Chapter 4

Support Vector Machine with Positive Definite Kernels

Contents

In this chapter, Support Vector Machine (SVM), a traditional kernel based method and a very good nonlinear classifier, is discussed to diagnose regular(common) skin disorders. SVM is originally designed for binary classification, but we have applied it for multiclass data (Dataset-I) using one-to-one algorithm (refer 2.4.1). Kernel

functions in SVM plays a very important role in classification accuracy. To diagnose skin disorders of Dataset-I, different kernel functions are used and also analyzed their effect by various accuracy measures of classification. Section 4.1 provides general introduction about SVM and its applications in various fields. A mathematical insight about the SVM is discussed in section 4.2. Section 4.3 discusses results to diagnose skin diseases using Support Vector Machine and the chapter ends with the summary.

4.1 Introduction

Support Vector Machine is a nonlinear generalization of the Generalized Portrait algorithm introduced by Vapnik and Lerner in 1963 [127]. They discussed that the pattern recognition problem includes, recognizing the patterns and classify them using generalized portrait algorithm. The history of SVM started in the beginning of 1960s when Vapnik *et.al.* have developed an algorithm to construct an optimal separating hyperplane for separable data [127]. But, they went unnoticed till 1992. In 1992 Boser *et. al.* have presented the algorithm to maximize the margin between training data set and decision boundary [12]. They have constructed the optimal separating hyperplane in Hilbert space using Mercer's theorem, which explicitly map the input vectors into higher dimension Hilbert space using kernel trick. In 1995 Cortes *et.al.* introduced the concept of soft margin which increase the generalisation capacity of SVM [22].

SVM is very popular classifier due to its robust mathematical theory and good classification accuracy for nonlinear data. It can deal with high dimensional data with less computational efforts [11]. SVM is not including knowledge about the geometry of the problem, just like other high performance classifiers, even though the classification accuracy of SVM is excellent. It is kernel based method where algorithm depends on the dot product of the data and uses the kernel induced feature space [18]. When data are non linear, it is not possible to separate data in input space with linear classifier. For such data dot product can be replaced by non linear functions (similarity measures), which pull the data into high dimension where classification with linear classifier is possible. When data is transformed into high dimension space using non linear function, it demands a lot computation power. SVM

overcome this problem using kernel trick in which instead of computing non linear function for training data in feature space, similarity between two training data points are measured by inner product in Hilbert space. Due to this, computational cost is reduced.

In SVM kernel functions are defined in reproducing kernel Hilbert space (RKHS)[118]. Kernels are Mercers kernel i.e., positive semi definite kernel and due to this SVM gives global optimum. When classification is to be done with SVM, proper choice of kernel function and its parameters are very important. Improper choice may reduce the performance of SVM.

High learning ability, good generalization in classification and regression makes SVM most popular learning algorithm. Therefore it applies to many real-life applications such as bioinformatics, in drug design, in classification of drug and nondrug problem [14], to diagnose diabetes and erythematous disease ([72], [10]). It is also useful in electrical load forecasting [52], pattern recognition, image processing, in qualitative and quantitative prediction from sensor data [64] and field of hydrology [106]. Mechanical property such as hot-rolled plain carbon steel can be predicted using SVM [134], to evaluate level of coal mine underground environment [146], for forecasting failures and reliability in engine system [138] SVM can be used. It is also useful to build credit scoring models assessing the risk of default of clients [128], in fault diagnosis [125] etc.

4.2 Mathematical Formulation of SVM

4.2.1 Linearly Separable Data

Let there be m separable training data $(\mathbf{x}_1, y_1)$,$(\mathbf{x}_2, y_2)$,...,$(\mathbf{x}_m, y_m)$. Construct an optimal hyperplane, which separate these training data (binary data) into positive and negative classes with the maximum margin, where the separating hyper plane is given by:

$$\langle \mathbf{w}^*, \mathbf{x} \rangle + b^* = 0 \tag{4.2.1}$$

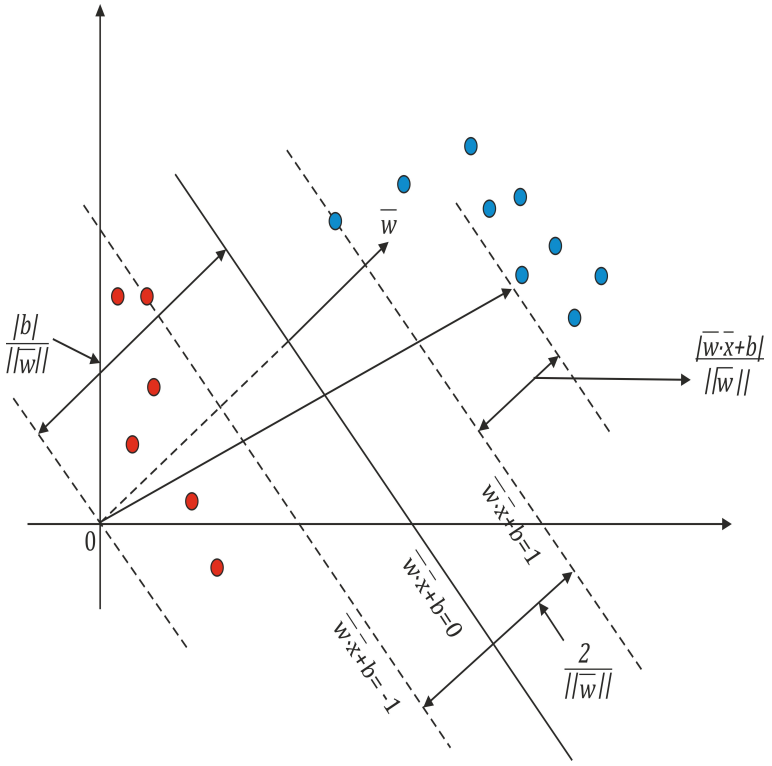

Figure 4.1: SVM-Maximum Margin Classifier

The goal is to find the optimal hyperplane, i.e. to find the values of the normal vector $\mathbf{w} = \mathbf{w}^*$ and the offset parameter $b = b^*$ in such a way that the margin is maximum, given by [129]

$$\text{maximize} \left(\min_{\{i:y_i=1\}} \left[\left\langle \frac{\mathbf{w}}{\|\mathbf{w}\|}, \mathbf{x}_i \right\rangle + b \right] - \max_{\{j:y_j=-1\}} \left[\left\langle \frac{\mathbf{w}}{\|\mathbf{w}\|}, \mathbf{x}_j \right\rangle + b \right] \right) \tag{4.2.2}$$

subject to the constraints

$$y_i \left(\mathbf{w}^T \mathbf{x}_i + b \right) \geq 1, \qquad i = 1, 2, ..., m. \tag{4.2.3}$$

This is equivalent to find minimum of the quadratic function

$$\min_{\{\mathbf{w},b\}} \frac{1}{2} \|\mathbf{w}\|^2 \tag{4.2.4}$$

subject to,

$$1 - y_i \left(\mathbf{w}^T \mathbf{x}_i + b \right) \leq 0, \quad i = 1, 2, ..., m.$$

The Lagrangian of the optimization problem defined as (4.2.4) is given by:

$$L(\mathbf{w}, \alpha, b) = \frac{1}{2}\|\mathbf{w}\|^2 - \sum_{i=1}^{m} \alpha_i \left[y_i \left(\langle \mathbf{w}, \mathbf{x}_i \rangle + b\right) - 1\right] \tag{4.2.5}$$

where, $\alpha_i \geq 0, \quad i = 1, 2, ..., m$ are Lagrange Multipliers. The $L(\mathbf{w}, \alpha, b)$ is to be minimize with respect to $\mathbf{w}$ and b.

Karush-Kuhn-Tucker (KKT) conditions (refer 2.2.1) are the necessary and sufficient conditions for convex optimization problem [107]. Optimization problem defined by (4.2.4) is a convex optimization problem. So, if $w = w^*$, satisfy the KKT conditions, then it becomes the optimum value. Now, applying KKT conditions to Lagrange's function defined by (equation 4.2.5), we have,

$$\frac{\partial L}{\partial \mathbf{w}} = 0 \Rightarrow \mathbf{w} - \sum_{i=1}^{m} \alpha_i \; y_i \; \mathbf{x}_i = 0.$$

and

$$\frac{\partial L}{\partial b} = 0 \Rightarrow \sum_{i=1}^{m} \alpha_i y_i = 0.$$

So, the solution of the quadratic form (refer 4.2.4) is

$$\mathbf{w} = \mathbf{w}^* = \sum_{i=1}^{m} \alpha_i \; y_i \; \mathbf{x}_i \tag{4.2.6}$$

The data points which satisfy $y_i \left(\langle \mathbf{w}, \mathbf{x}_i \rangle + b\right) = 1$ are the support vectors. So, the vector $\mathbf{w}^*$, gives the optimal hyperplane by maximizing the margin is a linear combination of only support vectors, for which $\alpha_i \neq 0, \quad i = 1, 2, ..m$ [12].

Some observations:

1. If $y_i \left(\langle \mathbf{w}, \mathbf{x}_i \rangle + b\right) > 1$, the data point is classified correctly and does not lie on the margin.

2. If $y_i \left(\langle \mathbf{w}, \mathbf{x}_i \rangle + b\right) = 1$, the data point lies exactly on the margin and is called support vector.

3. If $0 < y_i\left(\langle \mathbf{w}, \mathbf{x}_i \rangle + b\right) < 1$, the data point is within the margin, but still on the correct side of the decision boundary and classification is correct.

4. If $\mathbf{y}_i\left(\langle \mathbf{w}, \mathbf{x}_i \rangle + b\right) < 0$, then classification is incorrect.

Subsituting the value of $\mathbf{w}^*$ from (4.2.6) into the Lagrangian function given by (equation 4.2.5), the Lagrangian $L(\mathbf{w}, \alpha, b)$ is a function of α only. The corresponding dual optimization problem is:

$$\max_{\alpha} \quad \mathbf{w}\left(\alpha\right) = \sum_{i=1}^{m} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{m} y_i y_j \alpha_i \alpha_j \langle \mathbf{x}_i, \mathbf{x}_j \rangle \tag{4.2.7}$$

subject to the constraints

$$\sum_{i=1}^{m} \alpha_i y_i = 0, \quad \alpha_i \geq 0, \qquad i = 1, 2, ..., m.$$

The optimal value of the intercept term $b = b^*$ is,

$$b^* = -\frac{\min\limits_{\{i:y_i=1\}} \left[\left\langle \frac{\mathbf{w}^*}{\|\mathbf{w}^*\|}, \mathbf{x}_i \right\rangle\right] - \max\limits_{\{j:y_j=-1\}} \left[\left\langle \frac{\mathbf{w}^*}{\|\mathbf{w}^*\|}, \mathbf{x}_j \right\rangle\right]}{2} \tag{4.2.8}$$

Substituting the value of $\mathbf{w}^*$ from (4.2.6) and the value of the intercept b^* from (4.2.8) into the equation of separating plane given by (equation 4.2.1), the separating hyperplane is given by:

$$\langle \mathbf{w}^*, \mathbf{x} \rangle + b^* = \sum_{i=1}^{m} \alpha_i y_i < \mathbf{x}, \mathbf{x}_i > + b^* = 0. \tag{4.2.9}$$

This optimal hyperplane separates the positive and negative training data using the decision function:

$$f(\mathbf{x}) = sgn\left(\sum_{k \in SV} \alpha_k y_k < \mathbf{x}, \mathbf{x}_k > + b^*\right) \tag{4.2.10}$$

where, only support vectors (SV) $\mathbf{x}_k$ with non zero weights α_k, appears in the decision function.

The function $sgn(x)$ is defined as,

$$sgn(x) = \begin{cases} -1 & x < 0 \\ 0 & x = 0 \\ 1 & x > 0 \end{cases}$$

4.2.2 Nonlinearly Separable Data (Kernel Trick)

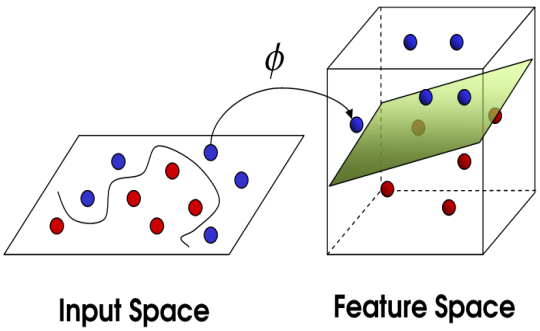

Figure 4.2: Hyperplane in Feature Space

When data is not separable, it can not be classified by linear classifier into input space. But when we pull the data into higher dimension space, they can be separated by linear classifier. Data are transformed from input space to higher dimension feature space using nonlinear function $\phi : \chi \to \mathcal{H}$, where χ is a non-empty set and $\mathcal{H}$ be a Hilbert space. To map data from input space to higher dimension feature space using non linear function it demands a very high computational power. In 1992 Boser *et.al* have given an effective way to construct the optimal separating hyperplane in Hilbert space [12]. They discussed how to map input vector $\mathbf{x}$ without explicitly mapping into the vectors $\mathbf{z}$ of Hilbert space. They used kernel trick, in which inner product of two data points are used to find similarity. So, to separate data points by hyper plane in high dimensional feature space, there is no need to calculate the non linear kernel function ϕ at each data point. Instead the inner product between two data points is to be taken, that reduces the computational cost. Also, the kernel function should be Mercer kernel (definition 2.1.6). i.e. for input data point $\mathbf{x} \in \chi$ (input space), there exists a positive definite function $k(\mathbf{x}, \mathbf{x}')$,

called the kernel function that defines the corresponding inner product in $Z \subset \mathcal{H}$ space, such that, $\langle \mathbf{z}, \mathbf{z}' \rangle = k(\mathbf{x}, \mathbf{x}') = \langle \phi(\mathbf{x}), \phi(\mathbf{x}') \rangle_{\mathcal{H}}$.

The feature space associated with a given kernel is called Reproducing Kernel Hilbert Space (RKHS) (definition 2.3.5).
Using kernel function, the dual problem (4.2.7) becomes,

$$\max_{\alpha} \quad \mathbf{w}(\boldsymbol{\alpha}) = \sum_{i=1}^{m} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{m} y_i y_j \alpha_i \alpha_j k(\mathbf{x}_i, \mathbf{x}_j) \tag{4.2.11}$$

subject to the constraints

$$\sum_{i=1}^{m} \alpha_i y_i = 0, \quad \alpha_i \geq 0, i = 1, 2, ..., m$$

where, $k(x_i, x_j)$ is a Mercer kernels and Mercer's kernels are positive (semi) definite.

Some popular Mercer's kernel are:

1. Linear Kernel $\mathbf{x}^T \mathbf{y} + a_0$
2. Polynomial Kernel $(\alpha \mathbf{x}^T \mathbf{y} + a_0)^d$,
   where d is the degree of the polynomial.
3. Radial Basis Function (RBF) $exp(-\gamma \|\mathbf{x} - \mathbf{y}\|^2)$,
   where the γ parameter is the radius of influence of support vectors.

Equation of the optimal separating hyperplane(equation 4.2.9) becomes,

$$\sum_{k \in SV} \alpha_k y_k k(\mathbf{x}, \mathbf{x}_k) + b^* = 0. \tag{4.2.12}$$

And the decision function (4.2.10) is given by,

$$f(\mathbf{x}) = sgn\left(\sum_{k \in SV} \alpha_k y_k k(\mathbf{x}, \mathbf{x}_k) + b^* \right) \tag{4.2.13}$$

4.2.3 Soft Margin - for Noisy Data

In 1995 Cortes *et.al.* have generalised the idea of maximal margin [22]. When training data is not separable without error, they have introduced the concept of soft margin, in which non separable data is separated with less error. They introduced some non negative numbers $\xi_i, \quad i = 1, 2, ..., m$, and the optimization problem (4.2.4) becomes,

$$\min_{\{\mathbf{w},b\}} \ \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{m}\xi_i \tag{4.2.14}$$

subject to the constraints,

$$\begin{aligned} y_i\left(\langle \mathbf{w}, \mathbf{x}_i\rangle + b\right) &\geq 1 - \xi_i, \quad i = 1, 2, ..., m \\ \xi_i &\geq 0, \quad i = 1, 2, ..., m \end{aligned}$$

where, C is a positive constant called regularization parameter which controls the tradeoff between complexity and classification accuracy. i.e. it controls the relative weighing to make the normal vector $\|\mathbf{w}\|^2$ small and also ensure that most of the data have functional margin at least 1.
The Lagrangian is,

$$L\left(\mathbf{w}, b, \alpha, \mu, \xi\right) = \frac{1}{2}\left(\mathbf{w}, \mathbf{w}\right) + C\sum_{i=1}^{m}\xi_i - \sum_{i=1}^{m}\alpha_i[y_i\left(\mathbf{w}, x_i\right) + b - 1 + \xi_i] - \sum_{i=1}^{m}\mu_i\xi_i \tag{4.2.15}$$

where, $\alpha_i \geq 0$ and $\mu_i \geq 0, \ i = 1, 2, ..., m$ are Lagrange multipliers.
Applying Karush-Kuhn-Tucker conditions (2.2.1) we obtain,

$$\mathbf{w}^* = \sum_{i=1}^{m}\alpha_i y_i \mathbf{x}_i, \quad \sum_{i=1}^{m}\alpha_i y_i = 0$$

and the dual problem (4.2.11) becomes,

$$\max_{\alpha} \quad \mathbf{w}\left(\boldsymbol{\alpha}\right) = \sum_{i=1}^{m}\alpha_i - \frac{1}{2}\sum_{i,j=1}^{m} y_i y_j \alpha_i \alpha_j k\left(\mathbf{x}_i, \mathbf{x}_j\right) \tag{4.2.16}$$

subject to,

$$\sum_{i=1}^{m}\alpha_i y_i = 0 \tag{4.2.17}$$

$$0 \leq \alpha_i \leq C \quad i = 1, 2, ..., m. \tag{4.2.18}$$

The dual optimization problem (4.2.16) which involves Lagrange's multipliers can be solved by Sequential Minimal Algorithm(SMO) invented by John in 1998 [104]. Applying SMO (refer section 4.2.4), Lagrange's multipliers are determined by solving (4.2.16). Using the Lagrange's multipliers, the normal to the hyperplane **w** and the offset parameter b, can be determined using the formula defined by (4.2.6) and (4.2.8).
If kernel function $k(x_i, x_j), i, j = 1, 2, ..., m$ in the optimization problem (4.2.16) is Mercer's Kernel then the problem (4.2.16) is convex optimization problem and therefore global optimum is achieved.

4.2.4 Sequential Minimal Algorithm (SMO)

The algorithm is discussed in [91]. The optimization problem defined in (4.2.16) is a function of Lagrange's multipliers only. There is a one-to-one relation between each Lagrange's multiplier and training data. So, when set of training data is large in size, the matrix formed by the Quadratic Programming (QP) Problem given by (4.2.16) is large in size. This is because the total number of elements in the matrix is the square of the number training examples. Due to this, it can not fit in the memory of an ordinary personal computer or workstation. SMO train Support Vector Machine by decomposing the QP problem (4.2.16) into QP subproblems of size two only, by keeping all other variables as constants. This QP subproblems then can be solved analytically. So, even though more subproblems are to be solved, they can be solved very fast and hence overall QP problem can be solved very quickly.
In SVM, QP problem defined by (4.2.16) is a function of m Lagrange's multipliers. i.e the objective function is of m variables given by,

$$\max_{\alpha} \quad \mathbf{w}(\alpha_1, \alpha_2, ..., \alpha_m). \tag{4.2.19}$$

Suppose, we keep $\alpha_3, \alpha_4, ..., \alpha_m$ fixed and optimize the problem (4.2.16) with respect to α_1 and α_2 only, then from the constraint (4.2.17),

$$\alpha_1 y_1 + \alpha_2 y_2 = -\sum_{i=3}^{m} \alpha_i y_i \tag{4.2.20}$$

Right hand side of equation (4.2.20) is constant, say ϑ i.e.

$$\alpha_1 y_1 + \alpha_2 y_2 = \vartheta \tag{4.2.21}$$

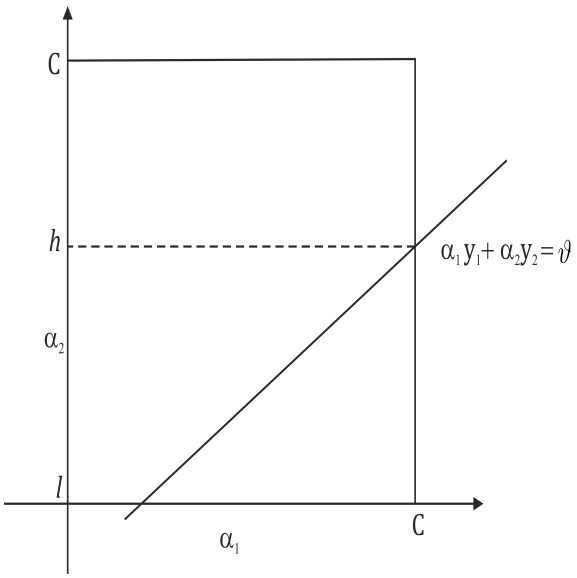

Figure 4.3: Diagram related to Sequential Minimal Algorithm

From the constraints (4.2.18), α_1 and α_2 must lie within the box $[0, C]$ x $[0, C]$ as well as they lie on the line given by (4.2.21). Both of these simultaneously satisfy if $l \leq \alpha_2 \leq h$ [104], where

$$l = max(0, \alpha_2 - \alpha_1), \qquad h = min(C, C + \alpha_2 - \alpha_1).$$

If $y_1 = y_2$ then

$$l = max(0, \alpha_1 + \alpha_2 - C), \qquad h = min(C, \alpha_1 + \alpha_2)$$

Now, from the equation of the line defined by the equation (4.2.21), α_1 can be written as,

$$\alpha_1 = (\vartheta - \alpha_2 \cdot y_2)\, y_1 \tag{4.2.22}$$

Then the optimization problem (4.2.19) becomes:

$$\max_{\alpha_2} \quad \mathbf{w}\left((\vartheta - \alpha_2 \cdot y_2)\, y_1, \alpha_2, ..., \alpha_m\right) \tag{4.2.23}$$

This is a quadratic function of only one variable α_2, which can be solved analytically. This can be expressed as $a_0\alpha_2^2 + a_1\alpha_2 + a_2$ and then be solved for some appropriate values of a_0, a_1 and a_2. Ignoring the box constraints (4.2.18), it is easy to maximize the quadratic function (4.2.23) by taking it's derivative to zero. Let the resulting value of α_2 as α_2^{new}. Since it is necessary to satisfy the box constraints, we determine the resulting value α_2^{opt} from α_2^{new} in such a way that it lie in the interval $[l, h]$, by defining

$$\alpha_2^{opt} = \begin{cases} h, & \text{if } \alpha_2^{new} > h \\ \alpha_2^{new}, & \text{if } l \leq \alpha_2^{new} \leq h \\ l, & \text{if } \alpha_2^{new} < l \end{cases}$$

Substituting the value of α_2^{opt} in the equation (4.2.22), we get the value of α_1^{new} as,

$$\alpha_1^{new} = \alpha_1 + \vartheta 1 \left(\alpha_2 - \alpha_2^{opt}\right) \tag{4.2.24}$$

where, $\vartheta 1 = y_1 y_2$.

Heuristic to Choose Multipliers

The method is discussed by Platt in [104]. SMO optimizes the objective function over two lagrange multipliers simultaneously. It is making heuristic choice for both Lagrange multipliers separately. First it iterates over the entire training set and the multiplier which violets the KKT conditions are eligible for optimization. Once the violated example is found then second heuristic choice is made for second multiplier. The feasibility of the dual optimization problem (4.2.16) is always maintained. For fast training, the choice is made for only those examples which are not bounded. The SVM is then updated using these two multiplier values. Again, heuristic choice for the multiplier searching is started and the process is repeated until the entire training set satisfy the KKT condition within some tolerance. During the process, the Lagrange multipliers which are at bounds are remain at bounds, but the Lagrange's

multipliers which are not at bounds will change. The algorithm will thus iterate over the entire non bounded subset.Then, SMO scan the entire training dataset again to search that any bound examples violets KKT condition due to optimization of non bounded subset.

4.3 SVM to diagnose Skin Disorders

Kernel plays a very important role in classification of nonlinear data when SVM is used as classifier. Improper choice of kernel function may reduce classification accuracy. Various diseases discussed in the Dataset-I of the Appendix-A are classified using different kernel functions and accuracy of the classifier is measured [99].

Following kernels are used in classification:

1. Linear Kernel: $\mathbf{x}^T\mathbf{y} + a_0$
2. Polynomial Kernel: $(\alpha\mathbf{x}^T\mathbf{y} + a_0)^d$,
   where d is the degree of the polynomial.
3. Radial Basis Function (RBF): $exp(-\gamma\|\mathbf{x} - \mathbf{y}\|^2)$,
   where parameter γ is the radius of influence of support vectors. If gamma is too large, the radius of influence of support vectors only includes support vector itself and no amount of regularization parameter C will be able to prevent over fitting. If the value of gamma is very small, the model is too constrained and cannot capture the complexity or shape of the data.
4. t-student kernel: $\frac{1}{\left(1+\|x-y\|^d\right)}$, d is degree.
5. Inverse Multiquadratic(IM): $\frac{1}{\left(a_0^2+\|x-y\|^2\right)}$, a_0 is a parameter of the kernel.

Numerical experiments are performed on MATLAB using LIBSVM 3.20 [76]. Linear Kernel, Polynomial kernel and Radial Basis Kernel are inbuilt kernel functions of LIBSVM. We have also incorporated other two functions viz., t-student kernel function and Inverse Multiquadratic kernel function in LIBSVM software. From Dataset-I 70% of the data are selected randomly, as training data while remaining 30% data are used for testing. Parameters of the kernels are set using grid search

method (refer 2.2.2). Classification accuracy defined by the formula (refer 2.4.3) and other statistical measures of accuracy such as, F-score(which is good measure of accuracy for imbalanced dataset) and G-score defined by the formulas given in (2.4.6), (2.4.7) respectively are measured using the confusion matrix of each kernel.

Table 4.1: Confusion Matrix for various Kernels

	Predicted +ve		Predicted −ve	
Actual +ve	TP	Linear : 122	FN	Linear : 19
		Polynomial :128		Polynomial :13
		RBF :128		RBF :13
		t Student :127		t Student :14
		IM :126		IM :15
Actual −ve	FP	Linear :19	TN	Linear :404
		Polynomial :13		Polynomial :410
		RBF :13		RBF :410
		t Student :14		t Student :409
		IM :15		IM :408

- TP : True Positive FN : False Negative
- FP : False Positive TN : True Negative

Table 4.2: Performance of SVM using various kernels

Kernel Function	Kernel Parameters	Accuracy	F-Score	G-Score
Linear	$a_0 = 1$	93.26%	86.52%	90.91%
Polynomial	α=2, $a_0 = 10$, $d = 3$	95.39%	90.78%	93.80%
Radial Basis Function	γ= 0.1	95.39%	90.78%	93.80%
t-Student	$d = 2$	95.04%	90.07%	93.32%
Inverse Multiquadratic	$a_0 = 10$	94.68%	89.36%	92.84%

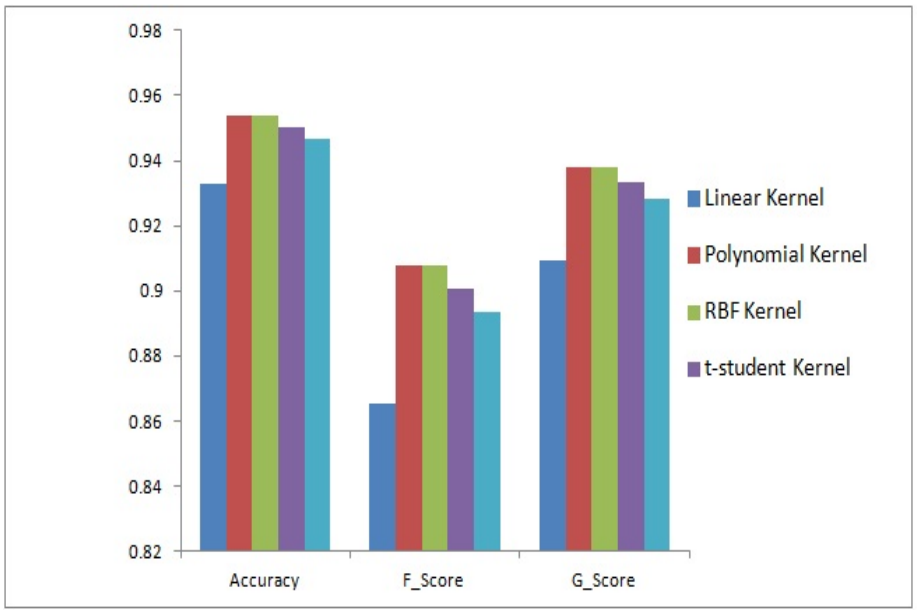

Figure 4.4: Graph of Accuracy, F-Score and G-Score for various kernels.

The results obtained are recorded in Table (4.2) and plotted graphically in figure (4.4) exhibit that the accuracy obtained from the Radial Basis Function (RBF) and Polynomial kernel is 95.39% (same for both kernels) using normal accuracy measure for 70%-30% training-testing data partitions. For same kernels 90.78% and 93.80%

accuracies are achieved using F-score and G-score statistical measures of accuracy. The accuracies achieved from these two kernel are highest than the other kernels for the Dataset-I.

4.4 Summary

In this chapter some common skin disorders viz. Bacterial infections, Fungal infections, Eczema and Scabies are diagnosed by using the powerful kernel based technique namely support vector machine. Since SVM is giving global optimum for positive definite kernel, here training of SVM is performed by using various positive definite kernels. During training phase, randomly 70% data of the Dataset-I discussed in Appendix-A is selected for training and remaining 30% data is used for testing. During training, parameters of the classifier are set using grid search method (refer 2.2.2) and 10 fold cross validation criteria (refer 2.4.8) is used for validation. classification results for various kernels are obtained by using various statistical measures such as Accuracy, F-score and G-score. It is observed skin disorders are diagnosed with highest classification accuracy using Radial Basis kernel and Polynomial kernel.