Dimensionality & Feature Reduction

Present databases are capable of enduring various data types. The multi-dimensionality of data entries makes a single record accommodate a great number of fields (attributes, features). Other problems can come from the insufficient amount of instances for sampling or the size of sampling data remains tremendous. Hence, feature reduction becomes an alternative, especially when the instance dimensionality is as big as hundreds or, even more, as thousands. Combination of several homogenous data sources or collaborating different datasets derived from the same theme can constitute high-dimensional datasets with ease. Feature reduction takes the responsibility in generating a smaller subset of essential classification factors.

To authenticate precisely discriminative features from others, proper data preprocessing is important. Good data preprocessing can refine the discriminative power of features while improper data preprocessing are likely to cause failure in discovering pivotal features. Generally, variety makes continuous data more controversial than discrete (nominal or categorical) data. Variety simplification for continuous data is commonly required before executing feature evaluation. Discretization is the process of transferring continuous data into discrete counterparts. Discretization can preserve the original data distribution.

The first phase of the data mining process is creating a set of features that the analyst can work with. In cases where the data is in raw and unstructured form (e.g., raw text, sensor signals), the relevant features need to be extracted for processing. In other cases where a heterogeneous mixture of features is available in different forms, an "off-the-shelf" analytical approach is often not available to process such data. In such cases, it may be desirable to transform the data into a uniform representation for processing. This is referred to as *data type porting*.

The first phase of feature extraction is a crucial one, though it is very application specific. In some cases, feature extraction is closely related to the concept of data type portability, where low-level features of one type may be transformed to higher-level features of another type. The nature of feature extraction depends on the domain from which the data is drawn: *Sensor data*, *Image data*, *Web logs*, *Network traffic*, *Document data, etc*.

Usage of appropriate data clustering schemes for accurate preprocessing of the raw feature values. A new heuristic algorithm that facilitates the selection of a compact subset of minimal-redundancy and maximal-relevance features.

Feature reduction (or data dimensionality reduction, DDR) is broadly categorized into feature transform (or feature extraction) and feature selection. Feature extraction can transform input data into a reduced representation set of factors. It is expected that the factor set will extract the relevant information from the input data to facilitate the classification task using this reduced representation instead of the full-size input.

Feature selection also known as variable selection, is the technique of removing nearly all irrelevant and redundant features from the original feature set. The selected features in the reduced set take the major responsibility of the classification work. Feature selection has many advantages such as alleviating the effect of the curse of dimensionality and enhancing generalization capability. In addition, feature selection can benefit classification performance by speeding up the learning process and improving model interpretability.

In feature selection, it is well known that the combinations of individual discriminative features do not necessarily lead to good classification performance. It is difficult to collect a subset of relevant features with respect to the class concept when they are totally independent. Redundancy among features is very common. Existing feature selection methods mainly exploit two strategies to reduce such problem:

individual evaluation and subset evaluation. Individual evaluation ranks features according to their importance in differentiating instances of different classes. It can only remove irrelevant features as redundant ones once features are ranked as similar. The conventional methods include information entropy and the Gini index. Subset evaluation searches for a compact subset of features that satisfies some goodness measure and can exclude irrelevant features as well as redundant ones.

Linear Regression

The modeling of the relationship between a response variable and a set of explanatory variables is one of the most widely used of all statistical techniques. We refer to this type of modeling as regression analysis. A regression model provides the user with a functional relationship between the response variable and explanatory variables that allows the user to determine which of the explanatory variables have an effect on the response. The regression model allows the user to explore what happens to the response variable for specified changes in the explanatory variables.

In this, there is a single independent variable and the equation for predicting a dependent variable *y* is a linear function of a given independent variable *x*.

In general, we write the prediction equation as:

$$\hat{y} = \hat{\beta}_0 + \hat{\beta}_1 * x$$

Where $\hat{\beta}_0$ is the intercept and $\hat{\beta}_1$ is the slope of the line.

The basic idea of simple linear regression is to use data to fit a prediction line that relates a dependent variable *y* and a single independent variable *x*. The first assumption in simple regression is that the relation is, in fact, linear. According to the **assumption of linearity,** the slope of the equation does not change as *x* changes. In the road resurfacing example, we would assume that there were no (substantial) economies or diseconomies from projects of longer mileage. There is little point in using simple linear regression unless the linearity assumption makes sense (at least roughly).

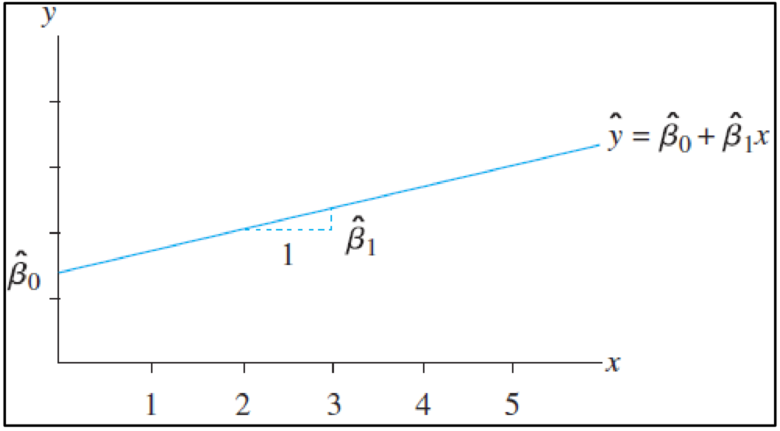


FIGURE: Linear prediction function

Linearity is not always a reasonable assumption, on its face. Assuming linearity, we would like to write *y* as a linear function of *x*: $y = \beta_0 + \beta_1 x$.

However, according to such an equation, *y* is an exact linear function of *x*; no room is left for the inevitable errors (deviation of actual *y* values from their predicted values). Therefore, corresponding to each *y* we introduce a **random error term** *i* and assume the model:

$$y = \beta_0 + \beta_1 x + \varepsilon$$

We assume the random variable *y* to be made up of a predictable part (a linear function of *x*) and an unpredictable part (the random error *i*). The coefficients and are interpreted as the true, underlying intercept and slope. The error term includes the effects of all other factors, known or unknown. In the road resurfacing project, unpredictable factors such as strikes, weather conditions, and equipment breakdowns would contribute to, as would factors such as hilliness or prerepair condition of the road—factors that might have been used in prediction but were not. The combined effects of unpredictable and ignored factors yield the random error terms.

Demerits of Linear Regression:

- Probabilities are bounded whereby 0≤p≤1.
- Probabilities should always be positive (>0) and less than 1.
- We need to design a function which satisfies the above characteristics.

Logistic Regression (or logit model or logit regression)

In these models, the dependent variables are categorical.

Types of Logistic Regression Models:

- **Binary (Dependent Variable is binary):** Used to estimate the probability of a binary response based on one or more predictor (or independent) variables (or features).

- **Multinomial** (Dependent Variable has more than 2 categories of outcomes)

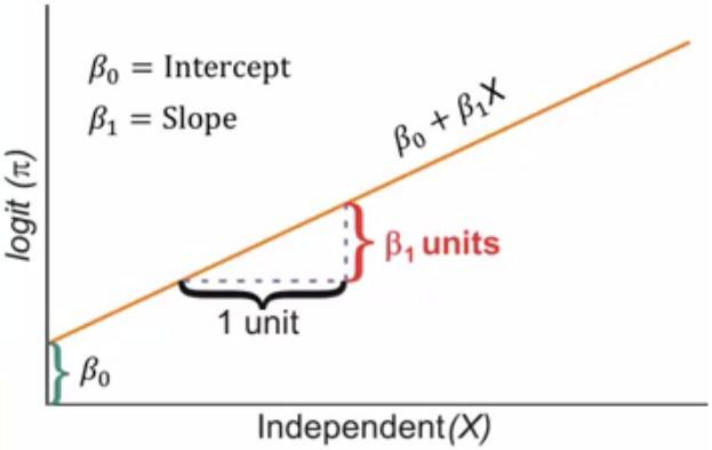


Figure: A simple Logistic Regression Model

Example: Logistic regression may be used to predict whether a patient has a given disease (e.g. diabetes) based on observed characteristics (age, sex, BMI results of blood tests, etc.) of the patient.

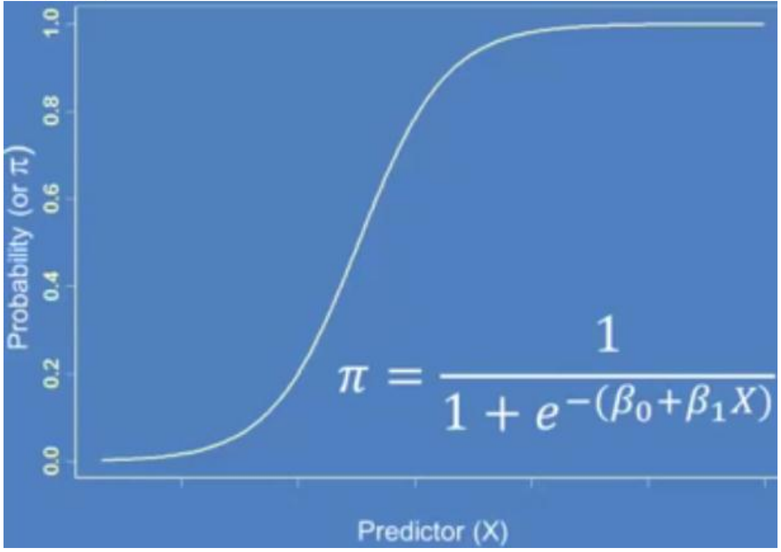
It makes use of one or more predictor variables that may be either continuous or categorical.

Used to predict binary dependent variables (treating the dependent variable as the outcome of the Bernoulli trial) rather than a continuous outcome.

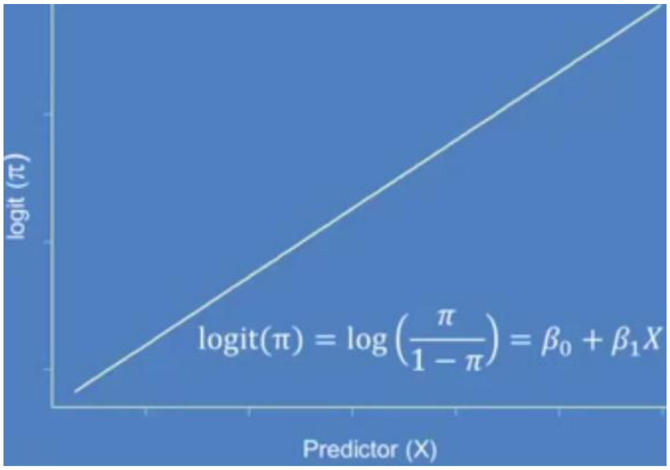
Dichotomous Dependent Variable: Generalizations of logistic model to more than 2 categories.

The question of interest for us to go for a Logistic Regression model is: The probability of experiencing the outcome of interest for a given value of independent variable.

The S-Curve: Indicates the relationship between a predictor (X) and the probability of experiencing the outcome of interest for a dichotomous dependent variable. The lie here is bounded below by zero and bounded above by 1.



It could be convenient if we could somehow transform this non-linear S-curve relationship between the predictor and the outcome so that it would be a simple linear equation similar to the one in Simple Linear Regression. Then it would be convenient to estimate the intercepts and slopes of the straight line. There is a convenient way to perform this transformation.



This is called the Logistic Function. If we apply the Logistic Function to the probability values and plot those values using a graph, we obtain the following figure:

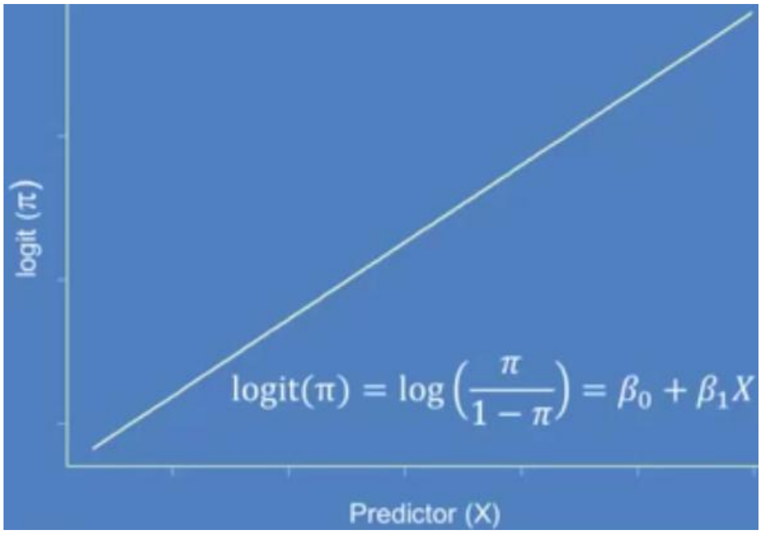


Figure: Linear in the Logit

Applying the Logistic Function to the probability values is referred to as taking the **Logit**. We refer to the new y-values as the Logit of the probability of experiencing the outcome of interest for a given value of independent variable (y) or Logit (Π).

Given this linear relationship between the Logit (Π) and the predictor variable (X), we can proceed as similar in Linear regression to estimate the β_0 & β_1 values. i.e . the Logit() transforms the probabilities to the log scale, a linear ready assumption of the relationship between Y and X is now also on the log scale.

The linear relationship utilizes the logit function our regression estimates would be on the log scale which will impact the interpretation of the coefficients. We can show these relationships graphically. In this the β_0 & β_1 values are same as the β_0 & β_1 values of Linear Regression, with the exception that the dependent variable is measured on the logit scale.

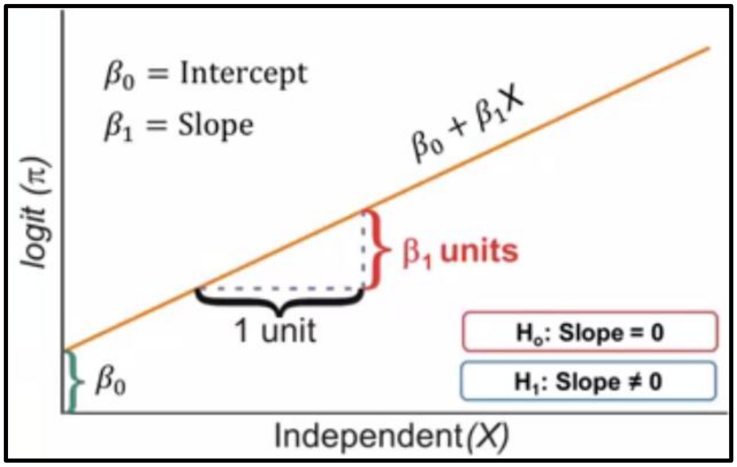


Figure: Simple Logistic Regression Model

The slope is represented by the change in the height of the orange line with every one unit change in the value of X. To determine the estimated Logit (Π) associated with a particular weight, simply replace the 'X' in the equation with the weight value of interest, multiply it by the slope and add the intercept. As with Linear Regression, the primary inference from the model is whether or not the slope coefficient is zero.

Factor Analysis

Factor analysis is a method for explaining the structure of data by explaining the correlations between variables. Factor analysis summarizes data into a few dimensions by condensing a large number of variables into a smaller set of latent variables or factors. It is commonly used in the social sciences, market research, and other industries that use large data sets.

To perform factor analysis, you need to decide how many factors to use, and determine loadings that make the most sense for your data.

1. Decide how many factors to use. The choice of the number of factors is often based on the proportion of variance explained by the factors, subject matter knowledge, and reasonableness of the solution.

- ➢ Try using the principal components extraction method without specifying the number of components.
- ➢ Examine the proportion of variability explained by different factors and narrow down your choice of how many factors to use. A scree plot can be useful here in visually assessing the importance of factors.
- ➢ Examine the fits of the different factor analyses. Communality values, the proportion of variability of each variable explained by the factors, can be especially useful in comparing fits. You might decide to add a factor if it contributes to the fit of certain variables.
- ➢ Try the maximum likelihood estimation method of extraction as well.

2. Evaluate your solution by trying multiple rotations. Johnson and Wichern suggest the varimax rotation. A similar result from different methods can lend credence to the solution you have selected. At this point you might want to interpret the factors using your knowledge of the data.

Principal Component Analysis (PCA)

Principal components analysis is a procedure for identifying a smaller number of uncorrelated variables, called "principal components", from a large set of data. The goal of principal components analysis is to explain the maximum amount of variance with the fewest number of principal components. Principal components analysis is commonly used in the social sciences, market research, and other industries that use large data sets. Principal components analysis is commonly used as one step in a series of analyses. You can use principal components analysis to reduce the number of variables and avoid multi-collinearity, or when you have too many predictors relative to the number of observations.

The methods used for the analysis of “large” data sets are: PCA and Projections to latent structures. These methods work computationally well for many variables and observations.

The simple concepts of latent variable models in multidimensional spaces, and projections of data onto these models, provide a strategy for utilizing this richness of information for:

- ➢ Summarizing data,
- ➢ Classification and Discriminant Analysis, and
- ➢ Modeling relationships between variables.

With more than around 100–200 variables, however, parameter plots (VIP, loadings and coefficients) tend to become messy and difficult to interpret; they become too crowded. Similarly, data sets with more than around 100–200 observations often give messy and hard to interpret score plots. Hierarchical multivariate models have advantages in the analysis and modeling of large data sets. These models provide an approach that is scalable, i.e., applicable to larger and larger data sets (in terms of larger and larger K).

Let $x^T = (x_1, x_2, \ldots, x_p)$ be a random vector with mean μ and covariance matrix Σ. PCA is a technique for dimensionality reduction from p dimensions to k < p dimensions. It tries to find, in order, the most informative k linear combinations of a set of variables $y_1, y_2, \ldots, y_k$. Here information will be interpreted as a percentage of the total variation (as previously defined) in The k sample PC's that "explain" x% of the total variation in a sample covariance matrix S may be similarly defined.

In streaming scenarios, recent data are generally considered more important than older data. This is because the data generating process may change over time, and the older data are often considered "stale" from the perspective of analytical insights. A uniform random sample from the reservoir will contain data points that are distributed uniformly over time. Typically, most streaming applications use a decay-based framework to regulate the relative importance of data points, so that more recent data points have a higher probability to be included in the sample. This is achieved with the use of a *bias function*.

The bias function associated with the r^{th} data point, at the time of arrival of the n^{th} data point, is given by $f(r, n)$. This function is related to the probability $p(r, n)$ of the r^{th} data point belonging to the reservoir at the time of arrival of the *n*th data point. In other words, the value of $p(r, n)$ is proportional to $f(r, n)$. It is reasonable to assume that the function $f(r, n)$ decreases monotonically with n (for fixed r), and increases monotonically with r (for fixed n). In other words, recent data points have a higher probability of belonging to the reservoir. This kind of sampling will result in a *bias-sensitive sample* $S(n)$ of data points.

Probabilistic Model-Based Algorithms

Probabilistic Models, such as hidden Markov models or Bayesian networks, are commonly used to model biological data. Much of their popularity can be attributed to the existence of efficient and robust procedures for learning parameters from observations. Often, however, the only data available for training a probabilistic model are incomplete. Missing values can occur, for example, in medical diagnosis, where patient histories generally include results from a limited battery of tests. Alternatively, in gene expression clustering, incomplete data arise from the intentional omission of gene-to-cluster assignments in the probabilistic model. The expectation maximization algorithm enables parameter estimation in probabilistic models with incomplete data.

Hard clustering algorithms are those algorithms in which each data point is deterministically assigned to a particular cluster.

Probabilistic model based algorithms are ***Soft algorithms*** in which each data point may have a nonzero assignment probability to many (typically all) clusters.

A soft solution to a clustering problem may be converted to a hard solution by assigning a data point to a cluster with respect to which it has the largest assignment probability.

Maximum Likelihood Estimation: The maximum likelihood method assesses the quality of a statistical model based on the probability it assigns to the observed data.

The broad principle of a mixture-based *generative* model is to assume that the data was generated from a mixture of k distributions with probability distributions Ģ$_1$ . . . Ģ$_k$. Each distribution Ģ$_i$ represents a cluster and is also referred to as a *mixture component*. Each data point $\overline{\mathbf{X}}_i$, where $i \varepsilon \{1 \ldots n\}$, is generated by this mixture model as follows:

1. Select a mixture component with prior probability $\alpha i = P(Ģ_i)$, where $i \in \{1 \dots k\}$. Assume that the r^{th} one is selected.
2. Generate a data point from $Ģ_r$.

This generative model will be denoted by *M*.

The different prior probabilities α_i and the parameters of the different distributions $Ģ_r$ are not known in advance.

Each distribution G_i is often assumed to be the Gaussian, although any arbitrary (and different) family of distributions may be assumed for each $Ģ_i$. The choice of distribution $Ģ_i$ is important because it reflects the user's a priori understanding about the distribution and shape of the individual clusters (mixture components).

The parameters of the distribution of each mixture component, such as its mean and variance, need to be estimated from the data, so that the overall data has the maximum likelihood of being *generated* by the model. This is achieved with the *expectation-maximization (EM)* algorithm. The parameters of the different mixture components can be used to describe the clusters. For example, the estimation of the mean of each Gaussian component is analogous to determine the mean of each cluster center in a *k*-representative algorithm. After the parameters of the mixture components have been estimated, the *posterior* generative (or assignment) probabilities of data points with respect to each mixture component (cluster) can be determined.

Assume that the probability density function of mixture component $Ģ_i$ is denoted by $f^i(\cdot)$. The probability (density function) of the data point X_j being generated by the model is given by the weighted sum of the probability densities over different mixture components, where the weight is the prior probability $\alpha_i = P(Ģ_i)$ of the mixture components:

$$f^{point}(\overline{X_j}|\mathcal{M}) = \sum_{i=1}^{k} \alpha_i \cdot f^i(\overline{X_j})$$

Gaussian Mixture Model:

It is a probabilistic model that assumes that all the data points are generated from a mixture of a finite number of Gaussian distributions with unknown parameters. One can think of the mixture models as generalizing k-means clustering to incorporate information about the covariance structure of the data as well as the centers of Latent Gaussians.

EM Algorithm: The expectation maximization algorithm alternates between the steps of guessing a probability distribution over completions of missing data given the current model (known as the E-step) and then re-estimating the model parameters using these completions (known as the M-step). The name 'E-step' comes from the fact that one does not usually need to form the probability distribution over completions explicitly, but rather need only compute 'expected' sufficient statistics over these completions. Similarly, the name 'M-step' comes from the fact that model re-estimation can be thought of as 'maximization' of the expected log-likelihood of the data.

The expectation maximization algorithm is a natural generalization of maximum likelihood estimation to the incomplete data case. In particular, expectation maximization attempts to find the parameters $\hat{\theta}$ that maximize the log probability $\log P(x;\theta)$ of the observed data.

Generally speaking, the optimization problem addressed by the expectation maximization algorithm is more difficult than the optimization used in maximum likelihood estimation. In the complete data case, the objective function $\log P(x,z;\theta)$ has a single global optimum, which can often be found in closed form. In contrast, in the incomplete data case the function log$P(x;\theta)$ has multiple local maxima and no closed form solution.

To deal with this, the expectation maximization algorithm reduces the difficult task of optimizing log$P(x;\theta)$ into a sequence of simpler optimization subproblems, whose objective functions have unique global maxima that can often be computed in closed form. These subproblems are chosen in a way that guarantees their corresponding solutions $\hat{\theta}^{(1)}$, $\hat{\theta}^{(2)}$,… and will converge to a local optimum of log$P(x;\theta)$. The objective function monotonically increases during each iteration of expectation maximization.

Independent component analysis (ICA) is a statistical and computational technique for revealing hidden factors that underlie sets of random variables, measurements, or signals.

ICA defines a generative model for the observed multivariate data, which is typically given as a large database of samples. In the model, the data variables are assumed to be linear mixtures of some unknown latent variables, and the mixing system is also unknown. The latent variables are assumed non-Gaussian and mutually independent, and they are called the independent components of the observed data. These independent components, also called sources or factors, can be found by ICA.

ICA is superficially related to principal component analysis and factor analysis. ICA is a much more powerful technique, however, capable of finding the underlying factors or sources when these classic methods fail completely.

The data analyzed by ICA could originate from many different kinds of application fields, including digital images, document databases, economic indicators and psychometric measurements. In many cases, the measurements are given as a set of parallel signals or time series; the term blind source separation is used to characterize this problem. Typical examples are mixtures of simultaneous speech signals that have been picked up by several microphones, brain waves recorded by multiple sensors, interfering radio signals arriving at a mobile phone, or parallel time series obtained from some industrial process.