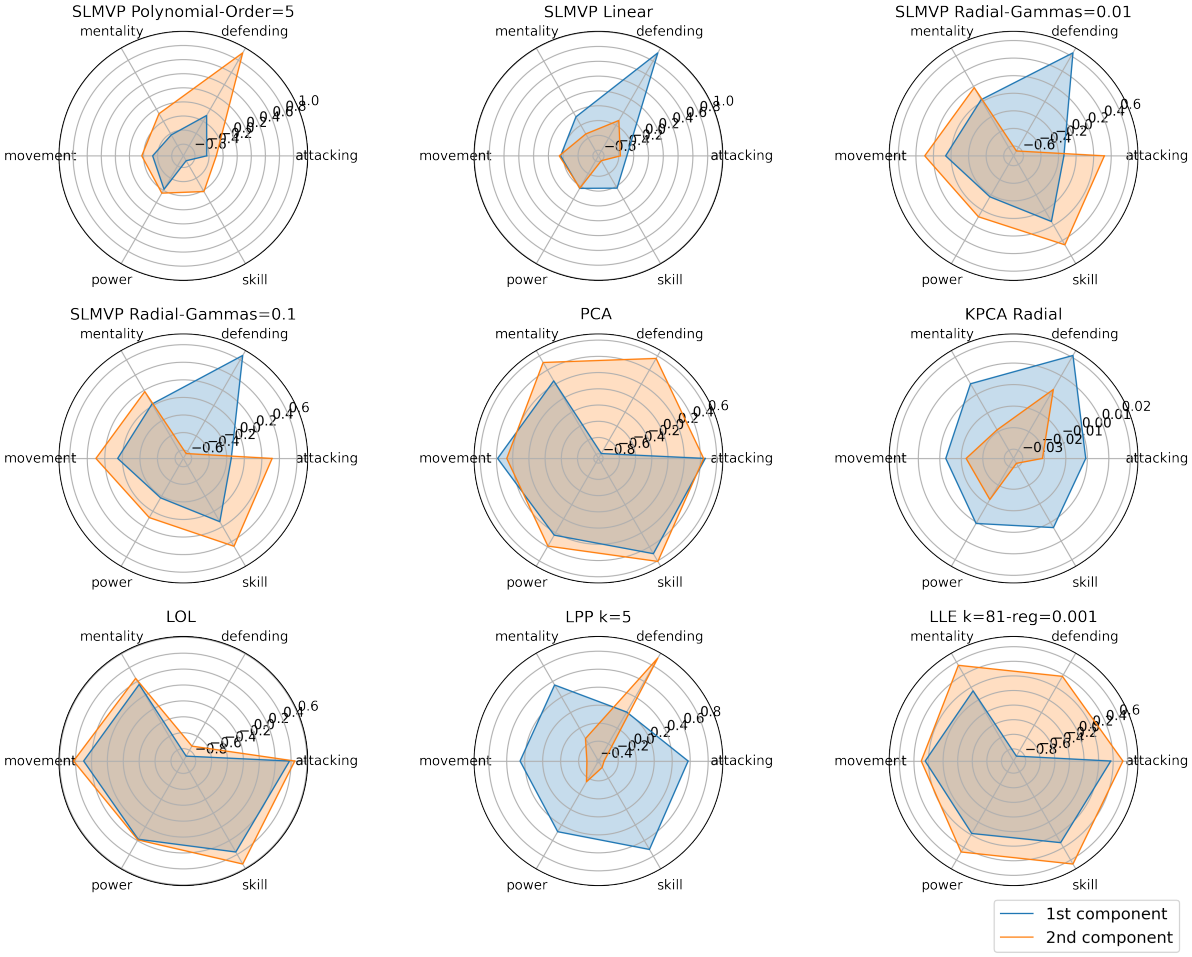


Figure 4.11: Visualization of the average correlation of the first two components with the features grouped by skill set.

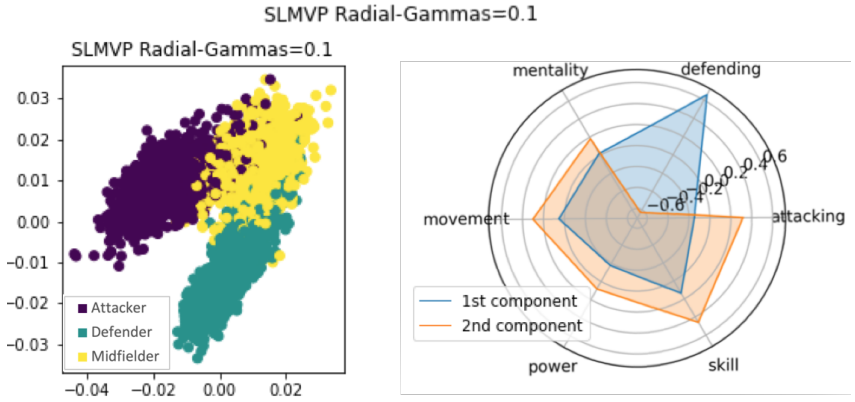


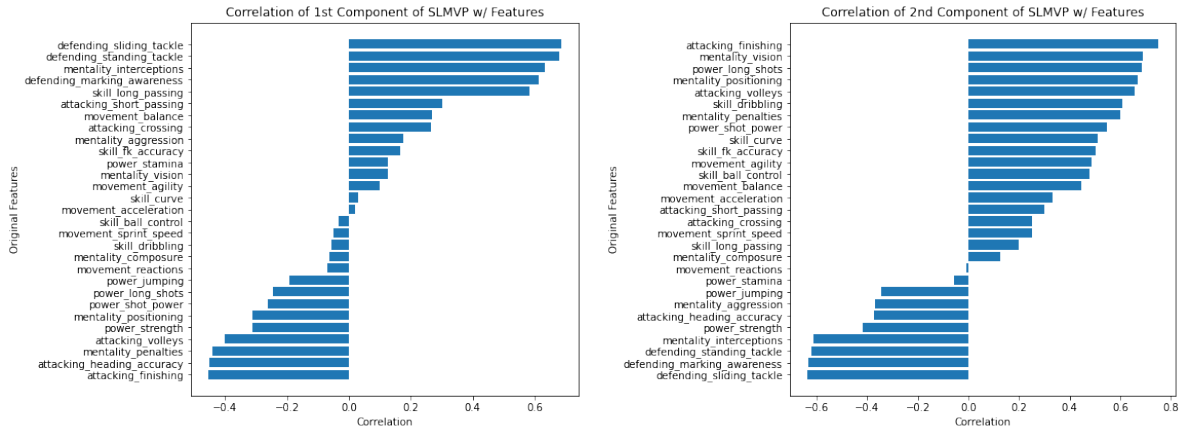
Figure 4.12: Visualization of the average correlation of the first two components with the features grouped by skill set.

the data points projected into the first 2 components, and a spider plot that conveys how these two components relate to the original features of the dataset. Figure 4.13

shows the correlations of the first and second component with the original features.

The following insights are drawn:

- The first component is positively correlated with *defending*. It separates the attackers from the defenders and midfielders. Attackers are shown to have less defending skills than the other groups.
- The second component is positively correlated with *attacking* and *skill*, and negatively correlated with *defending*. It separates the defenders from the attackers and midfielders.
- It is clear that attackers and defenders are separated by their defending and attacking skills. However the separation of midfielders with the other classes is not so straightforward.
- The bar plot in Figure 4.13 reveals that information. Besides being better at defending than attackers, midfielders are better at intercepting and making long passes. They are also better than defenders in their vision, positioning and long shots, whereas defenders are better than midfielders in heading accuracy and aggresiveness.

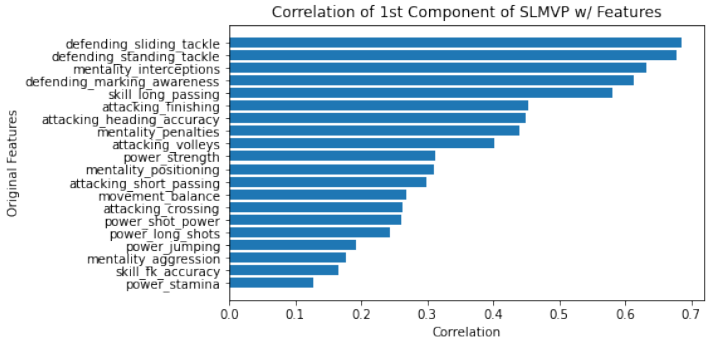


(a) 1st Component (b) 2nd component

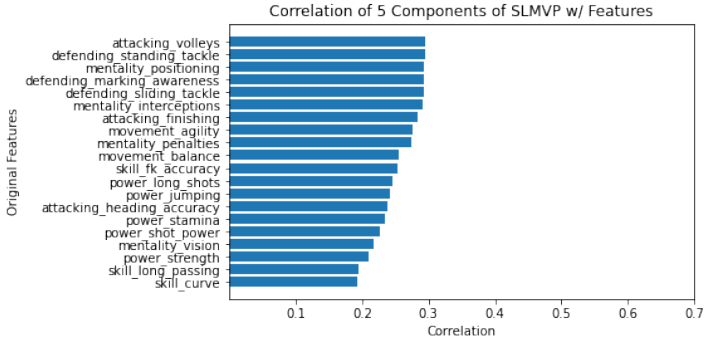
Figure 4.13: Correlations of the first and second component with the original features.

4.2.2.3 Choosing Features and Components

We can leverage the calculated absolute correlations to select the features that should be kept for a posterior machine learning task. Figure 4.14 shows the 20 features with the highest absolute correlations with the first component, in the case of the first subplot, and the average of the first 5 components, in the case of the second subplot. The features related to attacking and defending are the ones with the highest absolute correlations, which matches what was shown in Figure 4.12. Selecting the features with a higher correlation than 0.2 is reasonable.



(a) 1 component



(b) 5 components

Figure 4.14: Original features of the FIFA dataset ordered by their correlation to the components obtained with SLMVP with radial kernel and gamma values 0.1.

4.3 Insurance Company Benchmark (COIL 2000) Dataset

4.3.1 Classification Results

We conduct an evaluation and comparison of various techniques aimed at reducing the dimensionality of the COIL 2000 dataset introduced in Section 3.3.2.4. The evaluation results are presented in Table 4.3, where the techniques are ranked based on their accuracy. SLMVP is again the clear winner for this dataset. It achieves a higher accuracy than the other techniques for dimensions. It stands out at 3 dimensions, where it peaks at an accuracy of 77.5%. LOL follows closely in second place for all the tested dimensions except at 30 dimensions, where LLE takes its place.

4.3.2 Explanation

By looking at the scatter plots created by the different techniques in figures 4.16 and 4.17, we can compare how well they perform in capturing and preserving the structure of the original dataset. Here are the key observations we can make:

- Unlike with the previous datasets, these classes are more difficult to clearly separate. This is reasonable considering that the maximum accuracy that could be achieved in the classification task was 77.5%. Nonetheless, SLMVP and LOL do a good job separating those 70-80% of datapoints along the first two axes.
- The results from all the techniques, demonstrate a relatively even distribution

Dimensions	Dim. Technique	Dim. Params	Model	Accuracy
1	SLMVP	Radial-Gammas=0.01	KNN	0.760
	LOL		Naive Bayes	0.730
	KPCA	Linear	Naive Bayes	0.700
3	SLMVP	Radial-Gammas=0.01	LDA	0.775
	LOL		Naive Bayes	0.735
	LLE	k=28-reg=0.001	Naive Bayes	0.715
5	SLMVP	Linear	Naive Bayes	0.750
	LOL		Naive Bayes	0.725
	KPCA	Linear	AdaBoost	0.715
15	SLMVP	Radial-Gammas=0.01	KNN	0.740
	LOL		Random Forest	0.710
	KPCA	Radial-Gamma=0.007	Naive Bayes	0.705
30	SLMVP	Linear	KNN	0.755
	LLE	k=28-reg=0.001	LDA	0.740
	LPP	k=11	KNN	0.725

Table 4.3: Performance Comparison of Dimensionality Reduction Techniques on the COIL 2000 dataset. (See Annex Table 4) for full results.

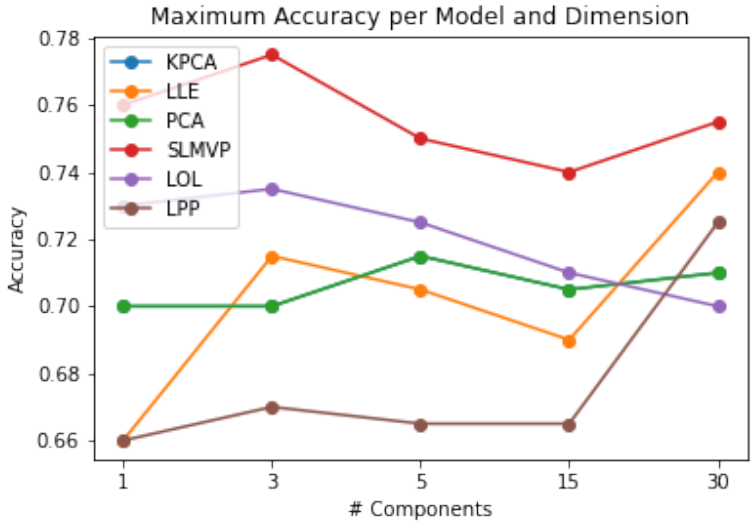


Figure 4.15: This figure shows how the accuracy of the best-performing model for each dimensionality reduction technique, evolves as the number of dimensions is increased.

of data points in the reduced space, indicating that it effectively captures the overall variance of the dataset.

4.3.2.1 Comparing Techniques

The heat map in Figure 4.18 visually represents the similarity and dissimilarity between different dimensionality reduction techniques based on their Spearman rank

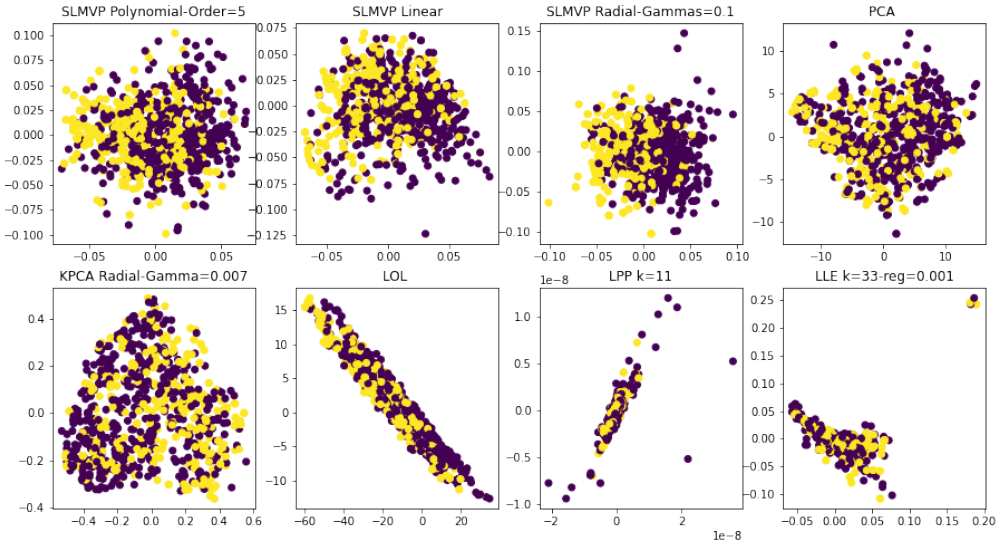


Figure 4.16: Visualization of the clusters projected into 2 dimensions that resulted from the application of the different dimensionality reduction techniques on the COIL 2000 dataset.

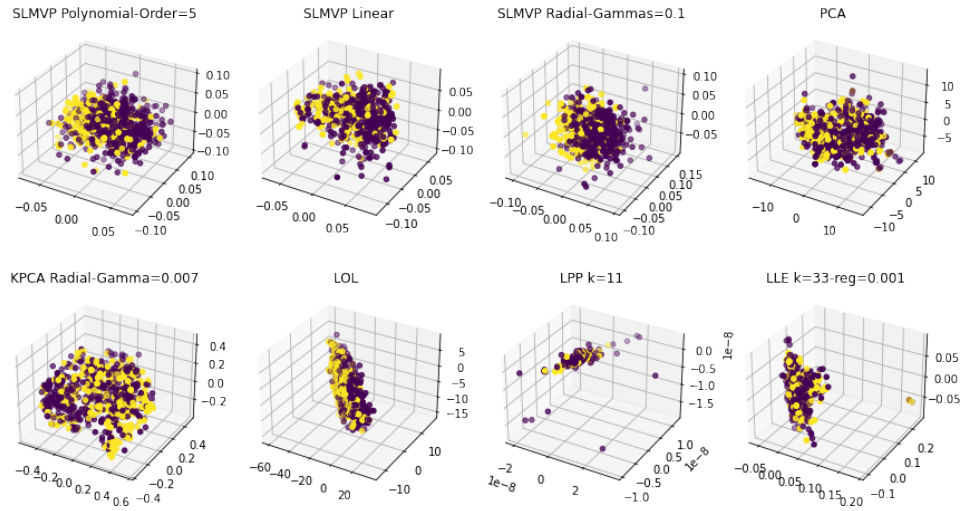


Figure 4.17: Visualization of the clusters projected into 3 dimensions that resulted from the application of the different dimensionality reduction techniques on the COIL 2000 dataset.

correlation coefficients. The correlation coefficients are calculated by measuring the absolute correlations between the data in the reduced space from each technique and the original data. Higher correlation coefficients indicate a stronger resemblance between the reduced-dimensional representations obtained using 2 dimensionality

reduction techniques. We can make the following observations:

- PCA, KPCA and LOL exhibit very high correlations between one another. KPCA was executed in its linear kernel configuration, which yields results that very closely resemble PCA. Furthermore, the similarity between PCA and LOL was also present in other datasets.
- SLMVP manifests a moderate correlation with PCA, KPCA and LOL.
- LPP stands out as yielding the most unique result, with very low correlations with other techniques.

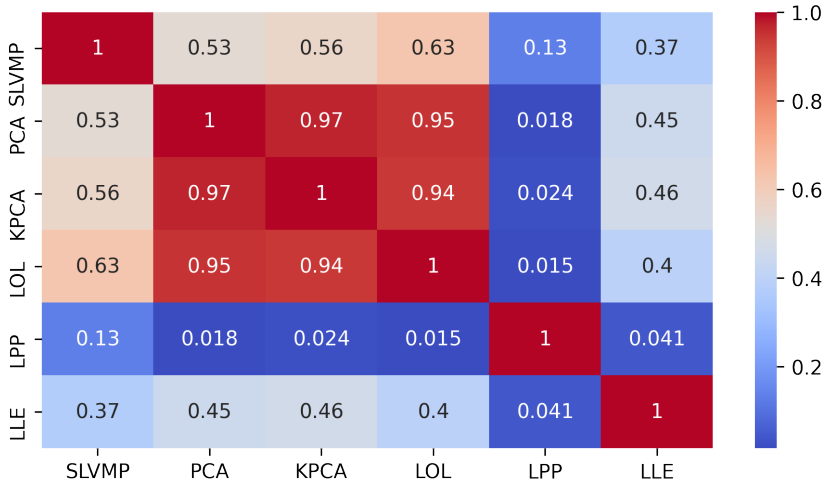


Figure 4.18: Heatmap of Spearman Rank Correlation Coefficients

4.3.2.2 Choosing Features and Components

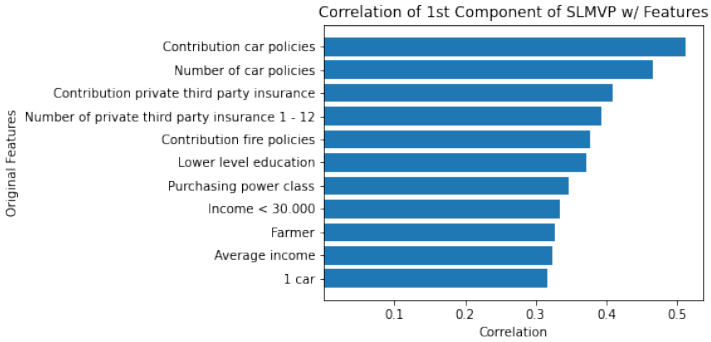
We can leverage the calculated absolute correlations to select the features that should be kept for a posterior machine learning task. Figure 4.19 shows the 15 features with the highest absolute correlations with the first component, in the case of the first subplot, and the average of the first 5 components, in the case of the second subplot.

In the first subplot, the features that show the highest absolute correlation with the component are related to whether the client already had some type of insurance. These are *Contribution car policies*, *Number of car policies*, and *Contribution private third party insurance*. We also observe features related to the education level and income of the client. The highest correlated features at 5 components do differ. We observe *Rented house* and *Home owners* at the top, as well as other features related to the marital status, or whether the client lives in a household with children.

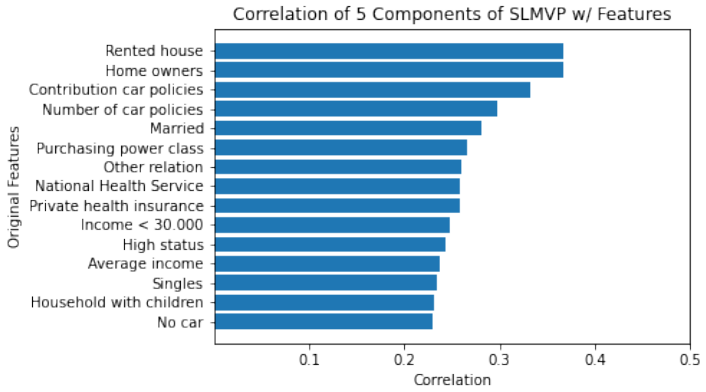
4.4 ORL Dataset

4.4.1 Classification Results

We conduct an evaluation and comparison of various techniques aimed at reducing the dimensionality of the Our Database of Faces (ORL) dataset introduced in Section 3.3.2.3. The evaluation results are presented in Table 4.4, where the techniques are ranked based on their accuracy.



(a) 1 component



(b) 5 components

Figure 4.19: Original features of the COIL 2000 dataset ordered by their correlation to the components obtained with SLMVP with radial kernel and gamma values 0.1.

For this dataset, at least 5 components are necessary for any dimensionality reduction technique machine learning model pair to reach a high accuracy (greater than 90%). LOL is the best performing technique at 5 dimensions. LOL, KPCA, and PCA obtain very high accuracies at 15 dimensions and reach a 100% at 30 dimensions. LLE and SLMVP underperform, perhaps due to their locality characteristic. The dataset is composed of images of 40 different faces, with variations in head rotation, tilt and pose, and locality can misguide the model.

4.4.2 Explanation

By looking at the scatter plots created by the different techniques in figures 4.21 and 4.22, we can compare how well they perform in capturing and preserving the structure of the original dataset. Here are the key observations we can make:

- The results from all the techniques, demonstrate a relatively even distribution of data points in the reduced space, indicating that it effectively captures the overall variance of the dataset.

- Only SLMVP with a radial kernel manages to clearly separate the classes. They do it in an interesting heart-like shape with 2 components.

Dimensions	Dim. Technique	Dim. Params	Model	Accuracy
1	LLE	k=18-reg=0.001	KNN	0.300
	KPCA	Linear	Random Forest	0.200
	PCA		XGBoost	0.175
3	KPCA	Linear	SVM	0.750
	LLE	k=18-reg=0.001	SVM	0.750
	PCA		SVM	0.750
5	LOL		SVM	0.925
	KPCA	Linear	SVM	0.900
	PCA		SVM	0.900
15	KPCA	Linear	SVM	0.975
	LOL		SVM	0.975
	PCA		SVM	0.975
30	KPCA	Linear	SVM	1.000
	LOL		LDA	1.000
	PCA		SVM	1.000

Table 4.4: Performance Comparison of Dimensionality Reduction Techniques on the ORL dataset. (See Annex Table 3) for full results.

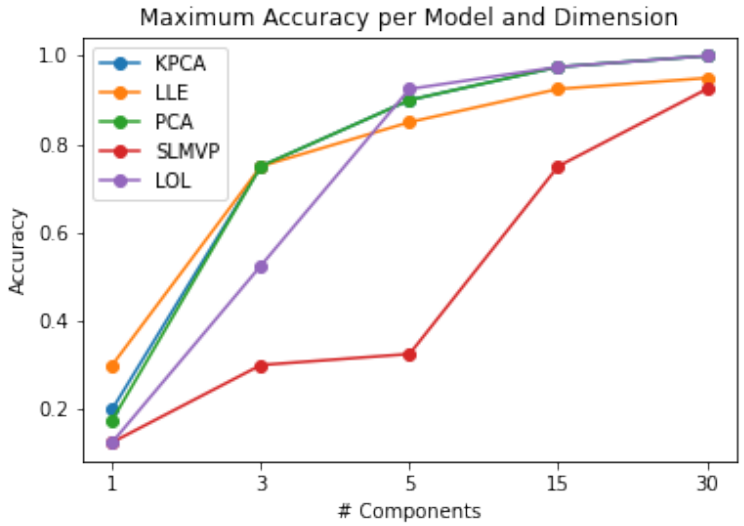


Figure 4.20: This figure shows how the accuracy of the best-performing model for each dimensionality reduction technique, evolves as the number of dimensions is increased.

4.4.2.1 Comparing Techniques

The heat map in Figure 4.23 visually represents the similarity and dissimilarity between different dimensionality reduction techniques based on their Spearman rank correlation coefficients. Higher correlation coefficients indicate a stronger resemblance between the reduced-dimensional representations obtained using 2 dimensionality reduction techniques. We can make the following observations:

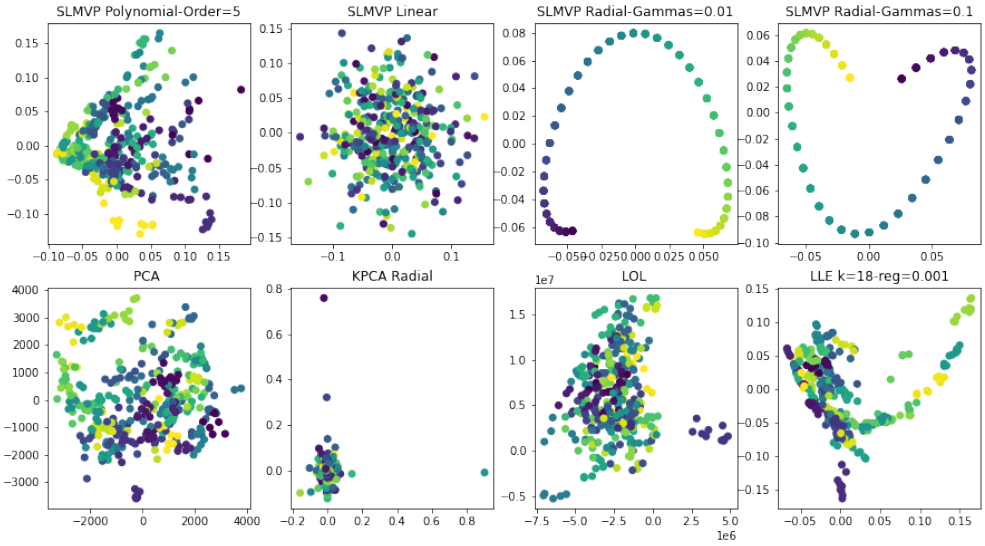


Figure 4.21: Visualization of the clusters projected into 2 dimensions that resulted from the application of the different dimensionality reduction techniques on the ORL dataset.

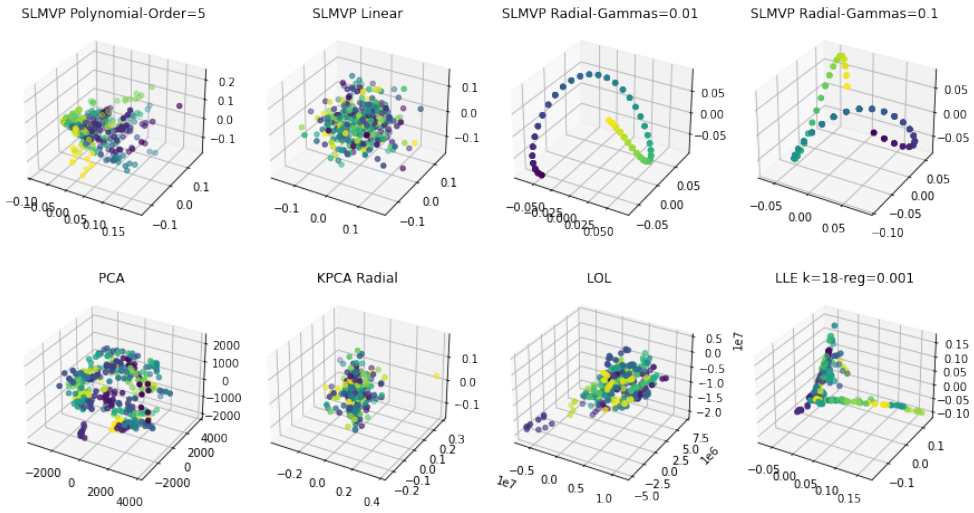
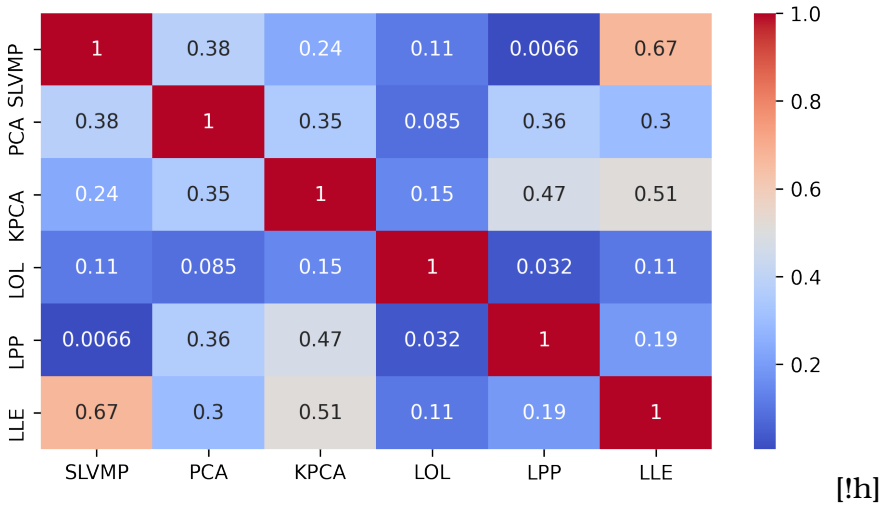


Figure 4.22: Visualization of the clusters projected into 3 dimensions that resulted from the application of the different dimensionality reduction techniques on the ORL dataset.

- LLE and SLMVP exhibit the highest correlations between one another. This might be due to the fact that both techniques are local and incorporate neighborhood information of the dataset.



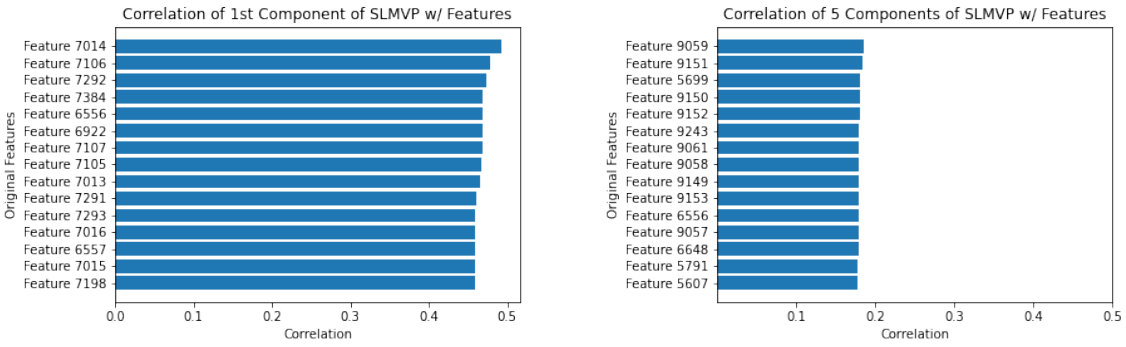
[!h]

Figure 4.23: Heatmap of Spearman Rank Correlation Coefficients

- KPCA, LLE, and LPP manifest moderate correlations.

4.4.2.2 Choosing Features and Components

We can leverage the calculated absolute correlations to select the features that should be kept for a posterior machine learning task. Figure 4.24 shows the 15 features with the highest absolute correlations with the first component, in the case of the first subplot, and the average of the first 5 components, in the case of the second subplot. For this image data, the correlations follow a smooth curve and it is difficult to determine how many features to reject. It seems reasonable to reject the features that correspond to the outer parts of the frame, but then again, those pixels might contain useful information.



(a) 1 component

(b) 5 components

Figure 4.24: Original features of the ORL dataset ordered by their correlation to the components obtained with SLMVP with radial kernel and gamma values 0.1.

Chapter 5

Conclusion

This master's thesis aimed to explore the explainability of dimensionality reduction techniques. Throughout the study, we explored the state-of-the art in the fields of dimensionality reduction and explainable artificial intelligence. We examined relevant techniques such as SLMVP, PCA or LOL, and assessed their effectiveness in reducing high-dimensional data while preserving essential information both through the explainability of the their components, and through their ability to reach high accuracies when paired with a machine learning model.

Some of the key findings of this research is that SLMVP is capable of effectively handling multilabel datasets, being able to separate the classes more clearly than the other tested techniques. SLMVP also excelled at segregating the clusters for single-label data, achieving the highest accuracy in 3 out of the 4 datasets employed. Furthermore, SLMVP is most similar to the techniques LOL and LPP in the significance that it gives to the original features. This is due to the fact that it possesses the characteristics of being local and supervised.

Moreover, the trade-off between complexity and explainability was discussed. While dimensionality reduction is most effective when dealing with datasets containing numerous features, it becomes challenging to present high-dimensional data through graphs and tables in a manner that humans can easily comprehend. However, this challenge was successfully overcome by utilizing the FIFA dataset, which conveniently groups its features into six distinct categories (such as *attacking*, *defending*, etc.) facilitating its explainability.

Additionally, the thesis introduced a way of interpreting the classes by examining the correlation between components and the original features. This correlation was then utilized to provide recommendations on which features should be selected and how many components should be retained for subsequent machine learning prediction tasks.

In conclusion, this thesis contributes to the understanding of dimensionality reduction techniques and highlights the importance of finding a balance between complexity, interpretability, and performance in machine learning.

Bibliography

[1] David M Blei, Andrew Y Ng, and Michael I Jordan. "Latent dirichlet allocation". In: *Journal of machine Learning research* 3.Jan (2003), pp. 993–1022.

[2] AT&T Laboratories Cambridge. https://cam-orl.co.uk/facedatabase.html. 1994.

[3] Tianqi Chen and Carlos Guestrin. "Xgboost: A scalable tree boosting system". In: *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*. 2016, pp. 785–794.

[4] Google Cloud. *Explaining model predictions on image data*. [Online; accessed July 13, 2023]. 2023. URL: https://cloud.google.com/blog/products/ai-machine-learning/explaining-model-predictions-on-image-data.

[5] Corinna Cortes and Vladimir Vapnik. "Support-vector networks". In: *Machine learning* 20 (1995), pp. 273–297.

[6] Evelyn Fix and Joseph Lawson Hodges. "Discriminatory analysis. Nonparametric discrimination: Consistency properties". In: *International Statistical Review/Revue Internationale de Statistique* 57.3 (1989), pp. 238–247.

[7] Yoav Freund and Robert E Schapire. "A desicion-theoretic generalization of online learning and an application to boosting". In: *European conference on computational learning theory*. Springer. 1995, pp. 23–37.

[8] Esteban García-Cuesta and José Antonio Iglesias. "User modeling: Through statistical analysis and subspace learning". In: *Expert Systems with Applications* 39.5 (2012), pp. 5243–5250.

[9] Esteban García-Cuesta et al. "A combination of supervised dimensionality reduction and learning methods to forecast solar radiation". In: *Applied Intelligence* 53.11 (2023), pp. 13053–13066.

[10] David Gunning et al. "XAI—Explainable artificial intelligence". In: *Science robotics* 4.37 (2019), eaay7120.

[11] Xiaofei He and Partha Niyogi. "Locality preserving projections". In: *Advances in neural information processing systems* 16 (2003).

[12] Tin Kam Ho. "Random decision forests". In: *Proceedings of 3rd international conference on document analysis and recognition*. Vol. 1. IEEE. 1995, pp. 278–282.

[13] I. T. Jolliffe. "Discarding Variables in a Principal Component Analysis. I: Artificial Data". In: *Journal of the Royal Statistical Society Series C: Applied Statistics* 21.2 (Dec. 2018), pp. 160–173. ISSN: 0035-9254. DOI: 10.2307/2346488. URL: https://doi.org/10.2307/2346488.

[14] Scott M Lundberg and Su-In Lee. "A unified approach to interpreting model predictions". In: *Advances in neural information processing systems* 30 (2017).

[15] Saumitra Mishra, Bob L Sturm, and Simon Dixon. “Local interpretable model-agnostic explanations for music content analysis.” In: *ISMIR*. Vol. 53. 2017, pp. 537–543.

[16] Karl Pearson. “On lines and planes of closest fit to systems of points in space”. In: *Philosophical Magazine* 2.11 (1901), pp. 559–572.

[17] Sam T Roweis and Lawrence K Saul. “Nonlinear dimensionality reduction by locally linear embedding”. In: *science* 290.5500 (2000), pp. 2323–2326.

[18] Bernhard Schölkopf, Alexander Smola, and Klaus-Robert Müller. “Nonlinear component analysis as a kernel eigenvalue problem”. In: *Neural computation* 10.5 (1998), pp. 1299–1319.

[19] Mukund Sundararajan, Ankur Taly, and Qiqi Yan. “Axiomatic attribution for deep networks”. In: *International conference on machine learning*. PMLR. 2017, pp. 3319–3328.

[20] Peter Van Der Putten and Maarten Van Someren. “A bias-variance analysis of a real world learning problem: The CoIL challenge 2000”. In: *Machine learning* 57 (2004), pp. 177–195.

[21] Michel Verleysen and Damien François. “The curse of dimensionality in data mining and time series prediction”. In: *Computational Intelligence and Bioinspired Systems: 8th International Work-Conference on Artificial Neural Networks, IWANN 2005, Vilanova i la Geltrú, Barcelona, Spain, June 8-10, 2005. Proceedings 8*. Springer. 2005, pp. 758–770.

[22] Joshua Vogelstein et al. “Linear Optimal Low Rank Projection Provably Outperforms Principal Components Analysis in High-Dimensional Multi-class Data”. In: (Sept. 2017).

Annex

Dimensions	Dim. Technique	Dim. Params	Model	Best Score
1	LLE	k=30-reg=0.001	XGBoost	0.90
	KPCA	Linear	XGBoost	0.63
	PCA		KNN	0.57
	SLMVP	Linear	XGBoost	0.56
	LPP	k=17	Decision Tree	0.47
	LOL		LDA	0.34
2	KPCA	Linear	KNN	0.96
	PCA		Random Forest	0.96
	SLMVP	Linear	LDA	0.96
	LLE	k=30-reg=0.001	KNN	0.91
	LPP	k=17	Random Forest	0.89
	LOL		LDA	0.64
3	KPCA	Linear	LDA	1.00
	LLE	k=30-reg=0.001	LDA	1.00
	PCA		LDA	1.00
	SLMVP	Linear	LDA	1.00
	LPP	k=17	LDA	0.96
	LOL		LDA	0.85
4	KPCA	Linear	LDA	1.00
	LLE	k=30-reg=0.001	KNN	1.00
	LPP	k=17	LDA	1.00
	PCA		LDA	1.00
	SLMVP	Polynomial-Order=5	LDA	1.00
	LOL		Naive Bayes	0.95
5	KPCA	Linear	LDA	1.00
	LLE	k=30-reg=0.001	KNN	1.00
	LPP	k=17	Random Forest	1.00
	PCA		LDA	1.00
	SLMVP	Linear	LDA	1.00
	LOL		KNN	0.99

Table 1: Artificial Dataset

Dimensions	Dim. Technique	Dim. Params	Model	Accuracy
1	SLMVP	Linear	KNN	0.86
	LOL		Naive Bayes	0.85
	KPCA	Linear	LDA	0.83
	PCA		LDA	0.83
	LLE	k=30-reg=0.001	AdaBoost	0.54
	LPP	k=5	KNN	0.50
2	SLMVP	Radial-Gammas=0.1	XGBoost	0.96
	LPP	k=5	Random Forest	0.90
	LOL		KNN	0.89
	KPCA	Linear	Decision Tree	0.84
	PCA		Decision Tree	0.84
	LLE	k=30-reg=0.001	Random Forest	0.58
3	SLMVP	Radial-Gammas=0.1	XGBoost	0.96
	KPCA	Linear	SVM	0.93
	LPP	k=5	SVM	0.93
	PCA		KNN	0.93
	LOL		KNN	0.89
	LLE	k=30-reg=0.001	SVM	0.85
4	SLMVP	Radial-Gammas=0.01	Random Forest	0.96
	LPP	k=5	XGBoost	0.94
	KPCA	Linear	SVM	0.90
	PCA		SVM	0.90
	LOL		SVM	0.89
	LLE	k=30-reg=0.001	SVM	0.84
5	SLMVP	Radial-Gammas=0.1	SVM	0.95
	LLE	k=30-reg=0.001	SVM	0.94
	LOL		XGBoost	0.91
	LPP	k=5	KNN	0.91
	KPCA	Linear	SVM	0.89
	PCA		SVM	0.89

Table 2: FIFA Dataset

Dimensions	Dim. Technique	Dim. Params	Model	Accuracy
1	LLE	k=18-reg=0.001	KNN	0.300
	KPCA	Linear	Random Forest	0.200
	PCA		XGBoost	0.175
	LOL		XGBoost	0.125
	SLMVP	Radial-Gammas=0.01	Naive Bayes	0.125
3	KPCA	Linear	SVM	0.750
	LLE	k=18-reg=0.001	SVM	0.750
	PCA		SVM	0.750
	LOL		KNN	0.525
	SLMVP	Polynomial-Order=5	XGBoost	0.300
5	LOL		SVM	0.925
	KPCA	Linear	SVM	0.900
	PCA		SVM	0.900
	LLE	k=18-reg=0.001	KNN	0.850
	SLMVP	Polynomial-Order=5	Random Forest	0.325
15	KPCA	Linear	SVM	0.975
	LOL		SVM	0.975
	PCA		SVM	0.975
	LLE	k=18-reg=0.001	SVM	0.925
	SLMVP	Radial-Gammas=0.1	Naive Bayes	0.750
30	KPCA	Linear	SVM	1.000
	LOL		LDA	1.000
	PCA		SVM	1.000
	LLE	k=18-reg=0.001	Random Forest	0.950
	SLMVP	Radial-Gammas=0.1	Naive Bayes	0.925

Table 3: ORL Dataset

Dimensions	Dim. Technique	Dim. Params	Model	Accuracy
1	SLMVP	Radial-Gammas=0.01	KNN	0.760
	LOL		Naive Bayes	0.730
	KPCA	Linear	Naive Bayes	0.700
	PCA		Naive Bayes	0.700
	LLE	k=28-reg=0.001	LDA	0.660
	LPP	k=11	LDA	0.660
3	SLMVP	Radial-Gammas=0.01	LDA	0.775
	LOL		Naive Bayes	0.735
	LLE	k=28-reg=0.001	Naive Bayes	0.715
	KPCA	Linear	Naive Bayes	0.700
	PCA		Naive Bayes	0.700
	LPP	k=11	Random Forest	0.670
5	SLMVP	Linear	Naive Bayes	0.750
	LOL		Naive Bayes	0.725
	KPCA	Linear	AdaBoost	0.715
	PCA		AdaBoost	0.715
	LLE	k=28-reg=0.001	Naive Bayes	0.705
	LPP	k=11	AdaBoost	0.665
15	SLMVP	Radial-Gammas=0.01	KNN	0.740
	LOL		Random Forest	0.710
	KPCA	Radial-Gamma=0.007	Naive Bayes	0.705
	PCA		LDA	0.705
	LLE	k=28-reg=0.001	Decision Tree	0.690
	LPP	k=11	Random Forest	0.665
30	SLMVP	Linear	KNN	0.755
	LLE	k=28-reg=0.001	LDA	0.740
	LPP	k=11	KNN	0.725
	KPCA	Radial-Gamma=0.007	Random Forest	0.710
	PCA		KNN	0.710
	LOL		KNN	0.700

Table 4: COIL2000 Dataset