Real Time Data Mining

Article · January 2017

Saed Sayad
Rutgers, The State University of New Jersey

REAL TIME DATA MINING

By

Saed Sayad

2.0

REAL TIME DATA MINING

REAL TIME DATA MINING

The Future is Here

Devoted to:

a dedicated mother, *Khadijeh Razavi Sayad*

a larger than life father, *Mehdi Sayad*

my beautiful wife, *Sara Pourmolkara*

and my wonderful children, *Sahar* and *Salar*

Foreword

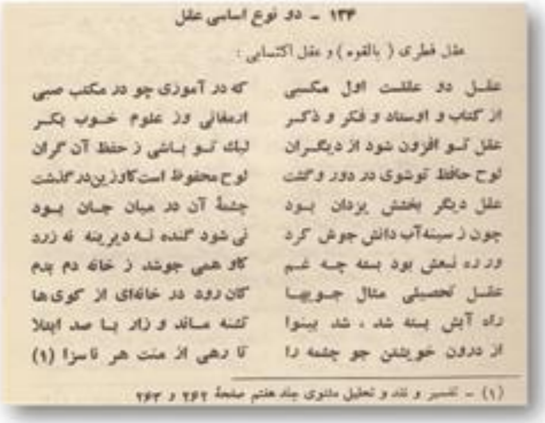
Data now appear in very large quantities and in real time but conventional data mining methods can only be applied to relatively small, accumulated data batches. This book shows how, by using a method termed the "Real Time Learning Machine" (RTLM) these methods can be readily upgraded to accept data as they are generated and to flexibly deal with changes in how they are to be processed. This real time data mining is the future of predictive modelling. The main purpose of the book is to enable you, the reader, to intelligently apply real time data mining to your own data. Thus, in addition to detailing the mathematics involved and providing simple numerical examples, a computer program is made freely available to allow you to learn the method using your data.

This book summarizes twenty years of method development by Dr. Saed Sayad. During that time he worked to apply data mining in areas ranging from bioinformatics through chemical engineering to financial engineering, in industry, in government and at university. This included the period since 1996 when Saed has worked with me to co-supervise graduate students in the Department of Chemical Engineering and Applied Chemistry at the University of Toronto. Portions of the method have now been previously presented at data mining conferences, published in scientific publications and used in Ph.D. thesis research as well as in graduate courses at the University of Toronto. This book integrates all of the previous descriptions and updates the nomenclature. It is intended as a practical instruction manual for data miners and students of data mining.

I have now so often witnessed the initial reaction to the numerous claims made for this very elegant method: incredulity. There have been so many much more complex attempts in the published literature to create practical real time data mining methods. None of these complex attempts have even come close to matching the capabilities of the RTLM. This book is uniquely important to the field of data mining: its combination of mathematical rigor, numerical examples, software implementation and documented references, finally comprehensively communicates what is Saed's invention of the RTLM. Most importantly, it does accomplish his purpose in authoring the book: it very effectively enables you to apply the method with confidence to your data. So, I sincerely hope that you will suspend your disbelief for a moment about the claims for the method, read this book, experiment with the software using your own data and then appreciate just how powerful is this apparently *"too simple"* real time data mining method.

Stephen T. Balke Ph.D., P.Eng.
Professor Emeritus
University of Toronto
Toronto, Ontario, Canada

Two Kinds of Intelligence



۱۳۴ ـ دو نوع اساسی عقل

عقل فطری (بالقوه) و عقل اکتسابی :

عقل دو عقلست اول مکسبی   که در آموزی چو در مکتب صبی
از کتاب و اوستاد و فکر و ذکر   ارمغانی وز علوم خوب بکر
عقل تو افزون شود از دیگران   لیک تو باشی ز حفظ آن گران
لوح حافظ توشوی در دور و گشت   لوح محفوظ است کاوزین درگذشت
عقل دیگر بخشش یزدان بود   چشمهٔ آن در میان جان بود
چون ز سینه آب دانش جوش کرد   نی شود گنده نـه دیرینه نه زرد
ور ره نبعش بود بسته چه غم   کاو همی جوشد ز خانه دم بدم
عقـل تحصیلی مثال جـویها   کان رود در خانه‌ای از کوی‌ها
راه آبش بسته شد ، شد بینوا   تشنه مـاند و زار بـا صد ابتلا
از درون خویشتن جو چشمه را   تا رهی از منت هر ناسزا (۱)

(۱) ـ تفسیر و نقد و تحلیل مثنوی جلد هشتم صفحهٔ ۲۶۲ و ۲۶۳

There are two kinds of intelligence: one acquired, as a child in school memorizes facts and concepts from books and from what the teacher says, collecting information from the traditional sciences With such intelligence you rise in the world.

You get ranked ahead or behind others in regard to your competence in retaining information. You stroll with this intelligence in and out of fields of knowledge, getting always more marks on your preserving tablets. There is another kind of tablet, one already completed and preserved inside you. A spring overflowing its spring box. A freshness in the center of the chest. This other intelligence does not turn yellow or stagnate. It's fluid, and it doesn't move from outside to inside through conduits of plumbing-learning. This second knowing is a fountainhead from within you, moving out.

From the translations of Rumi by Coleman Barks

CONTENTS

1.0 Introduction

Data mining is about explaining the past and predicting the future by exploring and analyzing data. Data mining is a multi-disciplinary field which combines statistics, machine learning, artificial intelligence and database technology.

Although data mining algorithms are widely used in extremely diverse situations, in practice, one or more major limitations almost invariably appear and significantly constrain successful data mining applications. Frequently, these problems are associated with large increases in the rate of generation of data, the quantity of data and the number of attributes (variables) to be processed: Increasingly, the data situation is now beyond the capabilities of conventional data mining methods.

The term "Real Time" is used to describe how well a data mining algorithm can accommodate an ever increasing data load instantaneously. However, such real time problems are usually closely coupled with the fact that conventional data mining algorithms operate in a batch mode where having all of the relevant data at once is a requirement. Thus, here **Real Time Data Mining** is defined as having all of the following characteristics, independent of the amount of data involved:

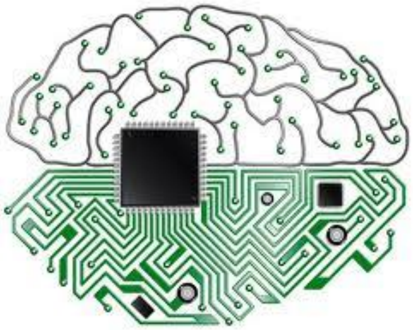
1. **Incremental learning (Learn)**: immediately updating a model with each new observation without the necessity of pooling new data with old data.
2. **Decremental learning (Forget)**: immediately updating a model by excluding observations identified as adversely affecting model performance without forming a new dataset omitting this data and returning to the model formulation step.
3. **Attribute addition (Grow)**: Adding a new attribute (variable) on the fly, without the necessity of pooling new data with old data.
4. **Attribute deletion (Shrink)**: immediately discontinuing use of an attribute identified as adversely affecting model performance.
5. **Scenario testing:** rapid formulation and testing of multiple and diverse models to optimize prediction.
6. **Real Time operation**: Instantaneous data exploration, modeling and model evaluation.
7. **In-Line operation**: processing that can be carried out in-situ (e.g.: in a mobile device, in a satellite, etc.).
8. **Distributed processing:** separately processing distributed data or segments of large data (that may be located in diverse geographic locations) and re-combining the results to obtain a single model.

9. **Parallel processing:** carrying out parallel processing extremely rapidly from multiple conventional processing units (multi-threads, multi-processors or **a specialized chip**).

Upgrading conventional data mining to real time data mining is through the use of a method termed the **Real Time Learning Machine** or **RTLM**. The use of the RTLM with conventional data mining methods enables "Real Time Data Mining".

The future of predictive modeling belongs to real time data mining and the main motivation in authoring this book is to help you to understand the method and to implement it for your application. The book provides previously published [1-6] and unpublished details on implementation of real time data mining. Each section is followed by a simple numerical example illustrating the mathematics. Finally, recognizing that the best way to learn and to appreciate real time data mining is to apply it to your own data, a software program that easily enables you to accomplish this is provided and is free for non-commercial applications.

The book begins by showing equations enabling real time data exploration previous to development of useful models. These "**Real Time Equations** (**RTE**)" appear similar to the usual ones seen in many textbooks. However, closer examination will reveal a slightly different notation than the conventional one. This notation is necessary to explain how "Real Time Equation" differs from conventional ones. Then, it details how a “**Basic Elements Table (BET)**” is constructed from a dataset and used to achieve scalability and real time capabilities in a data mining algorithm. Finally, each of the following methods is examined in turn and the real time equations necessary for utilization of the Basic Elements Table are provided: Naïve Bayesian, linear discriminant analysis, linear support vector machines, multiple linear regression, principal component analysis and regression, linear support vector regression, Markov chains and hidden Markov models.



2.0 The Real Time Learning Machine

In the previous section real time data mining algorithms defined as having nine characteristics, independent of the amount of data involved. There it is mentioned that conventional data mining methods are not real time methods. For example, while learning in nature is incremental, on-line and in real time as should real time algorithms be, most learning algorithms in data mining operate in a batch mode where having all the relevant data at once is a requirement. In this section we present a widely applicable novel architecture for upgrading conventional data mining methods to real time methods. This architecture is termed the "**Real Time Learning Machine**" (RTLM). This new architecture adds real time analytical power to the following widely used conventional learning algorithms:

- Naïve Bayesian
- Linear Discriminant Analysis
- Single and Multiple Linear Regression
- Principal Component Analysis and Regression
- Linear Support Vector Machines and Regression
- Markov Chains
- Hidden Markov Models

Conventionally, data mining algorithms interact directly with the whole dataset and must somehow accommodate the impact of new data, changes in attributes (variables), etc. An important feature of the RTLM is that, as shown in Figure 1, the modeling process is split into four separate components: **Learner**, **Explorer**, **Modeler** and **Predictor**. The data is summarized in the Basic Elements Table (BET) which is a relatively small table.

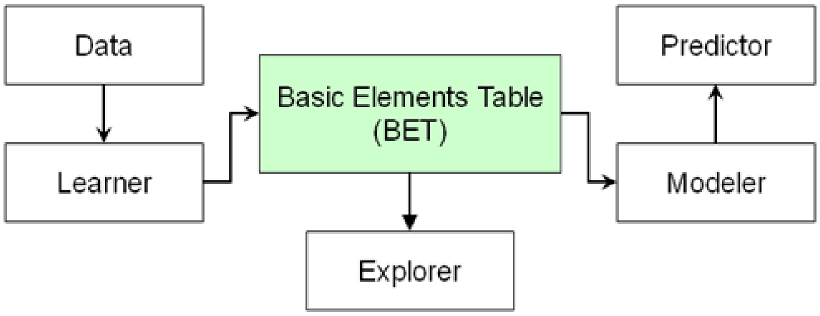


Figure 2.1 *Real Time Learning Machine (RTLM)*

The tasks assigned to each of the four real time components are as follows:

- **Learner:** updates (incrementally or decrementally) the Basic Elements Table utilizing the data in real time.
- **Explorer:** does univariate and bivariate statistical data analysis using the Basic Elements Table in real time.
- **Modeler:** constructs models using the Basic Elements Table in real time.
- **Predictor:** uses the models for prediction in real time.

The RTLM is not constrained by the amount of data involved and is a mathematically rigorous method for making parallel data processing readily accomplished. As shown in Figures 2 and 3, any size dataset can be divided to smaller parts and each part can be processed separately (multi-threads or multi-processors or **a specialized chip**) . The results can then be joined together to obtain the same model as if we had processed the whole dataset at once.

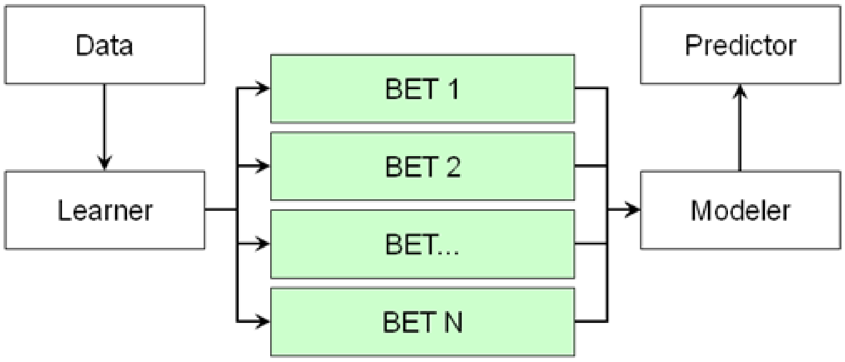


Figure 2.2 *Parallel Real Time Learning Machine*

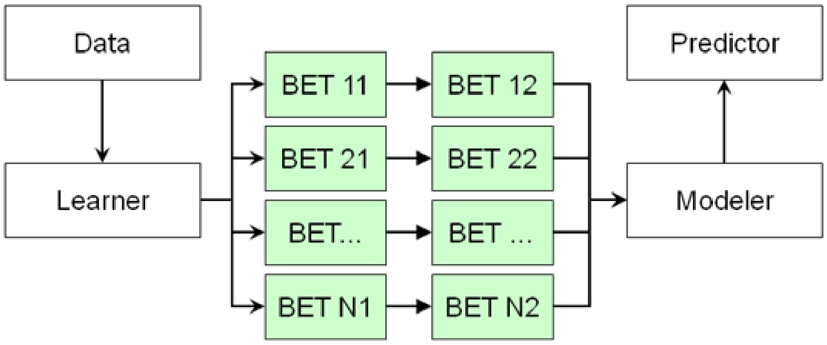


Figure 2.3 *Parallel Multi-layer Real Time Learning Machine*

2.1 The Basic Elements Table

The Basic Elements Table building block includes two attributes, X_i, X_j and one or more basic elements B_{ij}:

BET	X_j
X_i	B_{ij}

***Figure 2.4** Building block of the Basic Elements Table.*

where B_{ij} can consist of one or more following basic elements:

- N_{ij} : Total number of joint occurrence of two attributes
- $\sum X_i$ and $\sum X_j$: Sum of data
- $\sum X_i X_j$: Sum of multiplication
- $\sum X_i^2$ and $\sum X_j^2$: Sum of squared data
- $\sum \left(X_i X_j\right)^2$: Sum of squared multiplication

All above seven basic elements can be update in real time (incrementally or decrementally), using the following basic general real time equation.

➔ *General Real Time Equation*

$$B_{ij} := B_{ij} \pm B_{ij}^{new}$$

where:

- $B_{ij} = B_{ji}$
- (+) represents incremental and (-) decremental change of the basic elements.

The above seven basic elements are not the only ones; there are more elements which could be included in this list such as $\sum X^3$, $\sum X^4$ and more.

The number of attributes can also be updated in real time (incrementally or decrementally), simply by adding corresponding rows and columns and the related basic elements to the BET table.

2.2 Attribute Types

There are only two types of attributes in BET; **Numeric** and **Categorical (Binary)**. The numerical attributes can also be discretized (binning). The categorical attributes and the descretized version of the numerical attributes must be encoded into binary (0, 1). The following example shows how to transform a categorical attribute to its binary counterparts.

Temperature	Humidity	Play
98	88	no
75	70	yes
90	96	no
78	?	yes
65	60	yes

Table 2.1 *Original dataset.*

Temperature	Humidity	Play.yes	Play.no
98	88	0	1
75	70	1	0
90	96	0	1
78	?	1	0
65	60	1	0

Table 2.2 *Categorical attribute (Play) is transformed to two binary attributes.*

Temperature	Temp.hot	Temp.moderate	Temp.mild	Humidity	Play.yes	Play.no
98	1	0	0	88	0	1
75	0	1	0	70	1	0
90	1	0	0	96	0	1
78	0	1	0	?	1	0
65	0	0	1	60	1	0

Table 2.3 *Final transformed dataset with three new binary attributes which are created by discretizing Temperature.*

Note: Technically speaking, for a categorical attribute with *k* categories we only need to create *k-1* binary attributes but we do not suggest it for the RTLM implementation.

2.2.1 Missing Values

Simply, all the non-numeric values in the dataset are considered as missing values. For example, the “?” in the above dataset will be ignored by the RTLM Learner. There is no need for any missing values policy here, because RTLM can build a new model on the fly with excluding attributes with missing value.

Basic Elements Table - Example

The Basic Element Table for the above sample dataset with four basic elements is shown below. The RTLM Learner updates the basic elements table with any new incoming data.

N_{ij} $\sum X_i, \sum X_j$ $\sum X_i X_j$	X_1 Temp.	X_2 Temp.hot	X_3 Temp.moderate	X_4 Temp.mild	X_5 Humidity	X_6 Play.yes	X_7 Play.no
X_1 Temperature	5 406, 406 33638	5 406, 2 188	5 406, 2 153	5 406, 1 65	4 328, 314 26414	5 406, 3 218	5 406, 2 188
X_2 Temp.hot		5 2, 2 2	5 2, 2 0	5 2, 1 0	4 2, 314 184	5 2, 3 0	5 2, 2 2
X_3 Temp.moderate			5 2, 2 2	5 2, 1 0	4 1, 314 70	5 2, 3 2	5 2, 2 0
X_4 Temp.mild				5 1, 1 1	4 1, 314 60	5 1, 3 1	5 1, 2 0
X_5 Humidity					4 314, 314 25460	4 314, 2 130	4 314, 2 184
X_6 Play.yes						5 3, 3 3	5 3, 2 0
X_7 Play.no							5 2, 2 2

Table 2.4 *Basic Elements Table for the sample dataset.*

Here, it is shown how to compute some of the necessary statistics for many modeling algorithms using only the basic elements.

Numerical Variable

$$Average\ (Temperature) = \frac{\sum X_1}{N_{1,1}} = \frac{406}{5} = 81.2$$

$$Variance\ (Temperature) = \frac{\sum X_1 X_1 - \frac{(\sum X_1)^2}{N_1}}{N_1}$$

$$= \frac{33638 - \frac{(406)^2}{5}}{5} = 134.16$$

Categorical Variable

$$Count(Play.yes) = \sum X_6 = 3$$

$$Probability(Play.yes) = \frac{\sum X_6}{N_{6,6}} = \frac{3}{5} = 0.6$$

Numerical Variable and Numerical Variable

$$Covariance\ (Temperature, Humidity) = \frac{\sum X_1 X_5 - \frac{\sum X_1 \sum X_5}{N_{1,5}}}{N_{1,5}}$$

$$= \frac{26414 - \frac{(328\times 314)}{4}}{4} = 166.5$$

Numerical Variable and Categorical Variable

$$Average\ (Temperature|Play.yes) = \frac{\sum X_1 X_6}{\sum X_6} = \frac{218}{3} = 72.7$$

Categorical Variable and Categorical Variable

$$Probability(Temperature.mild|Play.no) = \frac{\sum X_4 X_7}{\sum X_7} = \frac{0}{2} = 0.0$$

3.0 Real Time Data Exploration

Data Exploration is about describing the data by means of statistical and visualization techniques. In this section the focus is on statistical methods and emphasize how specific statistical quantities can be calculated using “**Real Time Equations** (RTE)”. In subsequent sections it will be seen that the “Real Time Equations” enable upgrading of the conventional data mining techniques to their real time counterparts.

Real Time Data Exploration will be discussed in the following two categories:

1. Real Time Univariate Data Exploration
2. Real Time Bivariate Data Exploration

3.1 Real Time Univariate Data Exploration

Univariate data analysis explores attributes (variables) one by one using statistical analysis. Attributes are either numerical or categorical (encoded to binary). Numerical attributes can be transformed into categorical counterparts by discretization or binning. An example is “Age” with three categories (bins); 20-39, 40-59, and 60-79. Equal Width and Equal Frequency are two popular binning methods. Moreover, binning may improve accuracy of the predictive models by reducing the noise or non-linearity and allows easy identification of outliers, invalid and missing values.

3.1.1 Count

The total count of k subsets of the attribute X can be computed in real time. It means a dataset can be divided into k subsets and the count of the whole data will be equal to the sum of all its subsets count.

➔ *Real Time Equation 1: Count*

$$N := N \pm N^{new} \qquad (1)$$

The $\pm$ notation preceding N in the above equation means that the number of data in a subset is a positive quantity if those data are being added to the BET (incremental learning) or a negative quantity if those data are being subtracted from the BET (decremental learning).

3.1.2 Mean (Average)

The mean or average is a point estimation of a set of data. As normally written, average is not a real time equation because averages with different N cannot be added or subtracted incrementally.

$$\bar{X} = \frac{\sum X}{N}$$

However, using the same notation as used in the first real time equation we see that the summation can be written in a real time form as follows:

➔ *Real Time Equation 2: Sum of data*

$$\sum X := \sum X \pm \sum X^{new} \tag{2}$$

➔ *Real Time Equation 3: Mean*

$$\bar{X} = \frac{\sum X \pm \sum X^{new}}{N \pm N^{new}} \tag{3}$$

where:

- N: *Count (1)*

- $\sum X$: *Sum of data (2)*

Now “Mean” can be written as a real time quantity because the whole data is not required each time it is calculated. Only the values of the subset sums are required along with the count in each subset.

3.1.3 Variance

The variance is a measure of data dispersion or variability. A low variance indicates that the data tend to be very close to the mean, whereas high variance indicates that the data is spread out over a large range of values. Similarly, the variance equation is not real time by itself.

$$S^2 = \frac{\sum X^2 - \frac{(\sum X)^2}{N}}{N}$$

However, the sums involved can be written as sums over subsets rather than over the whole data.

➔ *Real Time Equation 4: Sum of Squared data*

$$\sum X^2 := \sum X^2 \pm \sum {X^2}^{new} \qquad (4)$$

Now, we can write real time equations for the variance and the standard deviation using the basic elements.

- *Real Time Equation 5: Variance*

$$S^2 = \frac{\left(\sum X^2 \pm \sum X^{2^{new}}\right) - \frac{\left(\sum X \pm \sum X^{new}\right)^2}{(N \pm N^{new})}}{(N \pm N^{new})} \tag{5}$$

Note: If the number of data is less than 30 we should replace N with N-1.

3.1.4 Standard Deviation

Standard deviation like variance is a measure of the variability or dispersion. A low standard deviation indicates that the data tend to be very close to the mean, whereas high standard deviation indicates that the data is spread out over a large range of values.

- *Real Time Equation 6: Standard Deviation*

$$S = \sqrt{S^2} = \sqrt{\frac{\left(\sum X^2 \pm \sum X^{2^{new}}\right) - \frac{\left(\sum X \pm \sum X^{new}\right)^2}{(N \pm N^{new})}}{(N \pm N^{new})}} \tag{6}$$

3.1.5 Coefficient of Variation

The coefficient of variation is a standardized measure of the dispersion or variability in data. CV is independent of the units of measurement.

Real Time Equation 7: Coefficient of Variation

$$CV = \frac{S}{\bar{X}} \times 100\% \qquad (7)$$

where:

- $\bar{X}$*: Average (3)*
- S*: Standard Deviation (6)*

3.1.6 Skewness

Skewness is a measure of symmetry or asymmetry in the distribution of data. The skewness equation is not real time by itself but its components are.

$$Skew = \frac{N}{(N-1)(N-2)} \sum \left(\frac{X - \bar{X}}{S} \right)^3$$

First we need to expand the aggregate part of the equation:

$$\sum \left(\frac{X - \bar{X}}{S} \right)^3 = \sum \left(\frac{X^3 - 3X^2\bar{X} + 3X\bar{X}^2 - \bar{X}^3}{S^3} \right)$$

$$= \frac{1}{S^3} \left(\sum X^3 - 3\bar{X} \sum X^2 + 3\bar{X}^2 \sum X - N\bar{X}^3 \right)$$

where:

- *N: Count (1)*
- *$\sum X$: Sum of data (2)*
- *$\bar{X}$: Average (3)*
- *S: Standard Deviation (6)*

Real Time Equation 8: Sum of data to the power of 3

$$\sum X^3 := \sum X^3 \pm \sum {X^3}^{new} \qquad (8)$$

Real Time Equation 9: Skewness

$$Skew = \frac{N}{(N-1)(N-2)} \times \frac{1}{S^3}\left(\sum X^3 - 3\bar{X}\sum X^2 + 3\bar{X}^2\sum X - N\bar{X}^3\right) \qquad (9)$$

3.1.7 Kurtosis

Kurtosis is a measure of whether the data are peaked or flat relative to a normal distribution. Like skewness, the standard equation for kurtosis is not real time equation but can be transformed to be real time.

$$Kurt = \left[\frac{N(N+1)}{(N-1)(N-2)(N-3)}\sum\left(\frac{X-\bar{X}}{S}\right)^4\right] - \frac{3(N-1)^2}{(N-2)(N-3)}$$

First we need to expand the aggregate part of the equation:

$$\sum\left(\frac{X-\bar{X}}{S}\right)^4 = \sum\left(\frac{X^4 - 4X^3\bar{X} + 6X^2\bar{X}^2 - 4X\bar{X}^3 + \bar{X}^4}{S^4}\right)$$

$$= \frac{1}{S^4}\left(\sum X^4 - 4\bar{X}\sum X^3 + 6\bar{X}^2\sum X^2 - 4\bar{X}^3\sum X + N\bar{X}^4\right)$$

where:

- N*: Count (1)*
- $\sum X$*: Sum of data (2)*
- $\bar{X}$*: Average (3)*
- $\sum X^2$*: Sum of Squared data (4)*
- S*: Standard Deviation (6)*
- $\sum X^3$*: Sum of data to the power of 3 (8)*

Real Time Equation 10: Sum of data to the power of 4

$$\sum X^4 := \sum X^4 \pm \sum {X^4}^{new} \tag{10}$$

Real Time Equation 11: Kurtosis

$$Kurt = \left[\frac{N(N+1)}{(N-1)(N-2)(N-3)} \times \frac{1}{S^4}\left(\sum X^4 - 4\bar{X}\sum X^3 + 6\bar{X}^2\sum X^2 - 4\bar{X}^3\sum X + N\bar{X}^4\right)\right] - \frac{3(N-1)^2}{(N-2)(N-3)} \tag{11}$$

3.1.8 Median

The median is the middle data point where below and above it, lie an equal number of data points. The median equation is not real time and cannot be directly transformed to. However, by using a discretized (binned) version of the attribute we can often have a good estimation of the median.

Real Time Equation 12: Median

$$Median = L_1 + \left[\frac{\frac{N+1}{2} - F_{j-1}}{N_j}\right] \times h \qquad (12)$$

- Figure out which bin contains the median by using the **(*N* + 1)/2** formula. ***N_j*** is the count for the median bin. ***N*** is the total count for all bins.
- Find the cumulative percentage of the interval (***F_{j-1}***) preceding the median group.
- ***h*** is the range in each bin.
- ***L₁*** is lower limit value in the median bin.

3.1.9 Mode

Mode like median cannot be transformed to real time. However, like median we can have a good estimation of mode by having the discretized (binned) version of a numerical attribute. When numerical attributes are discretized in bins, the mode is defined as the bin where most observations lie.

3.1.10 Minimum and Maximum

Minimum and Maximum can be updated in real time incrementally but not decrementally. It means if we lose an existing maximum or minimum value we would need to consider all historical data to replace them. One practical option is using the lower bound (minimum) and upper bound (maximum) of the discretized version of a numerical attribute.

Summary of Real Time Univariate Data Analysis

All the above univariate real time statistical equations are based on only five basic elements. To calculate any of the univariate quantities, we only need to save and update the following five elements in the Basic Elements Table (BET).

- *Real Time Equation 1: Count*

$$N := N \pm N^{new}$$

- *Real Time Equation 2: Sum of data*

$$\sum X := \sum X \pm \sum X^{new}$$

- *Real Time Equation 4: Sum of Squared data*

$$\sum X^2 := \sum X^2 \pm \sum {X^2}^{new}$$

- *Real Time Equation 8: Sum of data to the power of 3*

$$\sum X^3 := \sum X^3 \pm \sum {X^3}^{new}$$

- *Real Time Equation 10: Sum of data to the power of 4*

$$\sum X^4 := \sum X^4 \pm \sum {X^4}^{new}$$

Real Time Univariate Data Analysis - Examples

The following univariate real time statistical quantities are based on the Iris dataset found in the Appendix A. To calculate any of the univariate quantity, we only need to use the elements of the Basic Elements Table (BET) generated from the Iris dataset. All the BET elements are updateable in real time.

sepal_length	
indices	$i = 1\ and\ j = 1$
Count	$N = N_{1,1} = 150$
Mean	$\bar{X} = \frac{\sum X_1}{N} = \frac{876.5}{150} = 5.843$
Variance	$S^2 = \frac{\sum X_1^2 - \frac{(\sum X_1)^2}{N}}{N} = \frac{5223.85 - \frac{(876.5)^2}{150}}{150} = 0.681$
Standard Deviation	$S = \sqrt{S^2} = \sqrt{0.681} = 0.825$
Coefficient of Variation	$CV = \frac{S}{\bar{X}} \times 100\% = \frac{0.825}{5.843} \times 100\% = 14\%$

Skewness	$Skew = \frac{N}{(N-1)(N-2)}$ $\times \frac{1}{S^3}\left(\sum X_1^3 - 3\bar{X}\sum X_1^2 + 3\bar{X}^2\sum X_1\right.$ $\left. - N\bar{X}^3\right)$ $= \frac{150}{(150-1)(150-2)}$ $\times \frac{1}{0.825^3}(31745 - 3\times 5.843\times 5223.85$ $+ 3\times 5.843^2\times 876.5 - 150\times 5.843^3) = 0.32$
Kurtosis	$Kurt = \left[\frac{N(N+1)}{(N-1)(N-2)(N-3)}\right.$ $\times \frac{1}{S^4}\left(\sum X_1^4 - 4\bar{X}\sum X_1^3 + 6\bar{X}^2\sum X_1^2\right.$ $\left.\left. - 4\bar{X}^3\sum X_1 + N\bar{X}^4\right)\right] - \frac{3(N-1)^2}{(N-2)(N-3)}$ $= \left[\frac{150\times(150+1)}{(150-1)\times(150-2)\times(150-3)}\right.$ $\times \frac{1}{0.825^4}(196591.7 - 4\times 5.843\times 31745$ $+ 6\times 5.843^2\times 5223.85 - 4\times 5.843^3\times 876.5$ $\left. + 150\times 5.843^4)\right] - \frac{3\times(150-1)^2}{(150-2)\times(150-3)}$ $= -0.52$ *where*: $\sum X^4 = \sum \left(X_iX_j\right)^2 = \sum (X_1X_1)^2 = \sum (X_1^2)^2 = \sum X_1^4$

Figure 3.1 *Univariate analysis on a numerical attribute*

Iris_setosa	
indices	$i = 13 \; and \; j = 13$
Count	$N = N_{13,13} = 150$
Mean	$\bar{X} = Probability = \frac{\sum X_{13}}{N} = \frac{50}{150} = 0.333$
Variance	$S^2 = \frac{\sum X_{13}^2 - \frac{(\sum X_{13})^2}{N}}{N} = \frac{50 - \frac{(50)^2}{150}}{150} = 0.222$ $S^2 = P(1 - P) = 0.333(1 - 0.333) = 0.222$
Standard Deviation	$S = \sqrt{S^2} = \sqrt{P(1 - P)} = \sqrt{0.222} = 0.471$

Figure 3.2 *Univariate analysis on a categorical (binary) attribute.*

3.2 Real Time Bivariate Data Analysis

Bivariate data analysis is the simultaneous analysis of two attributes (variables). It explores the concept of relationship between two attributes, whether there is an association and the strength of this association, or whether there are differences between two attributes and the significance of these differences.

3.2.1 Covariance

Covariance measures the extent to which two numerical attributes vary together. That is a measure of the linear relationship between two attributes.

Real Time Equation 13: Covariance

$$Covar(X_i, X_j) = \frac{\sum X_i X_j - \frac{\sum X_i \sum X_j}{N_{ij}}}{N_{ij}} \qquad (13)$$

where:

- N: *Count (1)*
- $\sum X$ *and* $\sum Y$: *Sum of data (2)*

Real Time Equation 14: Sum of Multiplications

$$\sum X_i X_j := \sum X_i X_j \pm \sum X_i X_j{}^{new} \qquad (14)$$

The following real time equation is also very useful.

Real Time Equation 15: Sum of Squared Multiplication

$$\sum (X_i X_j)^2 := \sum (X_i X_j)^2 \pm \sum (X_i X_j)^{2^{new}} \qquad (15)$$

3.2.2 Linear Correlation Coefficient

Linear correlation quantifies the strength of a linear relationship between two attributes. When there is no correlation between two attributes, there is no tendency for the values of one quantity to increase or decrease with the values of the second quantity. The linear correlation coefficient measures the strength of a linear relationship and is always between -1 and 1 where -1 means perfect negative linear correlation and +1 means perfect positive linear correlation and zero means no linear correlation.

Real Time Equation 16: Linear Correlation Coefficient

$$R = \frac{Covar(X_i, X_j)}{\sqrt{S_i^2 \times S_j^2}} \qquad (16)$$

where:

- $Covar\left(X_i, X_j\right)$: *Covariance (13)*
- S_i^2 *and* S_j^2 : *Variance (5)*

3.2.3 Conditional Univariate Statistics

The following equations define univariate statistics for an attribute X_i given a binary attribute X_j when $X_j = 1$. Many of the bivariate statistics rely on these conditional univariate statistics.

➔ *Real Time Equation 17: Conditional Count*

$$Count\left(X_i \middle| X_j = 1\right) = N_{i|j} = \sum X_j \qquad (17)$$

➔ *Real Time Equation 18: Conditional Sum of data*

$$Sum\left(X_i \middle| X_j = 1\right) = \sum X_{i|j} = \sum X_i X_j \qquad (18)$$

➔ *Real Time Equation 19: Conditional Sum of Squared data*

$$Sum\ of\ Squares\left(X_i \middle| X_j = 1\right) = \sum X_{i|j}^2 = \sum (X_i X_j)^2 \qquad (19)$$

Real Time Equation 20: Conditional Mean

$$Mean\left(X_i \middle| X_j = 1\right) = \bar{X}_{i|j} = \frac{\sum X_{i|j}}{N_{i|j}} = \frac{\sum X_i X_j}{\sum X_j} \qquad (20)$$

Real Time Equation 21: Conditional Variance

$$Variance\left(X_i \middle| X_j = 1\right) = S_{i|j}^2 = \frac{\sum X_{i|j}^2 - \frac{\left(\sum X_{i|j}\right)^2}{N_{i|j}}}{N_{i|j}}$$
$$= \frac{\sum (X_i X_j)^2 - \frac{\left(\sum X_i X_j\right)^2}{\sum X_j}}{\sum X_j} \qquad (21)$$

Real Time Equation 22: Conditional Standard Deviation

$$S_{i|j} = \sqrt{S_{i|j}^2} = \sqrt{\frac{\sum X_{i|j}^2 - \frac{\left(\sum X_{i|j}\right)^2}{N_{i|j}}}{N_{i|j}}}$$
$$= \sqrt{\frac{\sum (X_i X_j)^2 - \frac{\left(\sum X_i X_j\right)^2}{\sum X_j}}{\sum X_j}} \qquad (22)$$

Complement Conditional Univariate Statistics

For real time predictive modeling we also need to define conditional univariate statistics for an attribute X_i given a binary attribute X_j *when* $X_j = 0$.

Real Time Equation 23: Complement Conditional Count

$$Count(X_i|X_j = 0) = N_{i|\bar{j}} = N_{ij} - \sum X_j \qquad (23)$$

Real Time Equation 24: Complement Conditional Sum of data

$$Sum(X_i|X_j = 0) = \sum X_{i|\bar{j}} = \sum X_i - \sum X_i X_j \qquad (24)$$

Real Time Equation 25: Complement Conditional Sum of Squared data

$$\begin{aligned} Sum\ of\ Squares(X_i|X_j = 0) &= \sum X_{i|\bar{j}}^2 \\ &= \sum X_i^2 - \sum (X_i X_j)^2 \qquad (25) \end{aligned}$$

Real Time Equation 26: Complement Conditional Mean

$$Mean(X_i|X_j = 0) = \bar{X}_{i|\bar{j}} = \frac{\sum X_{i|\bar{j}}}{N_{i|\bar{j}}} = \frac{\sum X_i - \sum X_i X_j}{N_{ij} - \sum X_j} \qquad (26)$$

- *Real Time Equation 27: Complement Conditional Variance*

$$Variance\left(X_i \middle| X_j = 0\right) = S_{i|\bar{j}}^2 = \frac{\sum X_{i|\bar{j}}^2 - \frac{\left(\sum X_{i|\bar{j}}\right)^2}{N_{i|\bar{j}}}}{N_{i|\bar{j}}}$$

$$= \frac{\left(\sum X_i^2 - \sum (X_i X_j)^2\right) - \frac{\left(\sum X_i - \sum X_i X_j\right)^2}{\left(N_{i,j} - \sum X_j\right)}}{\left(N_{ij} - \sum X_j\right)} \qquad (27)$$

- *Real Time Equation 28: Complement Conditional Standard Deviation*

$$S_{i|\bar{j}} = \sqrt{S_{i|\bar{j}}^2} = \sqrt{\frac{\sum X_{i|\bar{j}}^2 - \frac{\left(\sum X_{i|\bar{j}}\right)^2}{N_{i|\bar{j}}}}{N_{i|\bar{j}}}}$$

$$= \sqrt{\frac{\left(\sum X_i^2 - \sum (X_i X_j)^2\right) - \frac{\left(\sum X_i - \sum X_i X_j\right)^2}{\left(N_{i,j} - \sum X_j\right)}}{\left(N_{ij} - \sum X_j\right)}} \qquad (28)$$

3.2.4 *Z* test

The Z test assesses whether the difference between averages of two attributes are statistically significant. This analysis is appropriate for comparing the average of a numerical attribute with a known average or two conditional averages of a numerical attribute given two binary attributes (two categories of the same categorical attribute).

➔ *Real Time Equation 29: Z test - one group*

$$Z = \frac{\bar{X} - \mu_0}{\frac{S}{\sqrt{N}}} \qquad (29)$$

where:

- $\bar{X}$: *Mean or Conditional Mean (3 or 20)*
- S: *Standard Deviation or Conditional Standard Deviation (6 or 22)*
- N: *Count or Conditional Count (1 or 17)*
- μ_0: known average

The probability of *Z* (using normal distribution) defines the significance of the difference between two averages.

➔ *Real Time Equation 30: Z test - two groups*

$$Z = \frac{\bar{X}_1 - \bar{X}_2}{\sqrt{\frac{S_1^2}{N_1} + \frac{S_2^2}{N_2}}} \qquad (30)$$

where:

- $\bar{X}_1$*and* $\bar{X}_2$: *Conditional Mean (20)*

- S_1^2 *and* S_2^2 : *Conditional Variance (21)*
- N_1*and* N_2 : *Conditional Count (17)*

3.2.5 *T* test

The *T* test like *Z* test assesses whether the averages of two numerical attributes are statistically different from each other when the number of data points is less than 30. *T* test is appropriate for comparing the average of a numerical attribute with a known average or two conditional averages of a numerical attribute given two binary attributes (two categories of the same categorical attribute).

➔ *Real Time Equation 31: T test - one group*

$$t = \frac{\bar{X} - \mu_0}{\frac{S}{\sqrt{N}}} \qquad (31)$$

where:

- $\bar{X}$: *Mean or Conditional Mean (3 or 20)*
- S: *Standard Deviation or Conditional Standard Deviation (6 or 22)*
- N: *Count or Conditional Count (1 or 17)*
- μ_0: Known average

The probability of *t* (using *t* distribution with *N-1* degree of freedom) defines if the difference between two averages is statistically significant.

➔ *Real Time Equation 32: T test - two groups*

$$t = \frac{\bar{X}_1 - \bar{X}_2}{\sqrt{S^2\left(\frac{1}{N_1} + \frac{1}{N_2}\right)}} \qquad (32)$$

$$S^2 = \frac{(N_1 - 1)S_1^2 + (N_2 - 1)S_2^2}{N_1 + N_2 - 2}$$

where:

- $\bar{X}_1 and\ \bar{X}_2$: *Conditional Mean (20)*
- $S_1^2\ and\ S_2^2$: *Conditional Variance (21)*
- $N_1 and\ N_2$: *Conditional Count (17)*

3.2.6 *F* test

The *F*-test is used to compare the variances of two attributes. *F* test can be used for comparing the variance of a numerical attribute with a known variance or two conditional variances of a numerical attribute given two binary attributes (two categories of the same categorical attribute).

Real Time Equation 33: F test – one group

$$\chi^2 = \frac{(N-1)S^2}{\sigma_0^2} \qquad (33)$$

where:

- N : *Count or Conditional Count (1 or 17)*
- S^2 : *Variance or Conditional Variance (5 or 21)*
- σ_0^2 : known variance
- χ^2: has Chi^2 distribution with *N-1* degree of freedom

Real Time Equation 34: F test – two groups

$$F = \frac{S_1^2}{S_2^2} \qquad (34)$$

where:

- S_1^2 *and* S_2^2 : *Conditional Variance (21)*
- F : has *F* distribution with $N_1 - 1$ and $N_2 - 1$ degree of freedoms

3.2.7 Analysis of Variance (ANOVA)

ANOVA assesses whether the averages of more than two groups are statistically different from each other, under the assumption that the corresponding populations are normally distributed. ANOVA is useful for comparing averages of two or more numerical attributes or two or more conditional averages of a numerical attribute given two or more binary attributes (two or more categories of the same categorical attribute).

Source of Variation	Sum of Squares	Degree of Freedom	Mean Squares	F	Probability
Between Groups	SS_B	df_B	$MS_B = \frac{SS_B}{df_B}$	$F = \frac{MS_B}{MS_w}$	$P(F)$
Within Groups	SS_W	df_w	$MS_w = \frac{SS_w}{df_w}$		
Total	SS_T	df_T			

Figure 3.3 *Analysis of Variance and its components.*

Real Time Equation 35: Sum of Squares Between Groups

$$SS_B = \sum_{m=1}^{M} \frac{(\sum X)_m^2}{N_m} - \frac{(\sum_{m=1}^{M} (\sum X)_m)^2}{\sum_{m=1}^{M} N_m} \qquad (35)$$

$$df_B = M - 1$$

where:

- N : *Conditional Count (17)*
- $\sum X$: *Conditional Sum of data (18)*

Real Time Equation 36: Sum of Squares Within Groups

$$SS_w = \sum_{m=1}^{M} \left(\sum X^2\right)_m - \sum_{m=1}^{M} \frac{(\sum X)_m^2}{N_m} \qquad (36)$$

$$df_w = \sum_{m=1}^{M} N_m - M$$

where:

- N : *Conditional Count (17)*
- $\sum X$: *Conditional Sum of data (18)*
- $\sum X^2$: *Conditional Sum of Squared data (19)*

Real Time Equation 37: Sum of Squares Total

$$SS_T = SS_B + SS_w \qquad (37)$$

$$df_T = \sum_{m=1}^{M} N_m - 1$$

F : has F distribution with df_B and df_w degree of freedoms.

3.2.8 *Z* test – Proportions

The *Z* test can also be used to compare proportions. It can be used to compare a proportion from one categorical attribute with a known proportion or compare two proportions originated from two binary attributes (two categories of the same categorical attribute).

➔ *Real Time Equation 38: Z test - one group*

$$Z = \frac{\frac{n}{N} - P_0}{\sqrt{\frac{P_0(1 - P_0)}{N}}} \qquad (38)$$

where:

- N: *Count or Conditional Count (1 or 17)*
- n: *Sum of data or Conditional Sum of data (2 or 18)*
- P_0: known probability
- Z: has normal distribution

Real Time Equation 39: Z test - two groups

$$Z = \frac{\frac{n_1}{N_1} - \frac{n_2}{N_2}}{\sqrt{\hat{P}(1-\hat{P})\left(\frac{1}{n_1} + \frac{1}{n_2}\right)}} \qquad (39)$$

$$\hat{P} = \frac{n_1 + n_2}{N_1 + N_2}$$

where:

- N : *Conditional Count (17)*
- n : *Conditional Sum of data (18)*
- Z : has normal distribution

The probability of Z (using normal distribution) defines the significance of the difference between two proportions.

3.2.8 *Chi*^{2} test (Test of Independence)

The *Chi*^{2} test can be used to determine the association between categorical (binary) attributes. It is based on the difference between the expected frequencies and the observed frequencies in one or more categories in the frequency table. The *Chi*^{2} distribution returns a probability for the computed *Chi*^{2} and the degree of freedom. A probability of zero shows complete dependency between two categorical attributes and a probability of one means that two categorical attributes are completely independent.

➔ *Real Time Equation 40: Chi*^{2} *test (Test of Independence)*

$$\chi^2 = \sum_{i=1}^{r} \sum_{j=1}^{c} \frac{(n_{ij} - e_{ij})^2}{e_{ij}} \qquad (40)$$

$$e = \frac{n_{i.} n_{.j}}{n}$$

$$df = (r-1)(c-1)$$

where:

- $n_{ij} = \sum X_{i|j}$: *Conditional Sum of data (18)*
- e: expected frequency from the subset
- df: degree of freedom

- $r\ and\ c$: number of rows and columns
- χ^2 : has Chi^2 distribution with $(r-1)(c-1)$ degree of freedom

Summary of Real Time Bivariate Data Analysis

All the above real time bivariate statistical equations are based on only 5 basic elements. As in the case of the real time univariate statistical analysis, to calculate the required statistical quantities we only need to save and update these five elements in the basic elements table (BET).

- *Real Time Equation 1: Count*

$$N := N \pm N^{new}$$

- *Real Time Equation 2: Sum of data*

$$\sum X := \sum X \pm \sum X^{new}$$

- *Real Time Equation 4: Sum of Squared data*

$$\sum X^2 := \sum X^2 \pm \sum X^{2^{new}}$$

- *Real Time Equation 14: Sum of Multiplication*

$$\sum X_i X_j := \sum X_i X_j \pm \sum X_i X_j^{new}$$

Real Time Equation 15: Sum of Squared Multiplication

$$\sum \left(X_iX_j\right)^2 := \sum \left(X_iX_j\right)^2 \pm \sum {\left(X_iX_j\right)^2}^{new}$$

As a reminder, the following conditional basic elements are derived directly from the above five basic elements. These equations define univariate statistics for an attribute X_i given a binary attribute X_j when $X_j = 1$.

Real Time Equation 17: Conditional Count

$$Count\left(X_i \middle| X_j = 1\right) = N_{i|j} = \sum X_j$$

Real Time Equation 18: Conditional Sum of data

$$Sum\left(X_i \middle| X_j = 1\right) = \sum X_{i|j} = \sum X_iX_j$$

Real Time Equation 19: Conditional Sum of Squared data

$$Sum\ of\ Squares\left(X_i \middle| X_j = 1\right) = \sum X_{i|j}^2 = \sum (X_iX_j)^2$$

- *Real Time Equation 23: Complement Conditional Count*

$$Count\left(X_i \middle| X_j = 0\right) = N_{i|\bar{j}} = N_{ij} - \sum X_j$$

- *Real Time Equation 24: Complement Conditional Sum of data*

$$Sum\left(X_i \middle| X_j = 0\right) = \sum X_{i|\bar{j}} = \sum X_i - \sum X_i X_j$$

- *Real Time Equation 25: Complement Conditional Sum of Squared data*

$$Sum\ of\ Squares\left(X_i \middle| X_j = 0\right) = \sum X_{i|\bar{j}}^2 = \sum X_i^2 - \sum (X_i X_j)^2$$

Real Time Bivariate Data Analysis - Examples

The following bivariate real time statistical quantities are based on the Iris dataset in the Appendix A. To calculate any bivariate quantity, we only need to use the elements of the Basic Elements Table (BET) generated from the Iris dataset. All the BET elements are updateable in real time.

.

sepal_length (1), petal_length (3)	
indices	$i = 1 \; and \; j = 3$
Covariance	$Covar(X_1, X_3) = \frac{\sum X_1 X_3 - \frac{\sum X_1 \sum X_3}{N_{1,3}}}{N_{1,3}}$ $= \frac{3484.25 - \frac{876.5 \times 563.8}{150}}{150} = 1.265$
Linear Correlation	$R = \frac{Covar(X_1, X_3)}{\sqrt{S_1^2 \times S_3^2}} = \frac{1.265}{\sqrt{0.681 \times 3.092}} = 0.872$ *where:* $S_1^2 = \frac{\sum X_1^2 - \frac{(\sum X_1)^2}{N_{1,3}}}{N_{1,3}} = \frac{5223.85 - \frac{(876.5)^2}{150}}{150} = 0.681$ $S_3^2 = \frac{\sum X_3^2 - \frac{(\sum X_3)^2}{N_{1,3}}}{N_{1,3}} = \frac{2583 - \frac{(563.8)^2}{150}}{150} = 3.092$

Figure 3.4 *Bivariate analysis on two numerical attributes.*

sepal_length (1) , sepal_width_b1 (7) , sepal_width_b2 (8)	
indices	$i = 1 \; and \; j = 7, 8$
Z test	$Z = \frac{\bar{X}_1 - \bar{X}_2}{\sqrt{\frac{S_1^2}{N_1} + \frac{S_2^2}{N_2}}} = \frac{5.953 - 5.776}{\sqrt{\frac{0.450}{57} + \frac{0.811}{93}}} = 1.368$ $P(Z) = P(1.368) = 0.17$ *where:* $\bar{X}_1 = \bar{X}_{1\mid 7} = \frac{\sum X_1 X_7}{\sum X_7} = \frac{339.3}{57} = 5.953$ $\bar{X}_2 = \bar{X}_{1\mid 8} = \frac{\sum X_1 X_8}{\sum X_8} = \frac{537.2}{93} = 5.776$ $S_1^2 = S_{1\mid 7}^2 = \frac{\sum (X_1 X_7)^2 - \frac{(\sum X_1 X_7)^2}{\sum X_7}}{\sum X_7} = \frac{2045.37 - \frac{(339.3)^2}{57}}{57} = 0.450$ $S_2^2 = S_{1\mid 8}^2 = \frac{\sum (X_1 X_8)^2 - \frac{(\sum X_1 X_8)^2}{\sum X_8}}{\sum X_8} = \frac{3178.48 - \frac{(537.2)^2}{93}}{93} = 0.811$ $N_1 = N_{1\mid 7} = \sum X_7 = 57$ $N_2 = N_{1\mid 8} = \sum X_8 = 93$

Figure 3.5 *Bivariate analysis on one numerical attribute and one categorical (binary) attribute.*

<table>
<tr><th colspan="2">petal_width_b1 (11) , petal_width_b2 (12)
Iris_setosa (13) , Iris_versicolor (14) , Iris_virginica (15)</th></tr>
<tr><td><em>indices</em></td><td>$i = 11, 12 \; and \; j = 13, 14, 15$</td></tr>
<tr><td><em>Chi</em>²
<em>test</em></td><td>

$$\chi^2 = \sum_{i=1}^{r} \sum_{j=1}^{c} \frac{(n_{ij} - e_{ij})^2}{e_{ij}}$$

$$= \frac{(33.33)^2}{16.67} + \frac{(-16.67)^2}{16.67} + \frac{(-16.67)^2}{16.67} + \frac{(-33.33)^2}{33.33} + \frac{(16.67)^2}{33.33} + \frac{(16.67)^2}{33.33} = 150$$

$df = (2-1)\times(3-2) = 2$
$P(\chi^2, df) = P(150,2) = 0$

<table>
<tr><th>n_{ij}</th><th>Iris_setosa</th><th>Iris_versicolor</th><th>Iris_virginica</th><th></th></tr>
<tr><td>petal_width_b1</td><td>$\sum X_{11|13}$
$= 50$</td><td>$\sum X_{11|14} = 0$</td><td>$\sum X_{11|15}$
$= 0$</td><td>50</td></tr>
<tr><td>petal_width_b2</td><td>$\sum X_{12|13}$
$= 0$</td><td>$\sum X_{12|14}$
$= 50$</td><td>$\sum X_{12|15}$
$= 50$</td><td>100</td></tr>
<tr><td></td><td>50</td><td>50</td><td>50</td><td>150</td></tr>
</table>

</td></tr>
</table>

e_{ij}	Iris_setosa	Iris_versicolor	Iris_virginica	
petal_width_b1	$\frac{50\times50}{150} = 16.67$	$\frac{50\times50}{150} = 16.67$	$\frac{50\times50}{150} = 16.67$	50
petal_width_b2	$\frac{50\times100}{150} = 33.33$	$\frac{50\times100}{150} = 33.33$	$\frac{50\times100}{150} = 33.33$	100
	50	50	50	150

Figure 3.6 *Bivariate analysis on two categorical attributes.*

4.0 Real Time Classification

Classification refers to the data mining task of attempting to build a predictive model when the target is categorical. The main goal of classification is to divide a dataset into mutually exclusive groups such that the members of each group are as close as possible to one another, and different groups are as far as possible from one another. There are many different classification algorithms (e.g., Naïve Bayesian, Decision Tree, Support Vector Machines, etc.). However, not all classification algorithms can have a real time version. Here we discuss three classification algorithms which can be built and updated in real time using the Basic Elements Tables.

- Naïve Bayesian
- Linear Discriminant Analysis
- Linear Support Vector Machines

4.1 Naïve Bayesian

The Naïve Bayesian classifier is based on Bayes' theorem with independence assumptions between attributes. A Naïve Bayesian model is easy to build, with no complicated iterative parameters estimation which makes it particularly useful for very large datasets. Despite its simplicity, the Naïve Bayesian classifier often does surprisingly well and is widely used because it often outperforms more sophisticated classification methods.

Algorithm – Real Time

Bayes' theorem provides a way of calculating the posterior probability, $P(c|a)$, from the class (binary attribute) prior probability, $P(c)$, the prior probability of the value of attribute, $P(a)$, and the likelihood, $P(a|c)$. Naive Bayes classifier assumes that the effect of the value of attribute (a) on a given class (c) is independent of the values of other attributes. This assumption is called class conditional independence.

Bayes' Rule:

$$P(c|a) = \frac{P(c)\times P(a|c)}{P(a)}$$

Class conditional independence:

$$P(c|A) = P(c|a_1)\times P(c|a_2)\times P(c|a_3)\times \ldots\times P(c|a_n)\times P(c)$$

In the real time version of Bayesian classifiers we calculate the likelihood and the prior probabilities from the Basic Elements Table (BET) which can be updated in real time.

➔ *Real Time Equation 41: Likelihood*

$$P(a|c) = \bar{X}_{a|c} = \frac{\sum X_a X_c}{\sum X_c} \qquad (41)$$

➔ *Real Time Equation 42: Class Prior Probability*

$$P(c) = \bar{X}_c = \frac{\sum X_c}{N_c} \qquad (42)$$

Real Time Equation 43: Attribute-value Prior Probability

$$P(a) = \bar{X}_a = \frac{\sum X_a}{N_a} \qquad (43)$$

where:

- $\bar{X}_{a|c}$: *Conditional Mean (20)*
- $\bar{X}_c$ *and* $\bar{X}_a$: *Mean (3)*
- $\sum X_a X_c$: *Sum of Multiplication (14)*
- $\sum X_c$ *and* $\sum X_a$: *Sum of data (2)*
- N_c *and* N_a : *Count (1)*

If the attribute (*A*) is numerical the likelihood for its value (*a*) can be calculated from the normal distribution equation.

$$P(a|c) = \frac{1}{\sqrt{2\pi \times S^2}} e^{-\frac{(a-\bar{X})}{2S^2}}$$

Numerical Attribute Mean:

$$\bar{X} = \bar{X}_{a|c} = \frac{\sum X_a X_c}{\sum X_c}$$

Numerical Attribute Variance:

$$S^2 = S_{a|c}^2 = \frac{\sum(X_a X_c)^2 - \frac{(\sum X_a X_c)^2}{\sum X_c}}{\sum X_c}$$

where:

- $\bar{X}_{a|c}$: *Conditional Mean (20)*
- $S_{a|c}^2$: *Conditional Variance (21)*
- $\sum(X_a X_c)^2$: *Sum of Squared Multiplication (15)*
- $\sum X_a X_c$: *Sum of Multiplication (14)*
- $\sum X_c \ and \ \sum X_a$: *Sum of data (2)*
- $N_c \ and \ N_a$: *Count (1)*

In practice, there is no need to calculate *P(a)* because it is a constant value for all the classes and can be considered as a normalization factor.

Attribute Contribution

Kononenko's information gain as a sum of information contributed by each attribute (*A*) can offer an explanation on how values of the predictors influence the class probability.

$$log_2 P(c|a) - log_2 P(c)$$

The contribution of attributes can also be visualized by plotting **nomograms**. A nomogram plots log-odds ratios for each value of each attribute. Lengths of the lines correspond to spans of odds ratios, suggesting importance of the related predictor. It also shows impacts of individual values of the predictor.

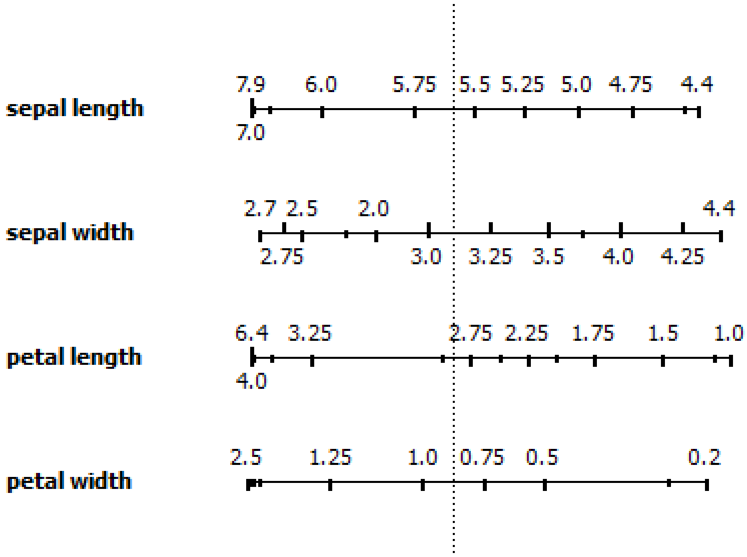


Figure 4.1 *Nomogram for the Iris dataset*

Both Kononenko's information gain and log-odds ratios can be calculated from the Basic Elements Table.

Real Time Naive Bayesian - Example

All the following real time statistical quantities are based on the Iris dataset in the Appendix A. To calculate any Bayesian quantity, we only need to use the elements of the Basic Elements Table (BET) generated from the transformed Iris dataset.

Attribute: sepal_length_b1 (binary) **Target (Class): Iris_versicolor (binary)**	
indices	$i = 5 \; and \; j = 14$
Posterior Probability	$P(c\|a) = P(Iris\_versicolor = 1\|speal\_length\_b1 = 1) = \frac{P(c)\times P(a\|c)}{P(a)} = \frac{0.333\times 0.52}{0.553} = 0.313$
Class Prior Probability	$P(c) = P(Iris\_versicolor \; = 1) = \bar{X}_{14} = \frac{\sum X_{14}}{N_{14,14}} = \frac{50}{150} = 0.333$
Likelihood	$P(a\|c) = P(speal\_length\_b1 = 1\|Iris\_versicolor = 1) = \bar{X}_{5\|14} = \frac{\sum X_5 X_{14}}{\sum X_{14}} = \frac{26}{50} = 0.52$

Attribute Prior Probability	$P(a) = P(sepal\_length\_b1 = 1) = \bar{X}_{15} = \frac{\sum X_5}{N_{5,5}}$ $= \frac{83}{150} = 0.553$

Figure 4.2 *A Naïve Bayesian classifier using a categorical (binary) attribute.*

Attribute: sepal_length (numeric) **Target (Class Attribute): Iris_versicolor (binary)**	
indices	$i = 1\ and\ j = 14$
Posterior Probability	$P(c \mid a) = P(Iris\_versicolor = 1 \mid speal\_length = 6)$ $= \frac{P(c)\times P(a \mid c)}{P(a)} = \frac{0.333\times 0.69}{0.431}$ $= 0.533$
Class Prior Probability	$P(c) = P(Iris\_versicolor = 1) = \bar{X}_{14} = \frac{\sum X_{14}}{N_{14,14}}$ $= \frac{50}{150} = 0.333$

Likelihood	$P(a\mid c) = P(sepal\_length = 6\mid Iris\_versicolor = 1)$ $= \frac{1}{\sqrt{2\pi\times S^2}} e^{-\frac{(x-\bar{X})}{2S^2}}$ $= \frac{1}{\sqrt{6.28\times 0.262}} e^{-\frac{(6-5.936)}{2\times 0.262}} = 0.69$ *where:* $\bar{X} = \bar{X}_{1\mid 14} = \frac{\sum X_1 X_{14}}{\sum X_{14}} = \frac{296.8}{50} = 5.936$ $S^2 = S^2_{1\mid 14} = \frac{\sum (X_1 X_{14})^2 - \frac{(\sum X_1 X_{14})^2}{\sum X_{14}}}{\sum X_{14}}$ $= \frac{1774.9 - \frac{(296.8)^2}{50}}{50} = 0.262$
Attribute Prior Probability	$P(a) = P(sepal\_length = 6) = \frac{1}{\sqrt{2\pi\times S^2}} e^{-\frac{(x-\bar{X})}{2S^2}}$ $= \frac{1}{\sqrt{6.28\times 0.681}} e^{-\frac{(6-5.843)}{2\times 0.681}} = 0.431$ *where:* $\bar{X} = \frac{\sum X}{N} = \frac{876.5}{150} = 5.843$ $S^2 = \frac{\sum X^2 - \frac{(\sum X)^2}{N}}{N} = \frac{5223.85 - \frac{(876.5)^2}{150}}{150} = 0.68$

Figure 4.3 *A Naïve Bayesian classifier using a numerical attribute.*

4.2 Linear Discriminant Analysis

Linear Discriminant Analysis (LDA) is a classification method originally developed in 1936 by R. A. Fisher. LDA is simple, mathematically robust and often produces models whose accuracy is as good as more complex methods.

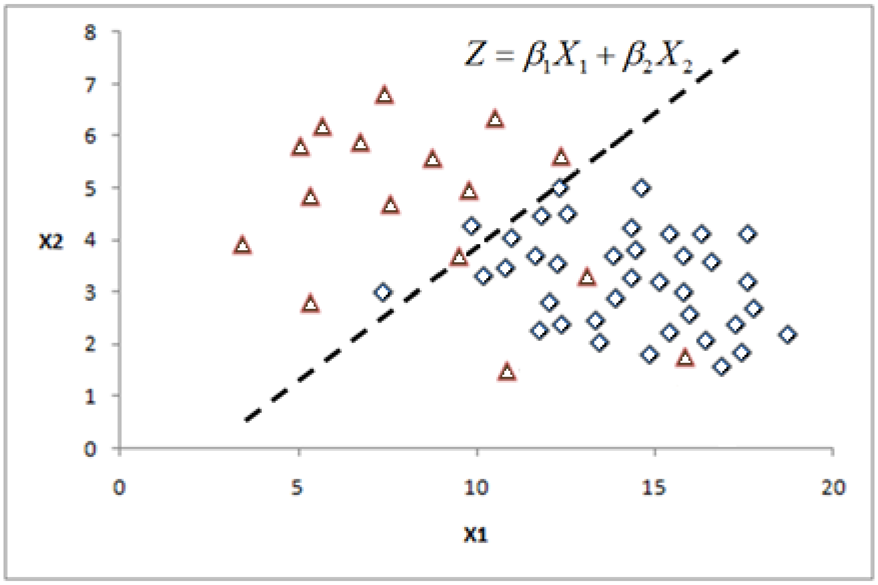


Figure 4.4 *LDA with two attributes separating two classes*

Algorithm – Real Time

LDA is based upon the concept of searching for a linear combination of attributes that best separates two classes (0 and 1) of a binary attribute.

$$Z = \beta_1 X_1 + \beta_2 X_2 + \beta_3 X_3 + \cdots + \beta_n X_n$$

To capture the notion of separability, Fisher defined the following score function.

$$S(\boldsymbol{\beta}) = \frac{\boldsymbol{\beta}^T \boldsymbol{\mu}_0 - \boldsymbol{\beta}^T \boldsymbol{\mu}_1}{\boldsymbol{\beta}^T \boldsymbol{C} \boldsymbol{\beta}} = \frac{\bar{Z}_0 - \bar{Z}_1}{Variance\ of\ Z\ within\ groups}$$

Pooled Covariance Matrix:

$$\boldsymbol{C} = \frac{1}{N_0 + N_1} (N_0 \boldsymbol{C}_0 + N_1 \boldsymbol{C}_1)$$

where:

- $\boldsymbol{\beta}$: Linear model coefficients vector
- $\boldsymbol{C}$: *Pooled covariance matrix*
- $\boldsymbol{\mu}_0$: *Complement Conditional Mean vector (26)*

- $\boldsymbol{\mu}_1$: *Conditional Mean vector (20)*
- N_0 : *Complement Conditional Count (23)*
- N_1 : *Conditional Count (17)*
- $\boldsymbol{C}_0$: *Complement Conditional Covariance matrix*
- $\boldsymbol{C}_1$: *Conditional Covariance matrix*

The conditional covariance ($\boldsymbol{C}_1$) or complement conditional covariance ($\boldsymbol{C}_0$) cannot be computed directly from the Basic Elements Table. However, the pooled covariance matrix ($\boldsymbol{C}$) can be derived from the BET using the following equation.

Real Time Equation 44: Pooled Covariance

$$C_{ij} = \frac{\sum X_i X_j - \frac{\sum X_{i|\bar{k}} \sum X_{j|\bar{k}}}{N_0} - \frac{\sum X_{i|k} \sum X_{j|k}}{N_1}}{N_0 + N_1 - 2} \qquad (44)$$

where:

- $\sum X_i X_j$: *Sum of Multiplication (14)*
- $\sum X_{i|\bar{k}}$ *and* $\sum X_{j|\bar{k}}$: *Complement Conditional Sum of data (24)*
- $\sum X_{i|k}$ *and* $\sum X_{j|k}$: *Conditional Sum of data (18)*
- $N_0 = N_{i|\bar{k}} = N_{j|\bar{k}}$: *Complement Conditional Counts (23)*

- $N_1 = N_{i|k} = N_{j|k}$: *Conditional Counts (17)*

The pooled variance equation (29) is valid if $N_{i|k} = N_{j|k}$ and two attributes should not have uneven number of missing values. Otherwise, it is better to use the overall covariance instead of the pooled covariance.

$$C_{ij} = \frac{\sum X_i X_j - \frac{\sum X_i \sum X_j}{N_{ij}}}{N_{ij}}$$

where:

- C_{ij} : *Covariance (13)*

Now, by having the covariance matrix and given the score function, the problem is to estimate the linear coefficients that maximize the score which can be solved by the following equations.

$$\boldsymbol{\beta} = \boldsymbol{C}^{-1}(\boldsymbol{\mu}_0 - \boldsymbol{\mu}_1)$$

A new data point is classified to class 1 if:

$$\boldsymbol{\beta}^T \left(\boldsymbol{X}\text{-}\frac{1}{2}(\boldsymbol{\mu}_0 - \boldsymbol{\mu}_1)\right) > -\,log\,\frac{P(c_0)}{P(c_1)}$$

$$P(c_0) = \frac{N_0}{N_0 + N_1} \qquad\qquad P(c_1) = \frac{N_1}{N_0 + N_1}$$

where:

- $\boldsymbol{X}$: Input data vector
- $P(c_0)\ and\ P(c_1)$: Probability of two classes (0 and 1)

How good is the model?

One way of assessing the effectiveness of the discrimination is to calculate the Mahalanobis distance between two groups.

$$\Delta^2 = \boldsymbol{\beta}^T(\boldsymbol{\mu_0} - \boldsymbol{\mu_1})$$

A distance greater than 3 indicates that two means differ by more than 3 standard deviations which means the overlap (probability of misclassification) is quite small.

4.2.1 Quadratic Discriminant Analysis (QDA)

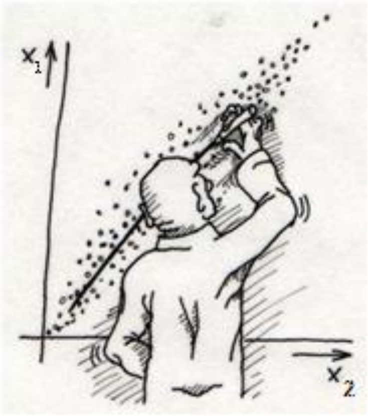
QDA is a general discriminant function with quadratic decision boundaries which can be used to classify datasets with two or more classes.

$$Z_k(\boldsymbol{X}) = -\frac{1}{2}(\boldsymbol{X} - \boldsymbol{\mu}_k)^T \boldsymbol{C}_k^{-1}(\boldsymbol{X} - \boldsymbol{\mu}_k) - \frac{1}{2}\ln|\boldsymbol{C}_k| + \ln P(c_k)$$

Where $\boldsymbol{C}_k^{-1}$ is the inverse covariance matrix for the class *k* and $\boldsymbol{\mu}_k$ is the means vector for class *k*. $|\boldsymbol{C}_k|$ is the determinant of the covariance matrix C_k and $P(c_k)$ is the prior probability of the class k. The classification rule is simply to find the class with highest Z value for the input data ($\boldsymbol{X}$).

The real time version of QDA using the Basic Elements Table is only possible if we replace the covariance matrix for each class $\boldsymbol{C}_k$ with the overall covariance matrix $\boldsymbol{C}$.

$$Z_k(\boldsymbol{X}) = -\frac{1}{2}(\boldsymbol{X}-\boldsymbol{\mu}_k)^T\boldsymbol{C}^{-\mathbf{1}}(\boldsymbol{X}-\boldsymbol{\mu}_k) - \frac{1}{2}\ln|\boldsymbol{C}_k| + \ln P(c_k)$$



Real Time Linear Discriminant Analysis - Example

All the following real time statistical quantities are based on the Iris dataset in the Appendix A. To calculate any LDA quantity, we only need to use the elements of the Basic Elements Table (BET) generated from the transformed Iris dataset.

<table>
<tr><th colspan="2">Attributes: sepal_length and petal_width
Target (Class Attribute): Iris_versicolor</th></tr>
<tr><td><b><i>indices</i></b></td><td>$i = 1, j = \ 4 \ and \ k = 14$</td></tr>
<tr><td><b><i>Classes</i></b></td><td>Iris_versicolor = 1
Iris_versicolor = 0</td></tr>
<tr><td><b><i>LDA</i></b></td><td>
<table>
<tr><th>Iris_versicolor</th><th>Count</th><th>Probability</th><th>Stats</th><th>sepal_length</th><th>petal_width</th></tr>
<tr><td rowspan="2">0</td><td rowspan="2">N_0
$= 100$</td><td rowspan="2">$P(0)$
$= 0.666$</td><td>$\boldsymbol{\mu}_0$</td><td>5.797</td><td>1.135</td></tr>
<tr><td>$\boldsymbol{C}_0$</td><td colspan="2">$\begin{bmatrix} 0.8936 & 0.7414 \\ 0.7414 & 0.8449 \end{bmatrix}$</td></tr>
<tr><td rowspan="2">1</td><td rowspan="2">N_1
$= 50$</td><td rowspan="2">$P(1)$
$= 0.333$</td><td>$\boldsymbol{\mu}_1$</td><td>5.936</td><td>1.326</td></tr>
<tr><td>$\boldsymbol{C}_1$</td><td colspan="2">$\begin{bmatrix} 0.2664 & 0.0558 \\ 0.0558 & 0.0391 \end{bmatrix}$</td></tr>
</table>
$$\boldsymbol{C} = \begin{bmatrix} 0.6846 & 0.5129 \\ 0.5129 & 0.5763 \end{bmatrix}$$
</td></tr>
</table>

	$\boldsymbol{\beta} = \boldsymbol{C}^{-1}(\boldsymbol{\mu}_0 - \boldsymbol{\mu}_1) = [0.1358 \quad -0.4522]$ $Z = 0.1358 \times sepal\_length - 0.4522 \times petal\_width$
N_0	$Count(X_1 \mid X_{14} = 0) = N_{1\mid\overline{14}} = N_{1,14} - \sum X_{14}$ $= 150 - 50 = 100$
N_1	$Count(X_1 \mid X_{14} = 1) = N_{1\mid 14} = \sum X_{14} = 50$
$\boldsymbol{\mu}_0$	$Mean(X_1 \mid X_{14} = 0) = \bar{X}_{1\mid\overline{14}} = \frac{\sum X_{1\mid\overline{14}}}{N_{1\mid\overline{14}}} = \frac{\sum X_1 - \sum X_1 X_{14}}{N_{1,14} - \sum X_{14}}$ $= \frac{876.5 - 296.8}{100} = 5.797$
	$Mean(X_4 \mid X_{14} = 0) = \bar{X}_{4\mid\overline{14}} = \frac{\sum X_{4\mid\overline{14}}}{N_{4\mid\overline{14}}} = \frac{\sum X_4 - \sum X_4 X_{14}}{N_{4,14} - \sum X_{14}}$ $= \frac{179.8 - 66.3}{100} = 1.135$
$\boldsymbol{\mu}_1$	$Mean(X_1 \mid X_{14} = 1) = \bar{X}_{1\mid 14} = \frac{\sum X_{1\mid 14}}{N_{1\mid 14}} = \frac{\sum X_1 X_{14}}{\sum X_{14}} = \frac{296.8}{50}$ $= 5.936$
	$Mean(X_4 \mid X_{14} = 1) = \bar{X}_{4\mid 14} = \frac{\sum X_{4\mid 14}}{N_{4\mid 14}} = \frac{\sum X_4 X_{14}}{\sum X_{14}} = \frac{66.3}{50}$ $= 1.326$

C	$C_{1,1} = \frac{\sum X_1 X_1 - \frac{\sum X_{1\vert\overline{14}} \sum X_{1\vert\overline{14}}}{N_0} - \frac{\sum X_{1\vert 14} \sum X_{1\vert 14}}{N_1}}{N_0 + N_1 - 2}$ $= \frac{5223.9 - \frac{579.7\times 579.7}{100} - \frac{296.8\times 296.8}{50}}{100 + 50 - 2}$ $= 0.6846$
	$C_{1,4} = C_{4,1} = \frac{\sum X_1 X_4 - \frac{\sum X_{1\vert\overline{14}} \sum X_{4\vert\overline{14}}}{N_0} - \frac{\sum X_{1\vert 14} \sum X_{4\vert 14}}{N_1}}{N_0 + N_1 - 2}$ $= \frac{1127.7 - \frac{579.7\times 113.5}{100} - \frac{296.8\times 66.3}{50}}{100 + 50 - 2}$ $= 0.5129$
	$C_{4,4} = \frac{\sum X_4 X_4 - \frac{\sum X_{4\vert\overline{14}} \sum X_{4\vert\overline{14}}}{N_0} - \frac{\sum X_{4\vert 14} \sum X_{4\vert 14}}{N_1}}{N_0 + N_1 - 2}$ $= \frac{302.3 - \frac{113.5\times 113.5}{100} - \frac{66.3\times 66.3}{50}}{100 + 50 - 2}$ $= 0.5763$

Figure 4.5 *A Linear Discriminant Analysis classifier using two numerical attributes.*

4.3 Real Time Linear Support Vector Machine

Support Vector Machine (SVM) performs classification by finding the hyperplane that maximizes the margin between the two classes. The vectors that define the hyperplane are the support vectors.

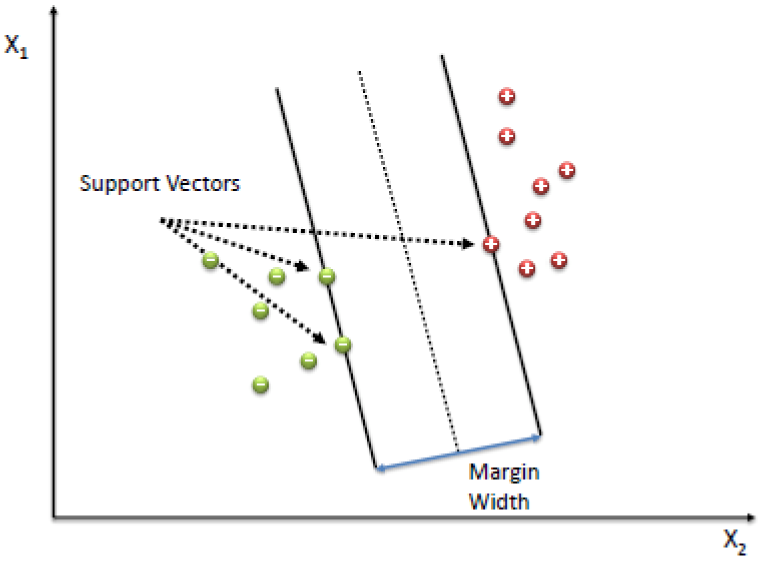


Figure 4.6 *The Support Vectors define an optimal hyperplane.*

Algorithm

- Define an optimal hyperplane: maximize margin.
- Extend the above definition for non-linearly separable problems: have a penalty term for misclassifications.

- Map data to high dimensional space where it is easier to classify with linear decision surfaces: reformulate problem so that data is mapped implicitly to this space.

To define an optimal hyperplane we need to maximize the width of the margin (*w*).

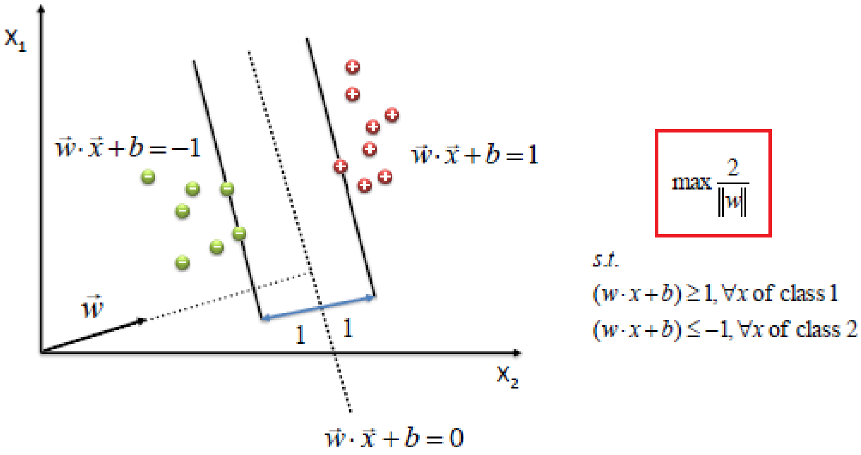

X_1
X_2
$\vec{w}\cdot\vec{x}+b=-1$
$\vec{w}\cdot\vec{x}+b=1$
$\vec{w}\cdot\vec{x}+b=0$
$\vec{w}$
1
1
$\max\frac{2}{\|w\|}$
$s.t.$
$(w\cdot x+b)\geq 1, \forall x$ of class 1
$(w\cdot x+b)\leq -1, \forall x$ of class 2

Figure 4.7 *An ideal SVM analysis produces a hyperplane that completely separates the vectors.*

We find *w* and *b* by solving the following objective function using Quadratic Programming.

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

$$s.t.\ y_i(w\cdot x_i+b)\geq 1,\ \ \forall x_i$$

The beauty of SVM is that if the data is linearly separable, there is a unique global minimum value. An ideal SVM analysis should produce a hyperplane that completely separates the vectors (cases) into two non-overlapping classes. However, perfect separation may not be possible, or it may result in a model with so many cases that the model does not classify correctly. In this situation SVM finds the hyperplane that maximizes the margin and minimizes the misclassifications.

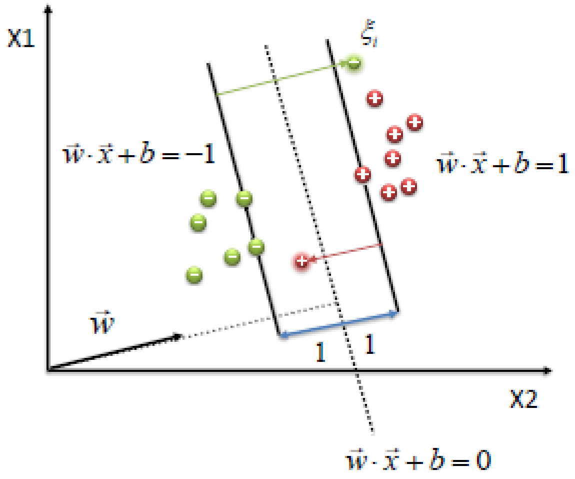

X1
ξ_i
$\vec{w}\cdot\vec{x}+b=-1$
$\vec{w}\cdot\vec{x}+b=1$
$\vec{w}$
1
1
X2
$\vec{w}\cdot\vec{x}+b=0$

Figure 4.8 *The slack variable ξ allows some instances to fall off the margin, but penalize them.*

The algorithm tries to maintain the slack variable to zero while maximizing margin. However, it does not minimize the number of misclassifications (NP-complete problem) but the sum of distances from the margin hyperplanes.

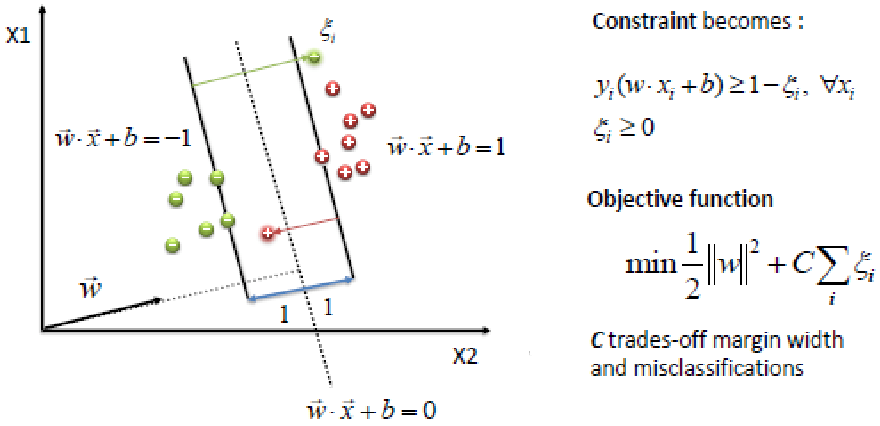

X1
ξ_i
$\vec{w}\cdot\vec{x}+b=-1$
$\vec{w}\cdot\vec{x}+b=1$
$\vec{w}$
1
1
X2
$\vec{w}\cdot\vec{x}+b=0$
Constraint becomes :
$y_i(w\cdot x_i+b)\geq 1-\xi_i,\ \forall x_i$
$\xi_i \geq 0$
Objective function
$\min\frac{1}{2}\|w\|^2+C\sum_i \xi_i$
C trades-off margin width and misclassifications

***Figure 4.9** The objective function penalizes for misclassified instances and those within margin.*

The **Linear Support Vector Machine** (LSVM) finds a hyperplane in the space of input data. The hyperplane splits the positive cases from the negative cases. The split will be chosen to have the largest distance from the hyperplane to the nearest of the positive and negative cases. Our implementation of the Real Time LSVM is based on Fung and Mangasarian solution [5].

Algorithm – Real Time

Linear Proximal SVM Algorithm defined by Fung and Mangasarian**:** *Given m data points in R^{n} represented by the m x n matrix **A** and a <u>diagonal</u> **D** of +1 and -1 labels denoting the class of each row of **A**, we generate the linear classifier as follows:*

The Linear Classifier:

$$sign(\boldsymbol{x}'\boldsymbol{\omega} - \gamma)\begin{cases} = +1, & then\ \boldsymbol{x} \in A + \\ = -1, & then\ \boldsymbol{x} \in A - \end{cases}$$

(i) Define ***E*** by:

$$\boldsymbol{E} = [\boldsymbol{A} \quad -\boldsymbol{e}]$$

where ***e*** is an m×1 vectors of ones.

(ii) Compute $\begin{bmatrix}\boldsymbol{\omega}\\ \gamma\end{bmatrix}$ by for some positive ***v***:

$$\begin{bmatrix}\boldsymbol{\omega}\\ \gamma\end{bmatrix} = \left(\frac{\boldsymbol{I}}{v} + \boldsymbol{E}'\boldsymbol{E}\right)^{-1} \boldsymbol{E}'\boldsymbol{D}\boldsymbol{e}$$

Typically ***v*** is chosen by means of a tuning (validating) set. And I is $speye(n + 1)$ which forms the sparse representation of $n + 1$.

(iii) Classify a new case by using the Linear Classifier and the solution $\begin{bmatrix} \boldsymbol{\omega} \\ \gamma \end{bmatrix}$ from the step above.

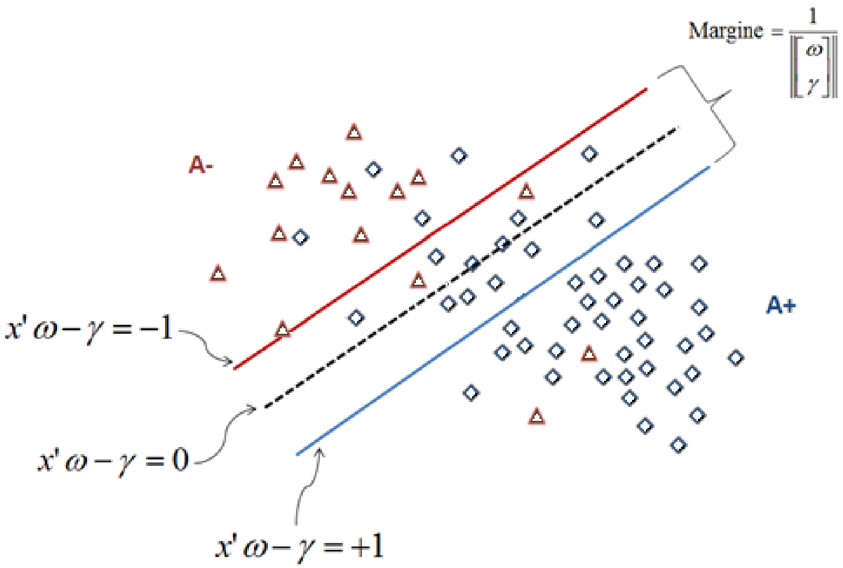

Margine = $\frac{1}{\left\| \begin{bmatrix} \omega \\ \gamma \end{bmatrix} \right\|}$
A-
A+
$x'\omega - \gamma = -1$
$x'\omega - \gamma = 0$
$x'\omega - \gamma = +1$

Figure 4.10 *Proximal Support Vector Machine classifier.*

As shown below the two main components of the proximal LSVM equation ($\boldsymbol{E'E}$ *and* $\boldsymbol{E'De}$) can directly be extracted from the Basic Elements Table (BET).

Real Time Equation 45: $\boldsymbol{E'E}$

$E'E$			
$\sum X_1X_1$	$\sum X_1X_i$	$\sum X_1X_n$	$-\sum X_1$
$\sum X_iX_1$	$\sum X_iX_i$	$\sum X_iX_n$	$-\sum X_i$
$\sum X_nX_1$	$\sum X_nX_i$	$\sum X_nX_n$	$-\sum X_n$
$-\sum X_1$	$-\sum X_i$	$-\sum X_n$	N

Real Time Equation 46: $\boldsymbol{E'De}$

$E'De$
$2\sum X_1X_{target} - \sum X_1$
$2\sum X_iX_{target} - \sum X_i$
$2\sum X_nX_{target} - \sum X_n$
$N_{target} - 2\sum X_{target}$

where:

- N : *Count (1)*
- $\sum X$: *Sum of data (2)*
- $\sum X_i X_j$: *Sum of Multiplication (14)*

The above equations are valid if all attributes have the same count (*N*) and also do not have uneven number of missing values. However, in practice the maximum count can be tried in this situation.

Real Time Support Vector Machine - Example

All the following real time statistical quantities are based on the Iris dataset in the Appendix A. To calculate any LSVM quantity, we only need to use the elements of the Basic Elements Table (BET) generated from the transformed Iris dataset.

<table>
<tr><th colspan="2">Attributes: sepal_length and petal_width
Target (Class Attribute): Iris_versicolor</th></tr>
<tr><td><i>indices</i></td><td>$i = 1, j = \ 4 \; and \; target = 14$</td></tr>
<tr><td><i>Classes</i></td><td>Iris_versicolor = 1
Iris_versicolor = 0</td></tr>
<tr><td><i>LSVM</i></td><td>$\begin{bmatrix}\boldsymbol{\omega}\\ \gamma\end{bmatrix} = \left(\frac{I}{v} + \boldsymbol{E'E}\right)^{-1} \boldsymbol{E'De} = \begin{bmatrix}\omega_1 = -0.0606\\ \omega_2 = 0.0560\\ \gamma = 0.0231\end{bmatrix}$</td></tr>
<tr><td>$\boldsymbol{E'E}$</td><td>
<table>
<tr><th colspan="3">$\boldsymbol{E'E}$</th></tr>
<tr><td>$\sum X_1X_1 = 5223.9$</td><td>$\sum X_1X_4 = 1127.7$</td><td>$-\sum X_1 = -876.5$</td></tr>
<tr><td>$\sum X_4X_1 = 1127.7$</td><td>$\sum X_4X_4 = 302.3$</td><td>$-\sum X_4 = -179.8$</td></tr>
<tr><td>$-\sum X_1 = -876.5$</td><td>$-\sum X_4 = -179.8$</td><td>$N = 150$</td></tr>
</table>
</td></tr>
</table>

$\boldsymbol{E'De}$	$\boldsymbol{E'De}$ $2\sum X_1X_{14} - \sum X_1 = 2\times 296.8 - 876.5 = -282.9$ $2\sum X_4X_{14} - \sum X_4 = 2\times 66.3 - 179.8 = -47.2$ $N_{14} - 2\sum X_{14} = 150 - 2\times 50 = 50$
	$v = \frac{1}{N_3} = \frac{1}{150} = 0.0067$; $I = speye(3)$

***Figure 4.11** SVM classifier using two numerical attributes.*

5.0 Real Time Regression

Regression refers to the data mining problem of attempting to build a predictive model when the target is numerical. The simplest form of regression, simple linear regression, fits a line to a set of data. Advanced techniques, such as multiple regression or decision trees use more than one attribute and allow for fitting of more complex models. There are many different regression algorithms (e.g., Linear or Non-Linear Regression, Decision Trees for Regression, Neural Networks, etc.). However, not all regression algorithms can have a real time version. Here we discuss three regression algorithms which can be built and updated in real time using the Basic Elements Tables.

- Simple and Multiple Linear Regression (SLR and MLR)
- Principal Components Analysis and Regression (PCA and PCR)
- Linear Support Vector Regression (SVR)

5.1 Real Time Simple Linear Regression

The Simple Linear Regression (SLR) is a method used to model the linear relationship between a target (dependent variable *Y*) and an attribute (independent variable *X*).

$$Y = b_0 + b_1 X + \varepsilon$$

➔ *Real Time Equation 47: Simple Linear Regression Slope*

$$b_1 = \frac{Covar(X, Y)}{Var(X)} \qquad (47)$$

➔ *Real Time Equation 48: Simple Linear Regression Intercept*

$$b_0 = \bar{Y} - b_1 \bar{X} \qquad (48)$$

where:

- b_0 *and* b_1 : Coefficients (intercept and slope)
- $Covar(X, Y)$: *Covariance (13)*
- $Var(X)$: *Variance (5)*
- $\bar{X}$ *and* $\bar{Y}$: *Averages (3)*

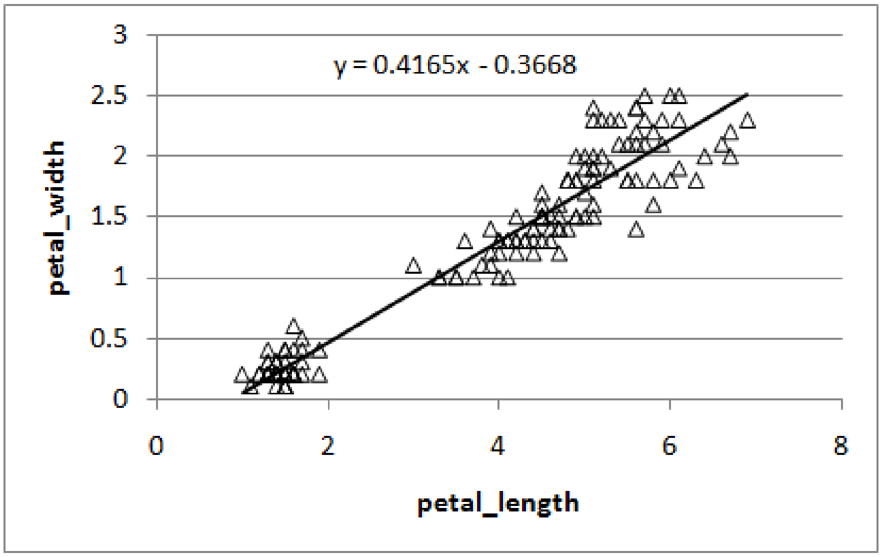


Figure 5.1 *Simple Linear Regression for "petal length" and "petal width" (see the Iris dataset in the Appendix A).*

Real Time Simple Linear Regression - Example

All the following real time statistical quantities are based on the Iris dataset in the Appendix A. To calculate any SLR quantity, we only need to use the elements of the Basic Elements Table generated from the Iris dataset.

petal_width (*Y*) **petal_length (*X*)**	
indices	$i = 3 \;\; and \;\; j = 4$
SLR	$petal\_width = b_0 + b_1 \times petal\_length$ $petal\_width = -0.3668 + 0.4165 \times petal\_length$
Coefficients	$b_1 = \frac{Covar(X_3, X_4)}{Var(X_3)} = \frac{1.288}{3.092} = 0.4165$ $b_0 = \bar{X}_4 - b_1\bar{X}_3 = 1.1987 - 0.4165 \times 3.7587 = -0.3668$
Covariance	$Covar(X_3, X_4) = \frac{\sum X_3 X_4 - \frac{\sum X_3 \sum X_4}{N_{3,4}}}{N_{3,4}}$ $= \frac{869.0 - \frac{563.8 \times 179.8}{150}}{150} = 1.288$

Variance	$Var(X_3) = \frac{\sum X_3^2 - \frac{(\sum X_3)^2}{N_{3,4}}}{N_{3,4}} = \frac{2583.0 - \frac{(563.8)^2}{150}}{150}$ $= 3.092$
Averages	$\bar{X}_3 = \frac{\sum X_3}{N_{3,4}} = \frac{563.8}{150} = 3.7587$ $\bar{X}_4 = \frac{\sum X_4}{N_{3,4}} = \frac{179.8}{150} = 1.1987$

***Figure** 5.2 Simple linear regression using two numerical attributes.*

5.2 Real Time Multiple Linear Regression

Multiple Linear Regression (MLR) is a method used to model the linear relationship between a target (dependent variable) and one or more attributes (independent variables).

Observed data:

$$Y = b_0 + b_1 X_1 + b_2 X_2 + \cdots + b_p X_p + \varepsilon$$

Predicted data:

$$Y' = b_0 + b_1 X_1 + b_2 X_2 + \cdots + b_p X_p$$

Error:

$$\varepsilon = Y - Y'$$

MLR is based on ordinary least squares (OLS), the regression model is fit such that the sum-of-squares of differences of observed and predicted values (error) is minimized.

$$minimize \sum (Y - Y')^2$$

➔ *Real Time Equation 49: Multiple Linear Regression Coefficients*

$$\boldsymbol{b} = \boldsymbol{C}_{XX}^{-1} \boldsymbol{C}_{XY} \qquad (49)$$

➔ *Real Time Equation 50: Multiple Linear Regression Intercept*

$$b_0 = \bar{Y} - \boldsymbol{b}\bar{\boldsymbol{X}} \qquad (50)$$

where:

- Y : Observed target (dependent variable) data
- Y' : Predicted target data
- $\boldsymbol{b}$: Linear regression model coefficients
- $\boldsymbol{C_{XX}^{-1}}$: Inverse covariance matrix between attributes (independent variables)
- $\boldsymbol{C_{XY}}$: Covariance vector between attributes and target

The MLR model is based on several assumptions (e.g., errors are normally distributed with zero mean and constant variance). Provided the assumptions are satisfied, the regression estimators are optimal in the sense that they are unbiased, efficient, and consistent. Unbiased means that the expected value of the estimator is equal to the true value of the parameter. Efficient means that the estimator has a smaller variance than any other estimator. Consistent means that the bias and variance of the estimator approach zero as the sample size approaches infinity.

Real Time Algorithm

There are just three steps to build a real time MLR model using the Basic Elements Table.

Step 1: Calculate the covariance matrix between attributes (independent variables).

$\boldsymbol{C_{XX}}$	X_1	X_j	X_p
X_1	C_{11}	C_{1j}	C_{1p}
X_i	C_{i1}	C_{ij}	C_{ip}
X_p	C_{p1}	C_{pj}	C_{pp}

where:

- C_{ij} : *Covariance (13)*
- $C_{ij} = C_{ji}$

Step 2: Calculate the covariance vector between the target (dependent variable) and the attributes (independent variables).

$\boldsymbol{C_{XY}}$	Y
X_1	C_{1y}
X_i	C_{iy}
X_p	C_{py}

where:

- C_{iy} : *Covariance (13)*

Step 3: Calculate the regression model coefficients.

$$\boldsymbol{b} = \boldsymbol{C}_{XX}^{-1}\boldsymbol{C_{XY}}$$

where:

- $\boldsymbol{C}_{XX}^{-1}$: Inverse covariance matrix between attributes
- $\boldsymbol{C_{XY}}$: Covariance vector between attributes and target

How good is the model?

R^2 (coefficient of determination) summarizes the explanatory power of the regression model and computed from the sums-of-squares terms.

Coefficient of Determination:

$$R^2 = \frac{SSR}{SST} = 1 - \frac{SSE}{SST}$$

Sum of Squares Total:

$$SST = \sum (Y - \bar{Y})^2$$

Sum of Squares Regression:

$$SSR = \sum (Y' - \bar{Y})^2$$

Sum of Squares Error:

$$SSE = \sum (Y - Y')^2$$

The coefficient of determination, R^2 describes the proportion of variance of the target explained by the regression model. If the regression model is “perfect”, *SSE* is zero, and R^2 is 1. If the regression model is a total failure, *SSE* is equal to *SST*, no variance is explained by the model, and R^2 is zero. It is important to keep in mind that there is no direct relationship between high R^2 and causation.

How significant is the model?

F-ratio estimates the statistical significance of the regression model and is computed from the sum of squares regression and total. The significance of the *F*-ratio is obtained by referring to the *F* distribution using degrees of freedom (df_1, df_2).

$$F = \frac{(n-1)SSR}{(n-p-1)SST}$$

$$df_1 = N - 1$$

$$df_2 = N - p - 1$$

The advantage of the *F*-ratio over R^2 is that the *F*-ratio incorporates sample size and number of attributes in assessment of significance of the regression model. A model can have a large R^2 and still not be statistically significant.

How significant are the coefficients?

If the regression model is significantly good, we can use *t*-test to estimate the statistical significance of each coefficient.

$$t_{(N-p-1)} = \frac{b_i}{S_e(b_i)} = \frac{b_i}{\sqrt{S_e^2(\boldsymbol{X}'\boldsymbol{X})^{-1}}}$$

Standard error of Estimation:

$$S_e = \sqrt{\frac{SSE}{N - p - 1}}$$

$$df = N - p - 1$$

Real Time Equation 51: Sum of Multiplications Matrix ($\boldsymbol{X'X}$*)*

$\boldsymbol{X'X}$	X_1	X_j	X_p
X_1	$\sum X_1X_1$	$\sum X_1X_j$	$\sum X_1X_p$
X_i	$\sum X_iX_1$	$\sum X_iX_j$	$\sum X_iX_p$
X_p	$\sum X_pX_1$	$\sum X_pX_j$	$\sum X_pX_p$

where:

- p *:* number of attributes
- N : *Count (1)*
- $\boldsymbol{X'X}$: *Sum of Multiplication (14)*

The significance of the *t*-test is obtained by referring to the *t* distribution using the degree of freedom (df).

Multicolinearity

A high degree of multicolinearity between attributes produces unreliable regression coefficient estimates. Signs of multicolinearity include:

1. High correlation between pairs of attributes.
2. Regression coefficients whose signs or magnitudes do not make good sense.
3. Statistically non-significant regression coefficients on important attributes.
4. Extreme sensitivity of sign or magnitude of regression coefficients to insertion or deletion of an attribute.

The diagonal values in the $(\boldsymbol{X'X})^{-1}$ matrix called **Variance Inflation Factors** (VIFs) and they are very useful measures of multicolinearity. VIF equals 1 means no colinearity but if any VIF exceed 5, multicolinearity is a problem and removing one or more of the independent attributes from the regression model is needed.

Model Selection

A frequent problem in data mining is to avoid attributes that do not contribute significantly to model prediction. First, it has been shown that dropping attributes that have insignificant coefficients can reduce the average error of predictions. Second, estimation of regression coefficients is likely to be unstable due to multicolinearity in models with many attributes. Finally, a simpler model is a better model with more insight into the influence of attributes in models. There are two main methods of model selection, Backward Elimination and Forward Selection.

Backward Elimination

We start with a model with all selected attributes and eliminate the least statistically significant attribute using t-test. We stop the elimination procedure when the least significant attribute has probability, $P(t, df)$ less than the predefined threshold (e.g., 0.05).

Forward Selection

We start with no attributes in the model, trying out the attributes one by one and including them in the model if they are statistically significant. In the first round of iterations, we build a simple linear regression model for each attribute, and calculate the improvement in the sum of squares of error for each of these resulting models relative to the intercept only model using F-test and select the attribute associated with the lowest P-value model as the first candidate attribute.

$$F = \frac{SSE\left(X_1, X_2, \dots, X_p\right) - SSE(X_1, X_2, \dots, X_p, X_{p+1})}{\frac{SSE(X_1, X_2, \dots, X_p, X_{p+1})}{N - p}}$$

$$Probability(F, df_1, df_2) = P(F, 1, N - p)$$

This one-term model provides a new starting model for the next round. We stop the forward selection procedure if in any rounds the lowest candidate P-value is not lower than the predefined probability.

Real Time Multiple Linear Regression - Example

All the following real time statistical quantities are based on the Iris dataset in the Appendix A. To calculate any MLR quantity, we only need to use the elements of the Basic Elements Table generated from the Iris dataset.

Attributes: sepal_length, sepal_width, and petal_length Target: petal_width	
indices	$i, j = 1, 2, 3, 4$
MLR	$petal\_width = -0.2498 - 0.2097\ sepal\_length$ $+ 0.2283\ sepal\_width$ $+ 0.5258\ petal\_length$
Coefficients	$\boldsymbol{b} = \boldsymbol{C}_{XX}^{-1}\boldsymbol{C}_{XY}$ $= \begin{bmatrix} 0.6811 & -0.0390 & 1.2652 \\ -0.0390 & 0.1868 & -0.3196 \\ 1.2652 & -0.3196 & 3.0924 \end{bmatrix}^{-1} \begin{bmatrix} 0.5135 \\ -0.1172 \\ 1.2877 \end{bmatrix}$ $= \begin{bmatrix} -0.2097 \\ 0.2283 \\ 0.5258 \end{bmatrix}$ $b_0 = \bar{Y} - \boldsymbol{b}\bar{\boldsymbol{X}} = 1.1987 - 1.4485 = -0.2498$

Covariance Matrix

$$Covar(X_i, X_j) = \frac{\sum X_i X_j - \frac{\sum X_i \sum X_j}{N_{ij}}}{N_{ij}}$$

$\boldsymbol{C}_{XX}$	X_1	X_2	X_3
X_1	$C_{1,1} = 0.6811$	$C_{1,2} = -0.0390$	$C_{1,3} = 1.2652$
X_2	$C_{2,1} = -0.0390$	$C_{2,2} = 0.1868$	$C_{2,3} = -0.3196$
X_3	$C_{3,1} = 1.2652$	$C_{3,2} = -0.3196$	$C_{3,3} = 3.0924$

Covariance Vector

$\boldsymbol{C}_{XY}$	X_4
X_1	$C_{1,4} = 0.5135$
X_2	$C_{2,4} = -0.1172$
X_3	$C_{3,4} = 1.2877$

Averages

$$\bar{Y} = \frac{\sum X_4}{N_{4,4}} = \frac{179.8}{150} = 1.1987$$

$$\bar{X}_1 = \frac{\sum X_1}{N_{1,1}} = \frac{876.5}{150} = 5.8433$$

$$\bar{X}_2 = \frac{\sum X_2}{N_{2,2}} = \frac{458.1}{150} = 3.0540$$

$$\bar{X}_3 = \frac{\sum X_3}{N_{3,3}} = \frac{563.8}{150} = 3.7587$$

$$\bar{\boldsymbol{X}} = [5.8433 \quad 3.0540 \quad 3.7587]$$

Figure 5.3 *Multiple linear regression using four numerical attributes.*

5.3 Real Time Principal Components Analysis

Principal component analysis (PCA) is a classical statistical method. This linear transform has been widely used in data analysis and data compression. The principal components (Eigenvectors) for a dataset can directly be extracted from the covariance matrix as follows:

Real Time Equation 52: Eigenvectors and Eigenvalues

$$\boldsymbol{Ce} = \boldsymbol{\lambda e} \qquad (52)$$

where:

1. $\boldsymbol{C}$: *Covariance matrix (13)*
2. $\boldsymbol{e}$: Eigenvectors
3. $\boldsymbol{\lambda}$: Eigenvalues

Eigenvectors and eigenvalues can be computed using a triangular decomposition module. By sorting the eigenvectors in the order of descending eigenvalues, we can find directions in which the data has the most significant amounts of its variation.

- Principal Components Analysis selects a new set of axes for the data

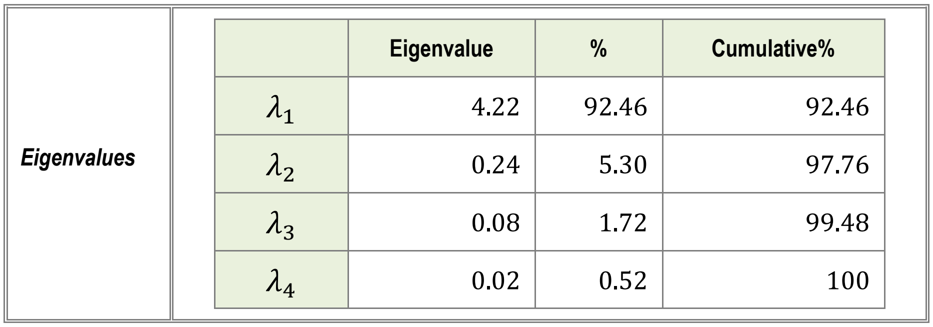
- These axes are selected in decreasing order of variance within the data
- The axes are also perpendicular to each other

Real Time Principal Components Analysis - Example

<table>
<tr><th colspan="2">Attributes: sepal_length, sepal_width, petal_length and petal_width</th></tr>
<tr><td><b><i>indices</i></b></td><td>$i, j = 1, 2, 3, 4$</td></tr>
<tr><td><b><i>PCA</i></b></td><td>$\boldsymbol{Ce = \lambda e}$</td></tr>
<tr><td><b><i>Covariance Matrix</i></b></td><td>

$$Covar(X_i, X_j) = \frac{\sum X_i X_j - \frac{\sum X_i \sum X_j}{N_{ij}}}{N_{ij}}$$

<table>
<tr><th>$\boldsymbol{C}_{XX}$</th><th>X_1</th><th>X_2</th><th>X_3</th><th>X_4</th></tr>
<tr><td>X_1</td><td>$C_{1,1} = 0.6811$</td><td>$C_{1,2} = -0.0390$</td><td>$C_{1,3} = 1.2652$</td><td>$C_{1,4} = 0.5135$</td></tr>
<tr><td>X_2</td><td>$C_{2,1} = -0.0390$</td><td>$C_{2,2} = 0.1868$</td><td>$C_{2,3} = -0.3196$</td><td>$C_{2,4} = -0.1172$</td></tr>
<tr><td>X_3</td><td>$C_{3,1} = 1.2652$</td><td>$C_{3,2} = -0.3196$</td><td>$C_{3,3} = 3.0924$</td><td>$C_{3,4} = 1.2877$</td></tr>
<tr><td>X_4</td><td>$C_{4,1} = 0.5135$</td><td>$C_{4,2} = -0.1172$</td><td>$C_{4,3} = 1.2877$</td><td>$C_{4,4} = 0.5785$</td></tr>
</table>
</td></tr>
<tr><td><b><i>Eigenvectors</i></b></td><td>
<table>
<tr><th></th><th>$\boldsymbol{e}_1$</th><th>$\boldsymbol{e}_2$</th><th>$\boldsymbol{e}_3$</th><th>$\boldsymbol{e}_4$</th></tr>
<tr><td>sepal_length</td><td>0.3616</td><td>0.6565</td><td>-0.5810</td><td>0.3173</td></tr>
<tr><td>sepal_width</td><td>-0.0823</td><td>0.7297</td><td>0.5964</td><td>-0.3241</td></tr>
<tr><td>petal_length</td><td>0.8566</td><td>-0.1758</td><td>0.0725</td><td>-0.4797</td></tr>
<tr><td>petal_width</td><td>0.3588</td><td>-0.0747</td><td>0.5491</td><td>0.7511</td></tr>
</table>
</td></tr>
</table>



		Eigenvalue	%	Cumulative%
Eigenvalues	λ_1	4.22	92.46	92.46
	λ_2	0.24	5.30	97.76
	λ_3	0.08	1.72	99.48
	λ_4	0.02	0.52	100

Figure 5.4 *Principal Components Analysis using four numerical attributes.*

5.4 Real Time Principal Components Regression

The real time **Principal Component Regression** (PCR) method combines the real time Principal Component Analysis (PCA) with the real time Multiple Linear Regression (MLR).

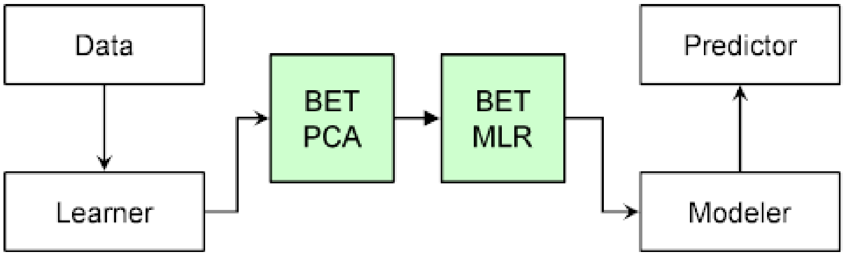


Figure 5.5 *Real Time Principal Components Regression*

First the Learner builds the first Basic Elements Table (BET PCR) from the original data and then by using the eigenvectors builds the second Basic Elements Table (BET MLR) using Scores.

➔ *Real Time Equation 53: PCR Scores*

$$\boldsymbol{S} = \boldsymbol{\lambda}_k(\mathbf{x} - \boldsymbol{\mu}_k) \qquad (53)$$

where:

- $\boldsymbol{S}$: Scores (transformed data vector)
- $\boldsymbol{\lambda}$: Top k eigenvectors
- $\mathbf{x}$: Raw data vector
- $\boldsymbol{\mu}$: Mean vector for the raw data

5.5 Real Time Linear Support Vector Regression

Our implementation of **Linear Support Vector Regression** (LSVR) is based on the Linear Proximal SVM algorithm (see LSVM).

The Linear Regression equation:

$$\boldsymbol{y} = \boldsymbol{x}'\boldsymbol{\omega} - \gamma$$

(i) Define *E* by:

$$\boldsymbol{E} = [\boldsymbol{A} \quad -\boldsymbol{e}]$$

where *e* is an m×1 vectors of ones.

(ii) Compute $\begin{bmatrix}\boldsymbol{\omega}\\ \gamma\end{bmatrix}$ by for some positive *v*:

$$\begin{bmatrix}\boldsymbol{\omega}\\ \gamma\end{bmatrix} = \left(\frac{\boldsymbol{I}}{v} + \boldsymbol{E}'\boldsymbol{E}\right)^{-1} \boldsymbol{E}'\boldsymbol{D}\boldsymbol{e}$$

Typically $\boldsymbol{v}$ is chosen by means of a tuning (validating) set. And I is $speye(n+1)$ which forms the sparse representation of $n+1$.

(iii) Predict a new output by using the Linear Regression and the solution $\begin{bmatrix}\boldsymbol{\omega}\\ \gamma\end{bmatrix}$ from the step above.

As shown below the two main components of the LSVR ($\boldsymbol{E'E}$ *and* $\boldsymbol{E'De}$) can directly be extracted from the Basic Elements Table.

Real Time Equation 54: $\boldsymbol{E'E}$

$\boldsymbol{E'E}$			
$\sum X_1X_1$	$\sum X_1X_i$	$\sum X_1X_n$	$-\sum X_1$
$\sum X_iX_1$	$\sum X_iX_i$	$\sum X_iX_n$	$-\sum X_i$
$\sum X_nX_1$	$\sum X_nX_i$	$\sum X_nX_n$	$-\sum X_n$
$-\sum X_1$	$-\sum X_i$	$-\sum X_n$	N

Real Time Equation 55: $\boldsymbol{E'De}$

$\boldsymbol{E'De}$
$\sum X_1X_{target}$
$\sum X_iX_{target}$
$\sum X_nX_{target}$
$-\sum X_{target}$

where:

- N : *Count (1)*
- $\sum X$: *Sum of data (2)*
- $\sum X_i X_j$: *Sum of Multiplication (14)*

Real Time Support Vector Regression - Example

All the following real time statistical quantities are based on the Iris dataset in the Appendix A. To calculate any LSVM quantity, we only need to use the elements of the Basic Elements Table (BET) generated from the transformed Iris dataset.

<table>
<tr><th colspan="2">Attributes: sepal_length, sepal_width, and petal_length
Target: petal_width</th></tr>
<tr><td><i>indices</i></td><td>$i, j = 1,2,3,4$</td></tr>
<tr><td><i>RSVM</i></td><td>$$\begin{bmatrix}\boldsymbol{\omega}\\ \gamma\end{bmatrix} = \left(\frac{I}{v} + \boldsymbol{E}'\boldsymbol{E}\right)^{-1} \boldsymbol{E}'\boldsymbol{De} = \begin{bmatrix}\omega_1 = -0.2056\\ \omega_2 = 0.2064\\ \omega_3 = 0.5205\\ \gamma = 0.1865\end{bmatrix}$$</td></tr>
<tr><td>$\boldsymbol{E}'\boldsymbol{E}$</td><td>
<table>
<tr><th colspan="4">$\boldsymbol{E}'\boldsymbol{E}$</th></tr>
<tr><td>$\sum X_1X_1 = 5223.9$</td><td>$\sum X_1X_2 = 2671.0$</td><td>$\sum X_1X_3 = 3484.3$</td><td>$-\sum X_1 = -876.5$</td></tr>
<tr><td>$\sum X_2X_1 = 2671.0$</td><td>$\sum X_2X_2 = 1427.1$</td><td>$\sum X_2X_3 = 1673.9$</td><td>$-\sum X_2 = -458.1$</td></tr>
<tr><td>$\sum X_3X_1 = 3484.3$</td><td>$\sum X_3X_2 = 1673.9$</td><td>$\sum X_3X_3 = 2583.0$</td><td>$-\sum X_3 = -563.8$</td></tr>
<tr><td>$-\sum X_1 = -876.5$</td><td>$-\sum X_2 = -458.1$</td><td>$-\sum X_3 = -563.8$</td><td>$N = 150$</td></tr>
</table>
</td></tr>
</table>

$\boldsymbol{E'De}$	$\boldsymbol{E'De}$ $\sum X_1X_4 = 1127.7$ $\sum X_2X_4 = 531.5$ $\sum X_3X_4 = 869.0$ $-\sum X_4 = -179.8$
	$v = 2\ ;\quad I = speye(4)$

Figure 5.6 *SVR model using four numerical attributes.*

6.0 Real Time Sequence Analysis

Sequence Analysis summarizes frequent sequences or states in data or predicts the next likely step of a new sequence. Here we discuss two well-known sequence analysis methods.

- Markov Chains
- Hidden Markov Models

6.1 Real Time Markov Chains

A Markov Chain is a graphical model in the form of a chain. Consider a sequence of "states" X and assume that the conditional probability of X_i depends only on the state immediately before it, X_{i-1}. A time independent Markov chain is called a stationary Markov chain which can be described by a transition probability matrix $\boldsymbol{T}$.

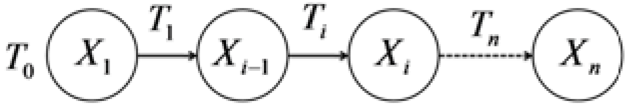


Figure 6.1 *A Markov Chain.*

where:

- $T_i = P(X_i|X_{i-1})$
- $T_0 = P(X_1)$

T_i is posterior probability and can be calculated from the Basic Elements Table as follows (also see real time Naïve Bayesian).

Probability State *i* given State *i-1:*

$$T_i = P(X_i|X_{i-1}) = \frac{P(X_i)\times P(X_{i-1}|X_i)}{P(X_{i-1})}$$

Probability State *i-1* given State *i :*

$$P(X_{i-1}|X_i) = \frac{\sum X_i X_{i-1}}{\sum X_i}$$

Probability State *i* :

$$P(X_i) = \frac{\sum X_i}{N_i}$$

Probability State *i-1*:

$$P(X_{i-1}) = \frac{\sum X_{i-1}}{N_{i-1}}$$

where:

- $\sum X_i X_{i-1}$: *Sum of Multiplication (14)*
- $\sum X_i$ *and* $\sum X_{i-1}$: *Sum of data (2)*
- N_i *and* N_{i-1} : *Count (1)*

6.2 Real Time Hidden Markov Models

Like Markov Chains, a Hidden Markov Model is a graphical model in the form of a chain. It consists of a sequence of "states" ***X*** and with the conditional probability of $X_i$depends only on the state immediately before it, X_{i-1}. In addition to a transition probability ***T***, the HMM model also involves a set of outputs ***Y***, and a time independent emission probability ***E***.

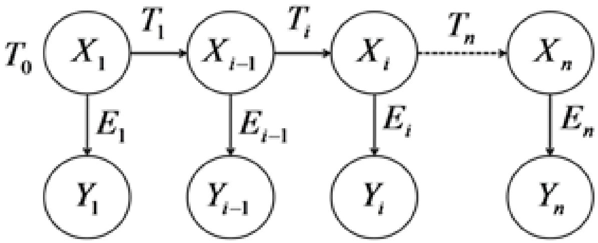


Figure 6.2 *A Hidden Markov Model.*

where:

- $T_i = P(X_i|X_{i-1})$
- $E_i = P(Y_i|X_i)$
- $T_0 = P(X_1)$

T and *E* are posterior probabilities and can be calculated from the Basic Elements Table (also see real time Naïve Bayesian).
Probability of State *i* given State *i-1:*

$$T_i = P(X_i|X_{i-1}) = \frac{P(X_i)\times P(X_{i-1}|X_i)}{P(X_{i-1})}$$

Probability of Output *i* given State *i:*

$$E_i = P(Y_i|X_i) = \frac{P(Y_i)\times P(X_i|Y_i)}{P(X_i)}$$

Probability of State *i-1* given State *i:*

$$P(X_{i-1}|X_i) = \frac{\sum X_i X_{i-1}}{\sum X_i}$$

Probability of State *i* given Output *i:*

$$P(X_i|Y_i) = \frac{\sum X_i Y_i}{\sum Y_i}$$

Probability of State *i:*

$$P(X_i) = \frac{\sum X_i}{N_i}$$

Probability of Output *i:*

$$P(Y_i) = \frac{\sum Y_i}{N_i}$$

Probability of State *i-1:*

$$P(X_{i-1}) = \frac{\sum X_{i-1}}{N_{i-1}}$$

where:

- $\sum X_i X_{i-1}$ and $\sum X_i Y_i$: *Sum of Multiplication (14)*
- $\sum X_i$, $\sum X_{i-1}$ and $\sum Y_i$: *Sum of data (2)*
- N_i and N_{i-1} : *Count (1)*

7.0 Real Time Parallel Processing

The Real Time Learning Machine (RTLM) is a rigorous architecture for making parallel data processing readily accomplished, especially for very large datasets. Any size dataset can be divided to smaller parts and each part can be processed separately. The results can then be joined together to obtain the same model as if we had the whole dataset at once.

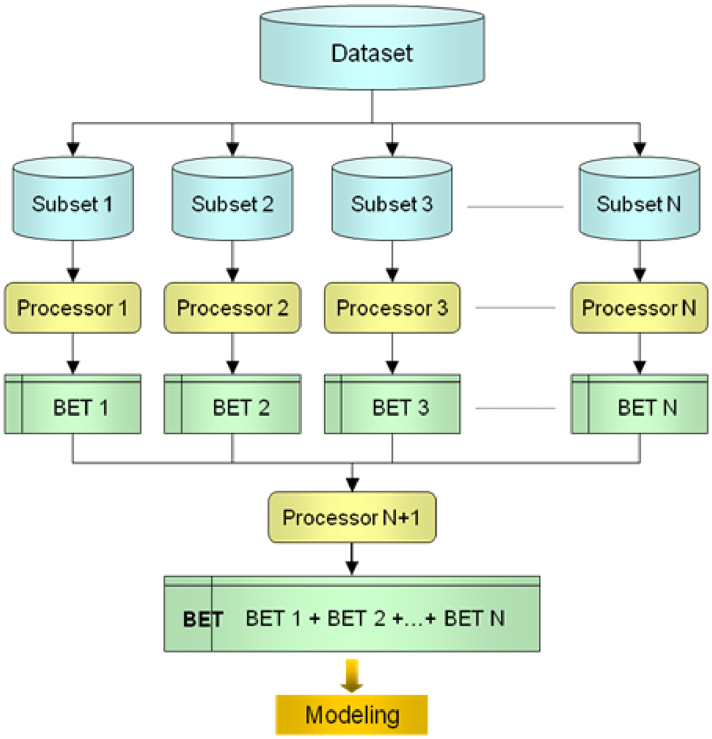


Figure 7.1 *Real Time Parallel Processing*

The processors 1, 2, 3… N build their corresponding Basic Elements Table and the processing time depends on the number of records in the subset but the processor N+1 only combines BETs from the previous step which usually takes about a few milliseconds. Using this architecture we can decrease the processing time linearly by dividing the dataset to any number of subsets. It means if it takes 10 hours to process a dataset with one billion records on a machine with one processor it will take just one hour on a machine with 10 processors by using RTLM.

References:

1. Shuo Yan, Saed Sayad, Stephen T. Balke, “Image Quality in Image Classification: Adaptive Image Quality Modification with Adaptive Classification”, Computers & Chemical Engineering, Computers and Chemical Engineering 33 (2009) 429–435.
2. Keivan Torabi, Saed Sayad and Stephen T. Balke, “On-line adaptive Bayesian classification for in-line particle image monitoring in polymer film manufacturing”, Computers & Chemical Engineering, Volume 30, Issue 1, 15 November 2005, Pages 18-27.
3. K. Torabi, S. Sayad and S.T., Balke, "Adaptive Bayesian classification for real-time image analysis in real- time particle monitoring for polymer film manufacturing”, Fifth International Conference on Data Mining, Text Mining and their Business Applications, Malaga, Spain, 2004. WIT Press.

4. Saed Sayad, Stephen T. Balke and Sina Sayad “An Intelligent Learning Machine”, 4th International Conference on Data Mining, Rio De Janeiro, Brazil, 1-3 December 2003.
5. Glenn Fung, O. L. Mangasarian; Incremental Support Vector Machine Classification, Proceedings of the Second SIAM International Conference on Data Mining, Arlington, Virginia, April 11-13, 2002,R. Grossman, H. Mannila and R. Motwani, editors, SIAM, Philadelphia 2002, 247-260.
6. Sayad S., Sayad M.H., and Sayad J.: “Neural Network with Variable Excitability of Input Units (NN-VEIN)”, Proc. 22nd Annual Pittsburgh Conference on Modeling and Simulation; Pittsburgh, 1991.

APPENDIX A:

The Iris dataset, the transformed Iris dataset and its related Basic Elements Table.

Iris Dataset

sepal length	sepal width	petal length	petal width	iris
5.1	3.5	1.4	0.2	Iris-setosa
4.9	3	1.4	0.2	Iris-setosa
4.7	3.2	1.3	0.2	Iris-setosa
4.6	3.1	1.5	0.2	Iris-setosa
5	3.6	1.4	0.2	Iris-setosa
5.4	3.9	1.7	0.4	Iris-setosa
4.6	3.4	1.4	0.3	Iris-setosa
5	3.4	1.5	0.2	Iris-setosa
4.4	2.9	1.4	0.2	Iris-setosa
4.9	3.1	1.5	0.1	Iris-setosa
5.4	3.7	1.5	0.2	Iris-setosa
4.8	3.4	1.6	0.2	Iris-setosa
4.8	3	1.4	0.1	Iris-setosa
4.3	3	1.1	0.1	Iris-setosa
5.8	4	1.2	0.2	Iris-setosa
5.7	4.4	1.5	0.4	Iris-setosa
5.4	3.9	1.3	0.4	Iris-setosa
5.1	3.5	1.4	0.3	Iris-setosa
5.7	3.8	1.7	0.3	Iris-setosa
5.1	3.8	1.5	0.3	Iris-setosa
5.4	3.4	1.7	0.2	Iris-setosa
5.1	3.7	1.5	0.4	Iris-setosa
4.6	3.6	1	0.2	Iris-setosa
5.1	3.3	1.7	0.5	Iris-setosa
4.8	3.4	1.9	0.2	Iris-setosa
5	3	1.6	0.2	Iris-setosa
5	3.4	1.6	0.4	Iris-setosa
5.2	3.5	1.5	0.2	Iris-setosa

5.2	3.4	1.4	0.2	Iris-setosa
4.7	3.2	1.6	0.2	Iris-setosa
4.8	3.1	1.6	0.2	Iris-setosa
5.4	3.4	1.5	0.4	Iris-setosa
5.2	4.1	1.5	0.1	Iris-setosa
5.5	4.2	1.4	0.2	Iris-setosa
4.9	3.1	1.5	0.1	Iris-setosa
5	3.2	1.2	0.2	Iris-setosa
5.5	3.5	1.3	0.2	Iris-setosa
4.9	3.1	1.5	0.1	Iris-setosa
4.4	3	1.3	0.2	Iris-setosa
5.1	3.4	1.5	0.2	Iris-setosa
5	3.5	1.3	0.3	Iris-setosa
4.5	2.3	1.3	0.3	Iris-setosa
4.4	3.2	1.3	0.2	Iris-setosa
5	3.5	1.6	0.6	Iris-setosa
5.1	3.8	1.9	0.4	Iris-setosa
4.8	3	1.4	0.3	Iris-setosa
5.1	3.8	1.6	0.2	Iris-setosa
4.6	3.2	1.4	0.2	Iris-setosa
5.3	3.7	1.5	0.2	Iris-setosa
5	3.3	1.4	0.2	Iris-setosa
7	3.2	4.7	1.4	Iris-versicolor
6.4	3.2	4.5	1.5	Iris-versicolor
6.9	3.1	4.9	1.5	Iris-versicolor
5.5	2.3	4	1.3	Iris-versicolor
6.5	2.8	4.6	1.5	Iris-versicolor
5.7	2.8	4.5	1.3	Iris-versicolor
6.3	3.3	4.7	1.6	Iris-versicolor
4.9	2.4	3.3	1	Iris-versicolor
6.6	2.9	4.6	1.3	Iris-versicolor
5.2	2.7	3.9	1.4	Iris-versicolor
5	2	3.5	1	Iris-versicolor
5.9	3	4.2	1.5	Iris-versicolor
6	2.2	4	1	Iris-versicolor

6.1	2.9	4.7	1.4	Iris-versicolor
5.6	2.9	3.6	1.3	Iris-versicolor
6.7	3.1	4.4	1.4	Iris-versicolor
5.6	3	4.5	1.5	Iris-versicolor
5.8	2.7	4.1	1	Iris-versicolor
6.2	2.2	4.5	1.5	Iris-versicolor
5.6	2.5	3.9	1.1	Iris-versicolor
5.9	3.2	4.8	1.8	Iris-versicolor
6.1	2.8	4	1.3	Iris-versicolor
6.3	2.5	4.9	1.5	Iris-versicolor
6.1	2.8	4.7	1.2	Iris-versicolor
6.4	2.9	4.3	1.3	Iris-versicolor
6.6	3	4.4	1.4	Iris-versicolor
6.8	2.8	4.8	1.4	Iris-versicolor
6.7	3	5	1.7	Iris-versicolor
6	2.9	4.5	1.5	Iris-versicolor
5.7	2.6	3.5	1	Iris-versicolor
5.5	2.4	3.8	1.1	Iris-versicolor
5.5	2.4	3.7	1	Iris-versicolor
5.8	2.7	3.9	1.2	Iris-versicolor
6	2.7	5.1	1.6	Iris-versicolor
5.4	3	4.5	1.5	Iris-versicolor
6	3.4	4.5	1.6	Iris-versicolor
6.7	3.1	4.7	1.5	Iris-versicolor
6.3	2.3	4.4	1.3	Iris-versicolor
5.6	3	4.1	1.3	Iris-versicolor
5.5	2.5	4	1.3	Iris-versicolor
5.5	2.6	4.4	1.2	Iris-versicolor
6.1	3	4.6	1.4	Iris-versicolor
5.8	2.6	4	1.2	Iris-versicolor
5	2.3	3.3	1	Iris-versicolor
5.6	2.7	4.2	1.3	Iris-versicolor
5.7	3	4.2	1.2	Iris-versicolor
5.7	2.9	4.2	1.3	Iris-versicolor
6.2	2.9	4.3	1.3	Iris-versicolor

5.1	2.5	3	1.1	Iris-versicolor
5.7	2.8	4.1	1.3	Iris-versicolor
6.3	3.3	6	2.5	Iris-virginica
5.8	2.7	5.1	1.9	Iris-virginica
7.1	3	5.9	2.1	Iris-virginica
6.3	2.9	5.6	1.8	Iris-virginica
6.5	3	5.8	2.2	Iris-virginica
7.6	3	6.6	2.1	Iris-virginica
4.9	2.5	4.5	1.7	Iris-virginica
7.3	2.9	6.3	1.8	Iris-virginica
6.7	2.5	5.8	1.8	Iris-virginica
7.2	3.6	6.1	2.5	Iris-virginica
6.5	3.2	5.1	2	Iris-virginica
6.4	2.7	5.3	1.9	Iris-virginica
6.8	3	5.5	2.1	Iris-virginica
5.7	2.5	5	2	Iris-virginica
5.8	2.8	5.1	2.4	Iris-virginica
6.4	3.2	5.3	2.3	Iris-virginica
6.5	3	5.5	1.8	Iris-virginica
7.7	3.8	6.7	2.2	Iris-virginica
7.7	2.6	6.9	2.3	Iris-virginica
6	2.2	5	1.5	Iris-virginica
6.9	3.2	5.7	2.3	Iris-virginica
5.6	2.8	4.9	2	Iris-virginica
7.7	2.8	6.7	2	Iris-virginica
6.3	2.7	4.9	1.8	Iris-virginica
6.7	3.3	5.7	2.1	Iris-virginica
7.2	3.2	6	1.8	Iris-virginica
6.2	2.8	4.8	1.8	Iris-virginica
6.1	3	4.9	1.8	Iris-virginica
6.4	2.8	5.6	2.1	Iris-virginica
7.2	3	5.8	1.6	Iris-virginica
7.4	2.8	6.1	1.9	Iris-virginica
7.9	3.8	6.4	2	Iris-virginica
6.4	2.8	5.6	2.2	Iris-virginica

6.3	2.8	5.1	1.5	Iris-virginica
6.1	2.6	5.6	1.4	Iris-virginica
7.7	3	6.1	2.3	Iris-virginica
6.3	3.4	5.6	2.4	Iris-virginica
6.4	3.1	5.5	1.8	Iris-virginica
6	3	4.8	1.8	Iris-virginica
6.9	3.1	5.4	2.1	Iris-virginica
6.7	3.1	5.6	2.4	Iris-virginica
6.9	3.1	5.1	2.3	Iris-virginica
5.8	2.7	5.1	1.9	Iris-virginica
6.8	3.2	5.9	2.3	Iris-virginica
6.7	3.3	5.7	2.5	Iris-virginica
6.7	3	5.2	2.3	Iris-virginica
6.3	2.5	5	1.9	Iris-virginica
6.5	3	5.2	2	Iris-virginica
6.2	3.4	5.4	2.3	Iris-virginica
5.9	3	5.1	1.8	Iris-virginica

Original and Transformed Attributes

Lable	Attribute	Type	Description
X1	sepal_length	Numeric	Original numeric attribute
X2	sepal_width	Numeric	Original numeric attribute
X3	petal_length	Numeric	Original numeric attribute
X4	petal_width	Numeric	Original numeric attribute

X5	sepal_lengt_b1	Binary	Discretized version of sepal_length:
			IF sepal_length < 6 THEN b1 = 1 ELSE b1=0
			IF sepal_lenghth >= 6THEN b2 = 1 ELSE b2 = 0

X6	sepal_lengt_b2	Binary	

X7	sepal_width_b1	Binary	Discretized version of sepal_width:
			IF sepal_width < 3 THEN b1 = 1 ELSE b1 = 1
			IF sepal_width >= 3 THEN b2 = 1 ELSE b2= 0

X8	sepal_width_b2	Binary	

X9	petal_length_b1	Binary	Discretized version of petal_length:
			IF petal_length < 4 THEN b1 = 1 ELSE b1 = 0
			IF petal_length >= 4 THEN b2=1 ELSE b2 = 0

X10	petal_length_b2	Binary	

X11	petal_width_b1	Binary	Discretized version of petal_width:
			IF petal_width < 1 THEN b1 = 1 ELSE b1 = 0
			IF petal_width >= 1 THEN b2 = 1 ELSE b2 = 0

X12	petal_width_b2	Binary	

X13	Iris_setosa	Binary	Encoded version of the Iris attribute:
			IF Iris =’setosa’ THEN Iris_setosa = 1 ELSE Iris_setosa = 0
			IF Iris =’versicolor’ THEN Iris_versicolor = 1 ELSE Iris_versicolor = 0
			IF Iris=’virginia’ THEN Iris_virginica = 1 ELSE Iris_virginia = 0

X14	Iris_versicolor	Binary	
X15	Iris_virginica	Binary	

Transformed Iris Dataset

X1	X2	X3	X4	X5	X6	X7	X8	X9	X10	X11	X12	X13	X14	X15
5.1	3.5	1.4	0.2	1	0	0	1	1	0	1	0	1	0	0
4.9	3	1.4	0.2	1	0	0	1	1	0	1	0	1	0	0
4.7	3.2	1.3	0.2	1	0	0	1	1	0	1	0	1	0	0
4.6	3.1	1.5	0.2	1	0	0	1	1	0	1	0	1	0	0
5	3.6	1.4	0.2	1	0	0	1	1	0	1	0	1	0	0
5.4	3.9	1.7	0.4	1	0	0	1	1	0	1	0	1	0	0
4.6	3.4	1.4	0.3	1	0	0	1	1	0	1	0	1	0	0
5	3.4	1.5	0.2	1	0	0	1	1	0	1	0	1	0	0
4.4	2.9	1.4	0.2	1	0	1	0	1	0	1	0	1	0	0
4.9	3.1	1.5	0.1	1	0	0	1	1	0	1	0	1	0	0
5.4	3.7	1.5	0.2	1	0	0	1	1	0	1	0	1	0	0
4.8	3.4	1.6	0.2	1	0	0	1	1	0	1	0	1	0	0
4.8	3	1.4	0.1	1	0	0	1	1	0	1	0	1	0	0
4.3	3	1.1	0.1	1	0	0	1	1	0	1	0	1	0	0
5.8	4	1.2	0.2	1	0	0	1	1	0	1	0	1	0	0
5.7	4.4	1.5	0.4	1	0	0	1	1	0	1	0	1	0	0
5.4	3.9	1.3	0.4	1	0	0	1	1	0	1	0	1	0	0
5.1	3.5	1.4	0.3	1	0	0	1	1	0	1	0	1	0	0
5.7	3.8	1.7	0.3	1	0	0	1	1	0	1	0	1	0	0
5.1	3.8	1.5	0.3	1	0	0	1	1	0	1	0	1	0	0
5.4	3.4	1.7	0.2	1	0	0	1	1	0	1	0	1	0	0
5.1	3.7	1.5	0.4	1	0	0	1	1	0	1	0	1	0	0
4.6	3.6	1	0.2	1	0	0	1	1	0	1	0	1	0	0
5.1	3.3	1.7	0.5	1	0	0	1	1	0	1	0	1	0	0
4.8	3.4	1.9	0.2	1	0	0	1	1	0	1	0	1	0	0
5	3	1.6	0.2	1	0	0	1	1	0	1	0	1	0	0
5	3.4	1.6	0.4	1	0	0	1	1	0	1	0	1	0	0
5.2	3.5	1.5	0.2	1	0	0	1	1	0	1	0	1	0	0
5.2	3.4	1.4	0.2	1	0	0	1	1	0	1	0	1	0	0
4.7	3.2	1.6	0.2	1	0	0	1	1	0	1	0	1	0	0
4.8	3.1	1.6	0.2	1	0	0	1	1	0	1	0	1	0	0
5.4	3.4	1.5	0.4	1	0	0	1	1	0	1	0	1	0	0

5.2	4.1	1.5	0.1	1	0	0	1	1	0	1	0	1	0	0
5.5	4.2	1.4	0.2	1	0	0	1	1	0	1	0	1	0	0
4.9	3.1	1.5	0.1	1	0	0	1	1	0	1	0	1	0	0
5	3.2	1.2	0.2	1	0	0	1	1	0	1	0	1	0	0
5.5	3.5	1.3	0.2	1	0	0	1	1	0	1	0	1	0	0
4.9	3.1	1.5	0.1	1	0	0	1	1	0	1	0	1	0	0
4.4	3	1.3	0.2	1	0	0	1	1	0	1	0	1	0	0
5.1	3.4	1.5	0.2	1	0	0	1	1	0	1	0	1	0	0
5	3.5	1.3	0.3	1	0	0	1	1	0	1	0	1	0	0
4.5	2.3	1.3	0.3	1	0	1	0	1	0	1	0	1	0	0
4.4	3.2	1.3	0.2	1	0	0	1	1	0	1	0	1	0	0
5	3.5	1.6	0.6	1	0	0	1	1	0	1	0	1	0	0
5.1	3.8	1.9	0.4	1	0	0	1	1	0	1	0	1	0	0
4.8	3	1.4	0.3	1	0	0	1	1	0	1	0	1	0	0
5.1	3.8	1.6	0.2	1	0	0	1	1	0	1	0	1	0	0
4.6	3.2	1.4	0.2	1	0	0	1	1	0	1	0	1	0	0
5.3	3.7	1.5	0.2	1	0	0	1	1	0	1	0	1	0	0
5	3.3	1.4	0.2	1	0	0	1	1	0	1	0	1	0	0
7	3.2	4.7	1.4	0	1	0	1	0	1	0	1	0	1	0
6.4	3.2	4.5	1.5	0	1	0	1	0	1	0	1	0	1	0
6.9	3.1	4.9	1.5	0	1	0	1	0	1	0	1	0	1	0
5.5	2.3	4	1.3	1	0	1	0	0	1	0	1	0	1	0
6.5	2.8	4.6	1.5	0	1	1	0	0	1	0	1	0	1	0
5.7	2.8	4.5	1.3	1	0	1	0	0	1	0	1	0	1	0
6.3	3.3	4.7	1.6	0	1	0	1	0	1	0	1	0	1	0
4.9	2.4	3.3	1	1	0	1	0	1	0	0	1	0	1	0
6.6	2.9	4.6	1.3	0	1	1	0	0	1	0	1	0	1	0
5.2	2.7	3.9	1.4	1	0	1	0	1	0	0	1	0	1	0
5	2	3.5	1	1	0	1	0	1	0	0	1	0	1	0
5.9	3	4.2	1.5	1	0	0	1	0	1	0	1	0	1	0
6	2.2	4	1	0	1	1	0	0	1	0	1	0	1	0
6.1	2.9	4.7	1.4	0	1	1	0	0	1	0	1	0	1	0
5.6	2.9	3.6	1.3	1	0	1	0	1	0	0	1	0	1	0
6.7	3.1	4.4	1.4	0	1	0	1	0	1	0	1	0	1	0
5.6	3	4.5	1.5	1	0	0	1	0	1	0	1	0	1	0

5.8	2.7	4.1	1	1	0	1	0	0	1	0	1	0	1	0
6.2	2.2	4.5	1.5	0	1	1	0	0	1	0	1	0	1	0
5.6	2.5	3.9	1.1	1	0	1	0	1	0	0	1	0	1	0
5.9	3.2	4.8	1.8	1	0	0	1	0	1	0	1	0	1	0
6.1	2.8	4	1.3	0	1	1	0	0	1	0	1	0	1	0
6.3	2.5	4.9	1.5	0	1	1	0	0	1	0	1	0	1	0
6.1	2.8	4.7	1.2	0	1	1	0	0	1	0	1	0	1	0
6.4	2.9	4.3	1.3	0	1	1	0	0	1	0	1	0	1	0
6.6	3	4.4	1.4	0	1	0	1	0	1	0	1	0	1	0
6.8	2.8	4.8	1.4	0	1	1	0	0	1	0	1	0	1	0
6.7	3	5	1.7	0	1	0	1	0	1	0	1	0	1	0
6	2.9	4.5	1.5	0	1	1	0	0	1	0	1	0	1	0
5.7	2.6	3.5	1	1	0	1	0	1	0	0	1	0	1	0
5.5	2.4	3.8	1.1	1	0	1	0	1	0	0	1	0	1	0
5.5	2.4	3.7	1	1	0	1	0	1	0	0	1	0	1	0
5.8	2.7	3.9	1.2	1	0	1	0	1	0	0	1	0	1	0
6	2.7	5.1	1.6	0	1	1	0	0	1	0	1	0	1	0
5.4	3	4.5	1.5	1	0	0	1	0	1	0	1	0	1	0
6	3.4	4.5	1.6	0	1	0	1	0	1	0	1	0	1	0
6.7	3.1	4.7	1.5	0	1	0	1	0	1	0	1	0	1	0
6.3	2.3	4.4	1.3	0	1	1	0	0	1	0	1	0	1	0
5.6	3	4.1	1.3	1	0	0	1	0	1	0	1	0	1	0
5.5	2.5	4	1.3	1	0	1	0	0	1	0	1	0	1	0
5.5	2.6	4.4	1.2	1	0	1	0	0	1	0	1	0	1	0
6.1	3	4.6	1.4	0	1	0	1	0	1	0	1	0	1	0
5.8	2.6	4	1.2	1	0	1	0	0	1	0	1	0	1	0
5	2.3	3.3	1	1	0	1	0	1	0	0	1	0	1	0
5.6	2.7	4.2	1.3	1	0	1	0	0	1	0	1	0	1	0
5.7	3	4.2	1.2	1	0	0	1	0	1	0	1	0	1	0
5.7	2.9	4.2	1.3	1	0	1	0	0	1	0	1	0	1	0
6.2	2.9	4.3	1.3	0	1	1	0	0	1	0	1	0	1	0
5.1	2.5	3	1.1	1	0	1	0	1	0	0	1	0	1	0
5.7	2.8	4.1	1.3	1	0	1	0	0	1	0	1	0	1	0
6.3	3.3	6	2.5	0	1	0	1	0	1	0	1	0	0	1
5.8	2.7	5.1	1.9	1	0	1	0	0	1	0	1	0	0	1

7.1	3	5.9	2.1	0	1	0	1	0	1	0	1	0	0	1
6.3	2.9	5.6	1.8	0	1	1	0	0	1	0	1	0	0	1
6.5	3	5.8	2.2	0	1	0	1	0	1	0	1	0	0	1
7.6	3	6.6	2.1	0	1	0	1	0	1	0	1	0	0	1
4.9	2.5	4.5	1.7	1	0	1	0	0	1	0	1	0	0	1
7.3	2.9	6.3	1.8	0	1	1	0	0	1	0	1	0	0	1
6.7	2.5	5.8	1.8	0	1	1	0	0	1	0	1	0	0	1
7.2	3.6	6.1	2.5	0	1	0	1	0	1	0	1	0	0	1
6.5	3.2	5.1	2	0	1	0	1	0	1	0	1	0	0	1
6.4	2.7	5.3	1.9	0	1	1	0	0	1	0	1	0	0	1
6.8	3	5.5	2.1	0	1	0	1	0	1	0	1	0	0	1
5.7	2.5	5	2	1	0	1	0	0	1	0	1	0	0	1
5.8	2.8	5.1	2.4	1	0	1	0	0	1	0	1	0	0	1
6.4	3.2	5.3	2.3	0	1	0	1	0	1	0	1	0	0	1
6.5	3	5.5	1.8	0	1	0	1	0	1	0	1	0	0	1
7.7	3.8	6.7	2.2	0	1	0	1	0	1	0	1	0	0	1
7.7	2.6	6.9	2.3	0	1	1	0	0	1	0	1	0	0	1
6	2.2	5	1.5	0	1	1	0	0	1	0	1	0	0	1
6.9	3.2	5.7	2.3	0	1	0	1	0	1	0	1	0	0	1
5.6	2.8	4.9	2	1	0	1	0	0	1	0	1	0	0	1
7.7	2.8	6.7	2	0	1	1	0	0	1	0	1	0	0	1
6.3	2.7	4.9	1.8	0	1	1	0	0	1	0	1	0	0	1
6.7	3.3	5.7	2.1	0	1	0	1	0	1	0	1	0	0	1
7.2	3.2	6	1.8	0	1	0	1	0	1	0	1	0	0	1
6.2	2.8	4.8	1.8	0	1	1	0	0	1	0	1	0	0	1
6.1	3	4.9	1.8	0	1	0	1	0	1	0	1	0	0	1
6.4	2.8	5.6	2.1	0	1	1	0	0	1	0	1	0	0	1
7.2	3	5.8	1.6	0	1	0	1	0	1	0	1	0	0	1
7.4	2.8	6.1	1.9	0	1	1	0	0	1	0	1	0	0	1
7.9	3.8	6.4	2	0	1	0	1	0	1	0	1	0	0	1
6.4	2.8	5.6	2.2	0	1	1	0	0	1	0	1	0	0	1
6.3	2.8	5.1	1.5	0	1	1	0	0	1	0	1	0	0	1
6.1	2.6	5.6	1.4	0	1	1	0	0	1	0	1	0	0	1
7.7	3	6.1	2.3	0	1	0	1	0	1	0	1	0	0	1
6.3	3.4	5.6	2.4	0	1	0	1	0	1	0	1	0	0	1

6.4	3.1	5.5	1.8	0	1	0	1	0	1	0	1	0	0	1
6	3	4.8	1.8	0	1	0	1	0	1	0	1	0	0	1
6.9	3.1	5.4	2.1	0	1	0	1	0	1	0	1	0	0	1
6.7	3.1	5.6	2.4	0	1	0	1	0	1	0	1	0	0	1
6.9	3.1	5.1	2.3	0	1	0	1	0	1	0	1	0	0	1
5.8	2.7	5.1	1.9	1	0	1	0	0	1	0	1	0	0	1
6.8	3.2	5.9	2.3	0	1	0	1	0	1	0	1	0	0	1
6.7	3.3	5.7	2.5	0	1	0	1	0	1	0	1	0	0	1
6.7	3	5.2	2.3	0	1	0	1	0	1	0	1	0	0	1
6.3	2.5	5	1.9	0	1	1	0	0	1	0	1	0	0	1
6.5	3	5.2	2	0	1	0	1	0	1	0	1	0	0	1
6.2	3.4	5.4	2.3	0	1	0	1	0	1	0	1	0	0	1
5.9	3	5.1	1.8	1	0	0	1	0	1	0	1	0	0	1

Basic Elements Table generated from the transformed Iris dataset

i	j	X_i	X_j	N_{ij}	$\sum x_i$	$\sum x_j$	$\sum x_i x_j$	$\sum (x_i x_j)^2$	$\sum x_i^2$	$\sum x_j^2$	$\sum x_i^3$
1	1	sepal_length	sepal_length	150	876.5	876.5	5223.9	196591.7	5223.9	5223.9	31745.0
1	2	sepal_length	sepal_width	150	876.5	458.1	2671.0	49251.8	5223.9	1427.1	31745.0
1	3	sepal_length	petal_length	150	876.5	563.8	3484.3	106644.7	5223.9	2583.0	31745.0
1	4	sepal_length	petal_width	150	876.5	179.8	1127.7	12633.4	5223.9	302.3	31745.0
1	5	sepal_length	sepal_lengt_b1	150	876.5	83.0	433.6	2279.9	5223.9	83.0	31745.0
1	6	sepal_length	sepal_lengt_b2	150	876.5	67.0	442.9	2943.9	5223.9	67.0	31745.0
1	7	sepal_length	sepal_width_b1	150	876.5	57.0	339.3	2045.4	5223.9	57.0	31745.0
1	8	sepal_length	sepal_width_b2	150	876.5	93.0	537.2	3178.5	5223.9	93.0	31745.0
1	9	sepal_length	petal_length_b1	150	876.5	61.0	309.2	1575.5	5223.9	61.0	31745.0
1	10	sepal_length	petal_length_b2	150	876.5	89.0	567.3	3648.4	5223.9	89.0	31745.0
1	11	sepal_length	petal_width_b1	150	876.5	50.0	250.3	1259.1	5223.9	50.0	31745.0
1	12	sepal_length	petal_width_b2	150	876.5	100.0	626.2	3964.8	5223.9	100.0	31745.0
1	13	sepal_length	Iris_setosa	150	876.5	50.0	250.3	1259.1	5223.9	50.0	31745.0
1	14	sepal_length	Iris_versicolor	150	876.5	50.0	296.8	1774.9	5223.9	50.0	31745.0
1	15	sepal_length	Iris_virginica	150	876.5	50.0	329.4	2189.9	5223.9	50.0	31745.0
2	2	sepal_width	sepal_width	150	458.1	458.1	1427.1	14682.2	1427.1	1427.1	4533.3
2	3	sepal_width	petal_length	150	458.1	563.8	1673.9	22772.2	1427.1	2583.0	4533.3
2	4	sepal_width	petal_width	150	458.1	179.8	531.5	2696.0	1427.1	302.3	4533.3
2	5	sepal_width	sepal_lengt_b1	150	458.1	83.0	259.4	830.8	1427.1	83.0	4533.3
2	6	sepal_width	sepal_lengt_b2	150	458.1	67.0	198.7	596.3	1427.1	67.0	4533.3
2	7	sepal_width	sepal_width_b1	150	458.1	57.0	150.5	400.2	1427.1	57.0	4533.3
2	8	sepal_width	sepal_width_b2	150	458.1	93.0	307.6	1026.8	1427.1	93.0	4533.3
2	9	sepal_width	petal_length_b1	150	458.1	61.0	198.3	660.1	1427.1	61.0	4533.3
2	10	sepal_width	petal_length_b2	150	458.1	89.0	259.8	767.0	1427.1	89.0	4533.3
2	11	sepal_width	petal_width_b1	150	458.1	50.0	170.9	591.3	1427.1	50.0	4533.3
2	12	sepal_width	petal_width_b2	150	458.1	100.0	287.2	835.8	1427.1	100.0	4533.3
2	13	sepal_width	Iris_setosa	150	458.1	50.0	170.9	591.3	1427.1	50.0	4533.3
2	14	sepal_width	Iris_versicolor	150	458.1	50.0	138.5	388.5	1427.1	50.0	4533.3

2	15	sepal_width	Iris_virginica	150	458.1	50.0	148.7	447.3	1427.1	50.0	4533.3
3	3	petal_length	petal_length	150	563.8	563.8	2583.0	68227.4	2583.0	2583.0	12974.0
3	4	petal_length	petal_width	150	563.8	179.8	869.0	8344.4	2583.0	302.3	12974.0
3	5	petal_length	sepal_lengt_b1	150	563.8	83.0	211.2	696.1	2583.0	83.0	12974.0
3	6	petal_length	sepal_lengt_b2	150	563.8	67.0	352.6	1886.9	2583.0	67.0	12974.0
3	7	petal_length	sepal_width_b1	150	563.8	57.0	257.0	1217.5	2583.0	57.0	12974.0
3	8	petal_length	sepal_width_b2	150	563.8	93.0	306.8	1365.5	2583.0	93.0	12974.0
3	9	petal_length	petal_length_b1	150	563.8	61.0	112.6	250.6	2583.0	61.0	12974.0
3	10	petal_length	petal_length_b2	150	563.8	89.0	451.2	2332.4	2583.0	89.0	12974.0
3	11	petal_length	petal_width_b1	150	563.8	50.0	73.2	108.6	2583.0	50.0	12974.0
3	12	petal_length	petal_width_b2	150	563.8	100.0	490.6	2474.4	2583.0	100.0	12974.0
3	13	petal_length	Iris_setosa	150	563.8	50.0	73.2	108.6	2583.0	50.0	12974.0
3	14	petal_length	Iris_versicolor	150	563.8	50.0	213.0	918.2	2583.0	50.0	12974.0
3	15	petal_length	Iris_virginica	150	563.8	50.0	277.6	1556.2	2583.0	50.0	12974.0
4	4	petal_width	petal_width	150	179.8	179.8	302.3	1108.5	302.3	302.3	563.5
4	5	petal_width	sepal_lengt_b1	150	179.8	83.0	58.1	71.5	302.3	83.0	563.5
4	6	petal_width	sepal_lengt_b2	150	179.8	67.0	121.7	230.8	302.3	67.0	563.5
4	7	petal_width	sepal_width_b1	150	179.8	57.0	82.6	130.1	302.3	57.0	563.5
4	8	petal_width	sepal_width_b2	150	179.8	93.0	97.2	172.2	302.3	93.0	563.5
4	9	petal_width	petal_length_b1	150	179.8	61.0	24.4	17.3	302.3	61.0	563.5
4	10	petal_width	petal_length_b2	150	179.8	89.0	155.4	285.0	302.3	89.0	563.5
4	11	petal_width	petal_width_b1	150	179.8	50.0	12.2	3.5	302.3	50.0	563.5
4	12	petal_width	petal_width_b2	150	179.8	100.0	167.6	298.8	302.3	100.0	563.5
4	13	petal_width	Iris_setosa	150	179.8	50.0	12.2	3.5	302.3	50.0	563.5
4	14	petal_width	Iris_versicolor	150	179.8	50.0	66.3	89.8	302.3	50.0	563.5
4	15	petal_width	Iris_virginica	150	179.8	50.0	101.3	208.9	302.3	50.0	563.5
5	5	sepal_lengt_b1	sepal_lengt_b1	150	83.0	83.0	83.0	83.0	83.0	83.0	83.0
5	6	sepal_lengt_b1	sepal_lengt_b2	150	83.0	67.0	0.0	0.0	83.0	67.0	83.0
5	7	sepal_lengt_b1	sepal_width_b1	150	83.0	57.0	28.0	28.0	83.0	57.0	83.0
5	8	sepal_lengt_b1	sepal_width_b2	150	83.0	93.0	55.0	55.0	83.0	93.0	83.0

5	9	sepal_lengt_b1	petal_length_b1	150	83.0	61.0	61.0	61.0	83.0	61.0	83.0
5	10	sepal_lengt_b1	petal_length_b2	150	83.0	89.0	22.0	22.0	83.0	89.0	83.0
5	11	sepal_lengt_b1	petal_width_b1	150	83.0	50.0	50.0	50.0	83.0	50.0	83.0
5	12	sepal_lengt_b1	petal_width_b2	150	83.0	100.0	33.0	33.0	83.0	100.0	83.0
5	13	sepal_lengt_b1	Iris_setosa	150	83.0	50.0	50.0	50.0	83.0	50.0	83.0
5	14	sepal_lengt_b1	Iris_versicolor	150	83.0	50.0	26.0	26.0	83.0	50.0	83.0
5	15	sepal_lengt_b1	Iris_virginica	150	83.0	50.0	7.0	7.0	83.0	50.0	83.0
6	6	sepal_lengt_b2	sepal_lengt_b2	150	67.0	67.0	67.0	67.0	67.0	67.0	67.0
6	7	sepal_lengt_b2	sepal_width_b1	150	67.0	57.0	29.0	29.0	67.0	57.0	67.0
6	8	sepal_lengt_b2	sepal_width_b2	150	67.0	93.0	38.0	38.0	67.0	93.0	67.0
6	9	sepal_lengt_b2	petal_length_b1	150	67.0	61.0	0.0	0.0	67.0	61.0	67.0
6	10	sepal_lengt_b2	petal_length_b2	150	67.0	89.0	67.0	67.0	67.0	89.0	67.0
6	11	sepal_lengt_b2	petal_width_b1	150	67.0	50.0	0.0	0.0	67.0	50.0	67.0
6	12	sepal_lengt_b2	petal_width_b2	150	67.0	100.0	67.0	67.0	67.0	100.0	67.0
6	13	sepal_lengt_b2	Iris_setosa	150	67.0	50.0	0.0	0.0	67.0	50.0	67.0
6	14	sepal_lengt_b2	Iris_versicolor	150	67.0	50.0	24.0	24.0	67.0	50.0	67.0
6	15	sepal_lengt_b2	Iris_virginica	150	67.0	50.0	43.0	43.0	67.0	50.0	67.0
7	7	sepal_width_b1	sepal_width_b1	150	57.0	57.0	57.0	57.0	57.0	57.0	57.0
7	8	sepal_width_b1	sepal_width_b2	150	57.0	93.0	0.0	0.0	57.0	93.0	57.0
7	9	sepal_width_b1	petal_length_b1	150	57.0	61.0	13.0	13.0	57.0	61.0	57.0
7	10	sepal_width_b1	petal_length_b2	150	57.0	89.0	44.0	44.0	57.0	89.0	57.0
7	11	sepal_width_b1	petal_width_b1	150	57.0	50.0	2.0	2.0	57.0	50.0	57.0
7	12	sepal_width_b1	petal_width_b2	150	57.0	100.0	55.0	55.0	57.0	100.0	57.0
7	13	sepal_width_b1	Iris_setosa	150	57.0	50.0	2.0	2.0	57.0	50.0	57.0
7	14	sepal_width_b1	Iris_versicolor	150	57.0	50.0	34.0	34.0	57.0	50.0	57.0
7	15	sepal_width_b1	Iris_virginica	150	57.0	50.0	21.0	21.0	57.0	50.0	57.0
8	8	sepal_width_b2	sepal_width_b2	150	93.0	93.0	93.0	93.0	93.0	93.0	93.0
8	9	sepal_width_b2	petal_length_b1	150	93.0	61.0	48.0	48.0	93.0	61.0	93.0
8	10	sepal_width_b2	petal_length_b2	150	93.0	89.0	45.0	45.0	93.0	89.0	93.0
8	11	sepal_width_b2	petal_width_b1	150	93.0	50.0	48.0	48.0	93.0	50.0	93.0

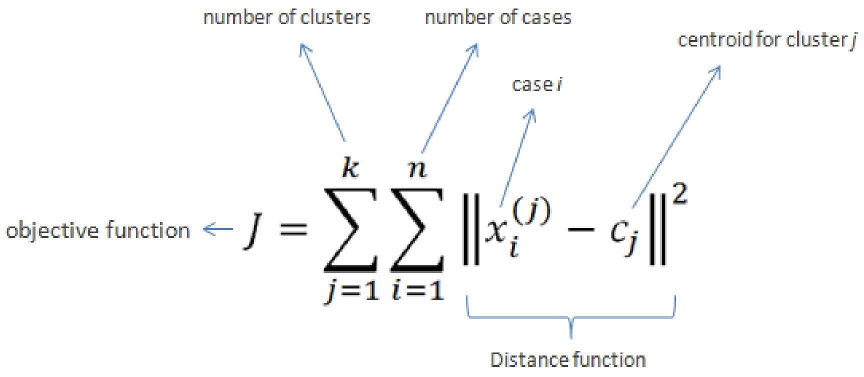
8	12	sepal_width_b2	petal_width_b2	150	93.0	100.0	45.0	45.0	93.0	100.0	93.0
8	13	sepal_width_b2	Iris_setosa	150	93.0	50.0	48.0	48.0	93.0	50.0	93.0
8	14	sepal_width_b2	Iris_versicolor	150	93.0	50.0	16.0	16.0	93.0	50.0	93.0
8	15	sepal_width_b2	Iris_virginica	150	93.0	50.0	29.0	29.0	93.0	50.0	93.0
9	9	petal_length_b1	petal_length_b1	150	61.0	61.0	61.0	61.0	61.0	61.0	61.0
9	10	petal_length_b1	petal_length_b2	150	61.0	89.0	0.0	0.0	61.0	89.0	61.0
9	11	petal_length_b1	petal_width_b1	150	61.0	50.0	50.0	50.0	61.0	50.0	61.0
9	12	petal_length_b1	petal_width_b2	150	61.0	100.0	11.0	11.0	61.0	100.0	61.0
9	13	petal_length_b1	Iris_setosa	150	61.0	50.0	50.0	50.0	61.0	50.0	61.0
9	14	petal_length_b1	Iris_versicolor	150	61.0	50.0	11.0	11.0	61.0	50.0	61.0
9	15	petal_length_b1	Iris_virginica	150	61.0	50.0	0.0	0.0	61.0	50.0	61.0
10	10	petal_length_b2	petal_length_b2	150	89.0	89.0	89.0	89.0	89.0	89.0	89.0
10	11	petal_length_b2	petal_width_b1	150	89.0	50.0	0.0	0.0	89.0	50.0	89.0
10	12	petal_length_b2	petal_width_b2	150	89.0	100.0	89.0	89.0	89.0	100.0	89.0
10	13	petal_length_b2	Iris_setosa	150	89.0	50.0	0.0	0.0	89.0	50.0	89.0
10	14	petal_length_b2	Iris_versicolor	150	89.0	50.0	39.0	39.0	89.0	50.0	89.0
10	15	petal_length_b2	Iris_virginica	150	89.0	50.0	50.0	50.0	89.0	50.0	89.0
11	11	petal_width_b1	petal_width_b1	150	50.0	50.0	50.0	50.0	50.0	50.0	50.0
11	12	petal_width_b1	petal_width_b2	150	50.0	100.0	0.0	0.0	50.0	100.0	50.0
11	13	petal_width_b1	Iris_setosa	150	50.0	50.0	50.0	50.0	50.0	50.0	50.0
11	14	petal_width_b1	Iris_versicolor	150	50.0	50.0	0.0	0.0	50.0	50.0	50.0
11	15	petal_width_b1	Iris_virginica	150	50.0	50.0	0.0	0.0	50.0	50.0	50.0
12	12	petal_width_b2	petal_width_b2	150	100.0	100.0	100.0	100.0	100.0	100.0	100.0
12	13	petal_width_b2	Iris_setosa	150	100.0	50.0	0.0	0.0	100.0	50.0	100.0
12	14	petal_width_b2	Iris_versicolor	150	100.0	50.0	50.0	50.0	100.0	50.0	100.0
12	15	petal_width_b2	Iris_virginica	150	100.0	50.0	50.0	50.0	100.0	50.0	100.0
13	13	Iris_setosa	Iris_setosa	150	50.0	50.0	50.0	50.0	50.0	50.0	50.0
13	14	Iris_setosa	Iris_versicolor	150	50.0	50.0	0.0	0.0	50.0	50.0	50.0
13	15	Iris_setosa	Iris_virginica	150	50.0	50.0	0.0	0.0	50.0	50.0	50.0
14	14	Iris_versicolor	Iris_versicolor	150	50.0	50.0	50.0	50.0	50.0	50.0	50.0

14	15	Iris_versicolor	Iris_virginica	150	50.0	50.0	0.0	0.0	50.0	50.0	50.0
15	15	Iris_virginica	Iris_virginica	150	50.0	50.0	50.0	50.0	50.0	50.0	50.0

APPENDIX B (New Subjects in Version 2):

Real Time Clustering

A cluster is a subset of data which are similar. Clustering (also called unsupervised learning) is the process of dividing a dataset into groups such that the members of each group are as similar (close) as possible to one another, and different groups are as dissimilar (far) as possible from one another. K-Means clustering intends to partition *n* objects into *k* clusters in which each object belongs to the cluster with the nearest mean. This method produces exactly *k* different clusters of greatest possible distinction. The best number of clusters *k* leading to the greatest separation (distance) is not known as a priori and must be computed from the data. The objective of K-Means clustering is to minimize total intra-cluster variance.



$$J = \sum_{j=1}^{k} \sum_{i=1}^{n} \underbrace{\left\| x_i^{(j)} - c_j \right\|^2}_{\text{Distance function}}$$

K-Means Algorithm:

1. Clusters the data into *k* groups where *k* is predefined.
2. Select *k* points at random as cluster centers.
3. Assign objects to their closest cluster center (*c*) per the Euclidean distance function.
4. Calculate the centroid (*c*) or mean of all objects in each cluster.
5. Repeat steps 2, 3 and 4 until the same points are assigned to each cluster in consecutive rounds.

Real Time K-Means Algorithm:

1. Clusters the data into *m* groups where *m* is predefined (e.g., 100).
2. Select *m* points at random as cluster centers.
3. Assign objects to their closest cluster center (*c*) per the Euclidean distance function.
4. Calculate the centroid (*c*) or mean of all objects in each cluster by incrementally updating the following equations.

$$\bar{X} = \frac{\sum X}{N}$$

$$N := N \pm N^{new}$$

$$\sum X := \sum X \pm \sum X^{new}$$

5. Repeat steps 2, 3 and 4 until the same points are assigned to each cluster in consecutive rounds.

Now, instead of keeping all the original training data we only keep *m* (e.g., 100) centroids which can be updated in real time using BET real time equations. For prediction, first we choose a much smaller number of clusters (*k*) and then we use the saved *m* centroids as the source data to build the new set of clusters.

Real Time Variable (Attribute) Selection for Classification

A frequent problem in data mining is to avoid variables that do not contribute significantly to model prediction. First, it has been shown that dropping variables (attributes/predictors) that have insignificant coefficients can reduce the average error of predictions. Second, estimation of model coefficients is likely to be unstable due to multicollinearity in models with many variables. Finally, a simpler model is a better model with more insight into the influence of variables in models. There are two main methods of model selection:

- Forward Selection, the best variables are entered in the model, one by one.
- Backward Elimination, the worst variables are eliminated from the model, one by one.

As explained above (page 79), one way of assessing the effectiveness of the discrimination of the LDA classification algorithm is to calculate the Mahalanobis distance between two groups.

$$\Delta^2 = \boldsymbol{\beta}^T(\boldsymbol{\mu_0} - \boldsymbol{\mu_1})$$

Using the Mahalanobis distance, we can construct a convenient and very fast way of selecting variables only using BET. Because there is no need for any external data, this method of variable selection can be done on the fly.

The real time forward selection technique begins with no variables in the LDA model. For each variable, the forward method calculates Δ^2 statistics that reflect the variable's contribution to the model if it is included. The Δ^2 values are compared to a predefined minimum contribution (e.g., 0.01) and if no Δ^2 statistic has a significance level greater than the minimum contribution, the forward selection stops. Otherwise, the forward method adds the variable that has the largest Δ^2 statistic to the model. The forward method then calculates Δ^2 statistics again for the variables remaining outside the model, and the evaluation process is repeated. Thus, variables are added one by one to the model until no remaining variable produces a significant Δ^2 statistic. Once an variable is in the model, it stays there.

The real time backward elimination method begins by calculating Δ^2 statistics for an LDA model, including all the variables. Then the variables are deleted from the model one by one until all the variables remaining in the model produce Δ^2 statistics significant at the predefined minimum contribution (e.g., 0.01). At each step, the variable showing the smallest contribution to the model is deleted.

Real Time Variable (Attribute) Selection for Regression

R^2 (coefficient of determination) summarizes the explanatory power of the regression model and is computed from the sums-of-squares terms (page 102). You can use the R^2 to finds subsets of variables that best predict a dependent variable (target) by linear regression in the given sample. However, we need a data set to perform this variable selection method using R^2, which means we cannot use this method in real time. However, there is a practical method of real time variable selection for regression, as follows:

1. Select the first variable with the largest linear correlation coefficient using real time linear correlation (page 43).
2. Add the best next variable to the first variable one by one and compute eigenvalues using real time PCA (page 111). The best next variable is the one the eigenvalues of which has the largest length $\sqrt{\boldsymbol{\lambda}^2}$.
3. Repeat Step 2 until there are no more variables *OR* the change in the length of eigenvalues is less than a predefined value (e.g., 0.01).

Real Time Variables Compression

Using BET elements, we can merge two or more variables on the fly.

$$Y = (X_1 + X_2)$$

$$\sum Y = \sum (X_1 + X_2) = \sum X_1 + \sum X_2$$

$$\sum Y^2 = \sum (X_1 + X_2)^2 = \sum {X_1}^2 + \sum {X_2}^2 + 2 \sum X_1 X_2$$

$$\sum Y X_3 = \sum (X_1 + X_2) X_3 = \sum X_1 X_3 + \sum X_2\, X_3$$

This unique feature can play a major role when you work with data with thousands of variables, and merging variables can make the model much smaller without excluding any variables.