Dimensionality Reduction for Studying Diffuse Circumgalactic Medium

GSoC 2021

Table of Contents

Contact Information

- **Name:** Sandeep Saurav
- **Country:** India
- **Email:** sandeep.saurav97@gmail.com
- **Phone:** +91-9654783365
- **Public repository:** https://github.com/Sandeep10021
- **Other Contacts:**
    - https://www.linkedin.com/in/sandeepsaurav/

Detailed Project Idea

- **Project Title:** Dimensionality Reduction for Studying Diffuse Circumgalactic Medium
- **Project Task:** Implement machine learning-based dimensionality reduction models applicable to the quasar absorption spectra datasets.
- **Project Description:**

Introduction

The gas surrounding galaxies outside their disks or interstellar medium and inside their virial radii is known as the circumgalactic medium (CGM). In recent years this component of galaxies has assumed an important role in our understanding of galaxy evolution owing to rapid advances in observational access to this diffuse, nearly invisible material. Observations and simulations of this component of galaxies suggest that it is a multiphase medium characterized by rich dynamics and complex ionization states. The CGM is a source for a galaxy's star-forming fuel, the venue for galactic feedback and recycling, and perhaps the key regulator of the galactic gas supply. Observations from all redshifts and from across the electromagnetic spectrum indicate that CGM gas has a key role in galaxy evolution.

Radiation from distant quasars interacts with the material lying along its path before it reaches the observer. The signatures of these interactions can be discerned from the observed quasar spectrum. The evidence of the interactions is the reduction in the number of photons at wavelengths specific to the interacting material in the form of absorption lines, reddening of the spectrum, and the 2175 Å bump in the rest frame of the absorber. In some cases, emission from the intervening material is also seen. Quasar absorption lines were discovered soon after the discovery of the first quasar. Thus, quasar spectra are store houses of vast amounts of information about the interstellar medium in the intervening galaxies as well about the intergalactic medium (IGM).

Machine Learning in Astronomy

- Astronomy is rife with tasks demanding human labor
    - Source identification
    - Continuum fitting
    - Line identification

- Machine Learning
    - Can perform many of these tasks
    - Automagically, repeatedly, better!

- Astrophysics and ML

  For Example
    - Deep Learning of DLAs using CNN model
    - Studying Diffuse Circumgalactic Medium using DL model

Problem With Many Input Variables

The performance of machine learning algorithms can degrade with too many input variables. If your data is represented using rows and columns, such as in a spreadsheet, then the input variables are the columns that are fed as input to a model to predict the target variable. Input variables are also called as features.

We can consider the columns of data representing dimensions on an n-dimensional feature space and the rows of data as points in that space. This is a useful geometric interpretation of a dataset. Having a large number of dimensions in the feature space can mean that the volume of that space is very large, and in turn, the points that we have in that space (rows of data) often represent a small and non-representative sample. This can dramatically impact the performance of machine learning algorithms fit on data with many input features, generally referred to as the "**curse of dimensionality**". Therefore, it is often desirable to reduce the number of input features. This reduces the number of dimensions of the feature space, hence the name "**dimensionality reduction**".

Dimensionality Reduction

Dimensionality reduction refers to techniques for reducing the number of input variables in training data.

When dealing with high dimensional data, it is often useful to reduce the dimensionality by projecting the data to a lower dimensional subspace which captures the “essence” of the data. This is called dimensionality reduction. High-dimensionality might mean hundreds, thousands, or even millions of input variables. Fewer input dimensions often mean correspondingly fewer parameters or a simpler structure in the machine learning model, referred to as degrees of freedom. A model with too many degrees of freedom is likely to overfit the training dataset and therefore may not perform well on new data. It is desirable to have simple models that generalize well, and in turn, input data with few input variables. This is particularly true for linear models where the number of inputs and the degrees of freedom of the model are often closely related.The fundamental reason for the curse of dimensionality is that high-dimensional functions have the potential to be much more complicated than low-dimensional ones, and that those complications are harder to discern. The only way to beat the curse is to incorporate knowledge about the data that is correct.

Dimensionality reduction is a data preparation technique performed on data prior to modeling. It might be performed after data cleaning and data scaling and before training a predictive model. Dimensionality reduction yields a more compact, more easily interpretable representation of the target concept, focusing the user’s attention on the most relevant variables. As such, any dimensionality reduction performed on training data must also be performed on new data, such as a test dataset, validation dataset, and data when making a prediction with the final model.

Techniques for Dimensionality Reduction

Dimensionality Reduction can be done in two ways either through feature selection or through feature transformation. There are many techniques that can be used for dimensionality reduction.

- **Missing Values Ratio** - Data columns with too many missing values are unlikely to carry much useful information. Thus, data columns with a ratio of missing values greater than a given threshold can be removed. The higher the threshold, the more aggressive the reduction.
- **Low Variance Filter** - Similar to the previous technique, data columns with little changes in the data carry little information. Thus, all data columns with a variance lower than a given threshold can be removed. Notice that the variance depends on the column range, and therefore normalization is required before applying this technique.
- **High Correlation Filter** - Data columns with very similar trends are also likely to carry very similar information, and only one of them will suffice for classification. Here we calculate the Pearson product-moment correlation coefficient between numeric columns and the Pearson's chi-square value between nominal columns. For the final classification, we only retain one column of each pair of columns whose pairwise correlation exceeds a given threshold. Notice that correlation depends on the column range, and therefore, normalization is required before applying this technique.
- **Random Forests/Ensemble Trees** - Decision tree ensembles, often called random forests, are useful for column selection in addition to being effective classifiers. Here we generate a large and carefully constructed set of trees to predict the target classes and then use each column's usage statistics to find the most informative subset of columns. We generate a large set (2,000) of very shallow trees (two levels), and each tree is trained on a small fraction (three columns) of the total number of columns. If a column is often selected as the best split, it is very likely to be an informative

column that we should keep. For all columns, we calculate a score as the number of times that the column was selected for the split, divided by the number of times in which it was a candidate. The most predictive columns are those with the highest scores.

- **Principal Component Analysis (PCA)** - Principal component analysis (PCA) is a statistical procedure that orthogonally transforms the original *n* numeric dimensions of a dataset into a new set of *n* dimensions called principal components. As a result of the transformation, the first principal component has the largest possible variance; each succeeding principal component has the highest possible variance under the constraint that it is orthogonal to (i.e., uncorrelated with) the preceding principal components. Keeping only the first $m < n$ principal components reduces the data dimensionality while retaining most of the data information, i.e., variation in the data. Notice that the PCA transformation is sensitive to the relative scaling of the original columns, and therefore, the data need to be normalized before applying PCA. Also notice that the new coordinates (PCs) are not real, system-produced variables anymore. Applying PCA to your dataset loses its interpretability. If interpretability of the results is important for your analysis, PCA is not the transformation that you should apply.
- **Backward Feature Elimination** - In this technique, at a given iteration, the selected classification algorithm is trained on *n* input columns. Then we remove one input column at a time and train the same model on *n-1* columns. The input column whose removal has produced the smallest increase in the error rate is removed, leaving us with *n-1* input columns. The classification is then repeated using *n-2* columns, and so on. Each iteration *k* produces a model trained on *n-k* columns and an error rate *e(k).* By selecting the maximum tolerable error rate, we define the smallest number of columns necessary to reach that classification performance with the selected machine learning algorithm.

- **Forward Feature Construction** - This is the inverse process to backward feature elimination. We start with one column only, progressively adding one column at a time, i.e., the column that produces the highest increase in performance. Both algorithms, backward feature elimination and forward feature construction, are quite expensive in terms of time and computation. They are practical only when applied to a dataset with an already relatively low number of input columns.
- **Linear Discriminant Analysis (LDA) -** A number *m* of linear combinations (discriminant functions) of the *n* input features, with $m < n$, are produced to be uncorrelated and to maximize class separation. These discriminant functions become the new basis for the dataset. All numeric columns in the dataset are projected onto these linear discriminant functions, effectively moving the dataset from the *n*-dimensionality to the *m*-dimensionality. In order to apply the LDA technique for dimensionality reduction, the target column has to be selected first. The maximum number of reduced dimensions *m* is the number of classes in the target column minus one, or if smaller, the number of numeric columns in the data. Notice that linear discriminant analysis assumes that the target classes follow a multivariate normal distribution with the same variance but with a different mean for each class.
- **Autoencoder -** An autoencoder is a neural network, with as many *n* output units as input units, at least one hidden layer with *m* units where $m < n$, and trained with the backpropagation algorithm to reproduce the input vector onto the output layer. It reduces the numeric columns in the data by using the output of the hidden layer to represent the input vector. The first part of the autoencoder - from the input layer to the hidden layer of *m* units - is called the encoder. It compresses the *n* dimensions of the input dataset into an *m*-dimensional space. The second part of the autoencoder - from the hidden layer to the output layer - is known as the decoder. The decoder expands the data vector from an *m*-dimensional

space into the original *n*-dimensional dataset and brings the data back to their original values.

- **t-distributed Stochastic Neighbor Embedding (t-SNE)** - This technique reduces the *n* numeric columns in the dataset to fewer dimensions *m (m < n)* based on nonlinear local relationships among the data points. Specifically, it models each high-dimensional object by a two- or three-dimensional point in such a way that similar objects are modeled by nearby points and dissimilar objects are modeled by distant points in the new lower dimensional space.

  In the first step, the data points are modeled through a multivariate normal distribution of the numeric columns. In the second step, this distribution is replaced by a lower dimensional t-distribution, which follows the original multivariate normal distribution as closely as possible. The t-distribution gives the probability of picking another point in the dataset as a neighbor to the current point in the lower dimensional space. The perplexity parameter controls the density of the data as the “effective number of neighbors for any point.” The greater the value of the perplexity, the more global structure is considered in the data. The t-SNE technique works only on the current dataset. It is not possible to export the model to apply it to new data.

Timeline

I have planned to spend 20-30 hours a week on this project so that within 10 weeks I can come up with a tested and working code. The following is the breakdown of the planning:

1. **Before 17th May**

   I will go through all the basics required for the project which includes understanding the physics behind it and will brush up all my concepts of python and machine learning.

2. **Community bonding period**

   17th May –7th June

   Understand the work going on in the community. Getting in contact with the mentors and creating a bond with them. Understand in detail the project by getting details and more insight into work. Deciding the perfect medium of contact and creating and maintaining a workflow sheet of works that need to be done. Understanding their expectations from my side and planning milestones and working with them accordingly.

3. **Coding Phase 1**

   7st June -12th July

   I. Week 1 (7th June - 14th June)
      - Analyse metadata dataset and develop code to extract essential features from dataset

   II. Week 2 (14th June - 21st June)
      - Develop code to clean and normalize extracted features

   III. Week 3 (21st June - 28th June)
      - Develop Dimensionality Reduction code for some of techniques

   IV. Week 4 (28th June - 12th July)
      - Test the Accuracy with different models, debug and document code

- Prepare work for Phase 1 submission along with a brief Phase 1 report

4. **Evaluation Phase 1**

   12th June – 16th July

   First evaluation period. Pull request/Submit the code for mentor and community evaluation. Discussion of result with mentor and proposed forward work.

5. **Coding Phase 2**

   16th July-16th August

   I. Week 5 (16th July - 23rd July)
      - Develop code for Dimensionality reduction with more techniques

   II. Week 6 (23rd July - 30th July)
      - Test Accuracy with different models and Compare all the techniques and choose the best.

   III. Week 7 (30th July - 6th August)
      - Finish implementing the framework with the following features
      - Retrain model by retrieving it and training it with new batch of metadata with known classes
      - Query test results for batch of record metadata

   IV. Week 8 (6th August - 16th August)
      - Test, debug code
      - Write pending documentation
      - Prepare Final Phase submission along with a detailed final phase report

6. **Final Evaluation**

   16th Aug – 23rd Aug

   - Final evaluation period. Mentors evaluate my final work product and documentation

Motivation

The reason for choosing for ML4SCI is my passion for science and enthusiasm for Open Source. ML4SCI is a combination of both science and open source and thus appeals to me a lot. The other factor which motivated me towards ML4SCI is my passion for Machine Learning and Python. I have been working with Python for the past 6 months and have taken up machine learning as my career objective. I have been learning many concepts of Machine Learning and have also applied some using Python. My inclination towards Machine Learning motivated me to take up 'Dimensionality Reduction for Studying Diffuse Circumgalactic Medium' project under ML4SCI. I am very enthusiastic about this technical challenge as it involves many dimensions in itself like, data analysis, language processing, implementation of ML concepts and frameworks. For the GSOC period I propose to Implement machine learning-based dimensionality reduction models applicable to the quasar absorption spectra datasets.

Product Outreach

My aim is to amalgamate my interest in Machine Learning, Image Processing, GIS and Remote Sensing with my skills in Software Development, Programming and Web Technologies and apply it in the field of Geoinformatics.

Being a part of such a helpful community is a great opportunity in itself and I would love to collaborate with others throughout my project timeline and even after that, as this is the true essence of Open Source culture. I would constantly contribute to the community by helping various other Open Source enthusiasts who wish to contribute by helping them out with their problems and at the same time encouraging them for participation. Also I would be submitting various reports on the work done by me on a regular basis.

Under the constant guidance of my mentors I would like to make the best out of my summers and do my part towards making this organization even better. Also I would be very much pleased to keep on contributing to this organization even after the program by being a mentor for Google Summer of Code and various other events organized by the organization so as to help various out there in starting there open source journey as

I think these projects have a huge potential to make a positive impact in the lives of the people even if it manages to touch a small user-base. It would be a matter of great pride for me if I can help add meaning to these cutting edge technologies by introducing them to one's life and help humanity.

Studies

- M.Tech
  - ❑ **Institute**: Indian Institute of Technology, Bombay (IITB), India
  - ❑ **Specialisation**: Geoinformatics and Natural Resource Engineering offered by Centre of Studies in Resources Engineering(established by ISRO)
  - ❑ **Expected Graduation Date:** 2022
  - ❑ **Courses:**
    - o Principles of Remote Sensing
    - o Principles of Geographic Information Systems
    - o Principles of Satellite Image Processing
    - o Advanced Methods in Satellite Image Processing
    - o Machine Learning
    - o Advances in Geospatial Standards,Interoperability and Knowledge Discovery
    - o Remote Sensing & GIS Applications to Mineral and Hydrocarbon Exploration
- B.Tech
  - ❑ **College:** Maharaja Agrasen Institute of Technology, Delhi
  - ❑ **Specialisation:** Electronics and Communication Engineering
  - ❑ **Graduation Date:** 2019

Current Experience and Relevant Projects

- Currently I am a **Teaching Assistant** in Centre of Studies in Resources Engineering, IIT Bombay. My role and responsibility involves assisting Professors in their research/project work related to machine learning, remote sensing, geo-informatics, geo-spatial data and satellite imagery. In the following year I will be allotted to mentor and guide the new batch of the department.
- In my "Geographic Information Systems" course project on **PgRouting**, I have created a fully working android application. The project aims at creating an android application for truck services and other heavy vehicles such that the customer should be able to hire a heavy vehicle nearby and track the goods to the destination. It is a fully functional GIS and GPS based android application using Google Maps for Android SDK, Google Places API, Google Directions API, Firebase Database, Paytm all in one SDK.
- In my "Advances in Geospatial Standards, Interoperability and Knowledge Discovery" course project on **OGC-Interoperability**, I have created a web client where I have integrated the OGC standard web services like **WMS, WFS** and **WCS**. I used **GeoServer** for fetching the APIs. Based on the capabilities of the web services I have developed an application which allows clients to generate requests for maps, features or coverage. Then I overlaid the requested layer on top of the **OpenLayers** Maps and displayed the map along with the attributes table to the user with an interactive UI/UX.
- In my "Machine Learning" course I have implemented a model for predicting **Predicting flight delays for the airport authorities** and analyzed flight delays for the airport authorities, to assist them in revamping their operations using Logistic Regression with model accuracy of 94%.
- I have implemented a linear regression model from scratch using numpy with **Principal Component Analysis** to reduce the data dimensionality, as a part of a Machine Learning course.

- ❏ Currently I am working on "Implement **Histogram of Oriented Gradients** method for extracting all instances of a particular object in the (high resolution) input image" as my Advanced Methods in Satellite Image Processing Project.
- ❏ Currently I am also working on **Natural Language Processing** based application for my Machine Learning project assignment.
- ❏ Currently I am also working on "**Diagnosis of Diabetic Retinopathy Using Deep Neural Networks"** as my Machine Learning Project.

Area of Interest

- ❏ Machine Learning
- ❏ Satellite image processing
- ❏ Geospatial Data Interoperability
- ❏ Geographic Information System (GIS)
- ❏ Remote Sensing

Programming and GIS

1. **Computing experience:** Windows , LINUX/UNIX
2. **GIS experience as a user:** QGIS, ESRI ArcMap desktop, ArcGIS online, Android Studio, SNAP, ENVI
3. **Programming Experience:**
    - **Professional working proficiency -** Python, C, C++, Java, Javascript, CSS, Html, Xml
    - **Database and Query Languages -** PostGreSQL, PostGIS, SQL, MySQL, Google data store, Real-time Firestore

Commitments

- ### How many hours will you work per week on your GSoC project?

  I can easily devote 20-30 hours of quality coding time a week. Except that, I am constantly learning from courses in my spare time and that would definitely help me further during GSoC.

- ### Other Commitments

  I have no other commitments during GSoC timeline. So that my aim to give GSoC maximum productivity is not hampered and therefore this won't hinder my performance or suggested work timeline in any way.

GSoC Participation

- ### Do you understand this is a serious commitment, equivalent to a full-time paid summer internship or summer job?

  Yes, I understand this and am willing to give my full commitment towards GSoC.

- ### Do you have any known time conflicts during the official coding period?

  No, I have no time conflicts during the official coding period of GSoC.

- ### Would the application contribute to your studies, if yes, how?

  Yes, I am pursuing my Masters in Geoinformatics Engineering and one of my core subjects is Machine Learning. Thus surely this project will help me improve my practical knowledge of the subject and polish my coding skills. This project will help me learn to work in a team. The Dimensionality Reduction Technique that I propose to create here can be used in some of my future projects, which will only enhance my project's capabilities. This would be my first contribution to the open source community.

- ## Have you ever participated in GSoC before?

  No, this is my first GSoC participation and I am very excited about it. I have submitted only one proposal for GSoC 2021.

Student Evaluation Test Link -

- https://github.com/Sandeep10021/ML4SCI_GSoC