PROJECT NAME – Astronomical Classification on Cloud

Navin Karri (MT2016070)

Piyush Singh Bora (MT2016104)

Vishwanath Acharya (MT2016003)

January 11, 2016

**Project Objective**:

* Classification of all unclassified objects available in SDSS to Stars and Quasars using machine learning classification algorithms on cloud.
* Extend these classification techniques to run on the cloud.
* For optimizing the techniques to run on multiple nodes in the cloud for better performance.

**Project Description**:

The photometric data available for previously identified objects in the SDSS catalogue is being used for classifying the large data sets downloaded from the same catalogue.

There has been prior work done on this project to classify stars and quasars using unclassified photometric data from SDSS catalogue. There work has been primarily on using combination of kNN and SVM algorithms to design an efficient algorithm to classify these data and implementing it in distributed environment. The implementation of these algorithms has been done on a sample of a data to check the performance of their algorithms.

Our work majorly focuses on getting the entire SDSS catalogue to the cloud and applying algorithm on the whole set of data objects and classify the entire dataset to Stars and Quasars category.

The data is trained on the labelled data sets to classify the large data sets available with us. We intend to use different classification algorithms such as kNN and SVM or an extension/combination of these algorithms run distributed on the cloud for better performance which would be far better in terms of running time than running the algorithms linearly on the large data set on a single node.

**Data Description**:

Our dataset will be taken from the SDSS (Sloan Digital Sky Survey) Spectroscopic Catalogue, primarily Quasar catalogue and Stellar parameter. SDSS spectroscopic catalogue contain parameters such as redshift, classification, velocity dispersion, quality flags.

Classified data from Spectroscopic catalogue:

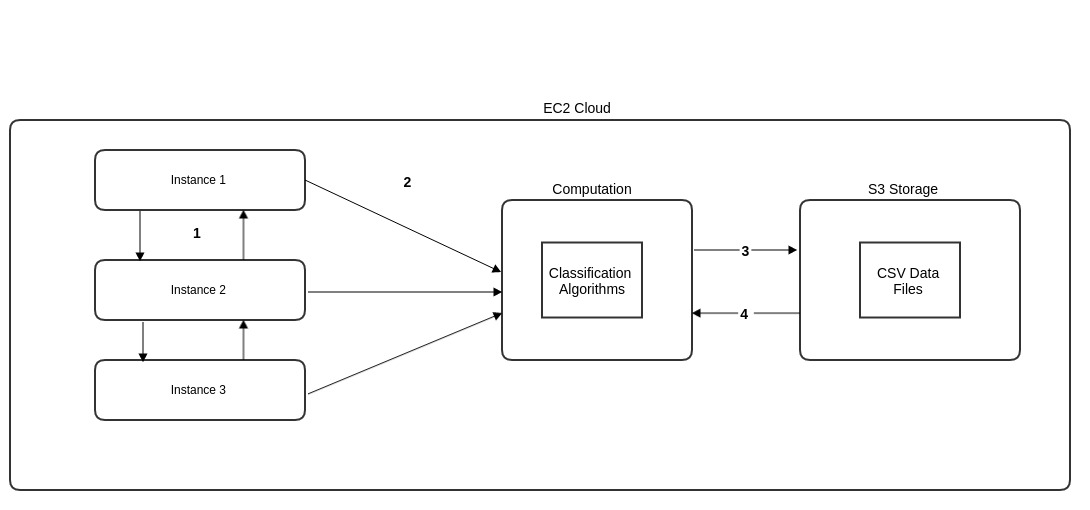
ObjID, u, g, r, i, z, class

Unclassified data from Photometric catalogue:

ObjID, u, g, r, i, z, class = null

**Conceptual Architecture**:

Distributed Cloud Architecture

1. Inter communication between the instances
2. Instance invokes algorithms
3. Algorithms fetches the data from the S3 Cloud Storage
4. Algorithms stores the result of the computation on the S3 storage

**Plan**:

First, we will set up S3 buckets for data storage in AWS EC2 and create multiple virtual instances to run distributed classification algorithms on the available data. The instances will communicate each other for any contingency and to operate smooth flow of computation across different nodes. We also would find if different combinations of the algorithms would give different levels of accuracy across nodes to see what gives the most accurate result to optimize our result. Finally, when the computation is completed the results are stored on to the S3 storage.

**Algorithmic Approach**:

Prior work has been done to develop an ensemble algorithm which is a combination of kNN and SVM algorithms. We will take this as a base and scale it up to cloud distributed environment.

We will design versions of multiple classification algorithms which can run on distributed cloud architectures. We will also develop an ensemble algorithm to efficiently identify quasars in a massive dataset of unclassified, photometrically defined objects.

We are planning to use distributed versions of kNN, SVM and ensemble algorithms to efficiently identify Quasars and Stars.

**Milestones**:

|  |  |  |
| --- | --- | --- |
| **Milestones** | **Start Data** | **End Date** |
| Upload data on S3 storage  Implementing algorithms as per single instance  Setting up multiple instances | 12th January | 1st February |
| Setting up distributed computing setup on cloud  Implementing Machine Learning Algorithms on distributed computing platform | 1st February | 15th February |
| Cross validation of classification models based on Cloud Platform | 15th February | 1st March |
| Optimization of the performance of the model | 1st Match | 15th March |
| Properly document the project's progress, make final refinements | 15th March | 31st March |
| Final Deployment of the Project | 1st April | 15th April |

**Technologies**:

* Amazon EC2 cloud platform
* Python
* Python Packages: Scikit-learn, dispy, boto
* Apache Spark MLlib
* Git

**Hardware Requirements**:

* Amazon EC2 Cloud