EPiC Series in Computing

Volume 97, 2024, Pages 81–90

Proceedings of 36th International Conference on
Computer Applications in Industry and Engineering


# An Evaluation of Strategies for Dimensionality Reduction

Gautam B. Singh

Computer Science and Engineering
Oakland University, Rochester, MI 48309, USA
singh@oakland.edu
https://www.oakland.edu

**Abstract**

The "curse of dimensionality" in machine learning refers to the increasing data training requirements for features collected from high dimensional spaces. Researchers generally use one of several dimensionality reduction methods to visualize data and estimate data trends. Feature engineering and selection minimize dimensionality and optimize algorithms. Dimensionality must be matched to the data to preserve information. This paper compares the final model evaluation dimensionality reduction methods. First, encode the data set in a smaller dimension to avoid the curse of dimensionality and train the model with a manageable number of features.

**Keywords:** High Dimensional Data, Machine Learning, Curse of Dimensionality, Dimensionality Reduction, Model Evaluation.

## 1 Introduction

High-dimensional data sets have become quite significant in data sciences, such as data sets including sensor data, financial data, and social network data. The intelligent machine learning modeling pipeline is a sequential workflow that outlines the steps involved in building, training, evaluating, and deploying such an intelligent model. The main stages for building the pipeline include data acquisition, data pre-processing, model training, and model evaluation. After the model's evaluation, the most promising model that meets the desired performance criteria is deployed in the production environment, where it can predict new, unseen data[3].

In machine learning, the term "curse of dimensionality" describes a phenomenon that occurs in high-dimensional spaces, where the volume of the space increases exponentially with the number of dimensions. In high-dimensional spaces, the number of data points required to maintain the same data density increases exponentially as the number of dimensions increases. As you add more features or dimensions to a dataset, the data needed to represent the feature space grows correspondingly at a rapid or possibly exponential rate [2], [14].

To mitigate the curse of dimensionality, researchers often employ dimensionality reduction techniques, such as Principal Component Analysis (PCA) or t-distributed Stochastic Neighbor Embedding (t-SNE), to reduce the number of dimensions while preserving important patterns in the data. Additionally, selecting relevant features and employing feature engineering can reduce the dimensionality and improve the performance of algorithms. However, balancing reducing dimensionality and preserving meaningful information is essential to avoid losing crucial insights from the data [4].

**Data Set:** This paper examines the various dimensionality reduction techniques and evaluates each with a machine learning model. Specifically, the idea is to establish a protocol to represent the data set in a lower dimension so that a manageable number of features must be learned to train the model using data after it has been transformed into the lower dimensional space.

Six dimensionality reduction techniques are applied to the wine data set, which comprises 178 samples represented on a 13-dimensional space. This data set results from a chemical analysis of wines grown in the same region of Italy but derived from three different cultivars. Each wine is measured on the following set of 13 features, each being a numerical value: (1) Alcohol; (2) Malic acid; (3) Ash; (4) Alkalinity of ash; (5) Magnesium; (6) Total phenols; (7) Flavanoids; (8) Non-flavanoid phenols; (9) Proanthocyanins; (10) Color intensity; (11) Hue; (12) OD280/OD315 of diluted wines, and (13) Proline

Each wine is classified into one of three class labels, or `1`, `2`, or `3`. The data set contains 59 instances of Class 1, 71 instances of Class 2, and 48 instances of Class 3. The data set is reduced to two dimensions (2D) using the following six dimensionality reduction techniques – Principal Component Analysis (PCA), Kernel Principal Component Analysis with Quadratic Polynomial kernel (KPCA1), Kernel Principal Component Analysis with Gaussian kernel (KPCA2), Multidimensional Scaling (MDS), Chi-Square ($\chi^2$), and ANOVA. Data in the reduced dimensions trains a classifier system and thus evaluates the performance of the classifier and, in turn, the dimensionality reduction technique.

## 2 Dimensionality Reduction Algorithm Review

Processing data matrix $\mathbf{X}$ comprising $n$ observations, each represented on a $p$ dimensional space:

$$\boldsymbol{X} = \begin{pmatrix} x_{1,1} & x_{1,2} & x_{1,3} & \dots & x_{1,p-1} & x_{1,p} \\ x_{2,1} & x_{2,2} & x_{2,3} & \dots & x_{2,p-1} & x_{2,p} \\ & & \dots & & & \\ x_{i,1} & x_{i,2} & x_{i,3} & \dots & x_{i,p-1} & x_{i,p} \\ & & \dots & & & \\ x_{n,1} & x_{n,2} & x_{n,3} & \dots & x_{n,p-1} & x_{n,p} \end{pmatrix}$$

The target variable is indicated by $Y$, an n-dimensional vector representing the desired outcome $y_i$ corresponding to the observation $X_i$.

Dimensionality refers to the number of input variables or features for a dataset. Dimensionality reduction refers to techniques that reduce the number of input variables for training data. When dealing with high-dimensional data, it is frequently advantageous to reduce the sample dimensions by projecting the data to a lower-dimensional subspace, as a greater number of features

make it difficult to effectively learn the correspondingly large number of predictive models. This is also commonly known as the "curse of dimensionality." The various dimensionality reduction approaches explored are described in the following.

## 2.1 Principal Component Analysis

Principal component analysis (PCA) is a statistical technique that employs an orthogonal transformation to morph a set of observations of potentially correlated variables into values of linearly uncorrelated variables called principal components [13], [12].

Consider a set of observations. The principal components for representing these observations along their eigenvectors involve the determination of the covariance matrix $S$ followed by the determination of the eigenvalue of the covariance matrix, $\boldsymbol{S}$.

$$\boldsymbol{S} = \frac{(\boldsymbol{X} - \boldsymbol{\mu})^T(\boldsymbol{X} - \boldsymbol{\mu})}{n-1}, \quad \text{where} \quad \boldsymbol{\mu} = \frac{\sum_n \boldsymbol{X}}{n}$$

And solving for the eigenbasis comprising the eigenvalues $\lambda$ and the eigenvectors $\mathbf{v}$, the PCA computes a new set of composite variables instead of your original variables. One of the characteristics of the new composite variables, also known as the Principal Components, is that they are entirely uncorrelated with one another. In addition, the Principal Components are arranged such that the first component captures the maximum amount of variance in the data set, the second the next highest degree of variance, and so on.

Since an orthonormal basis matrix forms unitary linear transformation, the linear transformation encoded in the covariance matrix $\boldsymbol{S}$ is a rotation or reflection and scaling of the data. Therefore, by using a few principal components corresponding to the higher eigenvalues, the data is transformed – notably, with a unitary transformation – into a space where the spread in the data is highest. In addition to eliminating redundancy due to feature dependencies, this also helps in a more straightforward determination of decision boundaries in classification problems, for example.

### 2.1.1 Kernel Principal Component Analysis (KPCA)

PCA rotates the original axes to align the coordinate system along the axes of maximum data variability. Since rotation is a linear transformation, the transformed coordinates are essentially a linear combination of coordinates in the original axes. As such PCA is unable to extract non-linear variability structures in the data. Kernel PCA (KPCA) is a non-linear version of the PCA aiming to capture higher-order data statistics by first mapping the data of the input space into another feature space using a non-linear mapping function $\Phi$.

The non-linear function $\Phi$ transforms the $p$ dimensional input data vector $\mathbf{x}$ into a feature from feature space as $\Phi(\mathbf{x})$. Next, the covariance matrix in this feature space is calculated, and eigenvalues and eigenvectors are calculated as outlined above.

In this paper, the Polynomial Kernel function with a 2-degree polynomial (KPCA1) and Gaussian Kernel function with $\gamma = (1/128^2)$ (KPCA2) were included in the comparisons.

## 2.2 Multi-Dimensional Scaling

Multidimensional Scaling (MDS) is a statistical methodology used to visualize and analyze relationships of similarity or dissimilarity between various objects or data points. The primary objective is to effectively depict high-dimensional data in a reduced-dimensional space while maintaining the accuracy of pairwise distances or similarities among the data points. In essence, MDS is a technique that simplifies intricate relationships among data points by converting them into a more straightforward geometric representation [5], [11].

MDS problem is posed as having a collection of $n$ objects with pairwise distances of $\{d_{ij}\}$, $1 \leq i, j \leq n$, which are then represented in the Euclidean space with points $\mathbf{y_1}, \mathbf{y_2} \ldots \mathbf{y_n} \in \Re^k$, such that:

$$||\mathbf{y_i} - \mathbf{y_j}|| = d_{ij}, \forall i, j$$

Several solutions may exist for representing the $y_i$s since translated values of any solution, where the same vector translates each of the $y$ points, will also be the MDS problem. Thus, an additional constraint is also placed on requiring the center of mass of all points to be the origin. This is achieved by applying a centering transformation to the distance matrix that recovers the centered cross-product matrix, $\mathbf{S} = \mathbf{X}\mathbf{X}^T$ from the square of the distance matrix $\mathbf{D} = \{d_{ij}^2\}$. [6].

$$\mathbf{S} = \frac{1}{2}\,\mathbf{C}\,\mathbf{D}\,\mathbf{C}^T \tag{1}$$

$$\text{where,} \quad \mathbf{C} = \mathbf{I}_n - \frac{1}{n}\mathbf{1}\mathbf{1}^T \tag{2}$$

MDS is generally used for visualization, which essentially depicts the first $k$ components, typically with $k = 2$, *or* 3, of the eigenvectors. One of the advantages of using MDS is that the determination of eigenvectors can also be used to reduce the dimensionality of the original data set using the distance matrix based on any of the distance metrics, such as Euclidean, Manhattan, Chebyshev, Minkowski, or Cosine distances.

## 2.3 Chi-Square

Another approach is to rank features based on their significance and ability to discriminate between classes. The chi-square statistic measures the difference between the observed and expected frequencies.

If the difference is large enough, it suggests that there might be a meaningful relationship between the variables rather than the difference being attributed only to random chance. The test result provides a *p-value*, which indicates the probability of obtaining this difference by chance alone [9], [15], [1].

The chi-square method for ranking features achieves this by computing the $\chi^2$ values and measuring the dependence of the feature val-

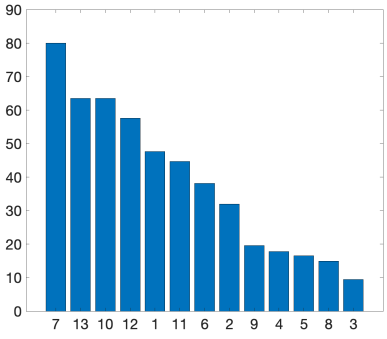


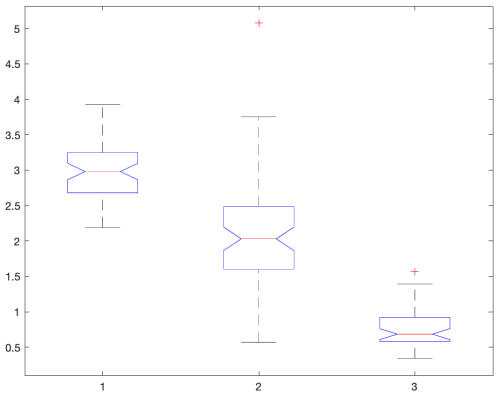
Figure 1: Chi-Square Values for features

ues and the response variables. When the feature values and the response values are independent, the $\chi^2$ values will be low since the observed frequencies will be relatively close to those expected under the assumption of independence of the feature value and the response variables.

The $\chi^2$ values reverse-sorted for the thirteen features are plotted in Fig. 1. As is evident from the figure, the $\chi^2$ for feature 7 is the highest. This implies that this feature value produces observations that are most distinct from those expected, thus carrying the most information. This highest information-carrying feature would be retained in the reduced dimension. The next feature to be retained is feature 13 since a 2D dataset representation is sought.

## 2.4 Analysis of Variance (ANOVA)

ANOVA, short for Analysis of Variance, is a statistical technique used to examine and evaluate the disparities between the means of two or more distinct groups. This assessment aims to evaluate the presence of statistically significant disparities in the means of different groups, helping to determine whether these differences are attributable to genuine effects or simple chance fluctuations [10], [8], [7].




**ANOVA Table**

| Source | SS | df | MS | F | Prob>F |
|---|---|---|---|---|---|
| Groups | 128.522 | 2 | 64.2612 | 233.93 | 3.59859e-50 |
| Error | 48.074 | 175 | 0.2747 | | |
| Total | 176.596 | 177 | | | |

Figure 2: ANOVA Analysis for Feature 7 (Flavanoids)

The ANOVA method for feature prioritization uses a similar idea as the $\chi^2$ method. However, the $F-$statistic of the ANOVA test compares the interclass and intraclass sum of square values, as in Eq. 5. The idea is to group any feature values by the class labels and test the null hypothesis that each group of observations comes from the same underlying population using the calculation of the $F$-statistic, which is computed by determining the sum of squares within classes $SSW$ and the sum of squares between classes $SSB$. Higher values of $F$-statistics imply that the null hypothesis should be rejected. Furthermore, the $F$ statistic may be used to order features and their subsequent selection.

As shown in Fig. 2, the feature with the highest F-statistic was feature 7, measuring the content of flavonoids. The flavonoid content in the three classes had different means as depicted in the figure. The $SS$ value for the `Group` shown in the output corresponds to the $SSW$ value, and the $SS$ value for `Error` corresponds to $SSB$. The $p-value$ that three means to belong to the same population is $3.6 \times 10^{-50}$, indicating that the mean value for feature 7 for the three classes is significantly different. Thus, feature 7 is one of the features in the reduced dimensions.

$$\text{SSW} = \sum_{j=1}^{k}\sum_{j=1}^{l}(X - \bar{X}_j)^2, \quad \text{df}_w = k - 1, \quad \text{MSW} = \frac{SSW}{df_w} \tag{3}$$

$$\text{SSB} = \sum_{j=1}^{k}(\bar{X}_j - \bar{X})^2, \quad \text{df}_b = n - k, \quad \text{MSB} = \frac{SSB}{df_b} \tag{4}$$

$$\text{F} - \text{Statistic} = \frac{MSW}{MSB} \tag{5}$$

# 3 Results

Decision boundaries for various dimensionality reduction techniques are analyzed. These boundaries are generated by creating a 2D mesh grid, each point of which is fed to the classifier. The predicted classification is then color-coded and displayed. The known data points are then overlaid on the plot, and the mismatched colors emerge as those points that fall outside their corresponding decision boundaries. The decision boundaries for the four-dimensionality reduction techniques discussed are shown in Fig. 3.

The efficacy of four unsupervised dimensionality reduction methods, PCA, two KPCA methods, and MDS, and two supervised dimensionality reduction methods, namely ChiSq and ANOVA, are studied while keeping the learning algorithm fixed as the quadratic discrimination method. The first four methods are based on analyzing the distances between the data points, while the latter two are statistical methods based on class labels. In all cases, the entire data set is reduced to a lower dimension, two-dimensional (2D) space. The machine learning algorithm is next trained only with the 2D data. Training is carried out with k-fold cross-validation being used for model selection, with the best model next being used to obtain error rates for training and the test set and for computation of the *F1 score*.

```
1.    Reduce of 13-dimensional data vectors to two-dimensional vectors
2.    Split the two-dimensional data into Training and Test sets
3.    Use 10-fold cross-validation of Training Set to train
      a Quadratic Discriminant Classifier
4.    Compute Error Rates for Training and Test Sets and
      F1 Scores for Test Set
```

## 3.1 Confusion Matrices

In the fields of machine learning and statistics, a confusion matrix is a tabular representation that is used to visualize the efficacy of a classification system. It presents a contrast between the predicted labels of a data set and the actual labels.

The concept of a confusion matrix is also generalized to multiclass classification problems. For the classification problem that involves assigning a label from $\mathcal{M}$ classes to an input sample, the confusion matrix will $\mathcal{M} \times \mathcal{M}$ grid. Similar to the case of a two-class problem, where classifier performance statistics were defined to measure the efficacy of detecting be class labeled `1` and class labeled `0`, $\mathcal{M}$ performance statistics are computed to measure the efficacy of the classifier

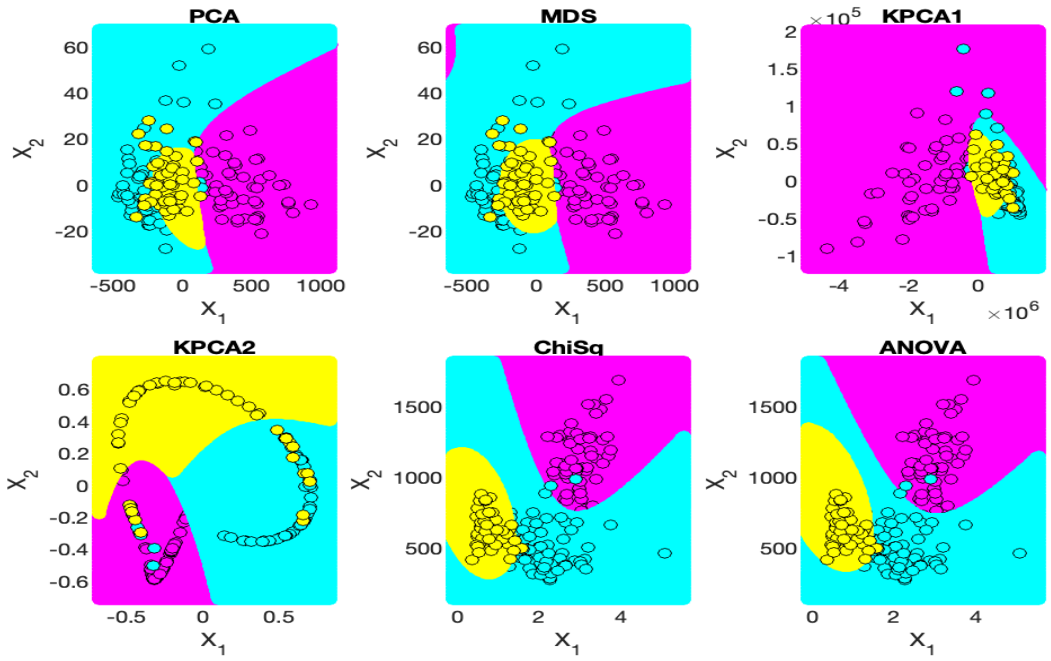
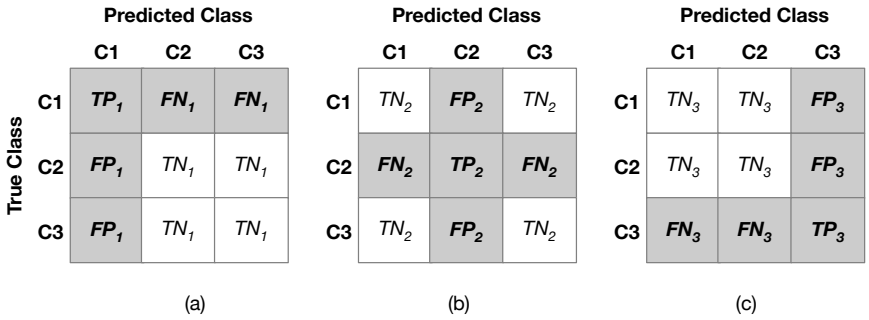



Figure 3: Two Dimensional Decision Surfaces for various dimensionality reduction algorithms.

in detecting each of the $\mathcal{M}$ classes.




$$\text{Precision}_c = \frac{TP_c}{TP_c + \sum FP_c} \tag{6}$$

$$\text{Recall}_c = \frac{TP_c}{TP_C + \sum FN_c} \tag{7}$$

$$\mathbf{F1_c} = 2 \cdot \frac{\text{Precision}_c \cdot \text{Recall}_c}{\text{Precision}_c + \text{Recall}_c} \tag{8}$$

Figure 4: Confusion Matrix with Three Classes

A confusion matrix for a three-class problem studied here is illustrated in Fig. 4. The $3 \times 3$ confusion matrix has elements $C_{i,j}$, with $1 \leq (i, j) \leq 3$, representing the number of instances of class $i$ that are labeled as class $j$, with diagonal elements of the confusion matrix representing the correct classification. Each of the figures Fig. 4(a), (b), and (c) are annotated from the perspective of calculating the precision and recall values for each of the three classes. Thus, the precision and recall values for each of the three classes $c$, with $1 \leq c \leq 3$, can be calculated as shown in Fig. 4.

## 3.2 Balancing Precision and Recall

Precision measures the accuracy of positive predictions. It is the ratio of correct positive predictions (true positives) to the total number of positive predictions (true positives plus false positives). In other terms, precision measures the proportion of positive cases predicted that were relevant or accurate.

However, recall emphasizes the comprehensiveness of positive predictions. It is the proportion of true positive predictions relative to the total number of actual positive cases (true positives plus false negatives). Recall evaluates the model's ability to identify as many relevant positive cases as possible.

To achieve a balance between precision and recall, a suitable compromise is often sought. When false positives are particularly costly, one may choose to prioritize precision. In other situations, it may be necessary to prioritize a high recall rate when it is important not to miss actual positive cases. Typically, a balance between these rates is obtained by adjusting the decision threshold of a classification model, with a higher decision threshold often favoring precision and a lowered threshold favoring recall.

Thus, precision and recall offer distinct perspectives on model performance, and an optimal equilibrium is determined by the application and the relative importance of avoiding false positives and false negatives. These measures are often combined by taking their harmonic mean of precision and recall.

The *F1-Score* is calculated taking the harmonic mean of precision and recall. The harmonic mean is calculated by obtaining the reciprocal of the arithmetic mean of a set of numbers' reciprocals. It is utilized when working with rates and widely used, for example, to determine the average speeds or to analyze the average of rates. Since the harmonic mean is the smallest of the three means and is highly influenced by the dataset's smaller values, the *F1-Score* is a conservative estimation of a classifier's performance.

$$\mathbf{F1} = \frac{2}{Precision^{-1} + Recall^{-1}} = 2 \cdot \frac{Precision \cdot Recall}{Precision + Recall}$$

## 3.3 Results

The benchmark results for the four dimensionality reduction techniques are included in Table 1. These results are graphically represented bar graphs in Fig. 5.

As is also evident in Fig. 3, the two geometrical methods, namely PCA and MDS, produce similar decision boundaries, as do the two statistical methods, namely the $\chi^2$ and ANOVA

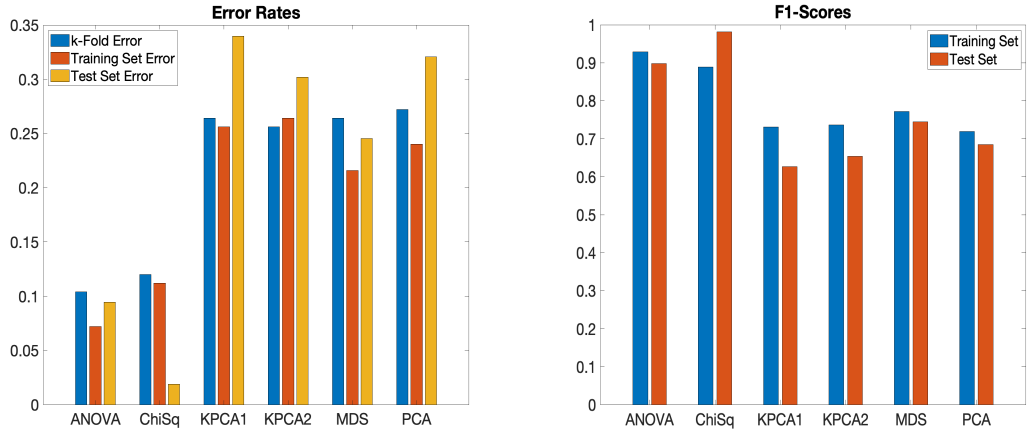



Figure 5: (a): Error rates for the various dimensionality reduction methods (b): *F1* Scores for the various dimensionality reduction methods.

Table 1: Results – Error Rates and F1 Scores.

| Method | kFold Error | Training Error | Test Error | Training F1 | Test F1 |
|---|---|---|---|---|---|
| **PCA** | 0.272 | 0.24 | 0.32 | 0.719 | 0.685 |
| **MDS** | 0.272 | 0.248 | 0.189 | 0.743 | 0.8 |
| **KPCA1** | 0.264 | 0.256 | 0.340 | 0.731 | 0.627 |
| **KPCA2** | 0.256 | 0.264 | 0.302 | 0.737 | 0.654 |
| **Chi-Sq** | 0.112 | 0.104 | 0.038 | 0.898 | 0.958 |
| **ANOVA** | 0.088 | 0.072 | 0.094 | 0.924 | 0.909 |

methods. The decision boundaries for KPCA2 method was somewhat unique. It should also be noted that the $\chi^2$ and ANOVA methods select feature #7 (flavonoid content) and feature #13 (proline content) as the two features to represent the data in 2D space. It should also be noted that the statistical methods produce slightly different decision boundaries and performance results only due to the random variations in the training and testing sets used in the two iterations for training the ML model.

Flavonoids possess several medicinal benefits, including anticancer, antioxidant, antiviral, and anti-inflammatory properties. Proline is an amino acid that helps in the formation of collagen, the regeneration of cartilage, the formation of connective tissue, the repair of skin damage and wounds, the healing of the gut lining, and the repair of joints. It is somewhat significant that these two ingredients, which are probably the only nutritionally significant components of wine, are also the most significant in determining its classification.

The performance of the geometrical methods (PCA and MDL) is comparable to each other and lower than the performance of the statistical methods ($\chi^2$ and ANOVA). It should be noted that both geometric methods are "unsupervised" methods. That is, a class label is not required for each sample point. In contrast, statistical methods are "supervised" methods that utilize associated class labels to derive a statistical inference protocol to select features. Finally, geometrical methods blend the feature values from all features to produce a lower-dimensional representation that is a linearly weighted combination of all feature values, while statistical methods select a smaller subset of features based on statistical criteria.

# 4 Conclusion

This paper reviewed four unsupervised methods (PCA, two KPCA methods, and MDS), and two supervised methods ($\chi^2$ and ANOVA) for feature reduction to alleviate the implications of the curse of dimensionality for building machine learning models for high-dimensional data. A 13-dimensional data set was reduced to a 2-D data set using each of these methods. The transformed data set was used to train a quadratic discriminant classifier, and the efficacy of machine learning for each transformation was analyzed. The performance of the ANOVA-based feature reduction exhibited superior performance and appeared promising, considering the cross-validation and testing error rates and the overall precision and accuracy measured with the $F1$ scores were the best.

# References

[1] George Argyrous and George Argyrous. The chi-square test for independence. *Statistics for Social Research*, pages 257–284, 1997.

[2] Visar Berisha, Chelsea Krantsevich, P Richard Hahn, Shira Hahn, Gautam Dasarathy, Pavan Turaga, and Julie Liss. Digital medicine and the curse of dimensionality. *NPJ digital medicine*, 4(1):153, 2021.

[3] David L Donoho et al. High-dimensional data analysis: The curses and blessings of dimensionality. *AMS math challenges lecture*, 1(2000):32, 2000.

[4] D Asir Antony Gnana, S Appavu Alias Balamurugan, and E Jebamalar Leavline. Literature review on feature selection methods for high-dimensional data. *International Journal of Computer Applications*, 136(1):9–17, 2016.

[5] Michael C Hout, Megan H Papesh, and Stephen D Goldinger. Multidimensional scaling. *Wiley Interdisciplinary Reviews: Cognitive Science*, 4(1):93–103, 2013.

[6] Donggun Kim and Kisung You. Pca, svd, and centering of data. 07 2023.

[7] Hae-Young Kim. Analysis of variance (anova) comparing means of more than two groups. *Restorative dentistry & endodontics*, 39(1):74–77, 2014.

[8] Tae Kyun Kim. Understanding one-way anova using conceptual figures. *Korean journal of anesthesiology*, 70(1):22–26, 2017.

[9] Mary L McHugh. The chi-square test of independence. *Biochemia medica*, 23(2):143–149, 2013.

[10] Amanda Ross and Victor L Willson. One-way anova. In *Basic and advanced statistical tests*, pages 21–24. Brill, 2017.

[11] Nasir Saeed, Haewoon Nam, Mian Imtiaz Ul Haq, and Dost Bhatti Muhammad Saqib. A survey on multidimensional scaling. *ACM Computing Surveys (CSUR)*, 51(3):1–25, 2018.

[12] Jonathon Shlens. A tutorial on principal component analysis. 04 2014.

[13] Alaa Tharwat. Principal component analysis-a tutorial. *International Journal of Applied Pattern Recognition*, 3(3):197–240, 2016.

[14] Michel Verleysen and Damien François. The curse of dimensionality in data mining and time series prediction. In *International work-conference on artificial neural networks*, pages 758–770. Springer, 2005.

[15] Minhaz Fahim Zibran. Chi-squared test of independence. *Department of Computer Science, University of Calgary, Alberta, Canada.*
*http://pages.cpsc.ucalgary.ca/ saul/wiki/uploads/CPSC681/topic-fahim-CHI-Square.pdf*, 1(1):1–7, 2007.