VIGNAN'S
Foundation for Science, Technology & Research
(Deemed to be University)
-Estd. u/s 3 of UGC Act 1956

Machine Learning

Unit – III

By
Mrs. P Jhansi Lakshmi
Assistant Professor
Department of CSE, VFSTR

Syllabus

UNIT – III

MULTIVARIATE METHODS: Multivariate data; Parameter estimation; Estimation of missing values multivariate normal distribution; Multivariate classification; Tuning complexity; Discrete features; multivariate regression.

DIMENSIONALITY REDUCTION: Subset selection; Principal components analysis; Feature embedding; Factor analysis; Singular value decomposition and matrix factorization; Multidimensional scaling; Linear discriminant analysis.

Multivariate Methods:
Multivariate data

Multivariate data

- In many applications, several measurements are made on each individual or event generating an observation vector. The sample may be viewed as a data matrix:

$$\mathbf{X} = \begin{bmatrix} X_1^1 & X_2^1 & \cdots & X_d^1 \\ X_1^2 & X_2^2 & \cdots & X_d^2 \\ \vdots & & & \\ X_1^N & X_2^N & \cdots & X_d^N \end{bmatrix}$$

- where the d columns correspond to d variables denoting the result of measurements made on an individual or event. These are also called *inputs*, *features*, *or attributes*.
- The N rows correspond to independent and identically distributed observations, examples, or instances on N individuals or events.

Multivariate data

- For example, in deciding a loan application, an observation vector is the information associated with a customer and is composed of *age*, *marital status*, *yearly income*, *and so forth*, and we have ***N*** such past customers.

- These measurements may be of different scales, for example, age in years and yearly income in monetary units. Some like age may be numeric, and some like marital status may be discrete.

- Typically these variables are correlated. If they are not, there is no need for a multivariate analysis.

- Our aim may be simplification, that is, summarizing this large body of data into relatively few parameters.

- Or our aim may be exploratory, and we may be interested in generating hypotheses about data.

- If the predicted variable is discrete, this is ***multivariate classification***, and if it is numeric, this is a ***multivariate regression*** problem.

Parameter Estimation

- The ***mean vector μ*** is defined such that each of its elements is the mean of one column of **X**:

$$E[x] = \mu = [\mu_1, \mu_2, \cdots \mu_d]^T$$

The variance of X_i is denoted as σ_i^2 , and the covariance of two variables X_i and X_j is defined as

Covariance: $\sigma_{ij} \equiv \text{Cov}(X_i, X_j) = E[(X_i - \mu_i)\ (X_j - \mu_j)] = E[X_i X_j] - \mu_i\ \mu_j$

With $\sigma_{ij} = \sigma_{ji}$,and when i = j, $\sigma_{ii} = \sigma_i^2$

Parameter Estimation

- With *d* variables, there are *d* variances and *d(d −1)/2* covariances, which are generally represented as a *d* X *d* matrix, named the ***Covariance Matrix,*** denoted as Σ, whose(i, j)th element is σ_{ij}:

$$\Sigma = \begin{bmatrix} \sigma_1^2 & \sigma_{12} & \cdots & \sigma_{1d} \\ \sigma_{21} & \sigma_2^2 & \cdots & \sigma_{2d} \\ \vdots & & & \\ \sigma_{d1} & \sigma_{d2} & \cdots & \sigma_d^2 \end{bmatrix}$$

- The diagonal terms are the variances, the off-diagonal terms are the covariances, and the matrix is symmetric.

- In vector-matrix notation $\boldsymbol{\Sigma} \equiv \text{Cov}(\boldsymbol{X}) = E[(\boldsymbol{X} - \boldsymbol{\mu})(\boldsymbol{X} - \boldsymbol{\mu})^T] = E[\boldsymbol{X}\boldsymbol{X}^T] - \boldsymbol{\mu}\boldsymbol{\mu}^T$

Parameter Estimation

- If two variables are related in a linear way, then the covariance will be positive or negative depending on whether the relationship has a positive or negative slope.
- But the size of the relationship is difficult to interpret because it depends on the units in which the two variables are measured.
- The correlation between variables X_i and X_j is a statistic normalized between -1 and $+1$, defined as:

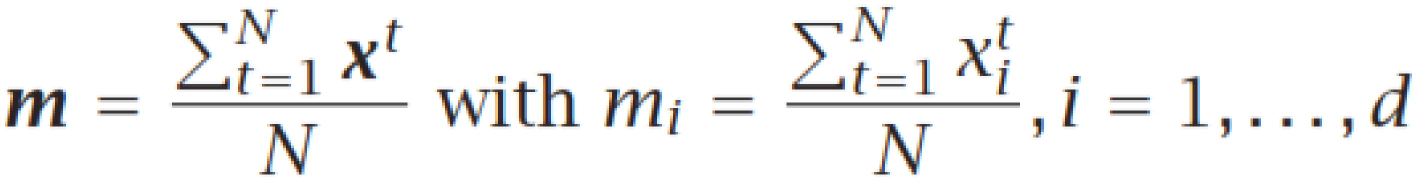
$$\text{Correlation: } \mathrm{Corr}(X_i, X_j) \equiv \rho_{ij} = \frac{\sigma_{ij}}{\sigma_i \sigma_j}$$

Parameter Estimation

- If two variables are independent, then their covariance, and hence their correlation, is 0.
- However, the converse is not true: The variables may be dependent (in a nonlinear way), and their correlation may be 0.
- Given a multivariate sample, estimates for these parameters can be calculated:
- The maximum likelihood estimator for the mean is the ***sample mean, m***.
- Its ith dimension is the average of the ith column of X

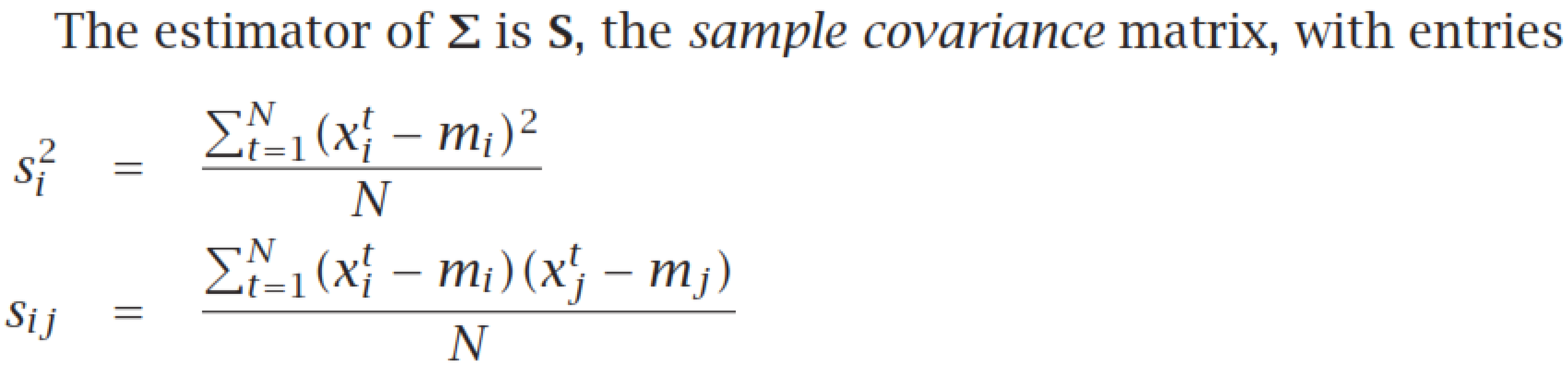
Parameter Estimation

- ***Sample Mean m:***



$$\boldsymbol{m} = \frac{\sum_{t=1}^{N} \boldsymbol{x}^t}{N} \text{ with } m_i = \frac{\sum_{t=1}^{N} x_i^t}{N}, i = 1, \ldots, d$$

- ***Sample Covariance Matrix:***



The estimator of **Σ** is **S**, the *sample covariance* matrix, with entries

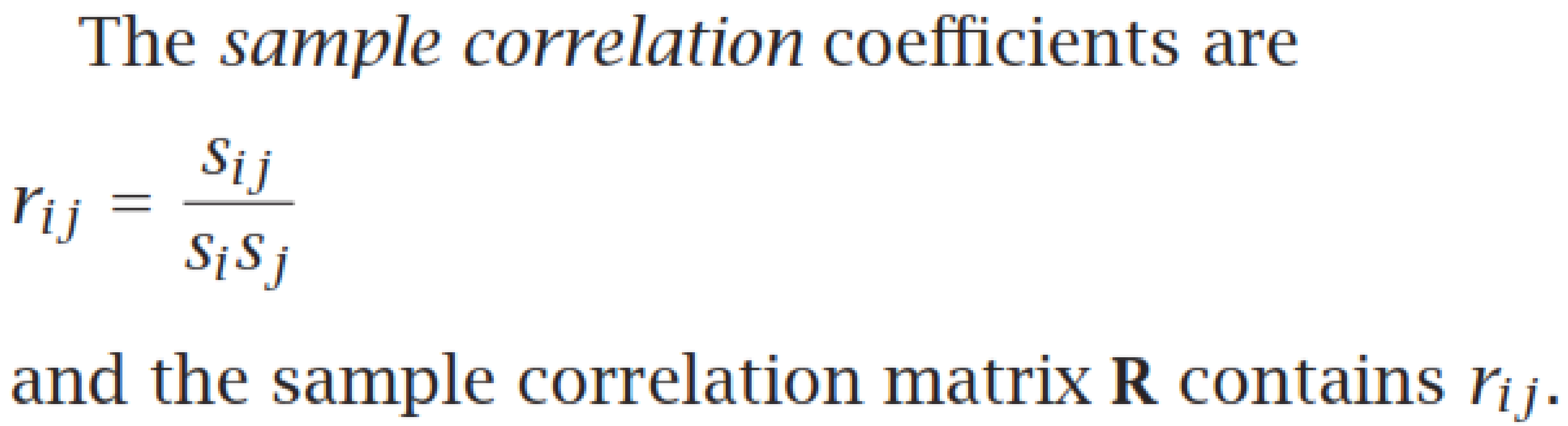
$$s_i^2 = \frac{\sum_{t=1}^{N} (x_i^t - m_i)^2}{N}$$

$$s_{ij} = \frac{\sum_{t=1}^{N} (x_i^t - m_i)(x_j^t - m_j)}{N}$$

Parameter Estimation

- These are biased estimates, but if in an application the estimates vary significantly depending on whether we divide by N or N − 1.

- ***Sample Correlation Coefficients:***



The *sample correlation* coefficients are

$$r_{ij} = \frac{s_{ij}}{s_i s_j}$$

and the sample correlation matrix **R** contains r_{ij}.

Estimation of Missing Values

- What to do if certain instances have missing attributes?
- Ignore those instances: not a good idea if the sample is small
- Use ‘missing’ as an attribute: may give information
- Imputation: Fill in the missing value
  - Mean imputation: Use the most likely value (e.g., mean)
  - Imputation by regression: Predict based on other attributes

Estimation of Missing Values

- Frequently, values of certain variables may be missing in observations.
- The best strategy is to discard those observations all together, but generally we do not have large enough samples to be able to afford this and we do not want to lose data as the non-missing entries do contain information.
- We try to fill in the missing entries by estimating them. This is called I***mputation***.
- In ***mean imputation***, for a numeric variable, we substitute the mean (average) of the available data for that variable in the sample.
- For a discrete variable, we fill in with the most likely value, that is, the value most often seen in the data.

Estimation of Missing Values

- In ***Imputation by Regression***, we try to predict the value of a missing variable from other variables whose values are known for that case.

- Depending on the type of the missing variable, we define a separate regression or classification problem that we train by the data points for which such values are known.

- If many different variables are missing, we take the means as the initial estimates and the procedure is iterated until predicted values stabilize.

- If the variables are not highly correlated, the regression approach is equivalent to ***mean imputation.***

Estimation of Missing Values

- Sometimes the missing attribute value may be important.
- For example, in a credit card application, if the applicant does not declare his or her telephone number, that may be a critical piece of information.
- In such cases, this is represented as a separate value to indicate that the value is missing and is used as such.

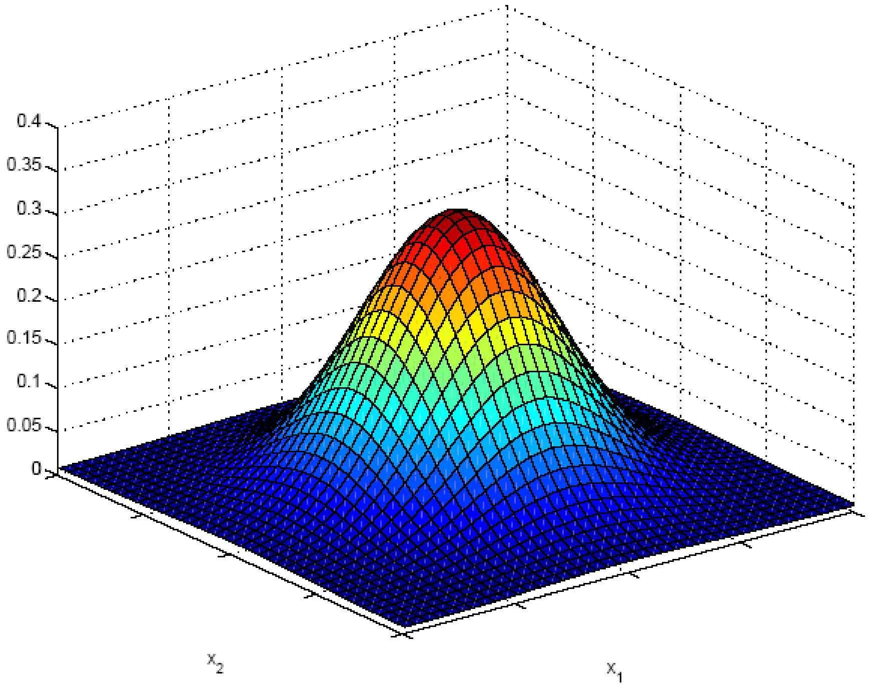
Multivariate Normal Distribution

- In the multivariate case where **x** is d-dimensional and normal distributed, we have:

$$p(\mathbf{x}) = \frac{1}{(2\pi)^{d/2}|\Sigma|^{1/2}} \exp\left[-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^T\Sigma^{-1}(\mathbf{x}-\boldsymbol{\mu})\right]$$

and we write $\mathbf{x} \sim \mathrm{N}_d(\boldsymbol{\mu}, \Sigma)$ where **μ** is the mean vector and **Σ** is the covariance matrix (see figure 5.1). Just as

$$\frac{(x-\mu)^2}{\sigma^2} = (x-\mu)(\sigma^2)^{-1}(x-\mu)$$



- is the squared distance from ***x to μ*** in standard deviation units.

Multivariate Normal Distribution

- Mahalanobis distance: $(\boldsymbol{x} - \boldsymbol{\mu})^T \sum^{-1} (\boldsymbol{x} - \boldsymbol{\mu})$

  measures the distance from $\boldsymbol{x}$ to $\boldsymbol{\mu}$ in terms of $\sum$ (normalizes for difference in variances and correlations)

- $(\boldsymbol{x} - \boldsymbol{\mu})^T \sum^{-1} (\boldsymbol{x} - \boldsymbol{\mu}) = c^2$ is the d-dimensional hyperellipsoid centered at **μ**, and its shape and orientation are defined by Σ.

- The use of the inverse of the covariance matrix has the effect of standardizing all variables to unit variance and eliminating correlations

Multivariate Normal Distribution d = 2

- Let us consider the bivariate case where d = 2 for visualization purposes (see figure 5.2).
- When the variables are independent, the major axes of the density are parallel to the input axes.
- The density becomes an ellipse if the variances are different.
- The density rotates depending on the sign of the covariance (correlation).

The mean vector $\mu^T = [\mu_1, \mu_2]$, and the covariance matrix $\boldsymbol{\Sigma}$ is usually expressed as

$$\Sigma = \begin{bmatrix} \sigma_1^2 & \rho\sigma_1\sigma_2 \\ \rho\sigma_1\sigma_2 & \sigma_2^2 \end{bmatrix}$$

Multivariate Normal Distribution

- The joint bivariate density can be expressed in the form

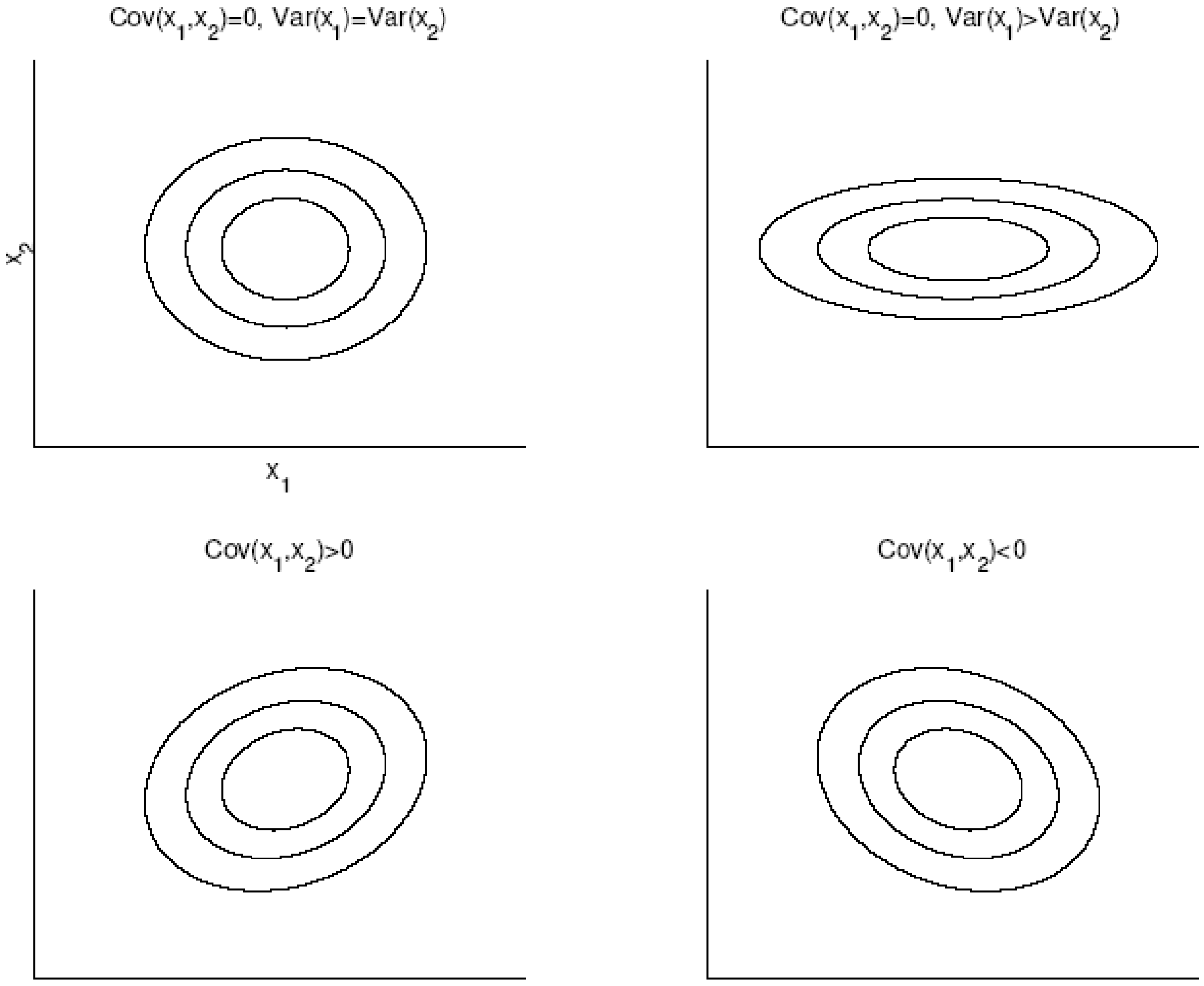
$$p(x_1, x_2) = \frac{1}{2\pi\sigma_1\sigma_2\sqrt{1-\rho^2}} \exp\left[-\frac{1}{2(1-\rho^2)}(z_1^2 - 2\rho z_1 z_2 + z_2^2)\right]$$

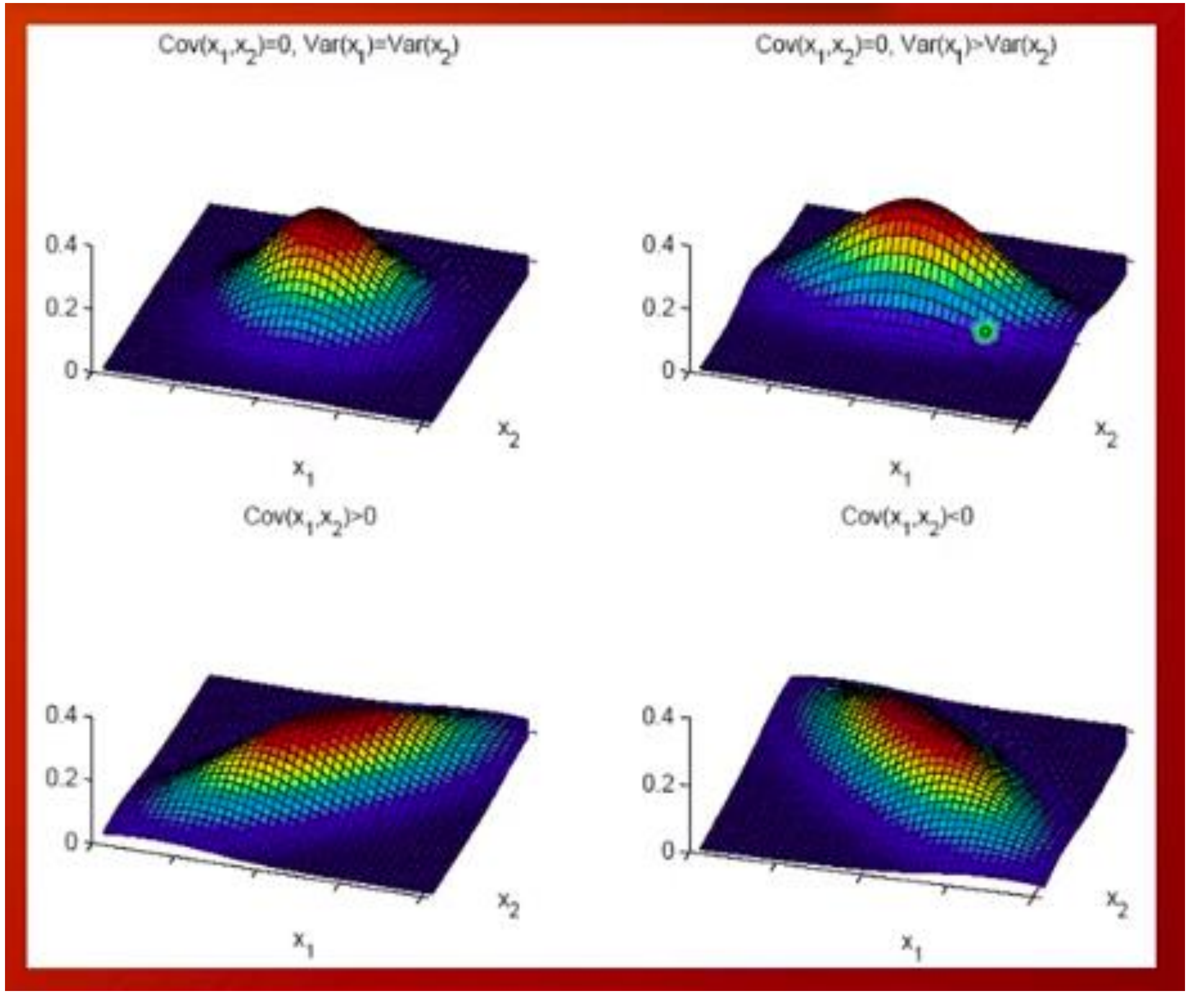
- Where $z_i = (x_i - \mu_i)/\sigma_i$, i = 1, 2, are standardized variables; this is called *z-normalization.*

- Remember that $z_1^2 + 2\rho z_1 z_2 + z_2^2 = Constant$ for |ρ| < 1, is the equation of an ellipse.

When ρ>0, the major axis of the ellipse has a positive slope and if ρ<0, the major axis has a negative slope.

Multivariate Normal Distribution d = 2





Multivariate classification

- When $x \in R^d$, if the class-conditional densities, $p(x|c_i)$ are taken as normal density, $N_d(\mu_i, \Sigma_i)$ i.e .,
- The class C_i is characterized by two parameters the **mean vector** for class C_i and the **covariance matrix** for class C_i
- If $p\,(\boldsymbol{x}\mid C_i) \sim N_d\,(\boldsymbol{\mu}_i\,, \sum_i)$ we have

$$p(\mathbf{x}|C_i) = \frac{1}{(2\pi)^{d/2}|\Sigma_i|^{1/2}}\exp\left[-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_i)^T\Sigma_i{}^{-1}(\mathbf{x}-\boldsymbol{\mu}_i)\right]$$

Multivariate classification

- Reasons for using normal distribution in multivariate classification
  - Simplicity of the equation for analytical developments.
  - To manipulate the log likelihood, and manageable in terms of mathematical developments.
  - And also it’s relatively good model for describing many natural phenomena
    - Observations are generally slight variations (Σ) of a mean observations (μ)
    - Robust model, allows good approximations
  - It requires the data to be grouped together
    - With several groups, we must use a *mixture distribution*, which is a linear combination of several densities.

Multivariate classification

- Let us say we want to predict the type of a car that a customer would be interested in.

- Different cars are the classes and **X** are observable data of customers, for example, age and income.

- μ_i is the vector of mean age and income of customers who buy car type *i* and Σ_i is their covariance matrix: σ_{i1}^2 and σ_{i2}^2 are the age and income variances, and σ_{i12} is the covariance of age and income in the group of customers who buy car type *i*.

Multivariate Parametric classification

- When we define the discriminant function as

$$g_i(\mathbf{x}) = \log p(\mathbf{x}|C_i) + \log P(C_i)$$

and assuming $p\,(\mathbf{x} \mid C_i) \sim N_d\,(\boldsymbol{\mu}_i\,,\sum_i)$

$$g_i(\mathbf{x}) = -\frac{d}{2}\log 2\pi - \frac{1}{2}\log|\Sigma_i| - \frac{1}{2}(\mathbf{x} - \mathrm{\mu}_i)^T \Sigma_i{}^{-1}(\mathbf{x} - \mathrm{\mu}_i) + \log P(C_i)$$

Estimation of parameters

- Given a training sample for K ≥ 2 classes, X= $\{x^t, r^t\}$, where $r_i^t = 1$, if $x^t \in c_i^t$ and 0 otherwise, estimates for the means and covariances are found using maximum likelihood separately for each class:

$$\hat{P}(C_i) = \frac{\sum_t r_i^t}{N}$$

$$\mathbf{m}_i = \frac{\sum_t r_i^t \mathbf{x}^t}{\sum_t r_i^t}$$

$$\mathrm{S}_i = \frac{\sum_t r_i^t (\mathbf{x}^t - \mathbf{m}_i)(\mathbf{x}^t - \mathbf{m}_i)^T}{\sum_t r_i^t}$$

Different S_i

- These are then plugged into the discriminant function to get the estimates for the discriminants. Ignoring the first constant term, we have

$$g_i(\mathbf{x}) = -\frac{1}{2}\log|\mathrm{S}_i| - \frac{1}{2}(\mathbf{x}-\mathbf{m}_i)^T{\mathrm{S}_i}^{-1}(\mathbf{x}-\mathbf{m}_i) + \log\hat{P}(C_i)$$

Expanding this, we get

$$g_i(\mathbf{x}) = -\frac{1}{2}\log|\mathrm{S}_i| - \frac{1}{2}\left(\mathbf{x}^T{\mathrm{S}_i}^{-1}\mathbf{x} - 2\mathbf{x}^T{\mathrm{S}_i}^{-1}\mathbf{m}_i + {\mathbf{m}_i}^T{\mathrm{S}_i}^{-1}\mathbf{m}_i\right) + \log\hat{P}(C_i)$$

which defines a *quadratic discriminant* that can also be

$$g_i(\mathbf{x}) \ = \mathbf{x}^T\mathrm{W}_i\mathbf{x} + {\mathbf{w}_i}^T\mathbf{x} + w_{i0}$$

Different S_i

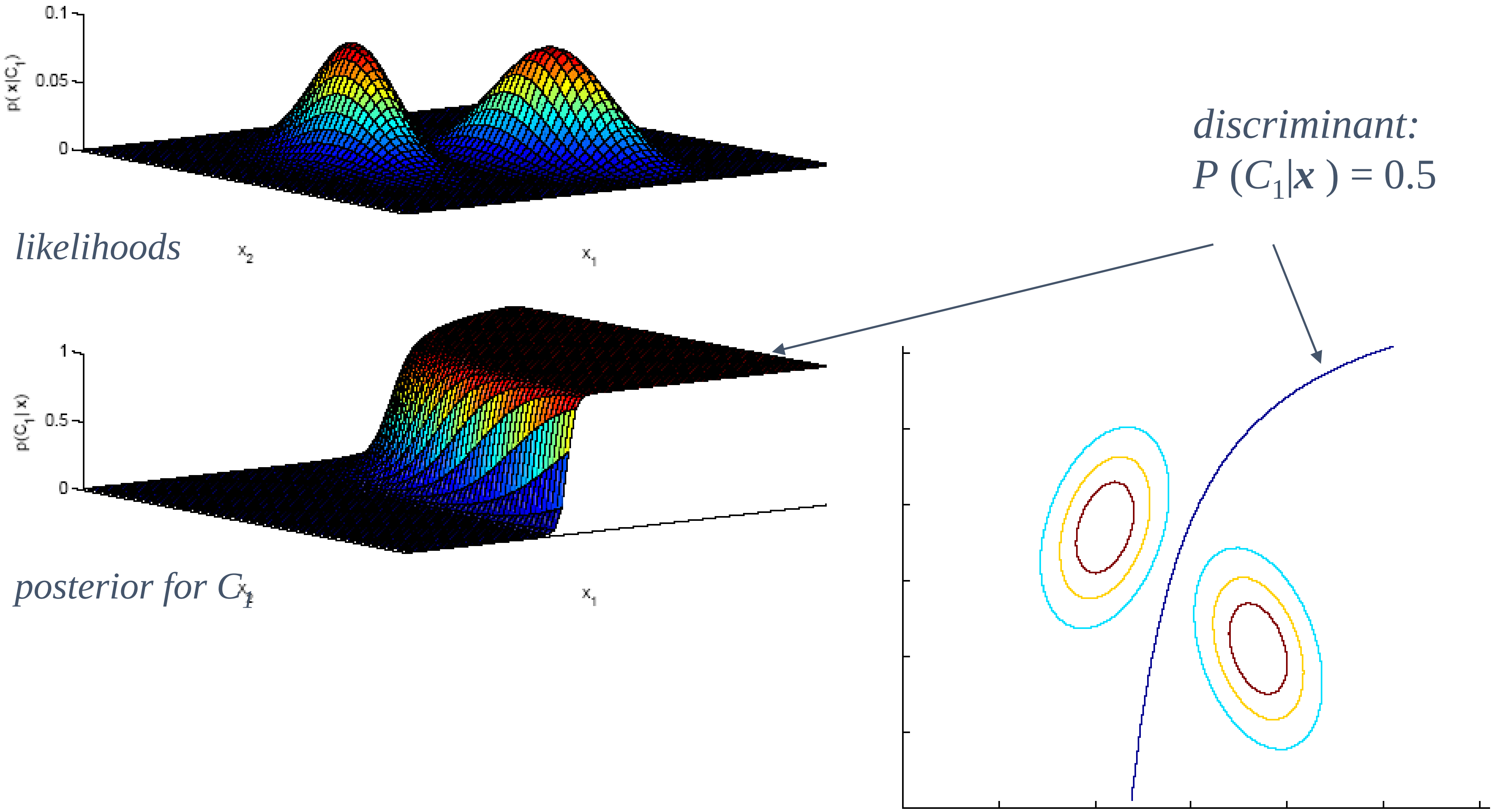
Where

$$\mathrm{W}_i = -\frac{1}{2}\mathrm{S}_i{}^{-1}$$

$$\mathbf{w}_i = \mathrm{S}_i{}^{-1}\mathbf{m}_i$$

$$w_{i0} = -\frac{1}{2}\mathbf{m}_i{}^T\mathrm{S}_i{}^{-1}\mathbf{m}_i - \frac{1}{2}\log|\mathrm{S}_i| + \log\hat{P}(C_i)$$

Different S_i



likelihoods

posterior for C_1

discriminant:
$P\ (C_1|\mathbf{x}\ ) = 0.5$

Common Covariance Matrix S

- Shared common sample covariance S

$$\mathrm{S} = \sum_i \hat{P}(C_i)\,\mathrm{S}_i$$

- Discriminant reduces to

$$g_i(\mathbf{x}) = -\frac{1}{2}(\mathbf{x}-\mathbf{m}_i)^T\mathrm{S}^{-1}(\mathbf{x}-\mathbf{m}_i) + \log \hat{P}(C_i)$$

which is a linear discriminant

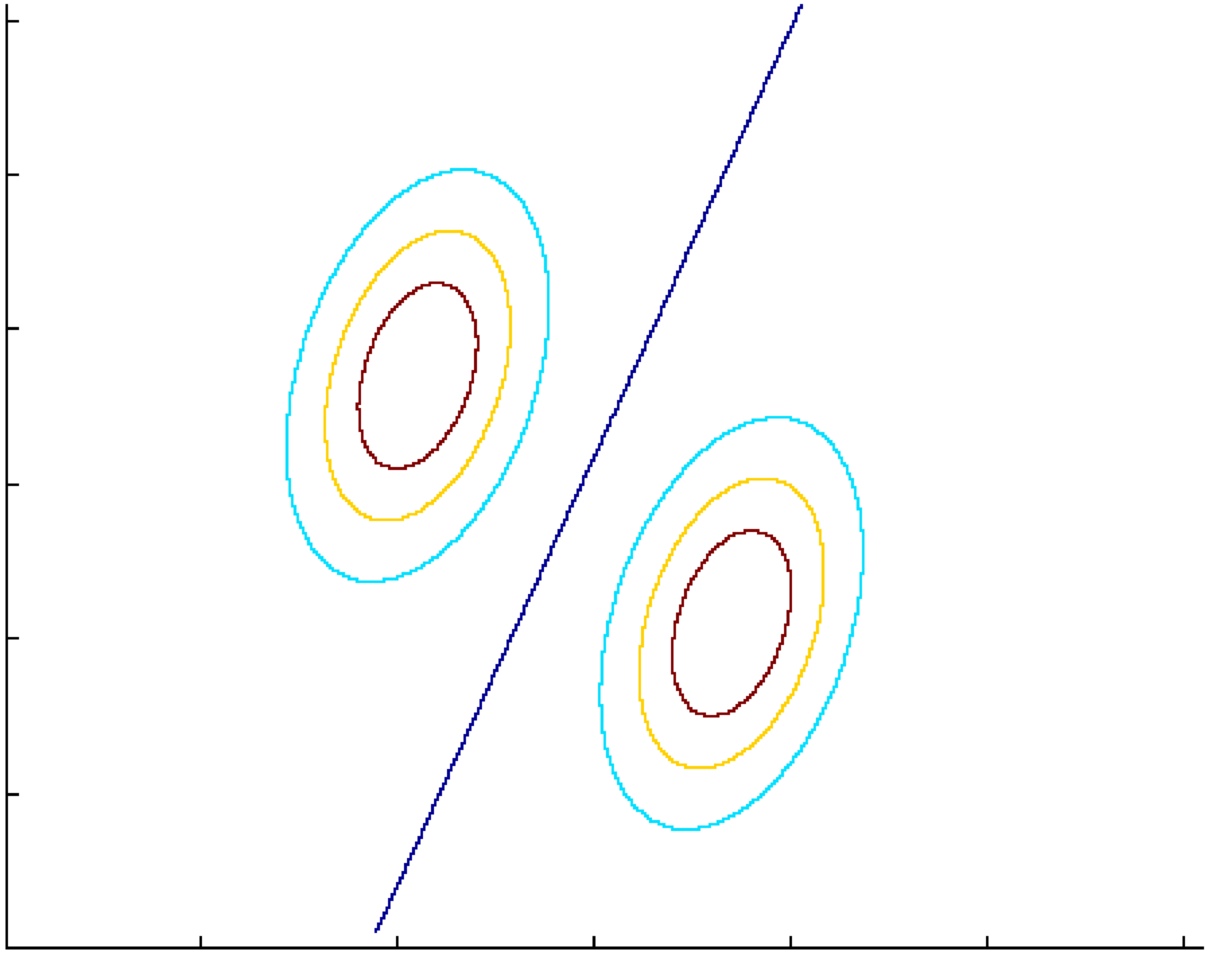
$$g_i(\mathbf{x}) = {\mathbf{w}_i}^T\mathbf{x} + w_{i0}$$

where

$$\mathbf{w}_i = \mathrm{S}^{-1}\mathbf{m}_i \quad w_{i0} = -\frac{1}{2}{\mathbf{m}_i}^T\mathrm{S}^{-1}\mathbf{m}_i + \log \hat{P}(C_i)$$

Although this function is quadratic in *x*, it yields a linear discriminant

Common Covariance Matrix S



Further Simplification: Independence (Diagonal S)

- If we share a common sample covariance S and the variables are independent, then the off-diagonal elements of S are zero.

- When x_j $j = 1,..d$, are independent, $\sum$ is diagonal

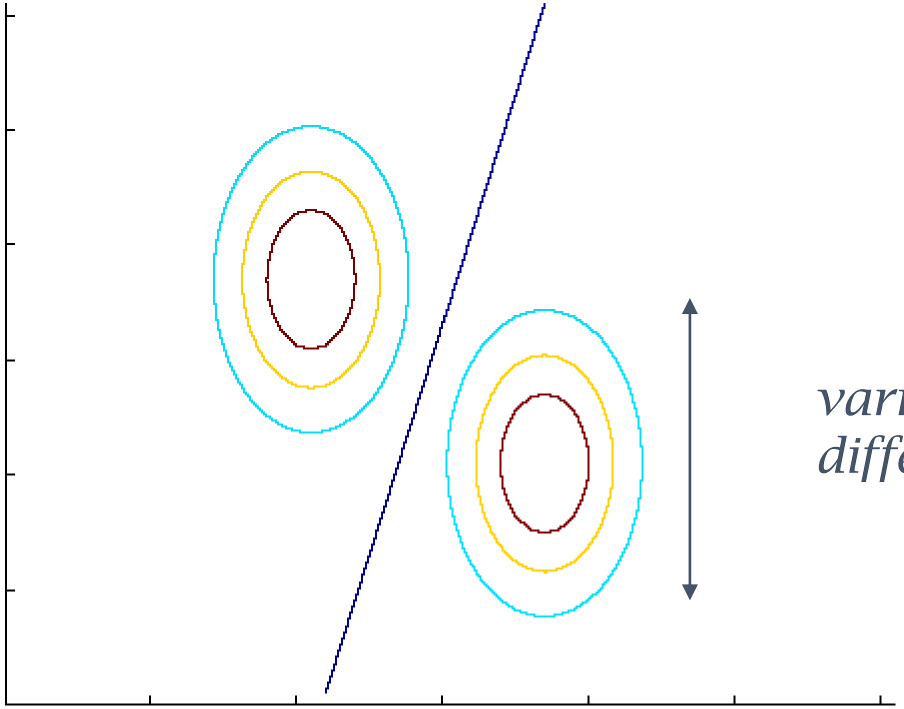
$$p\,(\mathbf{x}|C_i) = \prod_j p\,(x_j\,|C_i) \qquad \text{(Naive Bayes' assumption)}$$

Discriminant simplifies to

$$g_i(\mathbf{x}) = -\frac{1}{2}\sum_{j=1}^{d}\left(\frac{x_j^t - m_{ij}}{s_j}\right)^2 + \log \hat{P}(C_i)$$

- This is the Naive Bayes Classier
  - Each variable is an independent Gaussian
  - Distance measured in standard deviation units

Diagonal S



variances may be different

Diagonal S, equal variances

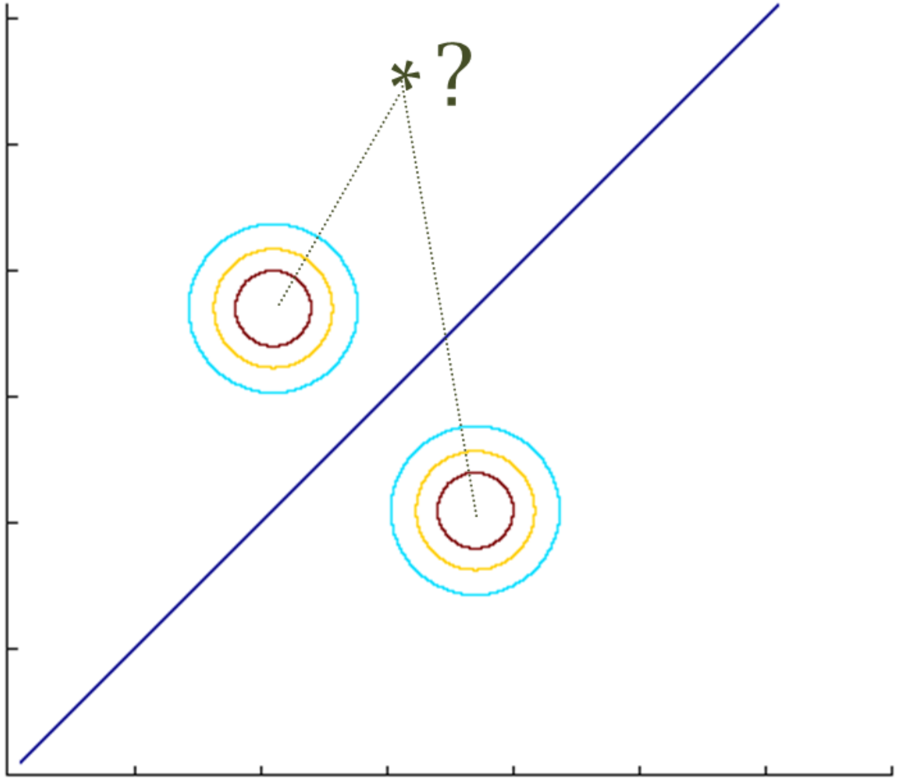
- If variances are also equal,
- Nearest mean classifier: Classify based on Euclidean distance to the nearest mean

$$g_i(\mathbf{x}) = -\frac{\|\mathbf{x}-\mathbf{m}_i\|^2}{2s^2} + \log \hat{P}(C_i)$$

$$= -\frac{1}{2s^2}\sum_{j=1}^{d}\left(x_j^t - m_{ij}\right)^2 + \log \hat{P}(C_i)$$

- Each mean can be considered a prototype or template and this is template matching

Diagonal S, equal variances



Model Selection

Assumption	*Covariance matrix*	*No of parameters*
Shared, Hyperspheric	$\mathbf{S}_i=\mathbf{S}=s^2\mathbf{I}$	1
Shared, Axis-aligned	$\mathbf{S}_i=\mathbf{S}$, with $s_{ij}=0$	d
Shared, Hyperellipsoidal	$\mathbf{S}_i=\mathbf{S}$	$d(d+1)/2$
Different, Hyperellipsoidal	$\mathbf{S}_i$	$K\ d(d+1)/2$

- As we increase complexity (less restricted S), bias decreases and variance increases

- Assume simple models (allow some bias) to control variance (regularization)

Model Selection

- we see how the number of parameters of the covariance matrix may be reduced. This is another example of bias/variance dilemma.
- When we make simplifying assumptions about the covariance matrices and decrease the number of parameters to be estimated

Discrete Features

- In some applications, we have discrete attributes taking one of ***n*** different values.

- For example, an attribute may be *color* ∈{*red*, *blue*, *green*, *black*}, or another may be *pixel* ∈{*on*, *off*}.

- Let us say x_i are binary (Bernoulli) where $p_{ij} \equiv p\left(x_j = 1|C_i\right)$

- If x_j are independent binary variables (Naive Bayes') we have

$$p(x|C_i) = \prod_{j=1}^{d} p_{ij}^{x_j}(1 - p_{ij})^{(1-x_j)}$$

Discrete Features

- This is another example of the naive Bayes' classifier where $p\left(x_{\text{j}}\middle|C_i\right)$ are Bernoulli. The discriminant function is
- $g_i(\mathbf{x}) = \log p(\mathbf{x}|C_i) + \log P(C_i)$

$$= \sum_j \left[x_j \log p_{ij} + \left(1 - x_j\right) \log\left(1 - p_{ij}\right)\right] + \log P(C_i)$$

which is linear. The estimator for p_{ij} is

$$\hat{p}_{ij} = \frac{\sum_t x_j^t r_i^t}{\sum_t r_i^t}$$

Discrete Features

- In the general case, instead of binary features, let us say we have the multinomial $X_j$chosen from the set $\{V_1, V_2, ---, V_{nj}\}$.

- We define new 0/1 dummy variables as

$$z_{jk}^{t} = \begin{cases} 1 & \text{if } x_j^t = v_k \\ 0 & \text{otherwise} \end{cases}$$

-

Discrete Features

Let p_{ijk} denote the probability that x_j belonging to C_i takes value V_k:

$$p_{ijk} \equiv p\left(z_{jk} = 1 | C_i\right) = p\left(x_j = v_k | C_i\right)$$

If the attributes are independent, we have

$$p(\mathbf{x}|C_i) = \prod_{j=1}^{d} \prod_{k=1}^{n_j} p_{ijk}^{z_{jk}}$$

The discriminant function is then

$$g_i(\mathbf{x}) = \sum_j \sum_k z_{jk} \log p_{ijk} + \log P(C_i)$$

The maximum likelihood estimator for p_{ijk} is

$$\hat{p}_{ijk} = \frac{\sum_t z_{jk}^t r_i^t}{\sum_t r_i^t}$$

Multivariate Regression

- In multivariate ***linear regression***, the numeric output *r* is assumed to be written as a linear function, that is, a weighted sum, of several input variables, x_1, x_2, ---, x_d, and noise.

- Actually in statistical literature, this is called multiple regression; statisticians use the term multivariate when there are multiple outputs.

- The multivariate linear model is

$$r^t = g(x^t | w_0, w_1, \ldots, w_d) + \varepsilon = w_0 + w_1 x_1^t + w_2 x_2^t + \cdots + w_d x_d^t + \varepsilon$$

Multivariate Regression

- As in the univariate case, we assume ε to be normal with mean 0 and constant variance, and maximizing the likelihood is equivalent to minimizing the sum of squared errors:

$$E(w_0, w_1, \ldots, w_d|\mathrm{X}) = \frac{1}{2}\sum_t [r^t - w_0 - w_1 x_1^t - \cdots - w_d x_d^t]^2$$

Multivariate Regression

- Taking the derivative with respect to the parameters, w_j, j = 0,...,d, we get these *normal equations:*

$$
\begin{aligned}
\sum_t r^t &= N w_0 + w_1 \sum_t x_1^t + w_2 \sum_t x_2^t + \cdots + w_d \sum_t x_d^t \\
\sum_t x_1^t r^t &= w_0 \sum_t x_1^t + w_1 \sum_t (x_1^t)^2 + w_2 \sum_t x_1^t x_2^t + \cdots + w_d \sum_t x_1^t x_d^t \\
\sum_t x_2^t r^t &= w_0 \sum_t x_2^t + w_1 \sum_t x_1^t x_2^t + w_2 \sum_t (x_2^t)^2 + \cdots + w_d \sum_t x_2^t x_d^t \\
&\vdots \\
\sum_t x_d^t r^t &= w_0 \sum_t x_d^t + w_1 \sum_t x_d^t x_1^t + w_2 \sum_t x_d^t x_2^t + \cdots + w_d \sum_t (x_d^t)^2
\end{aligned}
$$

Multivariate Regression

- Let’s define the following vectors and matrix:

$$\mathbf{X} = \begin{bmatrix} 1 & x_1^1 & x_2^1 & \cdots & x_d^1 \\ 1 & x_1^2 & x_2^2 & \cdots & x_d^2 \\ \vdots & & & & \\ 1 & x_1^N & x_2^N & \cdots & x_d^N \end{bmatrix}, \boldsymbol{w} = \begin{bmatrix} w_0 \\ w_1 \\ \vdots \\ w_d, \end{bmatrix}, \boldsymbol{r} = \begin{bmatrix} r^1 \\ r^2 \\ \vdots \\ r^N \end{bmatrix}$$

- Then the normal equations can be written as:

$$X^T X w = X^T r$$

- And we can solve for the parameters as:

$$w = (X^T X)^{-1} \, X^T r$$

DIMENSIONALITY REDUCTION

Why Reduce Dimensionality?

Dimensionality reduction is the process of reducing the number of variables under consideration by obtaining a smaller set of principal variables

- Reduces time complexity: Less computation
- Reduces space complexity: Fewer parameters
- Saves the cost of observing the feature
- Simpler models are more robust on small datasets
- Simpler explanation
- Easy to plot and analysis

Dimensionality Reduction

- There are two main methods for reducing dimensionality: ***Feature selection and Feature extraction.***
- In ***feature selection***, *we are interested in* finding ***k*** of the ***d*** dimensions that give us the most information, and we discard the other ***(d - k)*** dimensions.
- We discuss ***subset selection*** *as a* feature selection method.
- In ***feature extraction***, we are interested in finding a new set of ***k*** dimensions that are combinations of the original ***d*** dimensions.
- These methods may be supervised or unsupervised depending on whether or not they use the output information.

Feature Selection vs Extraction

- Feature selection: Choosing $k<d$ important features, ignoring the remaining $d - k$

  Subset selection algorithms (Filter method, Wrapper Methods, Embedded Methods)

- Feature extraction: Project the

  original x_i , i =1,...,d dimensions to

  new $k<d$ dimensions, z_j , j =1,...,k

Examples: Principal Component Analysis, Factor Analysis, Singular Value Decomposition.

Subset Selection

- ***Subset Selection*** is also known as variable selection or attribute selection or feature selection
- In ***subset selection***, we are interested in finding the best subset of the set of features.

Advantages:

- Simplification of models
- Shorter training times
- Enhanced generalization (Will remove overfitting problem)

Subset Selection

- The best subset contains the least number of dimensions that most contribute to accuracy.
- We discard the remaining, unimportant dimensions. Using a suitable error function, this can be used in both regression and classification problems.
- There are $\boldsymbol{2^d}$ possible subsets of $\boldsymbol{d}$ variables, but we cannot test for all of them unless $\boldsymbol{d}$ is small and we employ heuristics to get a reasonable (but not optimal) solution in reasonable (polynomial) time.

Subset Selection

There are two approaches:

- In ***forward selection***, we start with no variables and add them one by one, at each step adding the one that decreases the error the most, until any further addition does not decrease the error.

- In ***backward selection***, we start with all variables and remove them one by one, at each step removing the one that decreases the error, until any further removal increases the error significantly.

- In either case, checking the error should be done on a validation set distinct from the training set because we want to test the generalization accuracy.

- With more features, generally we have lower training error, but not necessarily lower validation error.

Subset Selection

- There are 2^d subsets of d features

Forward search: Add the best feature at each step

- Set of features F initially Ø.
- At each iteration, find the best new feature

  $j = \text{argmin}_i\, E\,(\,F \cup x_i\,)$

- Add x_j to F if $E\,(\,F \cup x_j\,) < E\,(\,F\,)$
- We stop if adding any feature does not decrease E

Subset Selection

Backward search: Start with all features and remove one at a time, if possible.

- Floating search (Add *k*, remove *l*): the number of added features and removed features can also change at each step.

- In **sequential backward selection**, we start with F containing all features and do a similar process except that we remove one attribute from F as opposed to adding to it, and we remove the one that causes the least error
  - Take all the features initially F.
  - At each iteration, remove the feature that cause the least error

    $j = \text{argmin}_i \, E\,(F - x_i)$
  - Remove x_j from F if $E\,(F - x_j) < E\,(F)$
  - We stop if removing a feature does not decrease the error

Principal Component Analysis

Principal Component Analysis

- In projection methods, we are interested in finding a mapping from the inputs in the original *d-dimensional* space to a new *(k < d)*-dimensional space, with minimum loss of information.

- The projection of $\boldsymbol{x}$ on the direction of $\boldsymbol{w}$ is $\mathbf{z} = \boldsymbol{w}^{T}\boldsymbol{x}$

- **Principal component analysis (PCA)** is an unsupervised method in that it does not use the output information; the criterion to be maximized is the variance.

- The principal component is w_1 such that the sample, after projection on to w_1.

- For a unique solution and to make the direction the important factor, we require $\|w_1\| = 1$.

Principal Component Analysis

- Find a low-dimensional space such that when $\boldsymbol{x}$ is projected there, information loss is minimized.
- The projection of $\boldsymbol{x}$ on the direction of $\boldsymbol{w}$ is: $z = \boldsymbol{w}^T\boldsymbol{x}$
- Find $\boldsymbol{w}$ such that Var(z) is maximized

$$
\begin{aligned}
\mathrm{Var}(z) = \mathrm{Var}(\boldsymbol{w}^T\boldsymbol{x}) &= \mathrm{E}[(\boldsymbol{w}^T\boldsymbol{x} - \boldsymbol{w}^T\boldsymbol{\mu})^2] \\
&= \mathrm{E}[(\boldsymbol{w}^T\boldsymbol{x} - \boldsymbol{w}^T\boldsymbol{\mu})(\boldsymbol{w}^T\boldsymbol{x} - \boldsymbol{w}^T\boldsymbol{\mu})] \\
&= \mathrm{E}[\boldsymbol{w}^T(\boldsymbol{x} - \boldsymbol{\mu})(\boldsymbol{x} - \boldsymbol{\mu})^T\boldsymbol{w}] \\
&= \boldsymbol{w}^\mathrm{T}\,\mathrm{E}[(\boldsymbol{x} - \boldsymbol{\mu})(\boldsymbol{x} - \boldsymbol{\mu})^T]\boldsymbol{w} = \boldsymbol{w}^T \sum \boldsymbol{w}
\end{aligned}
$$

where $\mathrm{Var}(\boldsymbol{x}) = \mathrm{E}[(\boldsymbol{x} - \boldsymbol{\mu})(\boldsymbol{x} - \boldsymbol{\mu})^T] = \sum$

Principal Component Analysis

- As we know that from equation **Var($\boldsymbol{w^T}, \boldsymbol{x}$)** $= \boldsymbol{w^T \Sigma w}$**,**

if $\boldsymbol{z_1} = \boldsymbol{w_1^t x}$ with Cov(x) = $\boldsymbol{\Sigma}$, then

$$Var(\boldsymbol{z_1}) = \boldsymbol{w_1^t \Sigma w_1}$$

- We seek $\boldsymbol{w_1}$such that $Var(\boldsymbol{z_1})$ is maximized subject to the constraint that $\boldsymbol{w_1^t w_1} = \boldsymbol{1}$ Writing this as a Lagrange problem, we have

Principal Component Analysis

Maximize Var(z) subject to $||\boldsymbol{w}||=1$

$$\max_{\mathbf{w}_1} \mathbf{w}_1^T \Sigma \mathbf{w}_1 - \alpha(\mathbf{w}_1^T \mathbf{w}_1 - 1)$$

Taking the derivative with respect to $\mathbf{w}_1$ and setting it equal to 0, we have

$2\Sigma\mathbf{w}_1 - 2\alpha\mathbf{w}_1 = 0$ and therefore

$\sum \boldsymbol{w}_1 = \alpha \boldsymbol{w}_1$ that is, $\boldsymbol{w}_1$ is an eigenvector of $\sum$

Choose the one with the largest eigenvalue for Var(z) to be max

Principal Component Analysis

Second principal component: Max Var(z_2), s.t., ||$\boldsymbol{w}_2$||=1 and orthogonal to $\boldsymbol{w}_1$

$$\max_{\mathbf{w}_2} \mathbf{w}_2^T \Sigma \mathbf{w}_2 - \alpha(\mathbf{w}_2^T \mathbf{w}_2 - 1) - \beta(\mathbf{w}_2^T \mathbf{w}_1 - 0)$$

Taking the derivative with respect to $\mathbf{w}_2$ and setting it equal to 0, we have

$$2\Sigma\mathbf{w}_2 - 2\alpha\mathbf{w}_2 - \text{ß}\mathbf{w}_1 = 0$$

Pre-multiply $\mathbf{w}_1^T$ by and we get $2\,\mathbf{w}_1^T\,\Sigma\mathbf{w}_2 - 2\alpha\,\mathbf{w}_1^T\,\mathbf{w}_2 - \text{ß}\mathbf{w}_1^T\mathbf{w}_1 = 0$

Principal Component Analysis

- Note that $\mathbf{w}_1^T\mathbf{w}_2 = 0$. $\mathbf{w}_1^T\,\Sigma\mathbf{w}_2$ is a scalar, equal to its transpose $\mathbf{w}_2^T\,\Sigma\mathbf{w}_1$.
- $\mathbf{w}_1$ is the leading eigenvector of Σ, $\Sigma\mathbf{w}_1 = \lambda_1 W_1$. Therefore

$$\mathbf{w}_1^T\,\Sigma\mathbf{w}_2 = \mathbf{w}_2^T\,\Sigma\mathbf{w}_1 = \lambda_1\mathbf{w}_2^T\mathbf{w}_1 = 0, \text{ Then ß} = 0 \text{ and}$$

$\sum \boldsymbol{w}_2 = \alpha\,\boldsymbol{w}_2$ that is, $\boldsymbol{w}_2$ is another eigenvector of $\sum$ with the second largest eigenvalue $\lambda_2 = \alpha$.

- Similarly, we can show that the other dimensions are given by the eigenvectors with decreasing eigenvalues.

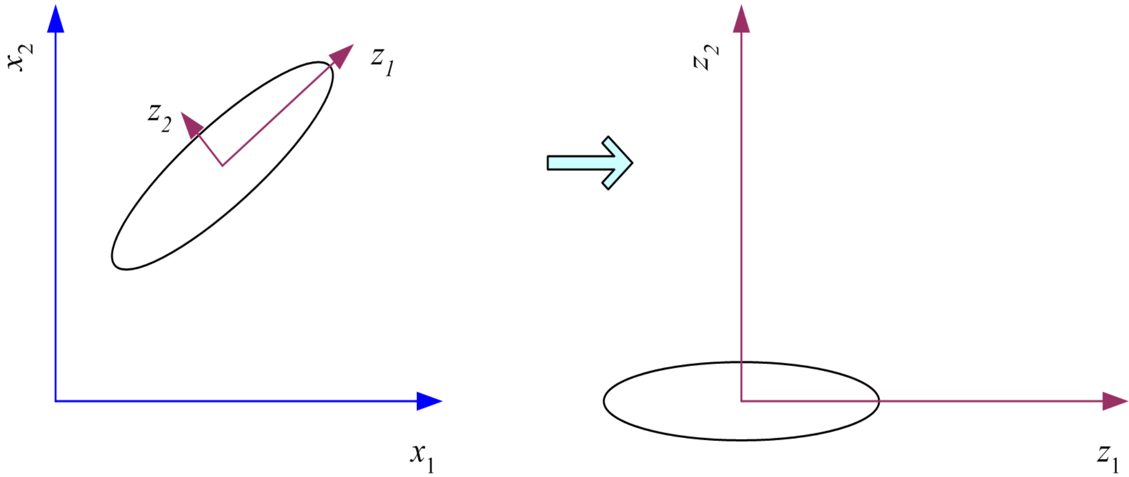
Principal Component Analysis

- The first eigenvector (the one with the largest eigenvalue), w_1, namely, the principal component, explains the largest part of the variance;

- The second explains the second largest; and soon.

We define $Z = w^T(x - m)$

  - where the columns of *W* are the eigenvectors of ∑ and *m* is sample mean
  - Centers the data at the origin and rotates the axes



How to choose k ?

- Proportion of Variance (PoV) explained

$$\frac{\lambda_1 + \lambda_2 + \cdots + \lambda_k}{\lambda_1 + \lambda_2 + \cdots + \lambda_k + \cdots + \lambda_d}$$

when λ_i are sorted in descending order

Linear Discriminant Analysis

Linear Discriminant Analysis