UNIT 13 FEATURE SELECTION AND EXTRACTION

13.1 INTRODUCTION

Data sets are made up of numerous data columns, which are also referred to as data attributes. These data columns can be interpreted as dimensions on an n-dimensional feature space, and data rows can be interpreted as points inside that space. One can gain a better understanding of a dataset by applying geometry in this manner. In point of fact, these characteristics are measurements of the same entity. It is possible for their existence in the algorithm's logic to get muddled, which will result in a change to how well the model functions.

Input variables are the columns of data that are fed into a model in order to provide a forecast for a target variable. However, if your data is given in the form of rows and columns, such as in a spreadsheet, then features is another term that can be used interchangeably with input variables. It is possible that the presence of a large number of dimensions in the feature space implies that the volume of that space is enormous. As a result, the points (data rows) in that space reflect a small and non-representative sample of the space's contents. It is possible for the performance of machine learning algorithms to degrade when there are an excessive number of input variables. The existence of an excessive number of input variables has a significant impact on the efficiency with which machine learning algorithms function. when it is used to data that contains a large number of input attributes; this phenomenon is referred to as the "curse of dimensionality." As a consequence of this, one of the most common goals is to cut down on the number of input features. The process of decreasing the number of dimensions that characterise a feature space is referred to as "dimensionality reduction," which is a phrase that was made up specifically to describe this phenomenon.

The usefulness of data mining can be hindered by an excessive amount of information on occasion. There are occasions when only a handful of the columns of data characteristics that have been compiled for the purpose of constructing and testing a model do not offer any information that is significant to the model. However, there are some that actually reduce the reliability and precision of the model.

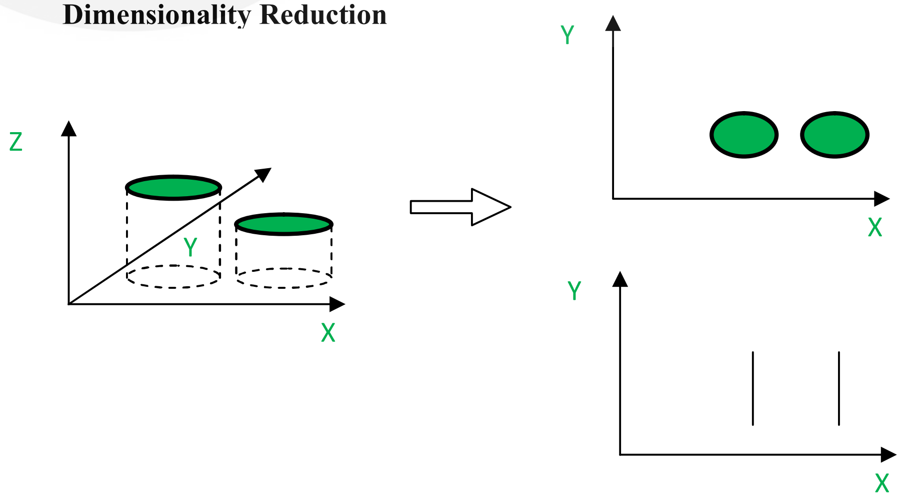
For instance, let's say you want to build a model that can forecast the incomes of people already employed in their respective fields. Therefore, data columns like cellphone number, house number, and so on will not truly contribute any value to the dataset, and they can therefore be omitted. This is because irrelevant qualities introduce noise to the data and affect the accuracy of the model. Additionally, because of the Noise, the size of the model as well as the amount of time and system resources required for model construction and scoring are both increased.

At this point in time, we are required to put the concept of Dimension Reductionality into practise. This can be done in one of two ways: either by selecting features to be extracted or by extracting features to be selected. Both of these approaches are broken down in greater detail below. The step of dimension reduction is one of the preprocessing phases that occurs during the process of data mining. This step is one of the preprocessing steps that may be beneficial in minimising the impacts of noise, correlation, and excessive dimensionality.

Some more examples are presented below to let you understand What does dimensionality reduction have to do with machine learning and predictive modelling?

- A simple issue concerning the classification of e-mails, in which we are tasked with deciding whether or not a certain email constitutes spam. can be brought up as a practical illustration of the concept of dimensionality reduction. This can include elements like whether or not the email has a standard subject line, the content of the email, whether or not it uses a template, and so on. However, some of these features may overlap with one another.
- A classification problem that involves humidity and rainfall can sometimes be simplified down to just one underlying feature as a result of the strong relationship that exists between the two variables. As a direct consequence of this, the number of characteristics could get cut down in some circumstances.
- A classification problem with three dimensions can be difficult to understand, whereas a problem with two dimensions can be translated to a fundamental space with two dimensions, and a problem with one dimension can be mapped to a line with one dimension. This concept is depicted in the diagram that follows, which shows how a three-dimensional feature space can be broken down into two one-dimensional feature spaces, with the number of features being reduced even further if it is discovered that they are related.

Dimensionality Reduction



In context of dimensionality reduction, various techniques like Principal Component Analysis, Linear Discriminant Analysis, Singular Value Decomposition are frequently used. In this unit we will discuss all the mentioned concepts, related to Dimension reductionality

13.2 Dimensionality Reduction

The Data mining and Machine Learning methodologies both have processing challenges when working with big amounts of data (many attributes). In point of fact, the dimensions of the feature space utilised by the approach, often referred to as the model attributes, play the most important function. Processing algorithms grow more difficult and time-consuming to implement as the dimensionality of the processing space increases.

These elements, also known as the model attributes, are the fundamental qualities, and they can either be variables or features. When there are more features, it is more difficult to see them all, and as a result, the work on the training set becomes more complex as well. This complexity was further increased when a significant number of characteristics were linked; hence, the classification became irrelevant as a result. In circumstances like these, the strategies for decreasing the number of dimensions can prove to be highly beneficial. In a nutshell, "the process of making a set of major variables from a huge number of random variables is what is referred to as dimension reduction." When conducting data mining, the step of dimension reduction can be helpful as a preprocessing step to lessen the negative effects of noise, correlation, and excessive dimensionality.

Dimension reduction can be accomplished in two ways :

- **Feature selection:** During this approach, a subset of the complete set of variables is selected; as a result, the number of conditions that can be utilised to illustrate the issue is narrowed down. It's normally done in one of three ways.:
  - Filter method
  - Wrapper method
  - Embedded method

- **Feature extraction:** It takes data from a space with many dimensions and transforms it into another environment with fewer dimensions.

13.2.1 Feature selection: It is the process of selecting some attributes from a given collection of prospective features, and then discarding the rest of the attributes that were considered. The use of feature selection can be done for one of two reasons: either to get a limited number of characteristics in order to prevent overfitting or to avoid having features that are redundant or irrelevant. For data scientists, the ability to pick features is a vital asset. It is essential to the success of the machine learning algorithm that you have a solid understanding of how to choose the most relevant features to analyse. Features that are irrelevant, redundant, or noisy can contaminate an algorithm, which can have a detrimental impact on the learning performance, accuracy, and computing cost. The importance of feature selection is only going to increase as the size and complexity of the typical dataset continues to balloon at an exponential rate.

Feature Selection Methods: Feature selection methods can be divided into two categories: supervised, which are appropriate for use with labelled data, and unsupervised, which are appropriate for use with unlabeled data. Filter methods, wrapper methods, embedding methods, and hybrid methods are the four categories that unsupervised approaches fall under.:

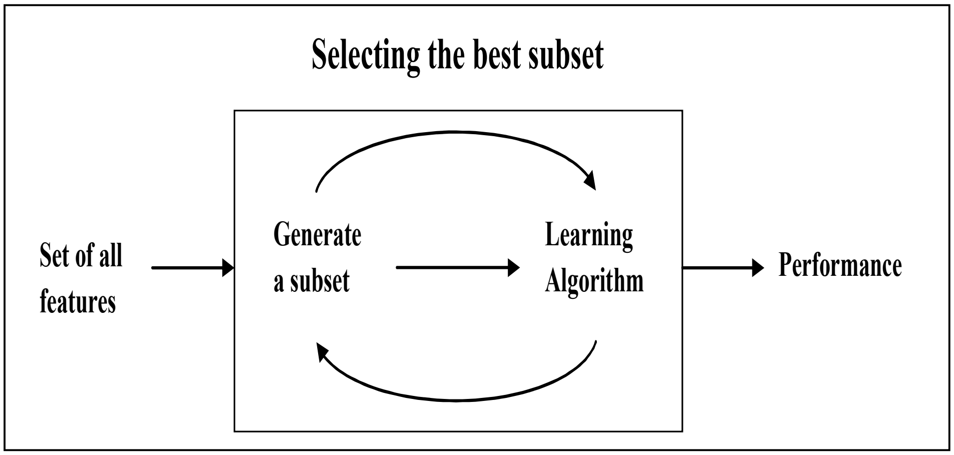
- **Filter methods**: Filter methods choose features based on statistics instead of how well they perform in feature selection cross-validation. Using a chosen metric, irrelevant attributes are found and recursive feature selection is done. Filter methods can be either univariate, in which an ordered ranking list of features is made to help choose the final subset of features, or multivariate, in which the relevance of all the features as a whole is evaluated to find features that are redundant or not important.

- **Wrapper methods**: Wrapper feature selection methods look at the choice of a set of features as a search problem. Their quality is judged by preparing, evaluating, and comparing a set of features to other sets of features. This method makes it easier to find possible interactions between variables. Wrapper methods focus on subsets of features that will help improve the quality of the results from the clustering algorithm used for the selection. Popular examples are Boruta feature selection and Forward feature selection.

- **Embedded methods:** Embedded feature selection approaches incorporate the feature selection machine learning algorithm as an integral component of the learning process. This allows for simultaneous classification and feature selection to take place within the method. Careful consideration is given to the extraction of the characteristics that will make the greatest contribution to each iteration of the process of training the model. A few examples of common embedded approaches are the LASSO feature selection algorithm, the random forest feature selection algorithm, and the decision tree feature selection algorithm.

Among all approaches the most conventional feature selection is feed forward feature selection.

Forward feature selection: The first step in the process of feature selection is to evaluate each individual feature and choose the one that results in the most effective algorithm model. This is referred to as "forward feature selection." After that step, each possible combination of the feature that was selected and a subsequent feature is analysed, and then a second feature is selected, and so on, until the required specified number of features is chosen. The operation of the forward feature selection algorithm is depicted here in the figure.



The procedure to follow in order to carry out forward feature selection

1. Train the model with each feature being treated as a separate entity, and then evaluate its overall performance.
2. Select the variable that results in the highest level of performance.
3. Carry on with the process while gradually introducing each variable.
4. The variable that produced the greatest amount of improvement is the one that gets kept.
5. Perform the entire process once more until the performance of the model does not show any meaningful signs of improvement.

Here, a fitness level prediction based on the three independent variables is used to show how forward feature selection works.

ID	Calories_burnt	Gender	Plays_Sport?	Fintess Level
1	121	M	Yes	Fit
2	230	M	No	Fit
3	342	F	No	Unfit
4	70	M	Yes	Fit
5	278	F	Yes	Unfit
6	146	M	Yes	Fit
7	168	F	No	Unfit
8	231	F	Yes	Fit
9	150	M	No	Fit
10	190	F	No	Fit

So, the first step in Forward Feature Selection is to train n models and judge how well they work by looking at each feature on its own. So, if you have three independent variables, we'll train three models, one for each of these three traits. Let's say we trained the model using the Calories Burned feature and the Fitness Level goal variable and got an accuracy of 87 percent.

ID	Calories_burnt	Gender	Plays_Sport?	Fintess Level
1	121	M	Yes	Fit
2	230	M	No	Fit
3	342	F	No	Unfit
4	70	M	Yes	Fit
5	278	F	Yes	Unfit
6	146	M	Yes	Fit
7	168	F	No	Unfit
8	231	F	Yes	Fit
9	150	M	No	Fit
10	190	F	No	Fit

Accuracy = 87%

We'll next use the Gender feature to train the model, and we acquire an accuracy of 80%. –

ID	Calories_burnt	Gender	Plays_Sport?	Fintess Level
1	121	M	Yes	Fit
2	230	M	No	Fit
3	342	F	No	Unfit
4	70	M	Yes	Fit
5	278	F	Yes	Unfit
6	146	M	Yes	Fit
7	168	F	No	Unfit
8	231	F	Yes	Fit
9	150	M	No	Fit
10	190	F	No	Fit

Accuracy = 80%

And similarly, the **Plays_sport** variable gives us an accuracy of **85%**–

ID	Calories_burnt	Gender	Plays_Sport?	Fintess Level
1	121	M	Yes	Fit
2	230	M	No	Fit
3	342	F	No	Unfit
4	70	M	Yes	Fit
5	278	F	Yes	Unfit
6	146	M	Yes	Fit
7	168	F	No	Unfit
8	231	F	Yes	Fit
9	150	M	No	Fit
10	190	F	No	Fit

Accuracy = 85%

At this point, we are going to select the variable that produced the most favourable results. If you take a look at this table, you'll notice that the variable titled "Calories Burned" alone has an accuracy rating of 87 percent, while the variable titled "Gender" has an accuracy rating of 80 percent, and the variable titled "Plays Sport" has an accuracy rating of 85 percent. When these two sets of data were compared, the winner was, unsurprisingly, the number of calories burned. As a direct result of this, we will select this variable.

Variable used	**Accuracy**
Calories_burnt	87.00%
Gender	80.00%
Plays_Sport?	85.00%

The next thing we'll do is repeat the previous steps, but this time we'll just add a single variable at a time. Because of this, it makes perfect sense for us to retain the Calories Burned variable as we proceed to add variables one at a time. Consequently, if we use gender as an illustration, we have an accuracy rate of 88 percent. –

ID	Calories_burnt	Gender	Plays_Sport?	Fintess Level
1	121	M	Yes	Fit
2	230	M	No	Fit
3	342	F	No	Unfit
4	70	M	Yes	Fit
5	278	F	Yes	Unfit
6	146	M	Yes	Fit
7	168	F	No	Unfit
8	231	F	Yes	Fit
9	150	M	No	Fit
10	190	F	No	Fit

Accuracy = 88%

We acquire a 91 percent accuracy when we combine Plays Sport with Calories Burnt. The variable that yields the greatest improvement will be kept. That makes natural sense. As you can see, when we combine Plays Sport with Calories Burnt, we get a better result. As a result, we'll keep it and use it in our model. We'll keep repeating the process till all the features are considered in improving the model performance

ID	Calories_burnt	Gender	Plays_Sport?	Fintess Level
1	121	M	Yes	Fit
2	230	M	No	Fit
3	342	F	No	Unfit
4	70	M	Yes	Fit
5	278	F	Yes	Unfit
6	146	M	Yes	Fit
7	168	F	No	Unfit
8	231	F	Yes	Fit
9	150	M	No	Fit

10	190	F	No	Fit

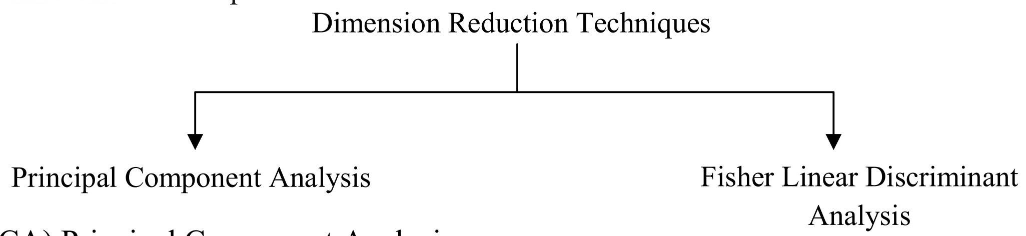
Accuracy = 91%

13.2.2 Feature extraction:

The process of reducing the amount of resources needed to describe a large amount of data is called "feature extraction." One of the main problems with doing complicated data analysis is that there are a lot of variables to keep track of. A large number of variables requires a lot of memory and processing power, and it can also cause a classification algorithm to overfit to training examples and fail to generalise to new samples. Feature extraction is a broad term for different ways to combine variables to get around these problems while still giving a true picture of the data. Many people who work with machine learning think that extracting features in the best way possible is the key to making good models. The data's information must be shown by the features in a way that fits the needs of the algorithm that will be used to solve the problem. Some "inherent" features can be taken straight from the raw data, but most of the time, we need to use these "inherent" features to find "relevant" features that we can use to solve the problem.

In simple terms *"feature extraction."* can be described as a technique for *Defining a set of features, or visual qualities, that best show the information.* Feature Extraction Techniques such as: PCA, ICA, LDA, LLE, t-SNE and AE. are some of the common examples in machine learning.

Feature extraction fills the following requirements:
It takes raw data, called features, and turns them into useful information by reformatting, combining, and changing the primary features into new ones. This process continues until a new set of data is created that the Machine Learning models can use to reach their goals.
Methods of Dimensionality Reduction : The following are two well-known and widely used dimension reduction techniques:


Dimension Reduction Techniques
Principal Component Analysis
Fisher Linear Discriminant Analysis

- (PCA) Principal Component Analysis
- (LDA) Fisher Linear Discriminant Analysis

The reduction of dimensionality can be linear or non-linear, depending on the method used. The most common linear method is called Principal Component Analysis, or PCA.

Check Your Progress - 1

Qn1. Define the term feature selection.

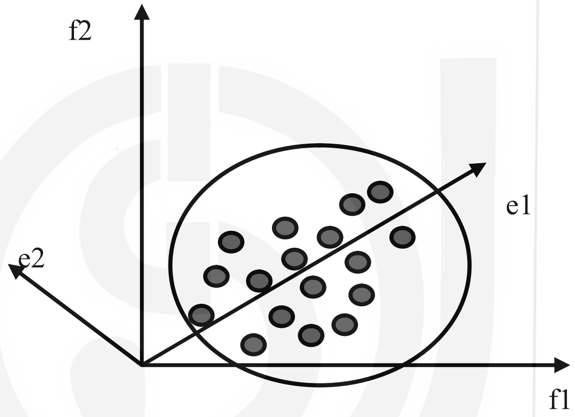
Qn2. What is the purpose of feature extraction in machine learning?

Qn3. Expand the following terms : PCA,LDA,GDA

Qn4. Name components of dimensionality reduction.

13.3 Principal Component Analysis

Karl Pearson was the first person to come up with this plan. It is based on the idea that when data from a higher-dimensional space is put into a lower-dimensional space, the lower-dimensional space should have the most variation. In simple terms, principal component analysis (PCA) is a way to get important variables (in the form of components) from a large set of variables in a data set. It tends to find the direction in which the data is most spread out. PCA is more useful when you have data with three or more dimensions.



When applying the PCA method, the following are the primary steps that should be followed:

1. Obtain the dataset you need.
2. Calculate the mean of the vectors ().
3. Deduct the mean of the given data from the total.
4. Complete the computation for the covariance matrix.
5. Determine the eigenvectors and eigenvalues of the matrix that represents the covariance matrix.
6. Creating a feature vector and deciding which components would be the major ones i.e. the principal components.
7. Create a new data set by projecting the weight vector onto the dataset.As a result, we have a smaller number of eigenvectors, and some data may have been lost in the process. However, the remaining eigenvectors should keep the most significant variances.

Merits of Dimensionality Reduction

- It helps to compress data, which reduces the amount of space needed to store it and the amount of time it takes to process it.

- If there are any redundant features, it also helps to get rid of them.

Limitations of Dimensionality Reduction

- You might lose some data.
- You might lose some data.
- PCA fails when the mean and covariance are not enough to describe a dataset.
- We don't know how many major parts we need to keep track of, but in practice, we follow some rules.

Below is the practice question for Principal Component Analysis (PCA) :

Problem-01: 2, 3, 4, 5, 6, 7; 1, 5, 3, 6, 7, 8 are the given data. Using the PCA Algorithm, calculate the primary component.

OR

Consider the two-dimensional patterns (2, 1), (3, 5), (4, 3), (5, 6), (6, 7), (8, 8) and (9, 10). (7, 8).

Using the PCA Algorithm, calculate the primary component.

OR

Calculate the principal component of following data-

Class1	**values**	**Class 2**	**values**
X	**2,3,4**	**X**	**5 , 6 , 7**
Y	**1,5,3**	**Y**	**6 , 7 , 8**

Answer :

Step-1: Get data.

The given feature vectors are- x1,x2,x3,x4,x5,x6 with the values given below:

$$\begin{bmatrix}2\\1\end{bmatrix} \begin{bmatrix}3\\5\end{bmatrix} \begin{bmatrix}4\\3\end{bmatrix} \begin{bmatrix}5\\6\end{bmatrix} \begin{bmatrix}6\\7\end{bmatrix} \begin{bmatrix}7\\8\end{bmatrix}$$

Step-2:

Find the mean vector (μ).

Mean vector (μ) = ((2 + 3 + 4 + 5 + 6 + 7) / 6, (1 + 5 + 3 + 6 + 7 + 8) / 6)= (4.5, 5)

Thus, Mean vector (μ) $= \begin{bmatrix}4.5\\5\end{bmatrix}$

Step-03:

On subtracting mean vector (μ) from the given feature vectors.

- $x_1 - \mu = (2 - 4.5, 1 - 5) = (-2.5, -4)$

same for others

Feature vectors (x_i) generated after subtraction are $\begin{bmatrix}-2.5\\-4\end{bmatrix}\begin{bmatrix}-1.5\\0\end{bmatrix}\begin{bmatrix}-0.5\\-2\end{bmatrix}\begin{bmatrix}0.5\\1\end{bmatrix}\begin{bmatrix}1.5\\2\end{bmatrix}\begin{bmatrix}2.5\\3\end{bmatrix}$

Step-04:

Now to find covariance matrix : Covariance Matrix $= \frac{\sum(X_i-\mu)(X_i-\mu)^t}{n}$

$$m_1 = (x_1-\mu)(x_1-\mu)^t = \begin{bmatrix}-2.5\\-4\end{bmatrix}\begin{bmatrix}-2.5 & -4\end{bmatrix} = \begin{bmatrix}6.25 & 10\\10 & 16\end{bmatrix}$$

$$m_2 = (x_2-\mu)(x_2-\mu)^t = \begin{bmatrix}-1.5\\0\end{bmatrix}\begin{bmatrix}-1.5 & 0\end{bmatrix} = \begin{bmatrix}2.25 & 0\\0 & 0\end{bmatrix}$$

$$m_3 = (x_3-\mu)(x_3-\mu)^t = \begin{bmatrix}-0.5\\-2\end{bmatrix}\begin{bmatrix}-0.5 & -2\end{bmatrix} = \begin{bmatrix}0.25 & 1\\1 & 4\end{bmatrix}$$

$$m_4 = (x_4-\mu)(x_4-\mu)^t = \begin{bmatrix}0.5\\1\end{bmatrix}\begin{bmatrix}0.5 & 1\end{bmatrix} = \begin{bmatrix}0.25 & 0.5\\0.5 & 4\end{bmatrix}$$

$$m_5 = (x_5-\mu)(x_5-\mu)^t = \begin{bmatrix}1.5\\2\end{bmatrix}\begin{bmatrix}1.5 & 2\end{bmatrix} = \begin{bmatrix}2.25 & 3\\3 & 4\end{bmatrix}$$

$$m_6 = (x_6-\mu)(x_6-\mu)^t = \begin{bmatrix}2.5\\3\end{bmatrix}\begin{bmatrix}2.5 & 3\end{bmatrix} = \begin{bmatrix}6.25 & 7.5\\7.5 & 9\end{bmatrix}$$

$$\text{Covariance Marix} = \frac{1}{6}\begin{bmatrix}17.5 & 22\\22 & 34\end{bmatrix}$$

$$\text{Covariance Matrix} = \begin{bmatrix}2.92 & 3.67\\3.67 & 5.67\end{bmatrix}$$

Step-05:

Eigen values and Eigen vectors of the covariance matrix.

$$\begin{vmatrix}2.92 & 3.67\\3.67 & 5.67\end{vmatrix} - \begin{vmatrix}\lambda & 0\\0 & \lambda\end{vmatrix} = 0$$

$$\begin{vmatrix}2.92-\lambda & 3.67\\3.67 & 5.67-\lambda\end{vmatrix} = 0$$

From here,

$$(2.92-\lambda)(5.67-\lambda) - (3.67 \text{x}\, 3.67) = 0$$

$$16.56 - 2.92\lambda - 5.67\lambda + \lambda^2 - 13.47 = 0$$

$\lambda^2 - 8.56\lambda + 3.09 = 0$

Solving this quadratic equation, we get $\lambda = 8.22, 0.38$

This, two eigen values are $\lambda_1 = 8.22$ and $\lambda_2 = 0.38$.

Clearly, the second eigen value is vary small compared to the first eigen value.

So, the second eigen vactor can be left out.

Eigen vector corresponding to the greatest eigen value is the principle component for the given data set.

So, we find the eigen vector corresponding to eigen value λ_1.

We use the following equation to find the eigen vector-

$$\mathrm{MX} = \lambda\mathrm{X}$$

Where-

- $M = Covariance\ Matrix\ ;\ X = Eigen\ vector\ , and\ \ \lambda = Eigen\ value$

Substituting the values in the above equation, we get-

On being substituting the values in the above equation, we get-

$$\begin{bmatrix} 2.92 & 3.67 \\ 3.67 & 5.67 \end{bmatrix} \begin{bmatrix} \mathrm{X1} \\ \mathrm{X2} \end{bmatrix} = 8.22 \begin{bmatrix} \mathrm{X1} \\ \mathrm{X2} \end{bmatrix}$$

Solving these, we get-

$2.92X_1 + 3.67X_2 = 8.22X_1$

$3.67X_1 + 5.67X_2 = 8.22X_2$

On simplification, we get-

$5.3X_1 = 3.67X_2 \dots\dots\dots (1)$

$3.67X_1 = 2.55X_2 \dots\dots\dots (2)$

From (1) and (2), $X_1 = 0.69X_2$

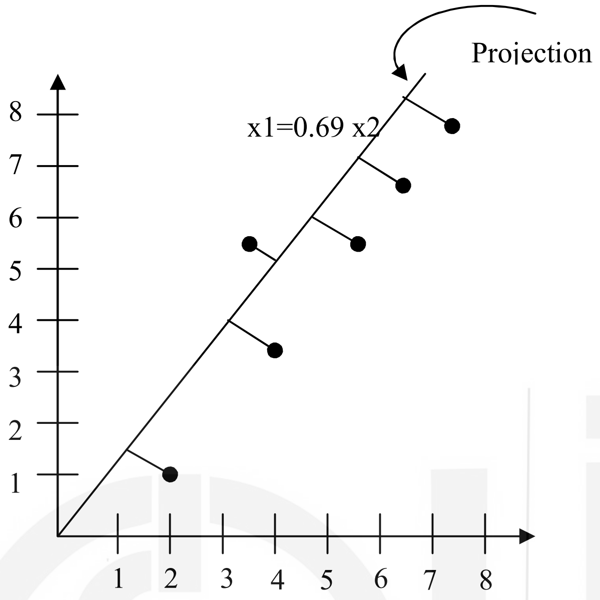
From (2), the eigen vector is-

Eigen Vector : $\begin{bmatrix} \mathrm{X1} \\ \mathrm{X2} \end{bmatrix} = \begin{bmatrix} 2.55 \\ 3.67 \end{bmatrix}$

Thus, PCA for the given problem is

Principle Component: $\begin{bmatrix} X1 \\ X2 \end{bmatrix} = \begin{bmatrix} 2.55 \\ 3.67 \end{bmatrix}$

Lastly, we project the data points onto the new subspace as-



Problem -02

Use PCA Algorithm to transform the pattern (2, 1) onto the eigen vector in the previous question.

Solution-

The given feature vector is (2, 1) i.e. Given Feature Vector: $\begin{bmatrix} 2 \\ 1 \end{bmatrix}$

The feature vector gets transformed to :

= Transpose of Eigen vector x (Feature Vector – Mean Vector)

$$= \begin{bmatrix} 2.55 \\ 3.67 \end{bmatrix}^T x \left(\begin{bmatrix} 2 \\ 1 \end{bmatrix} - \begin{bmatrix} 4.5 \\ 5 \end{bmatrix} \right) = \begin{bmatrix} 2.55 & 3.67 \end{bmatrix} x \begin{bmatrix} -2.5 \\ -4 \end{bmatrix} = -21.055$$

Check Your Progress -3

Qn1. What are the advantages of dimensionality reduction?

Qn2. What are the disadvantages of dimensionality reduction?

13.4 Linear Discriminant Analysis

In most cases, the application of logistic regression has been restricted to problems involving two classes of subjects. On the other hand, the Linear Discriminant Analysis is the linear classification method that is recommended to use when there are more than two classes.

The algorithm for linear classification known as logistic regression is known for being both straightforward and robust. On the other hand, there are a few restrictions or faults in the system that highlight the requirement for more complex linear classification algorithms. The following is a list of some of the problems:

- **Binary class Problems**. Concerns regarding the binary class is that the Logistic regression is utilised for issues that involve binary classification or two classes. It is possible to enhance it such that it can manage multiple-class categorization, but in practise, this is not very common.
- **Unstable, but with well-defined classes**. When the classes are extremely distinct from one another, logistic regression may become unstable.
- It is prone to instability when there are only a few occurrences. When there are not enough examples from which to draw conclusions about the parameters, the logistic regression model may become unstable.

In view of the limitations of logistic regression that were discussed earlier, the linear discriminant analysis is one of the prospective linear methods that can be used for multi-class classification. This is because it addresses each of the aforementioned concerns in their totality, which is the primary reason for its success (i.e. flaws of logistic regression). When dealing with issues that include binary categorization, two statistical methods that could be effective are logical regression and linear discriminant analysis. Both of these techniques are linear and regression-based.

Understanding LDA Models : In order to simplify the analysis of your data and make it more accessible, LDA will make the following assumptions about it:

1. The distribution of your data is Gaussian, and when plotted, each variable appears to be a bell curve.

2. Each feature has the same variance, which indicates that the values of each feature vary by the same amount on average in relation to the mean.

On the basis of these presumptions, the LDA model generates estimates for both the mean and the variance of each class. In the case where there is only one input variable, which is known as the univariate scenario, it is straightforward to think about this.

When the sum of values is divided by the total number of values, we are able to compute the mean value, or mu, of each input, or x, for each class(k), in the following manner.

$$mu_k = 1/nk * sum(x)$$

Where,
mu$_k$ represents the average value of x for class k and
nk represents the total number of occurrences that belong to class k.
When calculating the variance across all classes, the average squared difference of each individual result's distance from the mean is employed.

*sigma^2 = 1 / (n-K) * sum((x – mu)^2)*

Where sigma^2 represents the variance of all inputs (x), n represents the number of instances, K represents the number of classes, and mu is the mean for input x.

Now we will discuss How to use LDA to Make Predictions ?

LDA generates predictions by calculating the likelihood that each class will be given a fresh batch of data and then extrapolating from there. A forecast is created by selecting the output class that contains the events that are the most likely to occur. The Bayes Theorem is incorporated into the model in order to calculate the probabilities involved. Utilizing the likelihood of each class as well as the probability of data belonging to that class, Bayes's Theorem may be utilised to estimate the probability of the output class (k) given the input class (x). This is accomplished by using the following formula:

P(Y=x|X=x) = (PIk * fk(x)) / sum(PIl * fl(x))

The base probability of each class (k) that can be found in your training data is denoted by the symbol PIk (e.g. 0.5 for a 50-50 split in a two class problem). This concept is referred to as the prior probability within Bayes' Theorem.

PIk = nk/n

The value of f, which represents the estimated likelihood that x is a member of the class, is presented here as f(x). We make use of a Gaussian distribution function for the variable (x). By simplifying the previous equation and then introducing the Gaussian, we are able to arrive at the equation that is presented below. This type of function is referred to as a discriminate function, and the output classification (y) is determined by selecting the category that contains the greatest value:

D$_k$(x) = x * (mu$_k$/sigma^2) – (mu$_k$^2/(2*sigma^2)) + ln(PIk)

Where, D$_k$(x) is the discriminating function for class k given input x, and mu$_k$, sigma^2, and PIk are all estimated from your data.

Now to perform the above task we need to prepare our data first, so the question arises, How to prepare data suitable for LDA?

This section gives you some ideas to think about when getting your data ready to use with LDA.

Problems with Classification: LDA is used to solve classification problems where the output variable is a categorical one. This may seem obvious, as LDA works with both two and more than two classes.

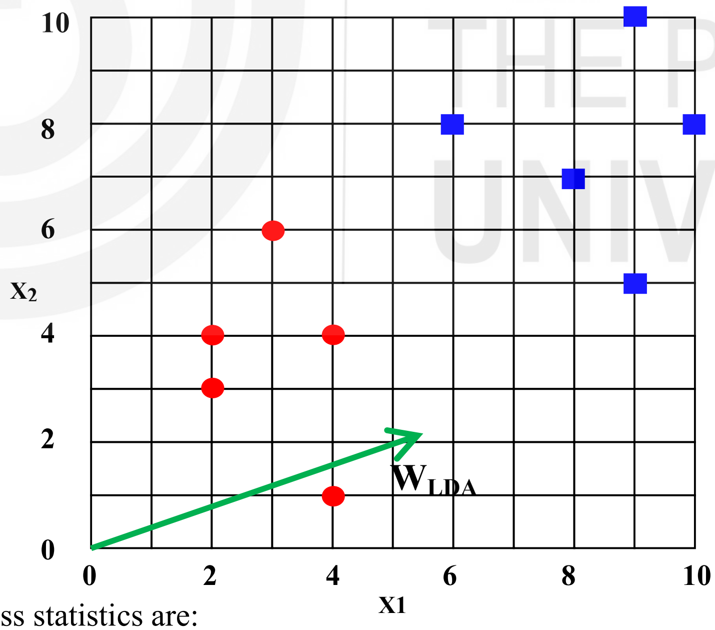
Gaussian Distribution: The standard way to use the model assumes that the input variables have a Gaussian distribution. Think about looking at the univariate distributions of each attribute and using transformations to make them look more like Gaussian distributions (e.g. log and root for exponential distributions and Box-Cox for skewed distributions).

Remove Outliers: Think about removing outliers from your data. These things can mess up the basic statistics like the mean and the standard deviation that LDA uses to divide classes.

Same Variance: LDA assumes that the variance of each input variable is the same. Before using LDA, you should almost always normalise your data so that it has a mean of 0 and a standard deviation of 1.

Below is a practice problems based on Linear Discriminant Analysis (LDA) -

Problem-2 : Compute the Linear Discriminant projection for the following two-dimensional datasetX1=(x1,x2)={(4,1),(2,4),(2,3),(3,6),(4,4)} & X2=(x1,x2)={(9,10),(6,8),(9,5),(8,7),(10,8)}



- The class statistics are:

$$S_1 = \begin{bmatrix} 0.80 & -0.40 \\ -0.40 & 2.60 \end{bmatrix}: S_2 = \begin{bmatrix} 1.84 & -0.04 \\ -0.04 & 2.64 \end{bmatrix}$$

$$\mu_1 = [3.00 \quad 3.60]: \mu_2 = [8.40 \quad 7.60]$$

- The within- and between-class scatter are

$$S_1 = \begin{bmatrix} 29.16 & 21.60 \\ 21.60 & 16.00 \end{bmatrix}; \ S_2 = \begin{bmatrix} 2.64 & -0.44 \\ -0.44 & 5.28 \end{bmatrix}$$

- The LDA projection is then obtained as the solution of the generalized eigenvalue problem

$$S_w^{.1} S_B v = \lambda v \Rightarrow \left| S_w^{.1} S_B - \lambda l \right| | = 0 \Rightarrow | \begin{vmatrix} 11.89 & 8.81 \\ 5.08 & 3.76 - \lambda \end{vmatrix} = 0 \Rightarrow \lambda = 15.65$$

$$\begin{bmatrix} 11.89 & 8.81 \\ 5.08 & 3.76 \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \end{bmatrix} = 15.65 \begin{bmatrix} v_1 \\ v_2 \end{bmatrix} \Rightarrow \begin{bmatrix} v_1 \\ v_2 \end{bmatrix} = \begin{bmatrix} 0.91 \\ 0.39 \end{bmatrix}$$

- On directly by

$$w^* = S_w^{-1}(\mu_1 - \mu_2) = [-0.91 \quad -0.39]^T$$

Check Your Progress -3

Q1. Define LDA.

Q2. Write any two limitations of LDA.

13.5 Single Value Decomposition

The Singular Value Decomposition (SVD) method is a well-known technique for decomposing a matrix into a large number of component matrices. This method is valuable since it reveals many of the interesting and helpful characteristics of the initial matrix. We can use SVD to discover the optimal lower-rank approximation to the matrix, determine the rank of the matrix, or test a linear system's sensitivity to numerical error.

Singular value decomposition is a method of decomposing a matrix into three smaller matrices.

$$\mathbf{A = USV^T}$$

Where:

- A : is an $m \times n$ matrix
- U : is an $m \times n$ *orthogonal* matrix
- S : is an $n \times n$ *diagonal matrix*
- V : is an $n \times n$ orthogonal matrix

The rationale for the transposition of the last matrix will be revealed later in the presentation. Also defined (in case your algebra is rusty) will be the word "orthogonal," as well as it describes the reason for having two outer matrices having the same feature.

The diagonal matrix, S, has been flattened into a vector, reducing the formula into a single summation. Singular values, or Si, are variables that are generally organized from largest to smallest.

Below is a practice problems based on single value decomposition

Problem-03: Find the SVD of the matrix $A = \begin{bmatrix} -3 & 1 \\ 6 & -2 \\ 6 & -2 \end{bmatrix}$

Solution : First, we'll work with $A^T A = \begin{bmatrix} 81 & -27 \\ -27 & 9 \end{bmatrix}$. The eigen values are $\lambda = 0,\ 90.$

For $\lambda = 0,$ the reduced matrix is $\begin{bmatrix} 1 & -1/3 \\ 0 & 0 \end{bmatrix}$, so $\mathrm{v} = \frac{1}{\sqrt{10}} \begin{bmatrix} 1 \\ 3 \end{bmatrix}$

For $\lambda = 90,$ the reduced matrix is $\begin{bmatrix} 1 & 3 \\ 0 & 0 \end{bmatrix}$, so $\mathrm{v} = \frac{1}{\sqrt{10}} \begin{bmatrix} -3 \\ 1 \end{bmatrix}$

Now, we can find the reduced SVD right away since $u_1 = \frac{1}{\sigma_1} AV_1 = \frac{1}{3} \begin{bmatrix} 1 \\ -2 \\ -2 \end{bmatrix}$

We now need a basis for the null space of

$$AA^T = \begin{bmatrix} 10 & -20 & -20 \\ -20 & 40 & 40 \\ -40 & 40 & 40 \end{bmatrix} \sim \begin{bmatrix} 1 & -2 & -2 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \Rightarrow \left\{ \begin{bmatrix} 2 \\ 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 2 \\ 0 \\ 1 \end{bmatrix} \right\}$$

Now the full SVD is given by:

$$A = \begin{bmatrix} -3 & 1 \\ 6 & -2 \\ 6 & -2 \end{bmatrix} = \begin{bmatrix} 1/3 & 2/\sqrt{5} & 2/\sqrt{5} \\ -2/3 & 1/\sqrt{5} & 0 \\ -2/3 & 0 & 1/\sqrt{5} \end{bmatrix} \begin{bmatrix} 2/\sqrt{10} & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} -3/\sqrt{10} & 1/\sqrt{10} \\ 1/\sqrt{10} & 3/\sqrt{10} \end{bmatrix}^T$$

Check Your Progress - 4

Qn1. Define SVD

13.6 SUMMARY

In this unit we learned about the concept of Dimension Reductionality, wherein we understood basics of Feature Selection and Feature extraction techniques. Thereafter an explicit discussion of Principal Component Analysis(PCA), Linear Discriminant Analysis(LDA) and Singular Value Decomposition(SVD) was given.

13.7 SOLUTIONS TO CHECK YOUR PROGRESS

Check Your Progress – 1

Refer Section 13.2 for detailed Solutions

Ans1. In machine learning and statistics, feature selection is the process of choosing a subset of relevant features to use when making a model. It is also called variable selection, attribute selection, or variable subset selection.

Ans2. The goal of Feature Extraction is to reduce the number of features in a dataset by making new features from the ones that are already there (and then discarding the original features).

Refer Section 13.2 for detailed Solutions

Ans3.Principal Component Analysis (PCA), Linear Discriminant Analysis (LDA) and Generalized Discriminant Analysis (GDA)

Ans4. Feature selection and feature extraction.

Check Your Progress - 2

Refer Section 13.3 for detailed Solutions

Ans1. Advantages of dimensionality reduction

- It reduces the time and space complexity.
- Getting rid of multi-colinearity makes it easier to understand how the machine learning model's parameters work.
- When there are only two or three dimensions, it's easier to see the data.

Ans2. Dimensionality Reduction's Drawbacks

- PCA likes to find linear relationships between variables, which isn't always a good thing.
- PCA doesn't work when the mean and covariance aren't enough to describe a dataset.
- In some problems, dimension reduction could also cause data loss.

Check Your Progress - 3

Refer Section 13.4 for detailed Solutions

Ans1. The LDA technique is a multi-class classification method that may be utilised to automatically perform dimensionality reduction. LDA cuts the number of features down from the initial number of features.

LDA projects the data into a new linear feature space, and obviously, the classifier will have a high level of accuracy if the data can be linearly separated.**Ans2**. Some of the limitations of Logistic Regression are as follows:

- Logistic regression is typically applied when attempting to solve problems involving binary or two-class classifications. Problems involving two classes Even while it is possible to extrapolate this information and apply it for multi-class classification, very few people actually do this. On the other hand, Linear Discriminant Analysis is regarded as a superior option whenever multi-class classification is necessary, and in the case of binary classifications, both logistic regression and LDA are utilised in the analysis process.

- Instability with Clearly Delineated Social Groups — Logistic Regression is known to be unreliable in situations when the classes are clearly differentiated from one another.

Check Your Progress - 4

Refer Section 13.5 for detailed Solutions

Ans1 The singular value decomposition is a factorization technique that can be used in linear algebra for real or complex matrices. It extends the application of the eigen decomposition of a square normal matrix to any m x n matrix by using an ortho normal eigen basis. It has something to do with the polar decomposition.

13.8 FURTHER READINGS

1. Machine learning an algorithm perspective, Stephen Marshland, 2nd Edition, CRC Press, 2015.
2. Machine Learning, Tom Mitchell, 1st Edition, McGraw- Hill, 1997.
3. Machine Learning: The Art and Science of Algorithms that Make Sense of Data, Peter Flach, 1st Edition, Cambridge University Press, 2012.