

DATA MINING TECHNIQUES TO CLASSIFY ASTRONOMY OBJECTS

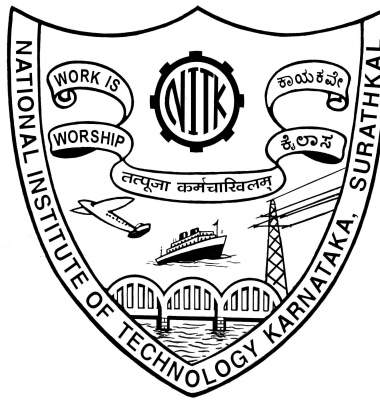
Project Report

Submitted by

V.SUBHASHINI

Under the guidance of

Dr. Ananthanarayana V. S.
Professor and Head
Department of Information Technology



DEPARTMENT OF INFORMATION TECHNOLOGY
NATIONAL INSTITUTE OF TECHNOLOGY KARNATAKA
SURATHKAL – 575 025, INDIA
2008-2009

Acknowledgment

I express my heartfelt gratitude to Dr. Ananthanarayana. V.S, HOD, Department of Information Technology, NITK Surathkal for his invaluable guidance, encouragement and support throughout this project.

V. SUBHASHINI
Student, B.Tech
Dept. of Information Technology
NITK, Surathkal

ABSTRACT

Astronomical images hold a wealth of knowledge about the universe and its origins; the archives of such data have been increasing at a tremendous rate with improvements in the technology of the equipments. Data mining would be an ideal tool to efficiently analyze these enormous datasets, help gather useful patterns and gain knowledge. Data mining tools for clustering, classification and regression are being employed by astronomers to gain information from the vast sky survey data. Some classical problems in this regard are the star-galaxy classification and star-quasar separation that have been worked upon by astronomers and data miners alike. Various methods like artificial neural networks, support vector machines, decision trees and mathematical morphology have been used to characterize these astronomical objects.

We first make a study of clustering and classification algorithms like Linear Vector Quantization, Multi-Layer Perceptrons, kd-trees, Bayesian Belief Networks and others that have been used by astronomers to classify celestial objects as Stars, Quasars, Galaxies or other astronomical objects. We then experiment with some of the existing datamining techniques like Support Vector Machines, Decision Trees and DBSCAN (Density Based Spatial Clustering with Noise) and explore their feasibility in clustering and classifying astronomy data objects. Our objective is to develop an algorithm to increase the classification accuracy of stars, galaxies and quasars and reduce computation time. We also compare how the use of different distance measures and new parameters affect the performance of the algorithm. Thus we also try to identify the most suitable distance measure and significant parameters that increase the accuracy of classification of the chosen objects.

Contents

1. Introduction	6
1.1 Sky survey data	6
2. Literature Survey	10
2.1 Review of existing systems	10
2.2 Problems with existing systems and suggestions	11
3. Methodology	12
3.1 Identify features or parameters that can aid in characterizing the classes .	13
3.2 Study of data-mining tools that aid in classifying astronomy objects . . .	13
3.3 Gather data on celestial objects from various surveys	13
3.4 Apply the algorithms on the chosen parameters, make improvisations. . .	14
3.5 Results and conclusions	14
4. Closer study of relevant data-mining algorithms	15
4.1 DBSCAN and GDBSCAN	15
4.2 Decision Trees	16
4.3 Comparison of the algorithms on the astronomy data set	17
5. Implementation	20
5.1 Data Structure: Tree to classify astronomy data	20
5.2 Distance measure: Use of different distance measures	21
5.3 Use k-means for better accuracy and identify optimum k.	24
5.4 Inclusion of more parameters	25
6. Conclusion	30
6.1 Advantages	30
6.2 Future work	32
7. References	33

List of Figures

1.1 Spectrum of a typical star	8
1.2 Spectrum of a typical galaxy	9
1.3 Spectrum of a typical quasar	9
4.1 DBSCAN clustering on 3 different databases	15
5.1 Diagrammatic representation of proposed structure	21
5.2 Accuracy vs K - compare distance measures	25
5.3 Accuracy vs K – all colors as parameters	26
5.4 Accuracy vs K – psf magnitude included with parameters	27
5.5 Accuracy vs K – combination of distances and parameters	28
6.1 Comparison of Euclidean and Sum of modulus distance measures . . .	31
6.2 Comparison of PSF and model magnitudes	31

Chapter 1

Introduction

In the recent years a number of multi-wavelength sky survey projects have been undertaken and are continuously generating a phenomenal amount of data. Astronomical images are available in large numbers from the various sky surveys Palomar sky survey (DPOSS), Sloan Digital Sky Survey (SDSS), 2micron-all sky survey (2MASS) to name a few. This has resulted in a “data-avalanche” in the field of astronomy and necessitates the need for efficient and effective methods to deal with the data. Classification, as a data mining task, is a key issue in this context where celestial objects are grouped into different kinds like, stars, galaxies and quasars depending on their characteristics. Data mining algorithms (like Decision tables, Bayesian networks, ADTrees, SVMs and k-d trees) have been applied successfully on astronomical data in the last few years. We could explore further on both supervised and unsupervised analysis methods to gain knowledge from this data. This project, not only looks towards increasing the efficiency of handling large datasets but could also help discover new patterns that may assist practicing scientists.

1.1 Sky Survey Data

A number of astronomy organizations all across the world are getting together and have developed various multi-wavelength sky survey projects such as, SDSS, GALEX, 2MASS, GSC-2, POSS2, RASS, FIRST, LAMOST and DENIS. The Faint Images of the Radio Sky at Twenty centimeters, FIRST, is a radio survey; Two Micron All-Sky survey (2MASS) collects data pertaining to the near infrared; RASS is employed to obtain X-ray strong objects; SDSS has data for the study of properties of various objects in five optical bandpasses. The data from these large digital sky surveys and archives run into terabytes. Moreover, with advancements in electronics and telescope equipments the amount of data gathered is rising at an enormous pace. As the data keeps mounting the task of organizing, classifying and searching for relevant information has become more challenging. Hence there is a need for efficient methods to analyze the data and this is where data-mining has become extremely relevant in today's astronomy.

Astronomy terminology and data relevant for classification

Electromagnetic waves are the source for almost all the knowledge that science has about the objects in space. This is primarily due to their ability to travel in vacuum, and the fact that most of space is vacuum. Most celestial objects emit electromagnetic waves, some of these radiations that reach earth are studied by scientists to gather information about the universe. The sky survey telescopes focus on portions of the sky to gather the electromagnetic waves from the region, this is then filtered to get specific details of the electromagnetic spectrum and is stored as electronic/digital data.

1.1.1 Flux and Magnitude

The spectrum of an object corresponds to the intensity of the light, for each wavelength, emitted by the object. A study of the spectrum gives us information about the object. **Flux** is a measure of the intensity of light, it is the amount of light that falls on a specified area of the earth in a

specified time. In astronomy the term **magnitude** is used with reference to the intensity of light from different objects, it is computed from the flux. Magnitude is a number that measures the brightness of a star or galaxy. In magnitude, higher numbers correspond to fainter objects, lower numbers to brighter objects; the very brightest objects have negative magnitudes. Also, the terms flux and magnitude are used with respect to each light wave, of a particular wavelength, belonging to the electromagnetic spectrum.

The magnitude is derived from the flux as:

$$m = -\log_{2.51} \left(F / F_0 \right),$$

m refers to the magnitude, F refers to the flux of the object under consideration and F_0 to the flux of the base object and all the three values - m, F and F_0 are with respect to a particular wavelength of light. As a standard, the base object is the star Vega in the constellation Lyra and its magnitude is considered as Zero for all wavelengths.

There are also other scales for computing the magnitude, such as the asinh magnitude(used by SDSS data), model magnitude, PSF magnitude (point-spread-function), Petrosian magnitude, Fiber magnitude, cmodel magnitude, reddening correction and so on. The type of magnitude used by the survey depends on whether we want to maximize some combination of signal-to-noise ratio, fraction of the total flux included, and freedom from systematic variations with observing conditions and distance.

1.1.2 Filter

It is expensive and quite unnecessary to gather and store information about every wavelength of the spectrum for each object. Hence, astronomers prefer to select a few interesting parts of the spectrum for their observation. This is done by the use of filters. A filter is a device that blocks all unwanted light waves and lets only the specified wavelength pass through for observation. The magnitude of a filter refers to the magnitude of the light corresponding to the wavelength allowed by the filter and it is this which is stored in the sky survey databases. e.g The SDSS uses five filters – Ultraviolet (u) 3543 Å(angstroms), Green (g) 4770 Å, Red (r) 6231 Å, Near Infrared (i) 7625 Å, and the Infrared (z) 9134 Å.

1.1.3 Color

In the real world color is a subjective judgment, i.e what one person calls "blue" may be a different shade than another person's "blue." But, colors are an important feature of objects. Its importance holds true in astronomy as well. However, physicists and astronomers need a definition for color that everyone can agree on. In astronomy, color is the difference in magnitude between two filters. The sky surveys choose the filters to view a wide range of colors, while focusing on the colors of interesting celestial objects. Color is symbolized by subtracting the magnitudes. Since all these quantities involve magnitude, they *decrease* with increasing light output. i.e If g and r are two magnitudes corresponding to green and red lights of specific wavelengths, g-r is a color, and a large value for g-r means that the object is more red than it is green. The reason for this can be seen from the fact that magnitudes are computed as logarithms.

1.1.4 Spectrum

Figure1.2: Spectrum of a typical galaxy

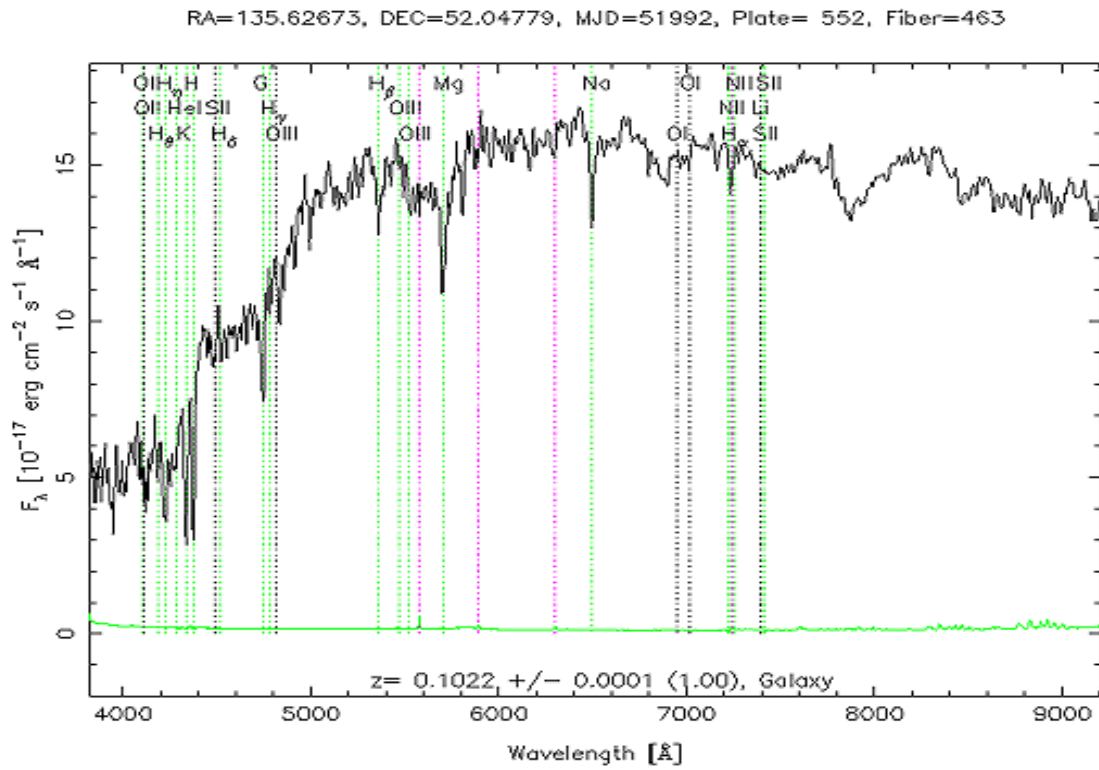
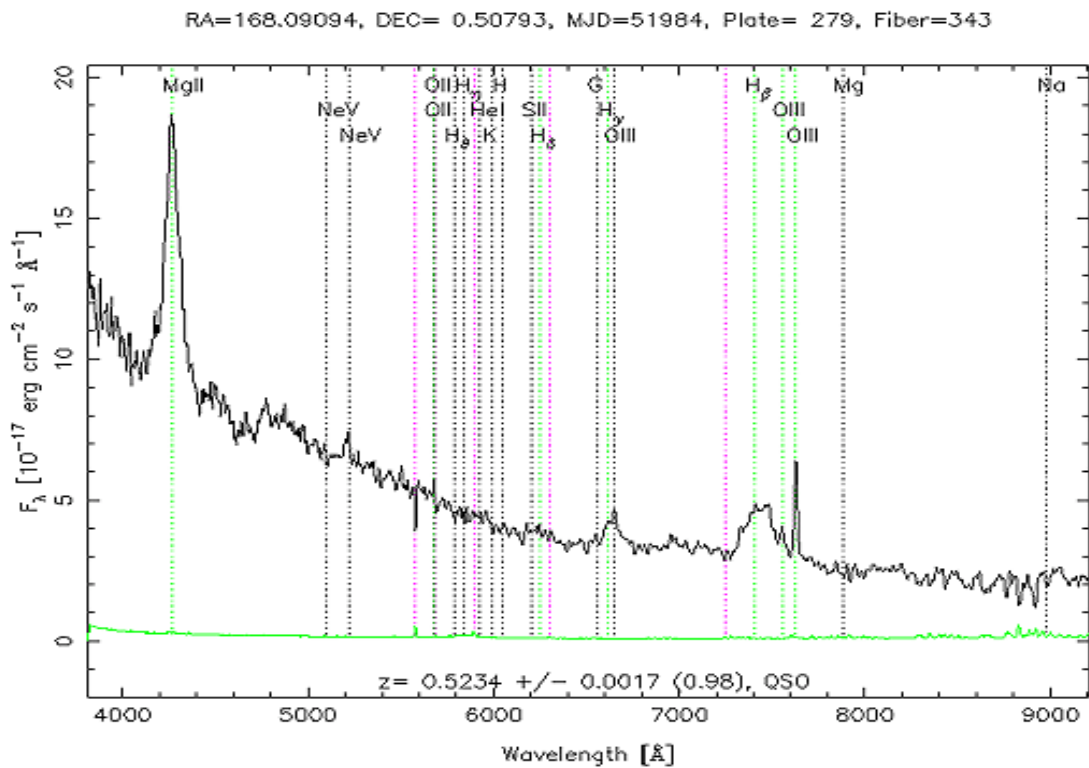


Figure1.3: Spectrum of a typical Quasar



Chapter 2

Literature Survey

Classification of astronomical objects was one of the earliest applications of datamining in astronomy. It was used in the development of an automatic star-galaxy classifier. As newer algorithms are being developed the classification results are also improving. The different data mining techniques used for classification have shown varying success rates. The success of the algorithm has also depended on the choice of the features or attributes chosen for the clustering. An interesting aspect is that, data mining tools have also been used to identify the relevant and significant features that help in better classification.

2.1 Review of existing systems

Neural Networks has proved to be a powerful tool in extracting necessary information and interesting patterns from large amounts of data in the absence of models describing the data. Automated clustering algorithms for classification of astronomical objects [7.6] have used supervised neural networks, to be specific, Linear Vector Quantization (LVQ), Single Layer Perceptron (SLP) and Support Vector Machines (SVM) for multi-wavelength data classification. Their classification was based on the properties of spectra, photometry, multi-wavelength etc. They used data from the X-ray(ROSAT), optical (USNO-A2.0) and infrared(2MASS) bands. The parameters they chose from different bands are $B - R$ (optical index), $B + 2.5\log(CR)$ (optical-X-ray index), CR , $HR1$ (X-ray index), $HR2$ (X-ray index), ext , $extl$, $J - H$ (infrared index), $H - Ks$ (infrared index), $J + 2.5\log(CR)$ (infrared-X-ray index). For supervised methods the input samples had to be tagged to know classes, for this they chose objects from the catalog of AGN [Veron Cetty & Veron, 2000].

They concluded that LVQ and SLP showed better performance when the features were fewer in number; and SVM showed a better performance when more features were included.

In 2006 with more data available about astronomy objects from radio surveys and Infrared surveys, Bayesian Belief Networks(BBN), Multi-Layer Perceptron (MLP) and Alternating Decision Trees(ADTrees) were used to separate quasars and non-quasars[7.5]. Radio and infrared rays are in different ends of the spectrum and can be used to better identify quasars. The radio data was obtained from the FIRST survey and infrared data from the 2MASS. They cross-matched the data from both surveys to obtain one-to-one entries -to get the entire set of radio and infrared details on the common objects- which were again matched with the Veron-Cetty & Veron catalog and the Tycho-2 catalog. The chosen attributes from different bands were $\log F_{peak}$ (F_{peak} : peak flux density at 1.4 GHz), $\log F_{int}$ (F_{int} : integrated flux density at 1.4 GHz), f_{maj} (fitted major axis before deconvolution), f_{min} (fitted minor axis before deconvolution), f_{pa} (fitted position angle before deconvolution), $j - h$ (near infrared index), $h - k$ (near infrared index), $k + 2.5\log F_{int}$, $k + 2.5\log F_{peak}$, $j + 2.5\log F_{peak}$, $j + 2.5\log F_{int}$. From the comparison of BBNs, ADTrees and MLP methods of classification it was concluded that all the three models achieved an accuracy between 94% and 96%. The comparison also re-established the fact that doing feature selection, to identify relevant features, increased the accuracy in all the three methods. Considering both accuracy and time, the ADTree algorithm proved to be the best of the three.

Based on the success of ADTree algorithm in identification of quasars, in 2007, Zhang & Zhao used Decision Tables for classifying point sources using the FIRST and 2MASS databases[7.4]. The features and data that they used was largely similar to what they had used for the BBN, MLP and ADTree models. Also, point source objects (like stars and quasars) tend to have their magnitude values falling in a particular range making the data more suitable to be classified using decision tree like structures. In addition to this they focussed on enhancing their feature selection technique, they used the Best-first search feature selection. By using the optimal features from the selection method they were able to conclude that the best-first search was more effective than the histogram technique (used previously by Zhang & Zhao). The Decision Table method also had a much higher accuracy (99%) in the case of quasars which proved that this was better than the previous models (BBN, MLP and ADTree).

In 2008, SDSS(Sloan Digital Sky Survey) made available the data of astronomical objects from the visible region of the spectrum. This provided a larger number of attributes to experiment with. Since SVMs had shown improved results (in 2004)with more attributes during the classification of these objects, Kd-tree and Support Vector machines were used for separating quasars from large survey databases [7.1]. Here the databases from SDSS and 2MASS catalogs were used. The features used in order to study the distribution of stars and quasars in the multi-dimensional space, were the different magnitudes: PSF magnitude (u^p, g^p, r^p, i^p, z^p), model magnitude (u, g, r, i, z) and model magnitude with reddening correction (u', g', r', i', z') from SDSS data, J, H and Ks magnitudes from 2MASS catalog. This study showed that both Kd-trees and SVMs are efficient algorithms for point source classification. It was also observed that, while SVM showed better accuracy, Kd-tree required less computation time. They also concluded that both the algorithms' performance was better when the features were fewer

2.2 Problems with existing systems and suggestions for improvement

2.2.1. Most algorithms have been used to classify just 2 object classes at a time.

Most algorithms have been used to classify just stars and galaxies or quasars and galaxies, But not all three together. There might be a possibility of using the same methods to cluster the objects differently, to characterize objects into stars, quasars, galaxies or others, provided that these objects are characterized by the same parameters and we can include these objects in our training process. If a single algorithm can be used to classify the objects into a wider category, it would save an enormous amount of time as compared to running many different algorithms on the same dataset.

2.2.2. Euclidean distance measure may not be the most suitable for astronomical objects.

Astronomy data parameters are of a very different nature compared to general numerical n-dimensional data. Although each astronomy object may be viewed as a point in n-dimensional space (n values for each of the attributes) it is probably not the right way to view them since these n values give a different meaning when we look at the spectra of the object and how the parameters define the shape of the spectra(or a crude curve). Euclidean distance measure is the common distance measure employed by all existing algorithms. However, by looking at the data from a different perspective it might be better to try and identify or develop a distance measure that is more suitable for astronomy data objects.

2.2.3. Feature selection

Most of the algorithms do not employ a feature selection mechanism. Feature selection mechanism is one which recognizes some parameters as more significant than others in order to discriminate between the type of objects. An optimum subset of the entire parameter list would be able to perform as well as or even better than the entire set. There can be a reduction in the time complexity and an increase in the accuracy by performing feature selection. Either by using methods like principal component analysis (PCA) or by experimenting with the various attributes we can try and identify the more significant features amongst them.

Chapter 3

Methodology

3.1 Identify features or parameters that can aid in characterizing the classes.

The survey data for each object would include all the features that have been captured. However, all of them might not be necessary to discriminate between our chosen classes. The parameters required for the classification are identified and chosen from the vast amount of data. Understanding the interdependencies among the different features is necessary for understanding the current models of classifications and identifying their flaws. This would also be one of the primary objectives. An increase in the accuracy of classification and reduction in computation time can be achieved by choosing an optimal set of significant characteristics. Hence it would be necessary for us to identify and select relevant parameters.

Stars, Quasars and Galaxies can be primarily identified from their spectrum. Hence, we choose to work with photometric parameters for the classification. Model magnitude values of the object at multiple wavelengths would be ideal attributes for classifying stars, galaxies and quasars since these are the values that are at different points in the spectrum. Color values (a ratio of magnitudes at 2 different wavelengths) will also aid in the classification as these would be better parameters for comparing the spectrum (curve-shape) of the objects. In addition to this, the use of red-shift values will also increase the accuracy of classifying quasars from other objects since quasars generally tend to have higher red-shifts.

3.2 Study of data mining tools that aid in classifying astronomy objects

We make a detailed study of the data mining techniques that have been used to classify the objects and identify the problems existing in them. Looking in to more recent developments, we try to identify relevant algorithms that can be used to classify astronomical objects. We would also make modifications to the existing algorithms and new algorithms to suit our needs. Our aim would be to come up with a set of algorithms that would bring improvements – in terms of accuracy or computation time - in the automated classification process.

3.3 Gather data on celestial objects from the various surveys.

There are a number of sky surveys that are conducted by various organizations which help in gathering information about the celestial sphere. Each of the surveys gather data pertaining to different frequencies in the spectrum. e.g Chandra Telescope gathers X-ray information, SDSS (Sloan Digital Sky Survey) gathers multi-wavelength data on 5 different bands, FIRST survey has radio data and so on. They each use different co-ordinate systems to map the sky. In order to classify an object into a specific class, we would need the properties of the radiations of the objects in all the different parts of the spectrum. We would need to correlate information from the different surveys and build a dataset for very large number of objects.

The Sloan Digital Sky Survey (SDSS) data would be able to satisfy our need for getting model magnitude values at multiple locations spread out in the spectra of an object. The SDSS uses five filters – Ultraviolet (u) 3543 Å(angstroms), Green (g) 4770 Å, Red (r) 6231 Å, Near Infrared (i) 7625 Å, and the Infrared (z) 9134 Å. The u, g, r, i and z values and the labels of the

objects from SDSS database will be the five basic parameters for the classification of stars, galaxies and quasars. From the 5 values we will also be able to derive the color values that can help enhance the classification. In addition to the model magnitudes we also use psf magnitudes (they are relative flux values obtained using a different function than what is used to obtain model magnitudes) as additional attributes in order to be able to obtain a clearer classification of galaxies.

3.4 Apply the algorithms on the chosen parameters, make improvisations.

The data is divided for training and testing purposes as required by the chosen algorithms. We can use existing softwares to test the results of some of the algorithms that have been used for classification by other astronomers. E.g the WEKA software has been used by some of the astronomers to classify stars and quasars using SVMs and kd-tree algorithms. Rapid Miner is another free and open software that could aid us in this regard. We would then build an application based on the newly developed algorithm and also include functions for the validation of its classification. This would give us the freedom to make many small changes with the algorithm – like using different distance measures or clustering techniques - and compare the performance.

3.5 Results and conclusion

The final phase is to compare the performance of our new algorithms with that of the existing ones with respect to classification of the astronomical objects. We need to see if there has been any improvement in the accuracy or reduction in time. We would also need to look into the success of our algorithms in classifying the new categories of objects.

Chapter 4

Closer study of relevant data mining algorithms

4.1 DBSCAN and GDBSCAN

DBSCAN is a Density Based Algorithm for Discovering Clusters in Large Spatial Databases[7.7] (Ester et al,1996). This algorithm uses the notion of density (the number of objects present in the neighbourhood space of a chosen object) to form clusters. This algorithm uses the following definitions:

1. **Eps-neighborhood** of a point: The Eps-neighborhood of a point p , denoted by $N_{Eps}(p)$, is defined by $N_{Eps}(p) = \{q \in D \mid \text{dist}(p,q) \leq Eps\}$.
2. **directly density-reachable**: A point p is directly density-reachable from a point q wrt. Eps, MinPts if
 - 1) $p \in N_{Eps}(q)$ and
 - 2) $|N_{Eps}(q)| \geq \text{MinPts}$ (core point condition).
3. **density-reachable**: A point p is density-reachable from a point q wrt. Eps and MinPts if there is a chain of points $p_1, \dots, p_n, p_1 = q, p_n = p$ such that p_{i+1} is directly density-reachable from p_i .
4. **density-connected**: A point p is density-connected to a point q wrt. Eps and MinPts if there is a point 'o' such that both, p and q are density-reachable from 'o' wrt. Eps and MinPts.
5. **cluster**: Let D be a database of points. A cluster C wrt. Eps and MinPts is a non-empty subset of D satisfying the following conditions:
 - 1) $\forall p, q$: if $p \in C$ and q is density-reachable from p wrt. Eps and MinPts, then $q \in C$. (Maximality)
 - 2) $\forall p, q \in C$: p is density-connected to q wrt. EPS and MinPts. (Connectivity)

The following images illustrate this:

Figure4.1: DBSCAN clustering on 3 different databases



The basic concept used in this algorithm is to cluster those objects that are density reachable and density connected, and it can also define objects that belonged to the border of a cluster. A detailed description of the algorithm can be found in the book *Data Mining Techniques* by Arun.K.Pujari(Pg.123).

Reasons for choosing to work with DBSCAN:

1. Astronomy objects of the same class tend to have similar color values (ratio of

magnitudes are similar although the magnitude itself might be different). The use of SVMs to classify astronomy objects using SDSS survey data has shown reasonably good results with respect to accuracy of classification [7.1]. SVMs generate a separating hyper-plane to classify objects, but based on closeness of color values, an algorithm that can find arbitrary clusters may be able to fare better. [i.e SVMs try to identify a plane in n-dimensions that helps in separating the different classes of objects from the supplied training data. Now, if the object classes can be identified by a single separating plane then considering the fact that the astronomy data objects chosen have fairly similar range of values, it might be possible to cluster them using a density based algorithm.]

2. DBSCAN also incorporates the concept of nearest neighbours (MinPts) which would help cluster astronomy objects, especially quasars better since the values of magnitudes and colors of the data object tend to fall within a small range.

GDBSCAN [7.8] , a further generalization of the DBSCAN algorithm gives greater freedom (or multiple options) with respect to the choice of defining the neighbourhood space of an object according to the application. This algorithm introduces the concept of adding weights to the parameters in computing distance between objects and in defining Eps which is used for computing the neighborhood density.

GDBSCAN being a more generalized approach of the DBSCAN includes different means for finding the density in the neighborhood space for n-dimensional data objects. The idea of giving different weights to the parameters can be used to cluster quasars better since most quasars have a higher red-shift value (z'), giving more weight-age to this factor may help us improve classification accuracy of quasars.

Advantages:

1. The space required for storing the data is less. Hence, its speed is also better. The operations that requires most time in this algorithm is the one used for identifying the objects within the neighborhood. By choosing appropriate Eps and MinPts, the number of times this operation needs to be executed can be reduced.

Disadvantages:

1. Choosing Eps for defining the neighborhood can be tricky. Finding the right value for Eps that is most appropriate for the given dataset is difficult. Also, the Eps value for quasar on one hand may be lower but stars and galaxies may require a higher value inorder to be clustered correctly. We need to find an Eps that is optimum in clustering all our data classes.

4.2 Decision Trees

A decision tree is a classification scheme which generates a tree and a set of rules, representing the model of different classes from a given data set. The decision tree model is constructed from the training set with the help of numerical and categorical attributes. Each node at a branch uses the value of the attributes to to derive the rules. The leaves of the tree contain the label of the classes. When the test data is input , the tree is traversed from the root and the branch chosen depends on the parameters under consideration in that node. Finally the test input is assigned the class of the leaf node it reaches after traversing a particular path of nodes from the root to

that leaf. Details about determining the splitting attribute and splitting criteria are referred from [7.9](pp 154-195).

The values for the magnitudes and colors of astronomy objects tend to fall within a close range. [The average range into which each parameter falls can be found from the training sample]. Based on this fact, it may be possible to form rules to group the objects [like, objects having $z \geq A$, $u-g > B$ and $i-z < C$ have a greater tendency of being galaxies]

Advantages:

1. Once the tree is constructed the time for testing the data inputs reduces greatly.
2. The storage space for the constructed tree is also less.

Disadvantages:

1. The number of scans required to construct the tree is very large, almost as many times as the number of attributes present. The astronomy dataset has 10 parameters (5magnitudes and 5 color values) and the label, hence the data would be scanned about 10 times.
2. Determination of splitting attributes and splitting criteria play a primary role in the final classification, and these are difficult operations. Especially for stars and galaxies there might not exist a clear splitting attribute.

4.3 Comparison of the algorithms on astronomy dataset

Dataset:

Data was obtained from the Sloan Digital Sky Survey(SDSS) data releases 6 and 7 (DR6 and DR7). (<http://cas.sdss.org/dr7/en/>, <http://www.sdss.org/dr7/>)

Objects that are intended to be classified: Stars, Quasars and Galaxies

Number of chosen objects of each class: **Stars- 3000, Quasars – 2000, Galaxies – 3000**

Parameters obtained from the survey: **model magnitudes – u, g, r, i, z**

Parameters Derived from the original parameters: **colors : u-g, g-r, r-i, i-z, u-r**

Training data: 5400 (1400Qso, 2000Stars, 2000Galaxies)

Test data : 2600 (600Qso, 1000Stars, 1000Galaxies)

4.3.1 DBSCAN

The experiment was performed using the DBSCAN clustering feature java package present with RapidMiner data mining software (<http://www.rapidminer.com>) which has been developed using open-source data mining library formerly known as YALE. (DBSCAN is an unsupervised algorithm)

Input: Training Data set (5400 objects)

Execution time (to obtain clusters): 153 secs

Number of clusters obtained: 9 clusters

Eps and MinPts default values were retained.

Cluster composition:

Cluster0 – 5354 items

Remaining clusters contained 4-6 items per cluster.

Conclusion: DBSCAN algorithm is not suitable for this data set. It is very difficult to identify the best values for Eps and MinPts that can cluster the objects into stars, quasars and galaxies.

4.3.2 Decision Tree

Decision Tree experiment was tested using RapidMiner software. The algorithm takes information from both numerical and categorical attributes. The chosen algorithm works similar to Quinlan's C4.5 or CART [7.9]. The actual type of the tree is determined by the criterion .e.g. Using gain_ratio or Gini for CART/C4.5[7.9].

Class Precision: It is the *number of relevant objects* retrieved by a search *divided by the total number of objects retrieved* by that search, expressed as percentage.

Class Recall: It is the *number of relevant documents* retrieved by a search *divided by the total number of existing relevant documents* (which should have been retrieved), expressed as percentage.

Input1: Data objects with parameters **u-g, g-r, r-i, i-z. u-r**

Execution Time: 703s

	True Qso	True Star	True Galaxy	Class Precision
Predicted Qso	1731	180	154	83.03%
Predicted Star	112	2014	976	64.94%
Predicted Galaxy	157	806	1870	66.01%
Class Recall	86.55%	67.13%	82.33%	

Table1: Decision tree result for Color parameters alone.

Accuracy of the system with colors as parameters: $(1731+2014+1870)/8000 = 70.19\%$

Input2: Data objects with parameters **u, g, r, i, z, u-g, g-r, r-i, i-z. u-r**

	True Qso	True Star	True Galaxy	Class Precision
Predicted Qso	1896	42	52	95.28%
Predicted Star	49	2129	709	73.74%
Predicted Galaxy	55	829	2239	71.69%
Class Recall	94.80%	70.97%	74.63%	

Table2: Decision tree result with parameters – magnitude and color.

Accuracy of the system with all ten parameters: $(1896+2129+2239)/8000 = 78.3\%$

4.3.3 Support Vector Machine(SVMs)

The experiment used the LibSVM Learner package of RapidMiner software. It applies the libsvm learner by Chih-Chuhn Chang and Chih-Jen Lin.

Input: Data objects with parameters **u, g, r, i, z, u-g, g-r, r-i, i-z, u-r**(magnitude and color)

	True Qso	True Star	True Galaxy	Class Precision
Predicted Qso	1822	408	68	79.29%
Predicted Star	112	670	531	51.03%
Predicted Galaxy	66	1922	2401	54.70%
Class Recall	91.10%	22.33%	80.03%	

Table3: SVM results with parameters – magnitude and color.

Accuracy of the system with all ten parameters: $(1822+670+2401)/8000 = \mathbf{61.16\%}$

Input: With parameter as colors alone

Output: The prediction of quasars is 0%.

Input: Data objects with parameters **u, g, r, i, z** (magnitude only)

	True Qso	True Star	True Galaxy	Class Precision
Predicted Qso	1832	490	87	76.05%
Predicted Star	94	228	188	44.71%
Predicted Galaxy	74	2282	2725	53%
Class Recall	91.60%	7.6%	90.83%	

Table4: SVM results for magnitude parameters alone.

Accuracy of the system with magnitude as parameters: $(1832+228+2725)/8000 = \mathbf{59.81\%}$

4.3.4 Conclusions from the experiments:

1. Decision Tree approach is more accurate than SVMs and DBSCAN algorithms. Support vector machines seem to be able to classify the objects marginally, however it fails to distinguish between stars and galaxies. DBSCAN clustering algorithm is unable to classify astronomy objects and it seems to dump all the objects into the same cluster.
2. Both decision trees and SVMs are able to identify quasars with a higher accuracy, irrespective of the choice of the parameters. Additionally decision tree approach to identify quasars is much better when both the color and magnitude parameters are used [It is able to classify 94.8% of the quasars correctly and of all the objects that it predicts as quasars 95.28% of them are truly quasars]
3. SVMs and Decision tree algorithms both perform better with more attributes, they have shown better results when both the magnitude and color parameters have been supplied as opposed to the case where only one of them (magnitude or color) is used.

Chapter 5

Implementation

This section is divided as follows:

Each sub-section from 5.1 to 5.4 represents different ideas that would be a part of the algorithm.

5.1 Describes the basic data structure that the algorithm follows.

5.2 Describes distance measures that can be used in the algorithm along with comparisons.

5.3 It introduces the idea of implementing k-means method and its results.

5.4 This sub-section introduces new attributes and their significance in the classification.

5.1 Data Structure: Tree to classify astronomy data

It has been found that Stars, Quasars and Galaxies have unique spectral properties (<http://cas.sdss.org/dr7/en/proj/advanced/quasars/spectracomparisons.asp>). The data being used for classification of the objects contains a small fraction of the spectral properties. [i.e. SDSS survey has the data for each object in 5 different wavelengths across the electromagnetic spectrum] The idea behind this algorithm is to compare the parameters (like curve matching, but we have very few points) and then classify the objects.

The attributes we would use for the comparison is the model magnitude and color of the objects. The magnitudes are u, g, r, i and z values. The color is the ratio of the magnitudes at two different wavelengths. (u-g, g-r, r-i, i-z, u-r).

5.1.1 Idea behind the algorithm:

We attempt to build a tree which will classify the input data depending on the closeness(distance measure) of the individual attributes.

Let us assume each object is represented as a tuple $\langle \text{objId}, a_1, a_2, \dots, a_n \rangle$. Now for each object we construct 'n' nodes containing the ObjectIds and values of one of the attributes in each node. Those nodes with $\langle \text{attribute}_i \rangle$ values that is common for more than 1 object (i.e there exist more than 1 object with the same value for attribute_i) will include more than one ObjectId. The root does not contain any attribute information, it is connected to all nodes with distinct $\langle \text{attribute}_1 \rangle$ values. The $\langle \text{attribute}_i \rangle$ nodes (A_i nodes) are connected to all those $\langle \text{attribute}_{i+1} \rangle$ nodes (A_{i+1} nodes) whose ObjectIds are included in the A_i node [i.e A_i node is connected to A_{i+1} node iff ($\exists \text{ ObjectId}_k$ such that $\text{ObjectId}_k \in A_{i+1}$ node AND $\text{ObjectId}_k \in A_i$ node)]. Finally, the A_n nodes are connected to the labels of the respective ObjectIds, the labels forming the leaves.

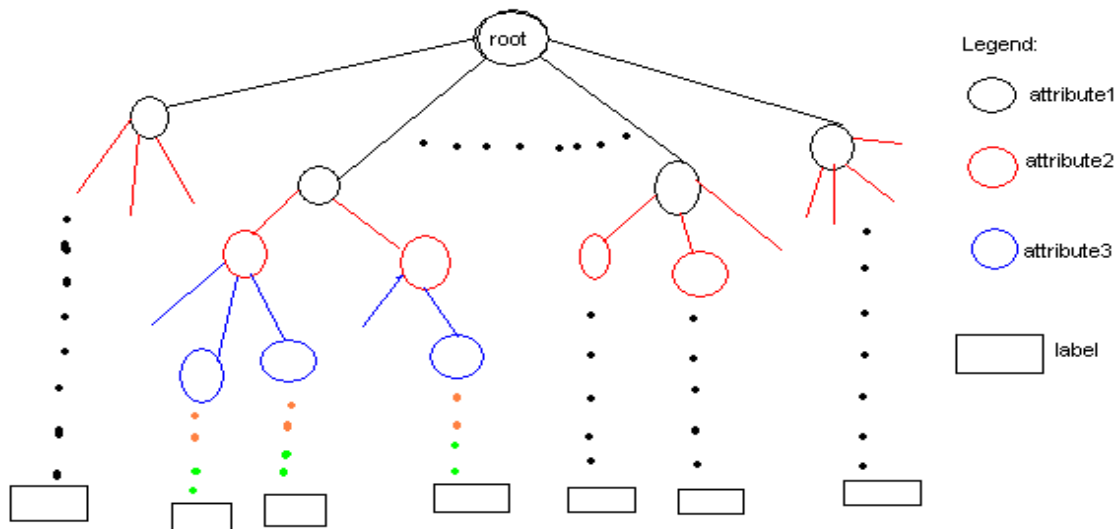
5.1.2 Classification of test data:

Each test data input is taken 1 at a time, we traverse all the branches of the tree and using the distance measure we identify the branch that is the closest to the input case. At each level i, we make a comparison of attribute_i of the test case with the value in the node to determine the closeness. After traversing all the branches and finding the closest match, we assign the label of

the closest match to the test input.

5.1.3 Diagrammatic Representation:

Figure 5.1: Diagrammatic Representation of proposed structure



Advantages:

1. The construction of the tree is fast. Since the training data is scanned only once, the tree can be constructed easily and quickly.
2. The tree can also reduce the number of computations performed for each test data. As we traverse down the tree to level i , if there are more paths for the next level, the computation done so far till level i can be saved and re-used for each of the branches. However, if our sample (training) data was just a list we might have to do the computations for each sample data individually.

Drawbacks:

1. The choice of attributes at each level can make a difference to the width of the tree. e.g. Level 1 has attribute1 nodes, level 2 has attribute2 nodes, so on. But we may be able to reduce the number of nodes in the tree by choosing common valued attributes at a higher level (i.e those attributes that have values which are the same for a large number of objects should be placed higher up in the tree). The next level of optimization could be to find such an optimal tree.

5.2 Distance measure: Use of different distance measures

Astronomy data parameters are of a very different nature compared to general numerical n -dimensional data. Although astronomy objects may be viewed as a point in n -dimensional space (n values for each of the attributes) it is probably not the right way to view them since these n

values give a different meaning when we look at the spectra of the object and how the parameters define the shape of the spectra(or a crude curve).

We could probably define a distance measure as follows:

5 basic parameters : u, g, r, i, z

Derived parameters (colors): u-g, g-r, r-i, i-z, u-r

Object O: <objectId, a1, a2, ..., an> [a1, a2, a3 ... are either u, g, r, i, z or the color values]

Distance between O1 and O2 can be given by:

1. $X_{ai} = O1_{ai} - O2_{ai}$

2. **Mean** distance of each of the parameters:

$$\mu = (1/n) * \sum_{i=1}^n (O1_{ai} - O2_{ai})$$

3. **Variance**

$$\sigma^2 = (1/n) * \sum_{i=1}^n (X_{ai} - \mu)^2$$

4. **Standard Deviation**

$$\sigma = \sqrt{\sigma^2}$$

Distance between O1 and O2 can be defined as μ or σ .

The 'variance' and standard deviation are values that would give us a picture of the displacement of 1object's spectra in comparison with an other objects spectra.

5. **Sum of modulus of differences**

$$D = \sum_{i=1}^n |O1_{ai} - O2_{ai}|$$

6. **Modulus of sum of distances**

$$D = \left| \sum_{i=1}^n (O1_{ai} - O2_{ai}) \right|$$

5.2.1 Comparison of different distance measures

Implementation: A single list of all test objects is formed and the test inputs are classified by comparing it with each of the training set objects. The experiment was done with both the euclidean distance measure as well as with the proposed standard deviation as the distance measure.

Training set: 5400(1400-Qso,2000-Star, 2000-Galaxy)
 Test set: 2600(600-Qso, 1000-Star, 1000-Galaxy)

Input: Data objects with parameters **u, g, r, i, z**
 Distance measure: **Euclidean**
 Execution Time: **1.984s**

	True Qso	True Star	True Galaxy	Class Precision
Predicted Qso	562	24	32	90.94%
Predicted Star	22	654	306	66.6%
Predicted Galaxy	16	322	662	71.69%
Class Recall	93.67%	65.40%	66.2%	

Table5: List data structure result with parameters – magnitude .

Accuracy with the magnitudes as parameters: = $(562+654+662)/(2600) = 72.2230\%$

Input: Data objects with parameters **u-g, g-r, r-i, i-z, u-r**
 Distance measure: **Euclidean**
 Execution Time: **2.131s**

	True Qso	True Star	True Galaxy	Class Precision
Predicted Qso	514	30	49	86.68%
Predicted Star	42	608	315	63.01%
Predicted Galaxy	44	362	636	61.04%
Class Recall	85.67%	60.8%	63.6%	

Table6: List data structure result with parameters – colors.

Accuracy of the system with the colors as parameters: $(514+608+636)/(2600) = 67.615\%$

Input: Data objects with parameters **u, g, r, i, z, u-g, g-r, r-i, i-z, u-r**
 Distance measure: **Euclidean**
 Execution Time: **3.181s**

	True Qso	True Star	True Galaxy	Class Precision
Predicted Qso	559	18	22	93.32%
Predicted Star	22	663	317	66.17%
Predicted Galaxy	19	319	661	66.17%
Class Recall	93.17%	66.30%	66.1%	

Table7: List data structure result with parameters – magnitude and colors .

Accuracy with the magnitude and colors as parameters: $= (559+663+661)/(2600) = 72.42\%$.

Results of the experiment with the proposed distance measure

Input: Data objects with parameters **u, g, r, i, z**

Distance measure: **Standard deviation**

Execution Time: **2.760s**

	True Qso	True Star	True Galaxy	Class Precision
Predicted Qso	465	160	109	63.35%
Predicted Star	108	537	514	46.33%
Predicted Galaxy	27	303	337	53.32%
Class Recall	77.5%	53.7%	37.7%	

Table8: Implementation List result with parameters – magnitude .

Accuracy with the magnitudes as parameters: $= (465+537+377)/2600 = 53.038\%$

Input: Data objects with parameters **u-g, g-r, r-i, i-z, u-r**

Distance measure: **Standard Deviation**

Execution Time: **2.807s**

	True Qso	True Star	True Galaxy	Class Precision
Predicted Qso	441	237	269	46.57%
Predicted Star	106	440	446	44.35%
Predicted Galaxy	53	323	285	43.12%
Class Recall	73.5%	44.0%	28.5%	

Table9: Implementation List result with parameters – colors.

Accuracy of the system with the colors as parameters: $(441+440+285)/2600 = 44.846\%$

5.3 Use k-means for better accuracy and identify an optimum k value.

In order to classify the test input, we consider the closest training data object. However, considering one closest object alone may not always be correct. It would be better to consider k-nearest neighbors for each test input and assign it the class that is most frequent among them. The most frequent object may also not always give the right class of the object because, among the nearest if the test data resembles 2 or 3 objects of it's actual class the perfectly but if there are more objects of another class in the k-nearest objects such an input would be assigned the wrong class. This can be rectified by considering the normal weights for each of the k-nearest neighbors and calculating the cumulative weights for each class. Then the test object is assigned that class that has maximum weight.

While using k-means technique we would also need to find the optimum k value (or range of values) for which the system gives more accurate results. This can easily be accomplished by assigning k values from a range (e.g. 1-20) and comparing the results. By drawing a graph of the k-value against the accuracy[Section 5.3.1] we can also make a study of how the accuracy varies across different k values, this would help us identify a value for k that gives the best results.

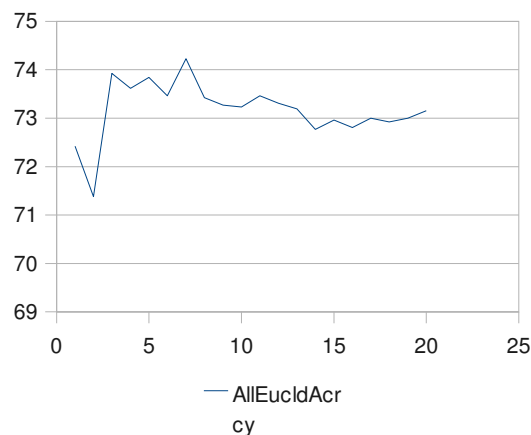
5.3.1 Accuracy vs k

Description of Graph: The graphs show the variation in accuracy with respect to change in 'k'.

x-axis: k-values [Range: 1 to 20]

y-axis: System Accuracy

1. Parameters: 5Model magnitudes+5colors
Distance measure: Euclidean



2. Parameters: 5magnitudes+5colors
Distance measure: Sum of modulus

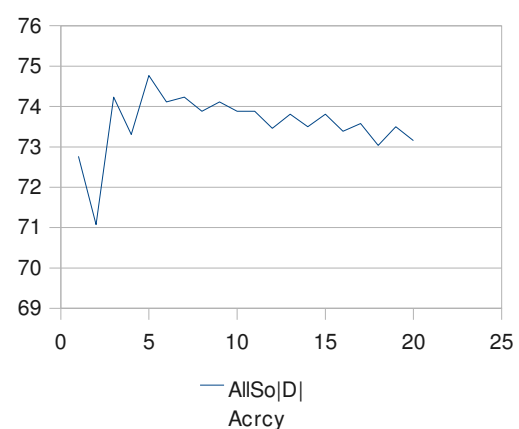


Figure5.2: The graphs compare the effect of 2 different distance measures in a system using the 5 model magnitudes and 5 colors as attributes.

5.4 Inclusion of more parameters

5.4.1 More derived parameters(colors)

Here we derive more attributes from the already existing ones (as a function of the existing parameters). Earlier we used u, g, r, i and z (the basic magnitudes) and u-g, g-r, r-i, i-z and u-r (colors) as the parameters. This set can be extended to include all colors i.e u-i, u-z, g-i, g-z and r-z can also be included.

1. Using Decision Trees:

Parameters: 5model magnitudes+10 colors

	True Qso	True Star	True Galaxy	Class Precision
Predicted Qso	1318	25	45	94.96%

Predicted Star	32	1478	570	71.06%
Predicted Galaxy	50	497	1385	71.69%
Class Recall	94.14%	73.90%	69.25%	

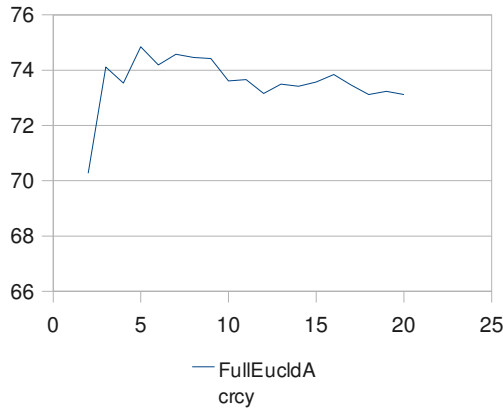
Table10: Implementation List result with parameters –magnitudes+10colors .

Accuracy with the all colors&magnitudes as parameters: = $(1920+1191+1065)/5400 = 77.43\%$

2. Using developed data structure

2.1. Parameters: 5Model magnitudes+10 colors

Distance measure: Euclidean



2.2. Parameters: 5magnitudes+10 colors

Distance measure: Sum of modulus

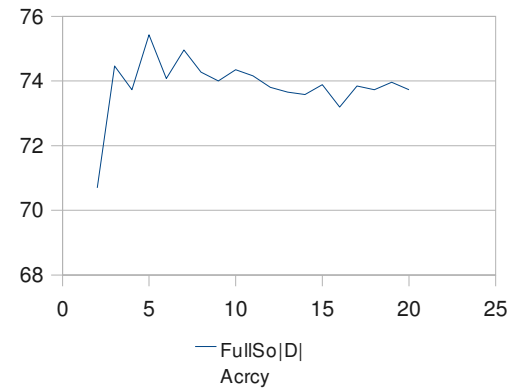


Figure5.3: The graphs show the variation in accuracy with respect to k-value in a system using the model magnitudes and all ten colors as attributes

5.4.2 New parameters

We can also use different set of magnitudes as attributes instead of just the model magnitudes. PSFmagnitudes are another set of magnitudes (relative flux values) of astronomical objects that are calculated using a function different from that used to obtain model magnitudes from flux values of objects. [psf magnitudes use point spread functions to calculate the magnitudes from the flux values] PSF magnitudes are more effective for identification of point source objects like stars and quasars.

1. Using Decision Trees:

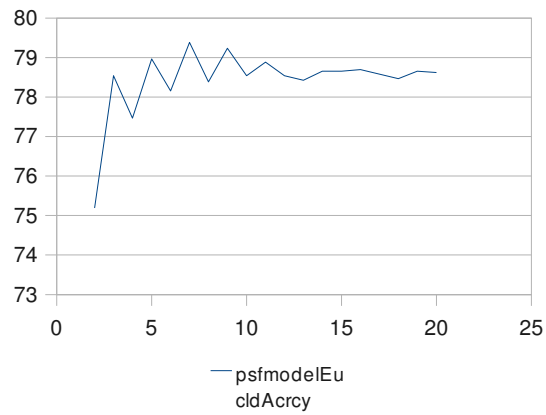
Parameters: 5PSF + 5model magnitudes

	True Galaxy	True Star	True Qso	Class Precision
Predicted Galaxy	1920	28	45	96.34%
Predicted Star	38	1191	290	78.41%
Predicted Qso	42	781	1065	56.41%
Class Recall	96.00%	59.55%	76.07%	

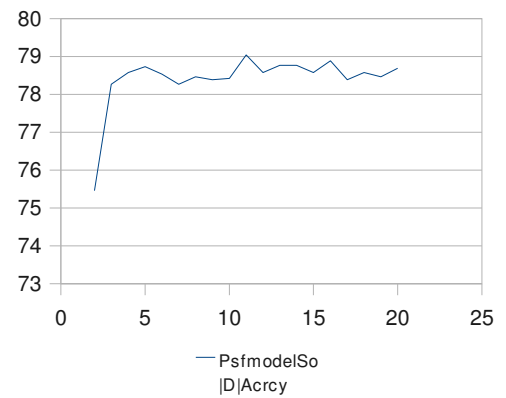
Table11: Implementation List result with parameters – psf+model magnitudes .

Accuracy with the magnitudes as parameters: = $(1920+1191+1065)/5400 = 77.33\%$

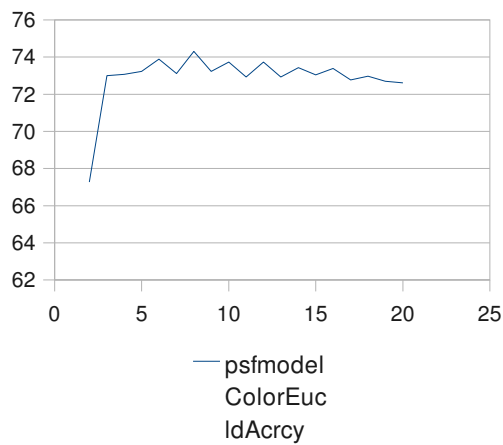
1. Parameters: 5PSF +5Model magnitudes
Distance measure: Euclidean



2. Parameters: 5PSF+5Model magnitudes
Distance measure: Sum of modulus



3. Parameters : 5 PSF + 5model magnitudes + 10 colors
Distance measure: Euclidean



Distance measure: Sum of modulus

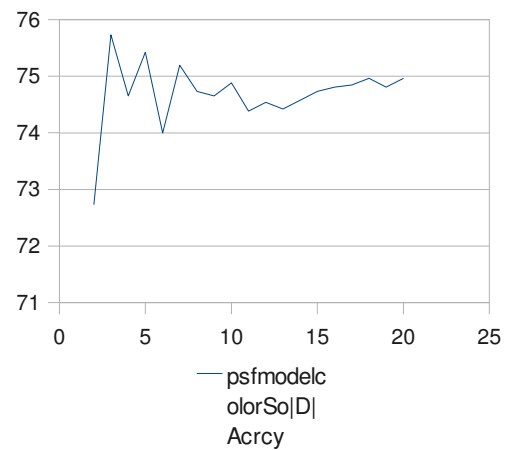
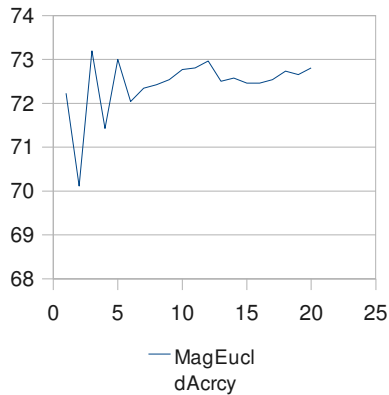


Figure5.4: The graphs show a comparison of the accuracy of the system when psf magnitude is also included in the combination of attributes.

Figure5.5: Graphs showing accuracy vs k-value with various combinations of input data parameters and distance measures. (X-axis: k value ; Y-axis: Accuracy)

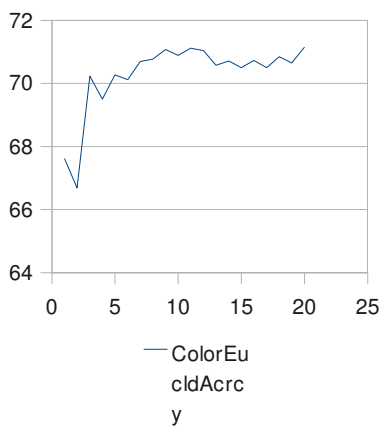
1. Parameters: Magnitudes only
Magnitudes+Colors

Distance measure: Euclidean



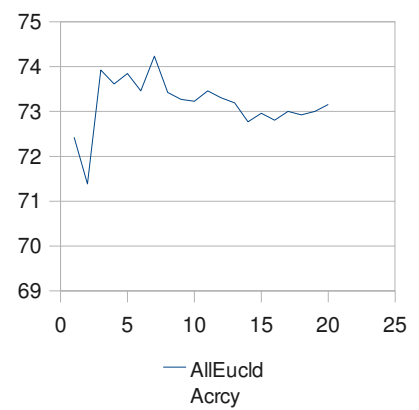
2. Parameters: 5 Colors only

Distance measure: Euclidean



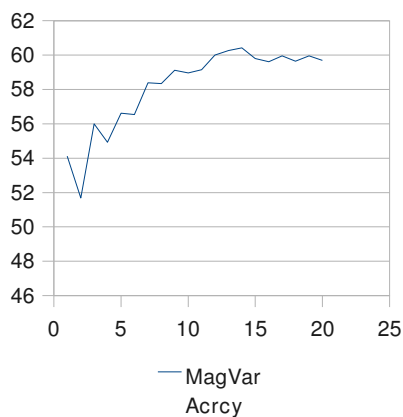
3. Parameters:

Distance measure: Euclidean



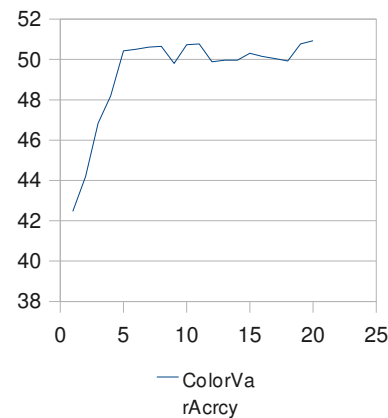
4. Parameters: Magnitudes only
Magnitudes+Colors

Distance measure: Variance



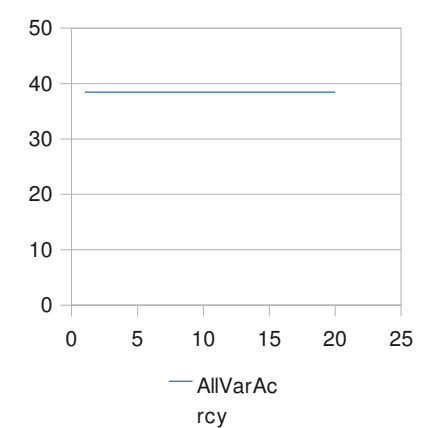
5. Parameters: 5 Colors only

Distance measure: Variance



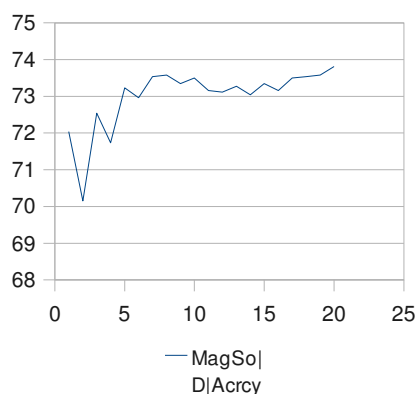
6. Parameters:

Distance measure: Variance



7. Parameters: Magnitudes only
Magnitudes+Colors

Distance measure: Sum of |Dist|



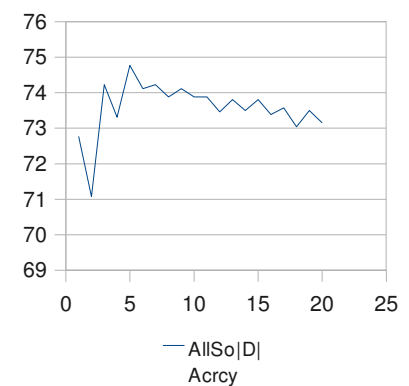
8. Parameters: 5 Colors only

Distance measure: Sum of |Dist|



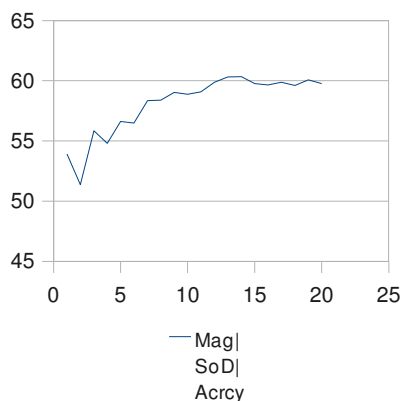
9. Parameters:

Distance measure: Sum of |Dist|



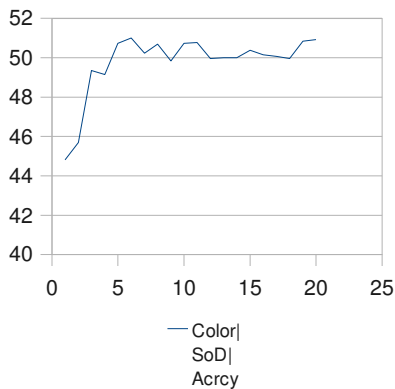
10. Parameters: Magnitudes only
Magnitudes+Colors

Distance measure: |Sum of Dist|



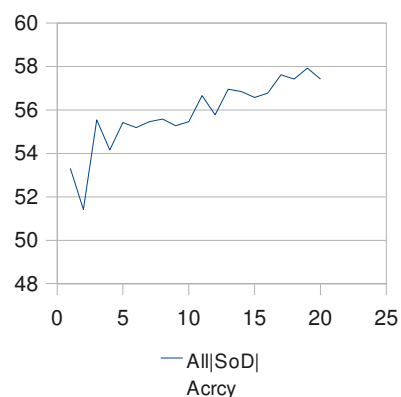
11. Parameters: 5 Colors only

Distance measure: |Sum of Dist|



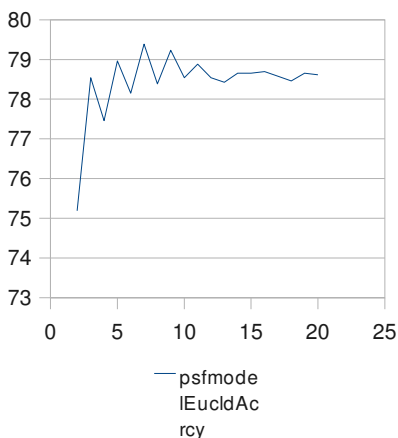
12. Parameters:

Distance measure: |Sum of Dist|



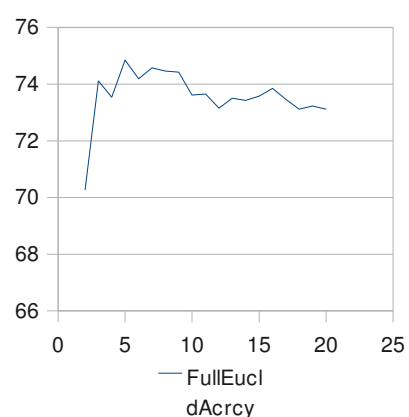
13. Parameters: psf+model_mags
psf+model+colors

Distance measure: Euclidean



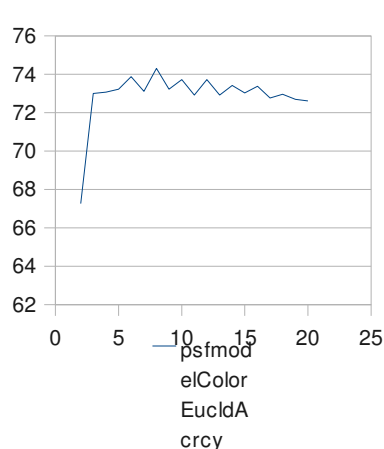
14. Parameters: 5mags+10Colors

Distance measure: Euclidean



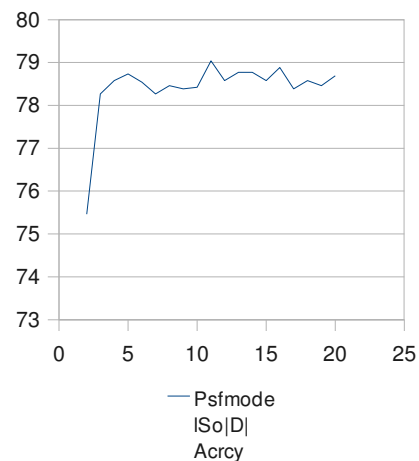
15. Parameters:

Distance measure: Euclidean



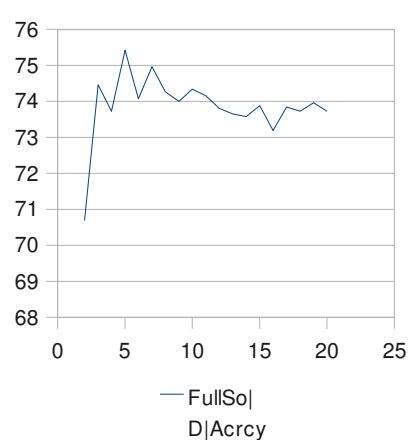
16. Parameters: psf+model_mags
psf+model+colors

Distance measure: Sum of |Dist|



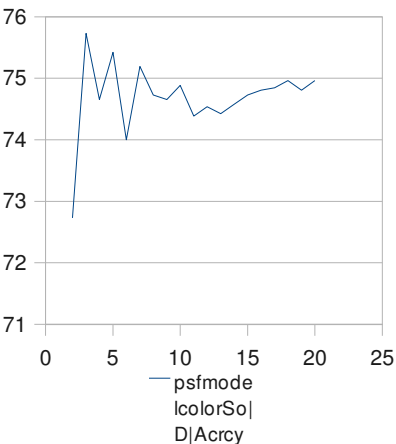
17. Parameters: 5mags+10Colors

Distance measure: Sum of |Dist|



18. Parameters:

Distance measure: Sum of |Dist|



Chapter 6

Conclusion

1. Decision Tree has the highest system accuracy followed by the list data structure, then it is the Support vector machine implementation and finally the list with the modified distance measure.
2. With respect to execution time the list data structure has the best time(<4secs for any given choice of distance measure).The other data structures involve more operations, additionally the decision tree, DBSCAN and SVM algorithms are implemented using Java, which would have contributed significantly to the execution time.
3. All the algorithms seem to be capable of identifying quasars better than stars and galaxies. Here again decision trees are able to classify galaxies marginally better than the other algorithms. And classification of stars and galaxies appears to be more difficult.
4. The list data structure has also performed better with more inputs i.e when both magnitude and colors are supplied than when only one of them is supplied as input. However, there does not seem to be much of a difference when magnitude alone is supplied(72.223%) and when all ten parameters are provided(72.4%)
5. Inclusion of more colors (u-i, u-z, g-i, g-z, r-z) has not improved the accuracy, in contrast these parameters appear to have decreased the accuracy to some extent. This is primarily because these extra colors that are being used as new parameters can actually be derived quite easily from the initial choice of colors (u-g, g-r, r-i, i-z). Hence the inclusion or removal of the additional colors should actually not bear an effect on the classification.
6. Psf magnitudes have indeed proved to be more significant in-order to obtain a better classification. Psf magnitudes have specifically helped in identifying galaxies more accurately as these magnitudes are better suited for separating point sources (like stars) from other astronomical objects.
7. The best result has been obtained when using psf magnitudes and model magnitudes alone without including any of the color attributes. This is because, the model magnitudes and color attributes help identify quasars more accurately but blur the differences between stars and galaxies, whereas psf magnitudes help separate point sources(stars&quasars) from non point sources(like galaxies). Together they are able to classify the objects better.

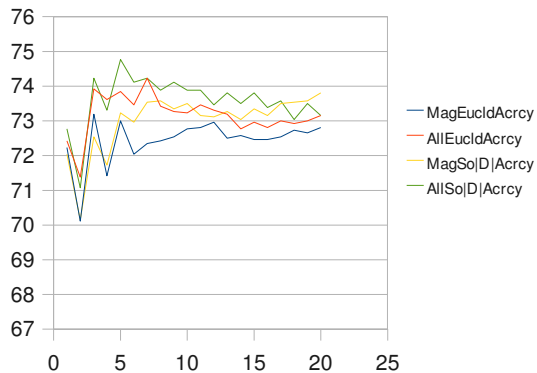
6.1 Advantages of proposed system

6.1.1 Overall time of execution is very small

With the proposed tree implementation, the time required to complete the classification process is very less (between 4 to 5 seconds in most cases irrespective of the distance measure used). Whereas in existing systems like SVMs and Decision trees a minimum of 150secs is required to obtain the classification. Thus, with respect to time the proposed system is clearly superior, and its accuracy also matches that of SVMs' and is not very far from that obtained using Decision trees.

Figure6.1: Graphs comparing performance of Sum of Modulus of distance and Euclidean distance measures with different parameters. (X-axis: k value ; Y-axis: Accuracy; Euclid-Euclidean, SolDl - sum of modulus of distance)

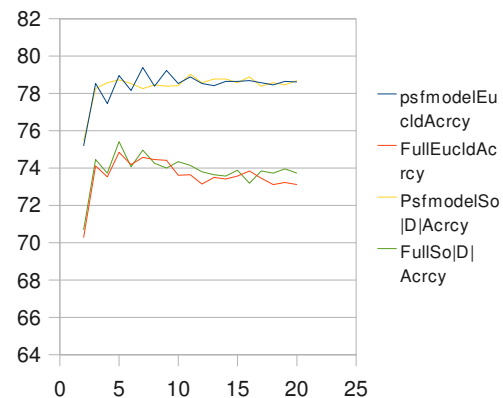
1. Parameters: model magnitudes(Mag), model magnitude & colors(All).



Inference:

Sum of modulus of distances (MagSolDlAcrcy and AllSolDlAcrcy) performs better than Euclidean distance(MagEuclidAcrcy and AllEuclidAcrcy) with respect to each of the parameters.

2. Parameters: PSF& model magnitudes (psfmodel), PSF&model magnitudes along with color(Full).



Inference:

Performance of Sum of modulus of distances (SolDlAcrcy) and Euclidean distance(EuclidAcrcy) are almost equivalent with both parameter sets, with Euclidean being only marginally better.

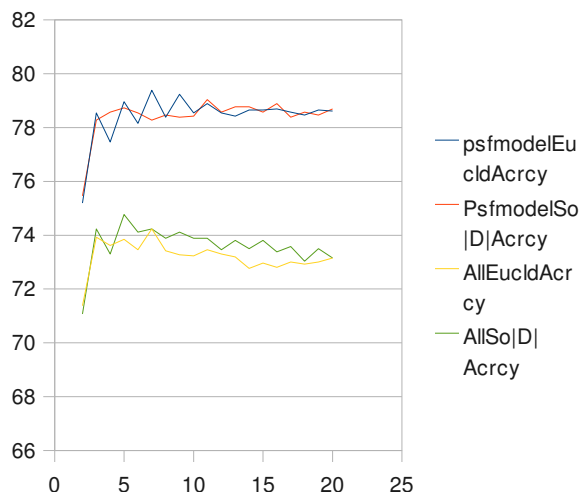


Figure6.2: Graph comparing performance with parameters - PSF magnitudes and model magnitudes vs colors and model magnitudes. (X-axis: k value ; Y-axis: Accuracy; Euclid-Euclidean, SolDl - sum of modulus of distance; psfmodel-PSFand Model magnitudes, All- model magnitudes and colors.)

Inference:

This graph clearly shows psf and model magnitudes when taken as parameters perform better than colors and model magnitudes as parameters.

6.1.2 Sum of modulus distance measure is advantageous

The time for execution is reduced by using 'sum of modulus of distances' as the distance measure instead of euclidean distance. At the same time it does not compensate much on the accuracy of the system itself [Figures 5.2, 5.3, 5.4, 5.5 and 6.1]. Thus in-order to obtain quicker results with more or less the same accuracy as that of euclidean distance we could use the sum of modulus as distance measure. Theoretically also calculating euclidean distance involves computation of squares and roots whereas the calculation of sum of modulus distances just performs addition and is hence computationally less intensive.

6.2 Future work

1. We can experiment with the different distance measures on the existing algorithms (like SVMs, kd-trees etc.) to view their impact on the classification.
2. We can try to classify the objects by clustering them and using representatives. [Form multiple clusters of objects of the same kind based on distance. Choose representative values for attributes of each cluster. Classify test objects depending on their proximity to the representatives.]
3. Make a more detailed study of why certain algorithms like DBSCAN were not effective in classifying the data.

Chapter 7

References

- [7.1] Gao, D., Zhang, Y., Zhao, Y., Support Vector Machines and Kd-tree for separating Quasars from Large Survey Databases, Chinese Journal of Astronomy and Astrophysics (ChJAA), 2008,
- [7.2] *Jiawei Han and Micheline Kamber, Data Mining Concepts and Techniques, Morgan Kaufmann Publishers, chapter 6, p 285-373*
- [7.3] Gao, D., Zhang, Y., Zhao, Y., The application of kd-tree in astronomy, Astronomical Data Analysis Software and Systems XVII , ASP Conference Series, Vol. XXX, 2008, P5.11
- [7.4] Zhang, Y., Zhao, Y., A Comparison of BBN, ADTree and MLP in separating Quasars from Large Survey Catalogues, ChJAA 7, 2007
- [7.5] Zhang, Y., Zhao, Y., Gao, D., Decision Table for classifying point sources based on FIRST and 2MASS databases, ChJAA 2006
- [7.6] Zhang Y., Zhao Y., 2004, Automated clustering algorithms for classification of astronomical objects, Chinese Journal of Astronomy&Astrophysics.
- [7.7] Martin Ester, Hans-Peter Kriegel, Sander J, Xiaowei Xu, 1996. “A density based algorithm for discovering clusters in large spatial databases with noise”. In Proc. of 2nd Int. Conf. on Knowledge Discovery and DataMining (KDD-96) Portland.OR pp. 226-231
- [7.8] Martin Ester, Hans-Peter Kriegel, Sander J, Xiaowei Xu, 1996. “Density Based Clustering in Spatial Databases. The Algorithm GDBSCAN and its Applications”.
- [7.9] Arun. K. Pujari, *Data Mining Techniques*, University Press. pp. 123-126, pp 153-195
- [7.10] <http://cas.sdss.org/dr7/en/proj/advanced/>