**BIRLA INSTITUTE OF TECHNOLOGY & SCIENCE, PILANI**

**BITS C464 – MACHINE LEARNING**

**I Semester 2014-2015**

**WORKSHEET #3**

**Multivariate Regression**

**OBJECTIVE:-**

* Multi Variate Linear Regression Model
* Multi Variate Gaussian model
* Analysis of different basis function on the multi variable data set(using Least Square Error)

In competitive programming contests the contestant’s computer program is evaluated against a set of pre-defined test cases, if the program’s output matches the test cases then it is accepted otherwise it is not. So it’s a binary situation where the program is either accepted or not accepted.

But there may be some minor errors in the program due to which can be rejected, for instance segmentation fault due to allocating small data buffer, a pointer pointing to memory other than the allocated segment. It could also have some logical error’s like divide by zero or square root of -1. But due to time constraint these error’s go undetected and the program is not submitted.

But it may be the case that the algorithm he developed is correct but due to these small errors it fails the test cases.

So a regression model can be build which takes as input a program and based on some features it assigns a score to it. So that problem can be eliminated and a program with correct algorithm can get high score inspite of these minute erroes.

The features can be number of lines, use of library functions, number of iterations and so on.

The regression model can be trained on features of some good quality/bad quality programs and their corresponding scores.

|  |  |  |  |
| --- | --- | --- | --- |
| Score | No. of lines of code | No. of iterations | Library files |
|  |  |  |  |

Here Score is the predicted value and (No of lines, No of Iterations, Library files) are input features.

This problem cannot be solved using the regression model we studied, as that model only had one predictor(one feature). This problem involves multiple predictors(features), so we need a different model to fit this problem .

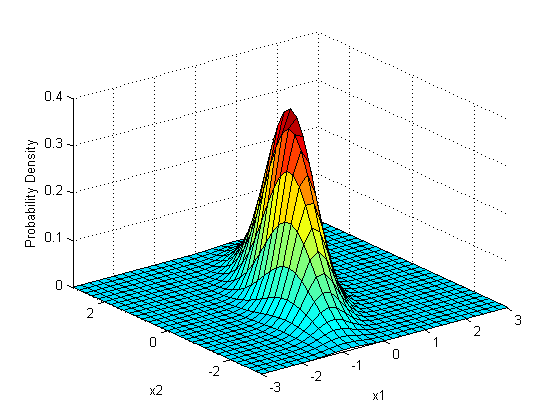
**Multivariable Normal distribution:**

The multivariate normal distribution is a generalization of the univariate normal to two or more variables. It is a distribution for random vectors of correlated variables, each element of which has a univariate normal distribution. In the simplest case, there is no correlation among variables, and elements of the vectors are independent univariate normal random variables.

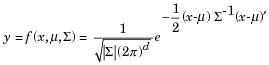
The multivariate normal distribution is parameterized with a mean vector, μ, and a covariance matrix, Σ. These are analogous to the mean μ and variance σ2 parameters of a univariate normal distribution. The diagonal elements of Σ contain the variances for each variable, while the off-diagonal elements of Σ contain the covariances between variables.

The multivariate normal distribution is often used as a model for multivariate data, primarily because it is one of the few multivariate distributions that is tractable to work with.

An example of a 2 dimensional Gaussian distribution is



Mathematically it can be expressed as



where x and μ are 1-by-d vectors and Σ is a d-by-d symmetric positive definite matrix.

To create a multivariable distribution in matlab:

mu = [0 0];

Sigma = [.25 .3; .3 1];

x1 = -3:.2:3; x2 = -3:.2:3;

F = mvnpdf([X1(:) X2(:)],mu,Sigma);

**Covariance Matrix :**

In probability theory and statistics, a covariance matrix (also known as dispersion matrix or variance–covariance matrix) is a matrix whose element in the *i*, *j* position is the covariance between the *i* th and *j* th elements of a random vector (that is, of a vector of random variables). Each element of the vector is a scalar random variable, either with a finite number of observed empirical values or with a finite or infinite number of potential values specified by a theoretical joint probability distribution of all the random variables.


\Sigma_{ij}
= \mathrm{cov}(X_i, X_j) = \mathrm{E}\begin{bmatrix}
(X_i - \mu_i)(X_j - \mu_j)
\end{bmatrix}


In matrix form:

Observe the diagonal elements are variance of the random variables and the off-diagnol elements are covariance’s.


\Sigma
= \begin{bmatrix}
 \mathrm{E}[(X_1 - \mu_1)(X_1 - \mu_1)] & \mathrm{E}[(X_1 - \mu_1)(X_2 - \mu_2)] & \cdots & \mathrm{E}[(X_1 - \mu_1)(X_n - \mu_n)] \\ \\
 \mathrm{E}[(X_2 - \mu_2)(X_1 - \mu_1)] & \mathrm{E}[(X_2 - \mu_2)(X_2 - \mu_2)] & \cdots & \mathrm{E}[(X_2 - \mu_2)(X_n - \mu_n)] \\ \\
 \vdots & \vdots & \ddots & \vdots \\ \\
 \mathrm{E}[(X_n - \mu_n)(X_1 - \mu_1)] & \mathrm{E}[(X_n - \mu_n)(X_2 - \mu_2)] & \cdots & \mathrm{E}[(X_n - \mu_n)(X_n - \mu_n)]
\end{bmatrix}.


**Multi Variate Linear Regression Models:**

* Multivariate Regression analysis is used to predict the value of one or more responses from a set of predictors.
* It can also be used to estimate the linear association between the predictors and responses.
* Predictors can be continuous or categorical or a mixture of both.

**Multiple Regression Analysis:**

* Let z1, z2, ..., zr be a set of r predictors believed to be related to a response variable Y .
* The linear regression model for the jth sample unit has the form

**Yj = β0 + β1zj1 + β2zj2 + ... + βrzjr + ej,**

* where e is a random error and the βi, i = 0, 1, ..., r are unknown and fixed regression coefficients
* Now suppose this problem is extended to case where a single response is also a vector .

y_{i,1} = \mathbf{x}_i^{\rm T}\boldsymbol\beta_{1} + \epsilon_{i,1}

….

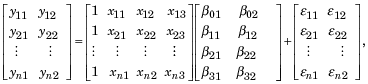
y_{i,m} = \mathbf{x}_i^{\rm T}\boldsymbol\beta_{m} + \epsilon_{i,m}

Here the ith response Yi is a vector of length m.

In vector notation, the vector Yi in row vector form can be written as:

\mathbf{y}_i^{\rm T} = \mathbf{x}_i^{\rm T}\mathbf{B} + \boldsymbol\epsilon_{i}^{\rm T}.

Consider a regression problem where predictor is a 3 variable input and response is of 2 variables.



Where X is the usual design matrix as in linear regression, but here stacked with all the variables.

where http://www.mathworks.in/help/stats/eqn1343662282.png

Where MVN is multivariable Gaussian distribution with zero expectation and sigma as covariance matrix.

The solution to this problem is

 \hat{\mathbf{B}} = (\mathbf{X}^{\rm T}\mathbf{X})^{-1}\mathbf{X}^{\rm T}\mathbf{Y}

Where B’ is the matrix which contains regression coefficients of the multivariable problem.

**Exercise:**

1. Analyze the multivariate Gaussian function by varying its parameters( Covariance, mean, number of univariate gaussians).
2. Load ‘imports-85.mat’ data file in matlab(preloaded in matlab). Extract last two columns (response vectors). And using remaining features as input features fit a multivariate regression model using different basis functions.