Estimating Photometric Redshifts using Machine Learning Techniques
PH 426 Project Report

Abhay Singh, Kumar Ayush, Piyush Bhatore, Sheshansh Agrawal

April 24, 2017

Abstract

We combine the spectroscopic data from PRIMUS and VIPERS survey for high redshift galaxies, and obtain photometric data from SDSS for these galaxies to create a dataset which can be used with machine learning methods to estimate photometric redshifts. This is presented in contrast to using the spectroscopic redshifts from SDSS itself, which is known to be accurate limited to lower redshift regimes. The performance of these two methods are compared for two methods: Artifical Neural Networks and K-Nearest Neighbours

Contents

1 Week 14 Jan-20 Jan

1.1 Summary of References

1.1.1 Csabail[1]

The paper is referenced with SDSS DR12 resources and it describes a new way to store the SpecZ data. Section 4, however shows how this novel data structure can be optimally used to calculate photometric redshifts using a k-clustering algorithm.

We first describe the data structure they use. It is essentially indexing your data in a multi-dimensional space, hence multi-indexing as in the title of the paper. They use a space partitioning technique known as k-d trees, where d is the dimension. When $d = 2$, these are known as quad trees, which will be known to computational physicists familiar with N body simulations, and algorithms such as fast multipole method (FMM). [1] choose $d = 5$ for every magnitude band (SDSS u,b,v,i,r).

Now, to estimate photometric redshifts, they use a k-nearest neighbour approach, where the unknown redshift is estimated with a low-order polynomial fit over k-nearest neighbours of the object in the color space. The multi-indexed data structure allows them to quickly determine these neighbours increasing the efficiency of k-nearest method by a huge margin.

They claim that their method helps reduce average error in photometric redshifts by 50%. Their algorithm is pasted below for reference

```
foreach (Galaxy g in UnknownSet)
{
    neighbors = NearestNeighbors(g, ReferenceSet)
    polynomCoeffs = FitPolynomial(neighbors.Colors,
                        neighbors.Redshift)
    g.Redshift = Estimate(g.colors,polynomCoeffs)
}
```

1.1.2 Cavouti[2]

This is much more recent than the previous paper and describes estimation of photometric redshifts with the KiDS ESO DR2 galaxies. Since this is not as popular as SDSS, we will briefly describe their data schema for photometric data, and then their machine learning approach, and then their results.

The KiDS ESO photometric data is stored in 4 bands (unlike the 5 bands of SDSS), which are u, g, r, i. The source positional and shape parameters are based on only the r-band. DR2 has 18 million cataloged objects. Counting only the galaxies as we want saves us 10 million, and then filtering based on the IMAFLAGS_ISO flag, we are left with 6 million objects. This flag determines if the source was observed in an unmasked region. They claim that their algorithm successfully works on estimating redshifts for about 1.14 million objects.

They use Multi Layer Perceptron with Quasi Newton Algorithm (MLPQNA)[3]. To be honest, we are not very clear on what they mean by using "quasi newton" learning rule, but their approach seems to be a simple feed-forward neural network with a non-linear activation function at each node.

They create training and test samples with relative sizes of 60% and 40%. The results are evaluated using the following measure

$$\Delta z = \frac{z_{spec} - z_{phot}}{1 + z_{spec}}$$

The mean and standard deviation of Δz gives an indication of the quality of the experiment. With their most restricted input data set, they report values of $9.9e-4$ and 0.0305 for the mean and stdev respectively.

2 Week 21 Jan-27 Jan

2.1 Importance

Looking at higher redshifts is equivalent to looking back in time. They are important to studies in cosmology, to help improve our knowledge about the origin of our universe. Numerous analytical studies in cosmology will benefit from more accurate redshifts, especially in the $z > 1$ region.

Spectroscopic redshifts are not feasible on a large scale because of the time required to obtain spectroscopic data. More than 70% of the SDSS time was spent on this. [1]. On the other hand, obtaining photometric data is fast and hence less expensive to the astronomical community. If one can map the spectroscopic redshifts into the photometric space, we can determine redshifts for objects just using their photometric data. Estimating these redshifts is thus, a task of utmost importance to the astronomy and astrophysics community.

2.2 Challenges and Gameplan

Methods like linear regression and k-nearest neighbour work well for $z < 1$, but suffer from a drastic increase in error for higher redshifts, which are of prime importance for cosmological studies. People have tried more complex methods such as artificial neural networks(ANN)[3] and support vector machines(SVM) and have succeeded to improve the estimates. However, we have a long way to go where we can push the accuracy to a satisfactory limit as required by cosmologists.

The properties(such as color-color diagram) of the photometric data is significantly different for higher redshifts as far as machine learning applications are concerned. This raises issues with incorporating the large data set with a single model. Additionally, often some stars are misclassified as galaxies at higher redshifts which distracts the learning algorithm as stars have completely different emissive properties as compared to galaxies.

Moreover, the accuracy of photometric data also falls at larger redshifts which should be taken into account for such models.

These couple of challenges are what we desire to solve in our project. We will proceed with SDSS DR12 data, filter the ”good” objects, and try various machine learning techniques on it. A part of our group will try to observe patterns in the data which might be helpful in classifying the data somehow beneficial for our learning model. In the modern world however, a basic prototype of feed-forward neural networks as in Brescia[3] can be implemented with packages like *keras* and *TensorFlow* in under 15 minutes. We thus want to try more complex neural networks on the data and document the results.

2.3 What we know

Empirically, we know some properties of color-color diagram for galaxies which can be useful to classify them, and use different model parameters for each class.

There is an approximate linear dependence on photometric magnitudes along with redshifts due to an increase in distance, but relying on this often leads to outliers due to a heavy dependence on the emission properties of the object, and intergalactic extinction. The emission properties can be theoretically modelled very well, but as explained in the previous section, misclassification of many stars as galaxies creates problems. It is indeed a challenge to weed out such objects during data preprocessing.

3 Week 28 Jan-3 Feb

3.1 Open Questions

1. **Are photometric redshifts good enough?** The first question to ask is, if the estimates of redshifts using photometric measurements provide a satisfactory and reliable solution to the spectroscopic measurements being expensive. These photometric redshifts are eventually used, among many other applications, to predict cosmological models, which are crucial to our understanding of fundamental physics. It is therefore important to have an idea on how tremendously inaccuracies in redshift estimations affect our interpretations of fundamental physics.

   There is no good answer to this. Astronomers have to rely on photometric redshifts, because at the moment, that's the best we can do.

2. **Why do people estimate the redshifts independently from other physical properties, and is it fundamentally wrong to do so?** [17] People often use empirical redshift estimates and then template spectra for type determination. Empirical redshift estimates which are a superset of machine learning techniques do not use any information about the physical properties.

   It sounds like a good idea to use template fitting techniques in combination with empirical techniques to ensure effects of physical properties are accounted, but empirical techniques have performed better, doubting the explicit introduction of physical correlations.

3. **What is the cause for redshift?** If two different stellar objects are observed at the same distance and their Red-Shift was due to recession, then their absorption lines should be equal, Hal Arp, and other astronomers have found that there are some stellar objects that share common visual detail and yet have completely different Red-Shifts.

   This is entirely a question of philosophy of science, and we do not intend to make an attempt to answer it here at this stage.

4. **What is the luminosity function and morphological distribution of galaxies above redshift 5?** [18] Since objects with higher redshifts are fainter, we have scanty knowledge about the physical properties about these making our task harder by forcing us to extrapolate our models where the fundamental assumptions behind them might also break.

5. **Can we ensure that our machine learning models really learn physical effects from training data, and can we put a measure upon it?** Generally speaking, machine learning methods do not know about the underlying physics of the problem: flux measurements arise from observing a red-shifted galaxy spectrum through known photometric band-passes. They will partially learn those effects from the training data, but they are not required to, which limits their robustness in regimes critical to cosmological applications. [20]

6. **Why are the current implementations of estimating redshift so different?** [17] Empirical methods map the relation of the observed and desired properties using a training set. Template-fitting techniques rely on prior knowledge encoded in the model's spectral energy distributions (SEDs) that can be matched to observations.

   Is there any scientific reason that these implementations have only diverged and there have only been few attempts to bring them together over the last decade, or this is for purely heuristic reasons?

7. **Does redshift actually depend on spatial information, and can existing machine learning methods benefit from it?** [4] A third class of methods for estimating photometric redshifts exists, and is sometimes referred to as "clustering redshifts" It exploits spatial information and the proximity of galaxies in real space (sky position and redshift) to constrain the redshifts of galaxies. Since it does not directly uses flux information we will not discuss it further, and we will assume that any flux-based photo-z method could be improved by adding spatial information.

8. **Can errors in photometric redshifts be estimated reliably?** Using photometric redshifts for cosmological applications would not be as huge a problem as we have reflected in some questions here if we can reliably estimate the size of errors in our estimates. However, empirical methods which entirely lack any physical basis have no way to provide a measure for the error.

   However, people have come up with innovative methods [23] to determine PDFs around the estimates.

9. **Can we estimate overconfidence in a particular method of photometric redshift estimation?**[22] In general the models do not analyze the PDF of the redshift given and only the most probable values. But is it possible to estimate the overconfidence over the redshift the model produces?

10. **The physics would be different for the higher redshift galaxies as they are older. Can our model for redshifts adapt to this problem?** The older galaxies have different compositions and shapes. It should be possible to take care of this factor and make the final redshift take care of this factor.

11. **What should be the ideal bin size in the parameter space while estimating redshifts?** [**21**] Larger bins cause loss of data and smaller bins mean more computation time. So, instead of randomly fixing the bin size in the parameters of the model can this be estimated based on some other characteristics of data?

12. **Can we precisely estimate a bias for the training data set using the astrophysical concepts involved?**
    Estimating a systematical bias by some methods is a very tough job as we don't have enough data to calibrate the training data given. Can the data be calibrated based on physical relations?

13. **Can we estimate and remove outliers from the training data set effectively making the use of the underlying physics?**
    Data cleaning is one the big challenges for any machine learning model. But here due to the astrophysical significance of the data we obtain, efforts have been made to find and eliminate outliers due to a particular phenomenon.

14. **Would the redshift dependence on wavelength cause problem in estimating higher redshifts (>1). How can it be solved?**
    Redshift in general is not independent of wavelength and it affects the training data we use in our model. So, making a model which takes care of these effects is quite a challenge.

15. **Can we distinguish faint objects based on their red shifts?**
    Classifying faint objects poses a problem as we want our training data to be as similar to the target as possible. To make our model precise and ensure that there is not a shortage of training data, we need a way to classify faint objects based on easily observable parameters.

16. **What is the true distribution of the rest equivalent width of Lyα as a function of redshift and luminosity? What determines this distribution?** [**18**] Some techniques for finding very high redshift galaxies rely only on colors while others rely on Lyα emission. The method of detecting galaxies creates a bias in the data which is reflected in our redshift estimates. It is important to understand the physical causes behind such bias, and this is one such example.

17. **Can there be a bias in estimation of photometric redshift due to non-uniform distribution of galaxies w.r.t redshifts?**

    The distribution of number of galaxies w.r.t redshifts is not uniform and it follows a certain dependence based on how the universe evolved. But this bias has to be taken care of in the model.

18. **Can the bias further be used to refine a cosmological model?**
    The training data contains a reflection of actually how the universe evolved and so the final model can be used to refine the cosmological model used to photometrically estimate the redshifts in first place.

19. **Do the magnitudes even contain data that can be used to estimate redshifts?**
    Though photometric estimation of redshifts has become a famous and widely accepted way. But why would flux data of stars even contain the information needed to estimate the redshifts and how can we explain the systematic outliers we get through this approach.

20. **Do we have an adequate set of spectroscopic templates to use in calibrating the photometric methods like Bruzual-Charlot model, model-free empirical fits?** [18] This is exactly similar to asking if we have enough training data to get the most satisfactory estimate on our redshifts when using machine learning techniques.

    A study of constraints on the PDF associated with the redshift estimates with respect to the size of calibration dataset would be helpful to answer this.

4 Week 4 Feb-10 Feb

We have decided to pursue the following three questions during our project:

1. **Does redshift actually depend on spatial information, and can existing machine learning methods benefit from it?** [4]

2. **The physics would be different for the higher redshift galaxies as they are older. Can our model for redshifts adapt to this problem?**

3. **Can we ensure that our machine learning models really learn physical effects from training data, and can we put a measure upon it?**

4.1 Using Spatial Information as a part of Feature Vector

Menard et. al. [4] make use of the angular clustering of galaxies to estimate redshifts with a reference or a set of reference populations for which redshifts are well determined. Even though such a technique is currently not being applied, the underlying idea has been discussed for several decades and in a few cases applications to data have pointed out some of its potential.

The implementation is discussed in detail in the paper. However, our approach is different. Traditionally, the feature set for machine learning methods in photometric redshift estimation is the set of colors obtained from the photometric magnitudes in the available bands. We intend to incorporate the spatial information as another parameter in the feature vector and then see if the variety of machine learning approaches show an improvement in the results.

5 Week 5 11 Feb - 17 Feb

5.1 Declaring Project Plan

Started project plan pdf. Main details from project plan are here The monthly schedule for our further project is
February(1 week) : Learning programming tools like keras
March(4 weeks) : Dataset cleaning, analyzing the data using the standard techniques and then repeating experiments including spacial parameters as an input.
April(3 weeks) : Comparing the results with spacial parameters with original results and then moving on to find the role of age of the galaxies(related to redshift) with the photometric redshifts.

6 Week 6 25 Feb - 3 Mar

6.1 Learning Keras

For this week, We learned the Keras, a python library, from its documentation https://keras.io/ .It is a high-level neural networks library, useful on our project. Its special type of models like Sequential(a linear stack of layers) will be the main data structure of our project.

7 Week 7 4 Mar - 10 Mar

7.1 Discussion with Mentor

Before we actually get down to do stuff, we had a discussion with our mentor Prof. Yogesh Wadadekar. The keypoints of the discussion are summarized as:

- It is a good idea to use spatial information of the galaxies as an addition to the input vector in machine learning approaches towards the problem. However, one has to be clear about the dimension of the spatial information that is being used i.e. we can have an estimate on the 3-dimensional separations between the galaxies and use that as an input feature or simply use the 2-dimensional angular separation on the sky. Both of these approaches will have varying effects, elaborated a bit below:
  - **3-dimensional distance**: If we suppose that we had means to accurately determine this without using redshift information for galaxies in our complete dataset, then this would be the most accurate measure of the clustering information and would be a helpful addition to the input vector. However, our current observation capabilities limit this to simply unfeasible.
  - **2-dimensional angular separation**: This is easy to determine, and captures some of the spatial clustering information that we are interested in. However, we were planning to use SDSS DR12 which covers a wide $14,555$sq. degrees of angular area. Prof. Yogesh pointed that SDSS suffers from a Malmquist bias and completely misses out on important data from high redshift galaxies which should be our primary interest.
- Prof. Yogesh suggests an alternative exploration path which might give us interesting results. He advised us to use two surveys namely VIPERS and PRIMUS which specialize in higher redshift regions as compared to SDSS. The strategy would be to use the galaxies in the common survey regions of VIPERS and PRIMUS, use the SDSS photometric data on them, and try to train an ANN using the dataset obtained.
  We were planning to include spatial information nevertheless when Prof. Yogesh pointed out that the area covered by these surveys is very less to give a good sample of the complete angular distribution of galaxies at various redshifts.

We chose to try this alternative path, i.e. matching SDSS with higher redshift surveys to obtain the machine learning dataset, as it had not been reported in literature within our search radar.

8 Week 8 11 Mar - 18 Mar

In this week, we read about the surveys PRIMUS and VIPERS. The following notes the relevant features of the two surveys.

8.1 VIPERS[24]

The VIMOS Public Extragalactic Redshift Survey (VIPERS) is an ongoing Large Programme to map in detail the large-scale distribution of galaxies at $0.5 < z < 1.2$. With a combination of volume and sampling

density that is unique for these redshifts, it focuses on measuring galaxy clustering and related cosmological quantities within the grand challenge of understanding the origin of cosmic acceleration. Moreover, VIPERS has been designed to also guarantee a broader legacy, allowing detailed investigations of the properties and evolutionary trends of z 1 galaxies. The survey strategy exploits the specific advantages of VIMOS, aiming at a final sample of nearly 100,000 galaxy redshifts to $i_{AB} = 22.5$, which represents the largest redshift survey ever performed with ESO telescopes.
A survey like the SDSS, for example, based on one million redshifts, has been able to measure to exquisite precision global galaxy population trends involving properties such as luminosities, stellar masses, colours and structural parameters. In more recent years, deeper redshift surveys over areas of 1-2 square degrees focused on exploring how this detailed picture emerged from the distant past. At the end of past decade it became therefore clear to us that a new step in deep redshift surveys was needed, in the direction of building a sample at z 1 with volume and statistics comparable to those of the available surveys of the local Universe. VIPERS was conceived to fill this gap by exploiting the unique capabilities of VIMOS. The survey will measure redshifts for 105 galaxies at $0.5 < z < 1.2$, covering an unprecedented volume. Its goals are to accurately and robustly measure galaxy clustering, the growth of structure and galaxy properties at an epoch when the Universe was about half its current age.
We use the PDR-2, i.e. the Public Data Release 2 from the survey. The survey area is split into two, called W1 and W4. Out of these, W1 is of our interest, as it matches with one of the PRIMUS filed that we're interested in. Roughly, it ranges from RA $30 - 38$degrees and δ -6 to δ -3. The target sample includes all galaxies with $i_{AB} < 22.5$, limited to having $z > 0.5$ through a robust ugri colour pre-selection.

8.2 PRIMUS[25]

Deep imaging surveys have covered as much volume at $z = 1$ as the SDSS has at $z = 0.1$. Much of this imaging is panchromatic, with UV and mid-IR data providing critical information about star-formation and stellar mass. The Spitzer SWIRE and GTO surveys and GALEX Deep Imaging Survey are imaging 60 square degrees to depths appropriate for $z = 1$ galaxies. However, these surveys lack redshifts.
The goal of the PRIsm MUlti-object Survey (PRIMUS) is to provide redshifts for a dense set of faint galaxies in the 10deg^2 of deep imaging accessible from Magellan that has panchromatic coverage. With a sample of $> 100,000$ PRIMIUS redshifts, it aims to quantify the clustering and multivariate properties of galaxies at $z = 0.2 - 1$ with similar precision as SDSS and 2dF, allowing detailed quantification of the evolution of the masses, luminosities, star formation rates, and clustering of galaxies with time.
The PRIMUS strategy is to acquire low resolution spectroscopy using a high-throughput prism behind an IMACS slit mask.
There are about 7 PRIMUS fields, depending upon the payload that collected the data or the target region. We are interested in the XMM-LSS field which centers around the same set of RA-Dec, $(-38, -6) to (-30, -4)$ as the VIPERS W1 field.

9 Week 9 19 Mar - 25 Mar

In this week, we created our dataset using VIPERS, PRIMUS and SDSS. The procedure is detailed below:

9.1 Merging VIPERS and PRIMUS

It is easy to obtain the data from VIPERS DR2 and PRIMUS in .fits format. The following python code extracts RA, Dec and redshift information from the two surveys and collects them into a single nd-array.

```
#!/usr/bin/env python
from astropy.io import fits
import numpy as np

hdulist_PRIMUS = fits.open('data/PRIMUS_2013_zcat_v1.fits');
```

```
data_PRIMUS = hdulist_PRIMUS[1].data
data_PRIMUS = np.column_stack((data_PRIMUS[’RA’],data_PRIMUS[’DEC’],
        data_PRIMUS[’Z’],data_PRIMUS[’FIELD’]))
data_PRIMUS = np.array(filter(lambda x:x[3].strip()==’xmm’,data_PRIMUS))[:,:3]
data_PRIMUS =  np.array(map(lambda x:[float(x[0]),
        float(x[1]),float(x[2])],data_PRIMUS))

hdulist_VIPERS = fits.open(’data/VIPERS_W1_SPECTRO_PDR2.fits’)
data_VIPERS = hdulist_VIPERS[1].data
data_VIPERS = np.column_stack((data_VIPERS[’alpha’],
        data_VIPERS[’delta’],data_VIPERS[’zspec’]))

from astropy import units as u
from astropy.coordinates import SkyCoord
PRIMUS_catalog = SkyCoord(ra=data_PRIMUS[:,0]*u.degree,
dec=data_PRIMUS[:,1]*u.degree)
VIPERS_catalog = SkyCoord(ra=data_VIPERS[:,0]*u.degree,
dec=data_VIPERS[:,1]*u.degree)

idx, d2d, d3d = PRIMUS_catalog.match_to_catalog_sky(VIPERS_catalog)
feasible_indices = np.array(map(lambda x:x[0],
        filter(lambda x:x[1].value>1e-3,zip(idx,d2d))))
data_VIPERS = data_VIPERS[feasible_indices]

data_HZ = np.vstack((data_PRIMUS,data_VIPERS))
```

9.2 Matching with SDSS

About SDSS DR13[26]

Data Release 13 is the first data release of the fourth phase of the Sloan Digital Sky Survey. It includes SDSS data taken through June 25, 2015, and encompasses more than one-third of the entire celestial sphere. It claims to cover a total of 2,401,952 unique galaxies.

The SDSS Data

We use the following query on Casjobs (SDSS Query Portal) to extract SDSS Photometric Data:

```
SELECT TOP 1000000 ra, dec, psfMag_u, psfMag_g, psfMag_r, psfMag_i, psfMag_z
INTO mydb.ph426_4 FROM DR13.PhotoObjAll
WHERE type=3 AND (ra BETWEEN 30.1892613 AND 38.802245)
AND (dec BETWEEN -5.9800618 AND -4.1715397)
AND psfMag_u IS NOT NULL
AND  psfMag_u>-5 AND psfMag_g>-5
AND psfMag_r>-5 AND psfMag_i>-5
AND psfMag_z>-5;
```

In addition to the above, we also extract SDSS Spectroscopic Data, in order to compare the accuracy of training on our merged dataset versus a pure SDSS dataset. We use the query

```
SELECT TOP 1000000 ra, dec, z, zErr,
psfMag_u, psfMag_g, psfMag_r, psfMag_i, psfMag_z
INTO mydb.ph426_3 FROM DR13.SpecPhotoAll
```

```
WHERE class = 'GALAXY' AND (ra BETWEEN 30.1892613 AND 38.802245)
AND (dec BETWEEN -5.9800618 AND -4.1715397)
AND zErr != 0 AND psfMag_u IS NOT NULL
AND z<1 AND z>0 AND zErr<0.01
AND psfMag_u>-5 AND psfMag_g>-5
AND psfMag_r>-5 AND psfMag_i>-5
AND psfMag_z>-5;
```

The following code added to previous python code matches the previous collected dataset with SDSS dataset and gives us our input dataset:

```
hdulist_SDSS = fits.open('data/SDSS_DR13_PhotoObjAll.fits')
data_SDSS = hdulist_SDSS[1].data
data_SDSS = np.column_stack((data_SDSS['ra'],data_SDSS['dec'],
data_SDSS['psfMag_u'],data_SDSS['psfMag_g'],data_SDSS['psfMag_r'],data_SDSS['psfMag_i'

PRIMUS_and_VIPERS_catalog = SkyCoord(ra=data_HZ[:,0]*u.degree, dec=data_HZ[:,1]*u.degr
SDSS_catalog = SkyCoord(ra=data_SDSS[:,0]*u.degree, dec=data_SDSS[:,1]*u.degree)

idx, d2d, d3d = PRIMUS_and_VIPERS_catalog.match_to_catalog_sky(SDSS_catalog)

feasible_indices = np.array(map(lambda x:(x[0],x[1]),
filter(lambda x:x[2].value<1e-3,zip(range(len(idx)),idx,d2d)))
# There are some duplicates which we keep, about 5% of data

data = np.column_stack((data_SDSS[feasible_indices[:,1]][:,2:7],
data_HZ[feasible_indices[:,0]][:,2]))

#pick only if redshift is greater than 0.0
data = data[data[:,5]>0.0,:]
```

There are some minor comments on the code which we find notable

1. We set the threshold for considering two sources as identical while matching to $1e-3$. Setting it lower would shrink our dataset a lot, while setting it higher would have more duplicates, which will bias the training. We have no good scientific reason for this value, and it is essentially arbitrary.

2. The final ‘data’ has 6 columns. The first five are the magnitudes reported in SDSS by fitting PSF on the source image. This is not always the best measure of an extended object’s magnitude, but this may also work in our favor since it captures some information about the source size. The real reason of selecting this set of magnitudes among many different types reported in the catalog is that it is most ‘complete’. Others have a lot of points set to null.

3. The sixth column is the spectroscopic redshift as measured by VIPERS or PRIMUS for the same source. We emphasize this because this is the scientific novelty of our project. We are comparing the accuracy of a model trained with a dataset where redshifts are taken from surveys specializing in high redshift region versus a dataset with SDSS redshift measurements. The test-data is same, while the training data is different. However, the training-data comes from the same source as test-data in the second method. We are not sure if this has any important implications.

10 Week 10 26 Mar - 1 Apr

I now describe the different machine learning methods applied to the two datasets and then we compare their results.

10.1 K nearest neighbors

We estimate the redshift of a data point based on the weighted mean of the K nearest neighbors of the point in the input space, weighted by their respective distances. The following code performs this:

```
from sklearn import neighbors
from sklearn.metrics import mean_squared_error

def knn_regression(K, training_data, labels,
        test_data, weights='distance'):

    knn = neighbors.KNeighborsRegressor(K, weights=weights)
    output = knn.fit(training_data, labels).predict(test_data)
    return output

output = knn_regression(21, X[:0.8*len(X)], Y[:0.8*len(Y)], X[0.8*len(X):])
```

We use 80% of the data as training while the remaining data as the test set.

10.2 Sequential Neural Networks

We train a ANN to get the output. Take a look at the following code. A detailed explanation is given ahead.

```
#normalizing stuff
minx, maxx = np.min(X), np.max(X)
miny, maxy = np.min(Y), np.max(Y)

X = (X-minx)/(maxx-minx)
Y = (Y-miny)/(maxy-miny)

model = Sequential()
model.add(Dense(100, input_dim=5, activation='linear'))
model.add(Dense(30, activation='linear'))
model.add(Dropout(0.1))
model.add(Dense(1, activation='sigmoid'))
model.compile(loss='mean_squared_error', optimizer=RMSprop(), metrics=['accuracy'])

#fit model
model.fit(X, Y, batch_size=10000, nb_epoch=100,verbose=0)

#get test data
hdulist_SDSS_test = fits.open('data/SDSS_DR13_SpecObjAll.fits')
data_SDSS_test = hdulist_SDSS_test[1].data
data_SDSS_test = np.column_stack((data_SDSS_test['psfMag_u'],data_SDSS_test['psfMag_g'
data_SDSS_test['psfMag_r'],data_SDSS_test['psfMag_i'],
data_SDSS_test['psfMag_z'],data_SDSS_test['z']))
X_test = data_SDSS_test[:,:5]
Y_test = data_SDSS_test[:,5:]

X_test = (X_test-minx)/(maxx-minx)
Y_test = (Y_test-miny)/(maxy-miny)

scores = model.evaluate(X_test, Y_test)
```

```
print "Hypothesis", scores
Yp = model.predict(X_test, batch_size=10000)
```

1. The code to normalize data is important because a neural network assumes no meaning to any input vector and all the components must be seen on an equal footing.

2. We then define a keras model, and add layers to it. The layer called Dropout randomly drops out certain part of the data. This is empirically known to improve results. The value of Dropout is to be determined by trial.

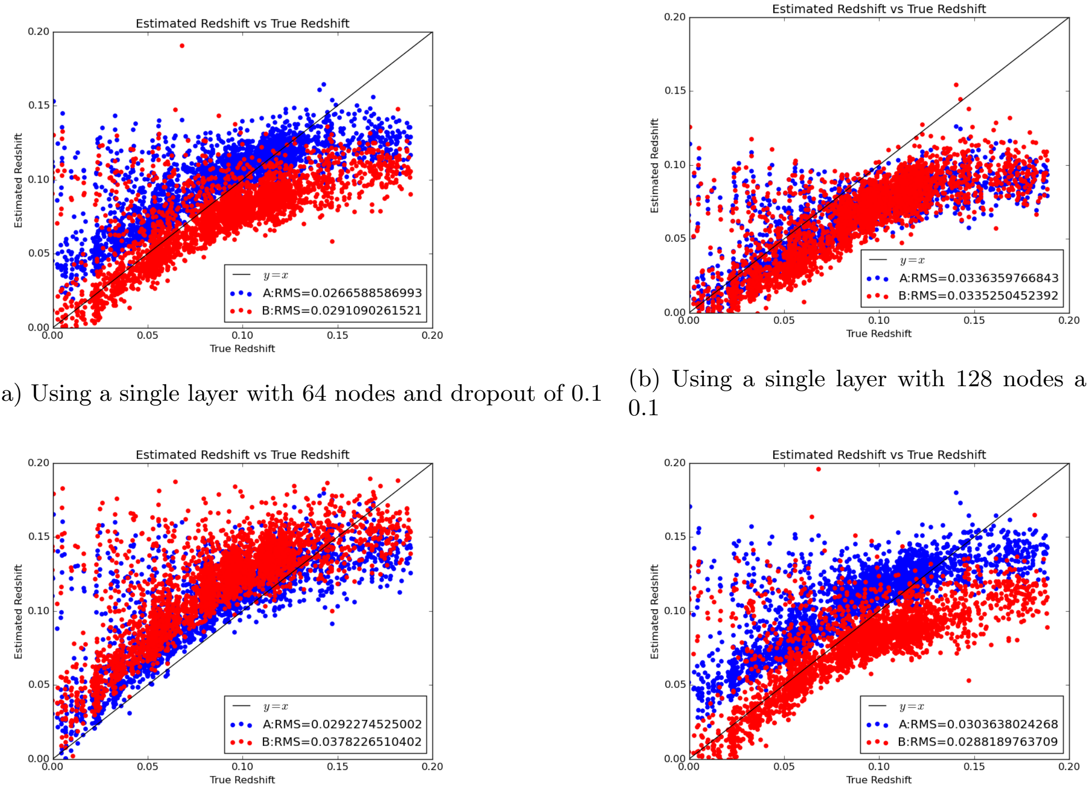
3. We then fit the data, and predict the outcome against our test batch, which we have to additionally normalize with the same normalizing constants as above.

4. There is a subtlety involved with the normalization. Suppose our training data has minimum and maximum values as 15 and 30. But, if in our test data we encounter a number whose value is 8, this moves the value outside the range of(0,1) and gives undesirable results. Thus, we may want to increase $\max - \min$ by 5.

11 Week 11-12 1 Apr - 14 Apr

Here we simply summarize our outputs for various situations described in captions. We will show plots of 'Estimated Redshift vs True Redshift' for most cases and only elaborate where necessary.

11.1 Neural Networks



(a) Using a single layer with 64 nodes and dropout of 0.1

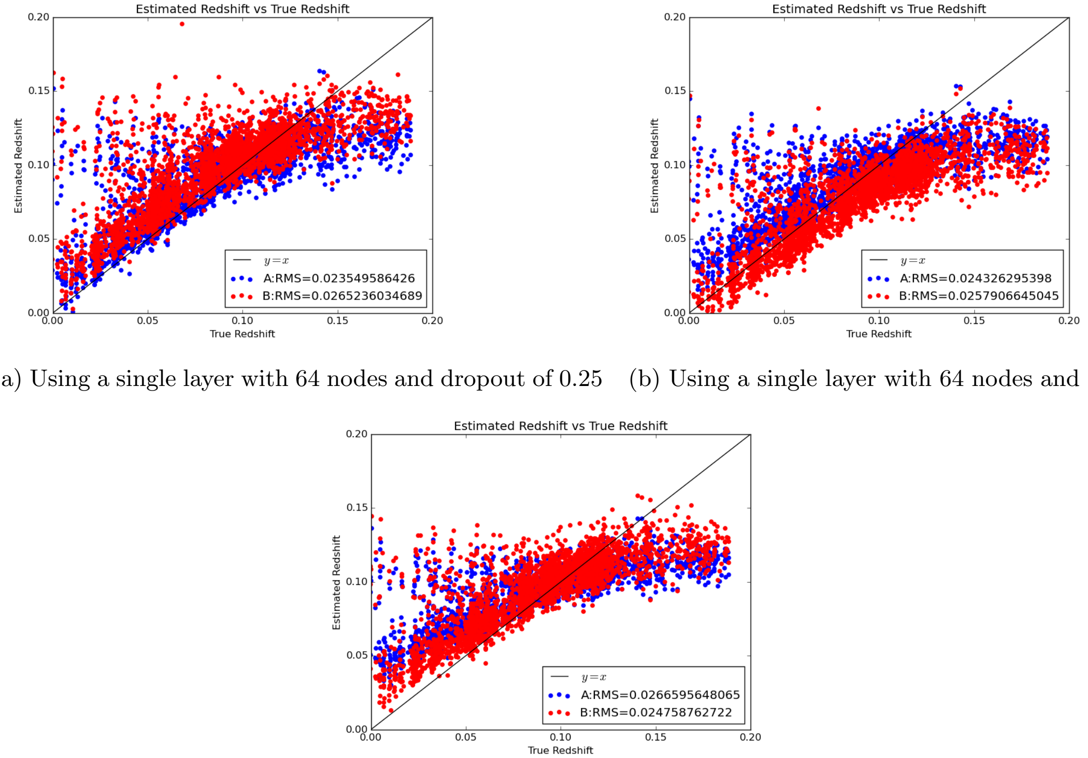
(b) Using a single layer with 128 nodes and dropout of 0.1

(c) Using two layers with 64 and 32 nodes each and a dropout of 0.1

(d) Using three layers with 64,32 and 16 nodes and a dropout of 0.1

Figure 1: Variation with layers and nodes, Method A involves training with mixed dataset, while Method B involves training with pure SDSS dataset

We notice a minor improvement by using method A in all the cases with fewer or equal to 2 layers. But the RMS error for both the methods are not different enough to make any satisfactory conclusions. We also report variations with the dropout rate for an ANN with 64 nodes and single layer, just for completeness sake. Variation with learning rate is not notable enough to be reported.



(a) Using a single layer with 64 nodes and dropout of 0.25

(b) Using a single layer with 64 nodes and dropout of 0.5

(c) Using a single layer with 64 nodes and dropout of 0.75

Figure 2: Variations with dropout rates, Method A involves training with mixed dataset, while Method B involves training with pure SDSS dataset

11.2 K-nearest neighbours

We use an additional function to determine optimum value of K which turns out to be 21.

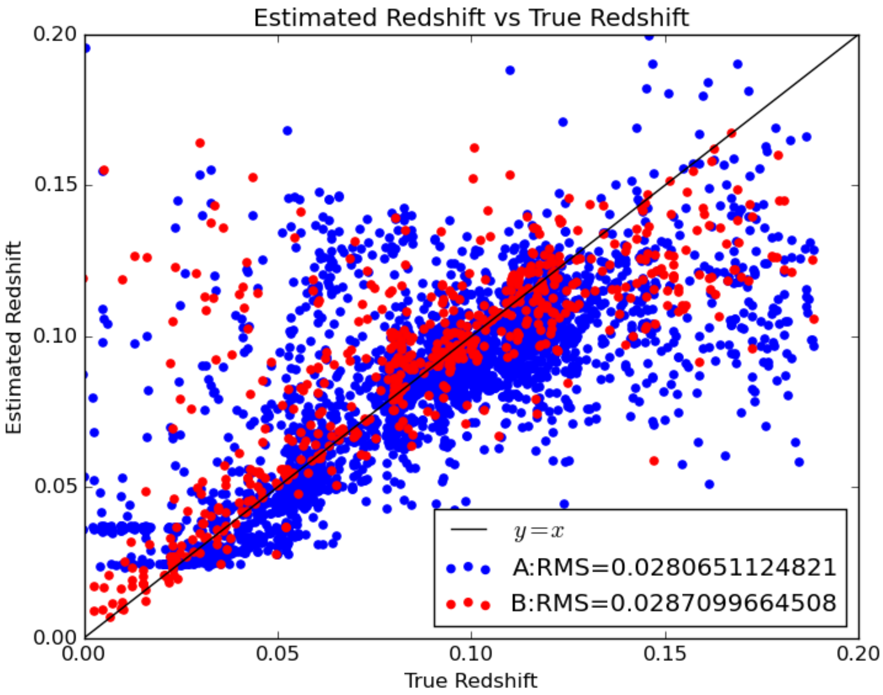


Figure 3: Using K-nearest neighbour with $K = 21$

We see that this method also gives close results with slight improvements using Method A, not conclusive.

12 Discussion

We will try to answer two questions here

12.1 Does our implementation test our hypothesis?

We wanted to train our dataset for high redshift galaxies and then test it versus training with low redshift galaxies. Unfortunately, the SDSS data that we gather has negligible points above $z = 0.2$ even for the Photometry database. And PRIMUS and VIPERS survey do not provide us with multi-band photometric data which we can use for training. This renders our effort to find out the ohotometric redshifts for $z > 0.5$ galaxies useless.

12.2 Are our results conclusive?

No, the difference in RMS error between Method A and B as described is too less to make any concluding remarks.

References

[1] I Csabai et al. *Multidimensional indexing tools for the virtual observatory.* Astron. Nachr. / AN 328, No. 8, 852 – 857 (2007) / DOI 10.1002/asna.200710817

[2] S. Cavuoti et al. *Machine Learning based photometric redshifts for the KiDS ESO DR2 galaxies.* arXiv:1507.00754v3, 30 Jul 2015 / Mon. Not. R. Astron. Soc. 000, 1–6 (2015)

[3] *Brescia, M., Cavuoti, S., Paolillo, M., Longo, G. & Puzia, T.. 2012a* MNRAS. 421, issue 2, 1155.

[4] Brice Menard et al. *Clustering-based Redshift Estimation:Method and Application to Data* arXiv:1303.4722

[5] Christopher B. Morrison et al. *The-wiZZ: Clustering redshift estimation for everyone* arXiv:1609.09085

[6] Stefano Cavuoti et al. *A cooperative approach among methods for photometric redshifts estimation: an application to KiDS data* arXiv:1612.02173

[7] Boris Leistedt et al. *Data-driven, interpretable photometric redshifts trained on heterogeneous and unrepresentative data* arXiv:1612.00847

[8] Adrian A. Collister et al. *ANNz: estimating photometric redshifts using artificial neural networks* arXiv:astro-ph/0311058

[9] Mi Dai et al. *Photometric classification and redshift estimation of LSST Supernovae* arXiv:1701.05689

[10] C.M. Harrison et al. *The KMOS Redshift One Spectroscopic Survey (KROSS): rotational velocities and angular momentum of z 0.9 galaxies* arXiv:1701.05561

[11] V. Scottez et al. *Clustering Based Redshift Estimation : Application TO VIPERS/CFHTLS* arXiv:1605.05501v2

[12] NARCISO BENITEZ *Bayesian Photometric Redshift Estimation* THE ASTROPHYSICAL JOURNAL, 536 : 571È583, 2000 June 20

[13] Yun Wang et al. *Analytic Photometric Redshift Estimator for Type Ia Supernovae From the Large Synoptic Survey Telescope* arXiv:1501.06839

[14] Stefano Cavuoti et al. *Photometric redshift estimation based on data mining with PhotoRApToR* arXiv:1501.06506

[15] Samuel Carliles et al. *Photometric Redshift Estimation on SDSS Data Using Random Forests* arXiv:0711.2477v1

[16] Tamás Budavári *PHOTOMETRIC REDSHIFTS 50 YEARS AFTER*

[17] Tamás Budavári *A UNIFIED FRAMEWORK FOR PHOTOMETRIC REDSHIFTS* arXiv:0811.2600v2

[18] Lisa J. Storrie-Lombardi *Photometric Redshifts and High-Redshift Galaxies* PUBLICATIONS OF THE ASTRONOMICAL SOCIETY OF THE PACIFIC, 111: 1188È1189, 1999 September

[19] Hiroaki Oyaizu et al. *PHOTOMETRIC REDSHIFT ERROR ESTIMATORS* The Astrophysical Journal, 689:709Y720, 2008 December 20

[20] Boris Leistedt *ROBUST PHOTOMETRIC REDSHIFTS AND REDSHIFT DISTRIBUTIONS* 2015/03/BLeistedt

[21] S. Cavuoti et. al. *METAPHOR: A machine learning based method for the probability density estimation of photometric redshifts* arXiv 1611.02162

[22] S. Cavuoti et. al. *METAPHOR: A machine learning based method for the probability density estimation of photometric redshifts* arXiv:1601.07857

[23] K.L. Polsterer et. al. *Uncertain Photometric Redshifts* arXiv:1608.08016

[24] INAF *The VIPERS Project*
`http://vipers.inaf.it/project.html`

[25] *PRIMUS: PRIsm Multi-object Survey*
`http://cass.ucsd.edu/ acoil/primus/Overview.html`

[26] *SDSS DR13*
`http://www.sdss.org/dr13/scope/`