



TATA INSTITUTE OF FUNDAMENTAL RESEARCH  
DEPARTMENT OF THEORETICAL PHYSICS

**Detecting hot baryons using cross-correlation  
of thermal SZ maps and Weak Lensing  
surveys**

PRANAV S

Supervised by  
PROF. SUBHABRATA MAJUMDAR

PROJECT WORK SUBMITTED IN PARTIAL FULFILMENT OF THE  
REQUIREMENTS FOR THE AWARD OF THE DEGREE OF  
MASTER OF SCIENCE IN PHYSICS

## Abstract

The knowledge of the baryonic and non-baryonic matter distribution of the universe is fundamental in understanding evolution and structure formation in the Universe. However, a large fraction of them cannot be detected directly. One of the major ways of indirectly detecting the baryon distribution is the cross-correlation between thermal Sunyaev Zel'dovich (tSZ) effect by the hot baryons and Weak Gravitational Lensing by the dark matter. There have been multiple attempts/detection of this cross-correlation in the past. However, newer and better data promises to yield better constraints as attempted in this thesis. Moreover, doing an independent analysis of this cross-correlation will also be useful in detecting systematic errors, if any, in the existing skymaps. We compute the tSZ skymaps using a methodology independent from the Planck collaboration's pipelines, and then cross-correlate these skymaps with tangential shear due to lensing potential, to compare with the existing constraints on halo astrophysics and cosmology and in the process detect for the first time, hot diffuse baryons present in conglomeration of galaxies. This work consists of two parts- independent generation of tSZ skymaps using Unsupervised machine learning (which was done partially in collaboration with Prof Rishi Khatri) and cross-correlating the tSZ skymaps with RCSLenS and KiDS weak lensing surveys. Finally, we attempt to connect the observed detection with physics modelling of diffuse baryons in halos.

### **Acknowledgements**

I would like to thank my supervisor, Prof Subhabrata Majumdar for introducing me to cosmology, suggesting this problem and providing theoretical insights into it. I am grateful for all his help and support during crucial times which made me progress faster. I would also like to thank Prof Rishi Khatri for his help and support on the generation of tSZ skymaps using machine learning, and Dr Shadab Alam from University of Edinburgh for all the discussions and technical help regarding the work on cross-correlations. I would also like to thank TIFR for providing me with the DTP High performance compute clusters **zeal** and **hobbes** which were crucial in carrying out the computations.

# Contents

<b>1</b>	<b>Introduction</b>	<b>2</b>
<b>2</b>	<b>Relevant Physics and Algorithms</b>	<b>3</b>
2.1	Weak Lensing . . . . .	3
2.2	SZ Effect . . . . .	5
2.3	Quick Introduction to Machine Learning . . . . .	5
2.3.1	Supervised Machine Learning . . . . .	6
2.3.2	Unsupervised Machine Learning . . . . .	6
2.3.3	Neural Networks . . . . .	6
2.4	Space Partitioning for quick computation of Correlation functions . . . . .	7
2.4.1	k-D Trees . . . . .	7
2.4.2	Finding the nearest neighbour . . . . .	8
2.4.3	Ball - Trees . . . . .	8
<b>3</b>	<b>New thermal SZ maps</b>	<b>9</b>
3.1	Generalized ILC . . . . .	9
3.1.1	Combining information from maps with different resolutions . . . . .	11
3.2	New tSZ maps with Unsupervised Machine Learning . . . . .	11
3.2.1	'k-means clustering' method . . . . .	11
3.2.2	Preprocessing . . . . .	12
3.2.3	Dimensionality Reduction . . . . .	13
3.2.4	Auto Encoders . . . . .	13
3.2.5	Self Organising Maps . . . . .	14
3.3	Results . . . . .	14
<b>4</b>	<b>Thermal-SZ Weak Lensing Cross Correlation</b>	<b>16</b>
4.1	Calculation from Data . . . . .	16
4.1.1	Observational Data and Sampling . . . . .	16
4.1.2	Two Point Correlation Functions . . . . .	18
4.1.3	Bias Correction . . . . .	18
4.1.4	Systematic Tests . . . . .	18
4.1.5	Effect of Masks . . . . .	19
4.2	Results . . . . .	19
4.3	Comparison with theory . . . . .	21
<b>5</b>	<b>Conclusion</b>	<b>22</b>
<b>A</b>	<b>m-maps</b>	<b>23</b>
<b>B</b>	<b>A single RCSLenS Field</b>	<b>24</b>

# Chapter 1

## Introduction

Though the general physical processes driving cosmological evolution and large scale structure formation is reasonably well understood, many of the details which are essential for understanding how galaxies and clusters of galaxies form and evolve are still unclear. One such important detail is the knowledge of the distribution of baryonic and dark matter in galaxies and clusters. We know that stellar mass accounts for only approximately 10% of the baryonic mass in the universe and the rest resides in diffuse components such as halos[2, 3]. This makes it hard to estimate the mass distribution of matter in galaxies and clusters.

Historically, diffuse components are measured via X-Ray emissions and thermal Sunyaev-Zel'dovich (tSZ) effect; but because of the sensitivity of the experiments, just relying on these will help us detect only the hot and dense diffuse components. A possible way of observing these diffuse components would be to use gravitational lensing fields; since these lensing fields provide an estimate of the matter distribution in the large scale structure of the universe. Especially, with the recent RCSLens dataset and the ongoing KiDS dataset (Section 4.1.1) from the galaxy surveys, weak lensing has become a precision tool in understanding large-scale structure. Whereas, our lack of understanding of baryonic physics at small scales, leads to uncertainty in our estimates of matter distribution using gravitational lensing.

The missing insights from both the tSZ probes and weak lensing datasets from sky surveys can be compensated by cross-correlating tSZ probes with the weak lensing fields. Since cross-correlations also have the advantage of being immune to systematic effects which doesn't correlate with the signals, it provides a powerful method for extracting information from these probes and help us understand the astrophysical processes governing these scales.

A recent attempt at cross-correlation between the two probes found that the data supports WMAP-7yr cosmology more than the Planck Cosmology[1]. While this result was found using the tSZ maps provided by the Planck collaboration, the results beg the question of systematic deviations existing in the skymaps provided by the Planck collaboration. It is therefore useful to consider independent algorithms which extract information from the multi frequency CMB observations such as WMAP and Planck. The methodology presented by us here is the first attempt in using unsupervised machine learning to perform component separation using a model independent methodology.

This work consists of 2 parts- (a) We use an independent methodology to extract the tSZ maps from Planck's frequency data. This provides us with hints, incase there exists systematic deviations from the Planck skymaps. (b) We compute cross-correlations between tSZ maps and Weak Lensing maps by various sky surveys.

This thesis is divided as follows, we initially review the relevant physics of the data and the machine learning concepts we would be using (Chapter 2). After which, we explain our component separation technique which uses unsupervised machine learning techniques (Chapter 3). We then compute the cross-correlation with various maps and datasets. (Chapter 4). We finally present our results and concluding remarks (Chapter 5).

## Chapter 2

# Relevant Physics and Algorithms

In this chapter, we will provide a quick introduction to the concepts involved. We first review Weak Lensing and SZ Effect (Sections 2.1 and 2.2). We then review machine learning concepts and techniques we will be using (Section 2.3), and then we finally review space partitioning algorithms and data structures which help us compute correlation functions efficiently (Section 2.4)

### 2.1 Weak Lensing

Weak Lensing is the phenomenon when a gravitational potential distorts the shape of astrophysical sources. We know that according to general relativity, light bends when passing through a gravitational potential. This creates a lens-like effect which distorts the sources in the background. If the lensing is weak enough that the images are simply distorted instead of forming multiple images, it is known as weak lensing.

Starting with the lensing equation  $\vec{\beta} = \vec{\theta} - \vec{\alpha}$ , where,  $\vec{\theta}$  is the true position of the source,  $\vec{\beta}$  is the observed position of the source and  $\vec{\alpha}$  is the deviation. The deviation  $\vec{\alpha}$  can be derived as a function of the gravitational potential, using the geodesic equation,

$$\alpha^i(\vec{\theta}) = \frac{\partial\Phi(\vec{\theta})/c^2}{\partial\theta^i} \quad (2.1)$$

where,  $\Phi$  is the projected gravitational potential[4] and  $c$  is the speed of light.

Since, we would be interested in the change in the flux of the incoming light due to the intermediate lensing potential, we would like to consider the change in the image size per unit change in the source size. This gives us a measure of distortion by the intermediate mass distribution,

$$\frac{\partial\beta_i}{\partial\theta_j} = \begin{pmatrix} 1 - \frac{\partial\alpha_1}{\partial\theta_1} & \frac{\partial\alpha_1}{\partial\theta_2} \\ \frac{\partial\alpha_2}{\partial\theta_1} & 1 - \frac{\partial\alpha_2}{\partial\theta_2} \end{pmatrix} \quad (2.2)$$

The distortion tensor ( $\Psi_{ij}$ ) is then defined as,

$$\Psi_{ij} \equiv \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} - \frac{\partial\beta_i}{\partial\theta_j} \quad (2.3)$$

which captures the deviation from identity. Note that, in the absence of intermediate lensing potentials, the distortion tensor goes to zero. This means, that the image we see would be the same size as the source. The elements of the distortion tensor<sup>1</sup> are given by,

$$\Psi_{ij} = \begin{pmatrix} \kappa + \gamma_1 & \gamma_2 \\ \gamma_2 & \kappa - \gamma_1 \end{pmatrix} \quad (2.4)$$

---

<sup>1</sup>The distortion tensor is symmetric because we can write the deviation  $\vec{\alpha}$  as a gradient of the projected gravitational potential

Here, the convergence term  $\kappa$  corresponds to the change in size of the object and the shear terms  $\gamma_1$  and  $\gamma_2$  corresponds to the shear in the particular direction. The shear terms contribute to the change in the shape of the objects. For a perfectly circular object, the ellipticities of the images which arise due to the shear terms, can be written in terms of  $\gamma_1$  and  $\gamma_2$ . In the weak lensing regime,  $\gamma_1 \approx e_1/2$  and  $\gamma_2 \approx e_2/2$  where,  $e_1$  and  $e_2$  are the ellipticities of the galaxies[4]

The shear terms  $\gamma_1$  and  $\gamma_2$ , along with the convergence  $\kappa$  can be written in terms of the deflection  $\alpha$  (Eq. (2.2)),

$$\begin{aligned}\kappa &= \frac{1}{2} \left( \frac{\partial\alpha_1}{\partial\theta_1} + \frac{\partial\alpha_2}{\partial\theta_2} \right) \\ \gamma_1 &= \frac{1}{2} \left( \frac{\partial\alpha_1}{\partial\theta_1} - \frac{\partial\alpha_2}{\partial\theta_2} \right) \\ \gamma_2 &= \frac{\partial\alpha_1}{\partial\theta_2} = \frac{\partial\alpha_2}{\partial\theta_1}\end{aligned}\tag{2.5}$$

which can then be written in terms of the gravitational potential,

$$\begin{aligned}\kappa &= \frac{1}{2c^2} \left( \frac{\partial^2\Phi}{\partial\theta_1^2} + \frac{\partial^2\Phi}{\partial\theta_2^2} \right) \\ \gamma_1 &= \frac{1}{2c^2} \left( \frac{\partial^2\Phi}{\partial\theta_1^2} - \frac{\partial^2\Phi}{\partial\theta_2^2} \right) \\ \gamma_2 &= \frac{1}{2c^2} \left( \frac{\partial^2\Phi}{\partial\theta_1\partial\theta_2} \right)\end{aligned}\tag{2.6}$$

These relationships between the gravitational potential and the distortion tensor, allows us to understand various astrophysical and cosmological processes by computing the mass distribution from the shear.

Different linear combinations of  $\gamma_1$  and  $\gamma_2$  are defined in order to remove the polar dependence on these terms,

$$\begin{aligned}\gamma_T &= -\gamma_1 \cos(2\phi) - \gamma_2 \sin(2\phi) \\ \gamma_X &= -\gamma_1 \cos(2\phi) + \gamma_2 \sin(2\phi)\end{aligned}\tag{2.7}$$

Since, for the case of spherically symmetric mass distributions,  $\gamma_x$  is zero, it serves as a very useful *null test*. Any deviation from zero for  $\gamma_x$  can be considered as systematic error in our calculations.

While trying to compute the shear from observed ellipticities of the galaxies, we have no way of knowing in advance, the intrinsic ellipticities of the galaxies to study the effect of shear. The only way to eliminate the effect of these intrinsic ellipticities is to compute the average of shear over a particular region. Since, the intrinsic ellipticities of the galaxies are randomly oriented; while computing the average of a particular component of shear the intrinsic component cancels out. A way of computing these averages is correlation functions, which quantifies the correlation between the shears of galaxies separated by a distance  $r$ . The correlation function would be zero without the presence of an lensing field, since the intrinsic ellipticities are randomly oriented. When there is an intermediate gravitating body, the shears are correlated because of it.

In the following example, for the case of cross-correlation between  $\gamma_T$  and some scalar field  $y$ , let us consider the observed shear of the galaxies consisting of two parts. One from the intrinsic shear of the galaxies and the second from the lensing gravitational body,

$$\begin{aligned}\langle\gamma_T^{obs}y\rangle &= \langle\gamma_T^{shear}y\rangle + \langle\gamma_T^{intrinsic}y\rangle \\ &= \langle\gamma_T^{shear}y\rangle + \cancel{\langle\gamma_T^{intrinsic}y\rangle} \rightarrow 0 \\ &= \langle\gamma_T^{shear}y\rangle\end{aligned}\tag{2.8}$$

this way, the effect of weak lensing can be computed directly from the ellipticities without worrying about accounting for the intrinsic shapes of galaxies. Note that, even in the case of auto-correlation (correlation function between two shears), the intrinsic part cancels out but we are susceptible to the systematic noise in our measurements of shear which can be correlated.

## 2.2 SZ Effect

Compton scattering is one of the major astrophysical processes which couple matter and radiation. Sunyaev - Zel'dovich effect (SZ Effect) is one such example of Compton scattering at low energies, which electrons in clusters of galaxies get scattered by the Cosmic Microwave background radiation. This induces a temperature anisotropy at a frequency  $\nu$ . We would specifically be interested in the *comptonization parameter*  $y$ , which is a dimensionless measure of the time spent by the radiation in an electron distribution along a particular line of sight, which is related to the temperature anisotropy  $\Delta T$ ,

$$\frac{\Delta T(\vec{r})}{T_0} = y(\vec{r})S\left(\frac{h\nu}{k_B T_0}\right) \quad y(\vec{r}) = \int n_e(\vec{r})\sigma_T dl \frac{k_B T_e(\vec{r})}{m_e c^2} \quad (2.9)$$

where,

- $S(x)$  is the tSZ spectral dependence  $S(x) = x \coth(x/2) - 4$ [5],
- $n_e$  is the electron concentration,
- $\sigma_T$  is the thompson scattering cross section (since for low energies, we can take the non relativistic approximation),
- $T_e$  is the temperature of the electron cloud,
- $T_0$  is the present-day CMB temperature,
- $h$  and  $k_b$  are the Planck and Boltzmann constant respectively.

When we construct skymaps for the SZ effect, we are using the change in intensity to measure this comptonization parameter to give us insights into the distribution of electron clouds in the universe.

To use these maps to compute astrophysical or cosmological parameters, we compute the cross-correlation between the two quantities. Cross-correlation is a powerful statistical tool since it helps us avoid systematic biases, which might be present in these two quantities individually. Cross-correlating the quantities mean that the systematic biases won't affect our final outcome as the systematic biases from two different experiments could be assumed to be uncorrelated with each other. (See Fig. 4.4)

## 2.3 Quick Introduction to Machine Learning

Machine learning is a type of an algorithm which, rather than explicitly coding an algorithm for a particular function/behaviour, gives the computer the ability to *learn* that function. Machine learning involves a training set, from which the computer *learns* the function of interest, and a validation set where the computer applies whatever it has learned from the training set and gives us results. Whenever, we say that the computer *learns*, we mean a minimization of some cost function, by varying the parameters of the function the computer is trying to learn. This cost function captures the deviation from the desired behavior of the algorithm after learning.

We would be particularly interested in a classifier, which can be defined as a function ( $f : \mathcal{D} \rightarrow [0, 1]^n$ ) which divides the input data (from the domain  $\mathcal{D}$  which is usually  $\mathbb{R}^m$ ) into  $n$  different classes, by assigning a probability for each of the classes.

We would also be only implementing hard clustering, where the final result is a single cluster value instead of a set of probabilities for each cluster. This is implemented by assigning the value of the cluster to be the one with maximum probability. In our case, we will be dividing the sky into different classes based

on their foreground properties, while using the different frequencies as input. Based on our training set, machine learning can be divided into two categories,

- Supervised Machine Learning
- Unsupervised Machine Learning

### 2.3.1 Supervised Machine Learning

A machine learning algorithm is said to perform supervised learning when the training set is different from the validation set and we have the expected learning outcomes as part of our training set. These labeled training sets consist of an input and an expected output value as part of the examples, and the algorithm learns the function which relates the input value to the output value based on these training examples. In this case, the cost function is some sort of deviation from the output value compared to the given output value in the training examples.

### 2.3.2 Unsupervised Machine Learning

Unsupervised learning is when a machine learning algorithm has no prior information about the data set in terms of a training set and the data learns and validates from the same set. In this method only the raw data is given to the algorithm, and it tries to *learn* patterns inherent in the data, without any labels to learn from. This would be really useful for our purposes since it allows us to partition the sky based on the frequency without any model dependent inputs. We would specifically be using clustering algorithms and dimensionality reduction algorithms.

**Clustering Algorithms** Partition the data into different classes based on how every data point is located with respect to it's neighbours. These algorithms have a varying point in the *data space* which is representative of the cluster and assign the cluster to the available data based on it's relationship to this point.

**Dimensionality Reduction** Clustering algorithms in higher dimensions tend to be heavily dependent on the initial seed for the clusters. It is therefore, incredibly useful to reduce the dimensionality of the data by summarizing the information in fewer number of variables and then performing the clustering in a lower dimensional space.

### 2.3.3 Neural Networks

A neural network is a special case of a machine learning algorithm, which has a specific functional form. The advantage a neural network provides is based on a mathematical theorem that this specific functional form can approximate any continuous function[6].

A single layer neural network is defined by the following function,

$$\vec{c} = \sigma(M_1\vec{v} + \vec{b}_1) \tag{2.10}$$

Where,

- $\vec{v}$  is the Input Data with the dimensionality of our Input variables space,
- $\vec{c}$  is the Output data with the dimensionality of our number of classes,
- $\vec{b}$  is the bias vector with the dimensionality of our number of classes,
- $\sigma$  is a sigmoidal<sup>2</sup> function which acts component wise (also known as the activation function),
- $M$  is a matrix whose elements we will tune as parameters.

The parameters (the elements of the matrix M) can then be tuned based on a cost function, which after training will help us get the final matrix, which we can use to classify our data. A multi-layer neural network can be constructed by composition of this, an arbitrary number of times. We use Rectified Linear Unit (ReLU:  $\max(0, x)$ ) as our activation function.

---

<sup>2</sup>A sigmoid function is a bounded, differentiable, real function that is defined for all real input values and has a non-negative derivative at each point.

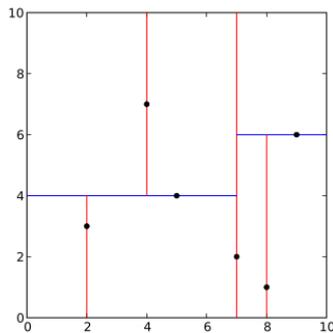
## 2.4 Space Partitioning for quick computation of Correlation functions

In order to compute the correlation function, the nearest neighbours of every galaxy needs to be known. Computationally, looping over all the galaxies and computing the pairwise distance is a very costly. We would need an algorithm, which lets us know the nearest neighbour without explicitly computing the distances. There is a class of algorithms which help us do this, called *Space Partitioning Algorithms*. Space partitioning systems are often hierarchical, where the algorithm is applied recursively in order to partition the regions. These regions are in general stored in the form of a *Tree*<sup>3</sup>. Finally when the recursion stops at the individual points instead of regions. These end points of the *tree* are known as *leaves*. We would specifically be interested in an algorithm called k-D Trees.

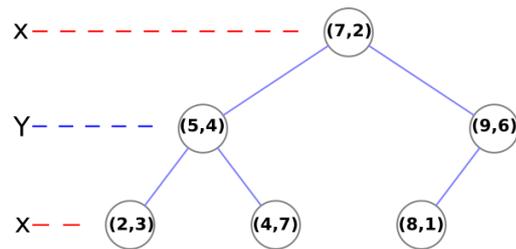
### 2.4.1 k-D Trees

k-D trees is a space partitioning data structure, which constructs the tree based on the proximity of the points without explicitly computing the distances between the points. k-D Trees are a special case of space partitioning, called binary space partitioning, where each region is divided into two regions recursively. This also implies, that each node in the *tree* we are constructing will be connected to two other nodes. In this structure, each leaf-node will be a point from the given data. Each non-leaf-node, will also implicitly divide the space with a hyper plane, which is parallel to the axis and passes through the point corresponding to the space. The Algorithm goes as follows,

1. Choose the median point along a particular axis, and make that into a parent node.
2. Create 2 regions, based on the hyperplane which divides the same axis along the median point.
3. Create two sub-trees for these two regions, and repeat from step 1, while cycling through all possible axis for the choice of axis.



(a) A k-d tree partitioning a 2-d space



(b) A visual representation of the data structure

Figure 2.1: In this example, we can see k-D trees, in action. In Fig. 2.1a we see how every node in a k-d tree also acts as a hyperplane which divides the space into two parts. This is important since, it helps us ignore entire segments when traversing the tree. In Fig. 2.1b, We see how the tree like structure of the data structure enables it. Note that each parent node only contains children on their side of the division

Finding the nearest neighbours then becomes merely the act of traversing the trees. This immensely speeds up the computation of correlation functions.

<sup>3</sup>A Tree is a data structure, which is defined recursively, where the values are stored as a collection of nodes, each with a root value and also acts as a parent for a sub-tree, represented by a collection of linked nodes

## 2.4.2 Finding the nearest neighbour

By efficiently, utilizing the properties of the tree; Nearest Neighbour (NN) searches can be done efficiently by quickly eliminating large chunks of data. The process of searching for nearest neighbours is analogous to the process of searching for particular value on the tree. This involves traversing the tree starting from the root, and choosing the branch at every node. Since, in the case of a binary tree; this involves ignoring the other branch altogether, we can ignore huge chunks of data without explicitly comparing with the current neighbours. The Algorithm goes as follows,

1. Start from the root-node of the tree.
2. Compute the distance to the node. Store this value as the best value.
3. Compute the distances to all the children nodes. If any of the values is lesser than the best value. Choose this node and repeat step 2.
4. Exit the loop when all the children's distances are greater than the best value, or if you reach the leaves of the tree.

If instead of searching for the nearest neighbour, we want all the neighbours till a particular distance. We can also exit the loop once that particular distance is reached and return all the nodes, belonging to a particular node all the way to the leaves.

We wrote the code in `Python` implementing the k-d trees algorithm to compute correlation functions.

## 2.4.3 Ball - Trees

Ball Tree is a variant of the k-D Trees data structure, where each node is partitions the data into two disjoint sets each associated with a different hypersphere. Each node in this tree defines the smallest ball, which contains all the data points in it's subtree. The algorithm for construction and searching for nearest neighbour is similar to the k-d trees algorithm. We use a `Python` library (`treecorr`) which computes two

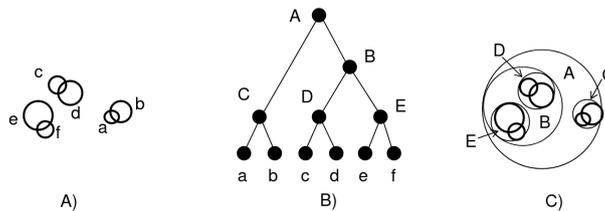


Figure 2.2: Ball Trees: (A) Initial Hyper-balls (These initializations can be overlapping, but it is made sure that each hypersphere contains only one point; It's center) (B & C) Ball-Tree structure constructed for the available data (Image Courtesy:[7])

point correlation functions, by implementing the ball-tree algorithm.[8]. All the subsequent correlation functions computed in this thesis uses `treecorr`.

# Chapter 3

## New thermal SZ maps

One of the ways of removing foregrounds from our data is to have a parametric model for the foregrounds based on known physics and fit the data to these foreground models to get an estimate of the parameters and use them to eliminate the foregrounds in the data. This is the methodology followed by Commander pipeline of the Planck collaboration[9] and LIL Method[10]. These methods suffer from being model dependent and the parameters are insufficient to model the foregrounds because various physical processes contribute to the foreground in a single pixel, making it hard to model these with a single parametric model. Therefore, it is useful to look at algorithms which are model independent and just use the data to estimate the foreground characteristics.

Various such *blind* component separation algorithms have been proposed where only the spectrum of the signal is needed[11, 12] and have been applied in various variations such as Harmonic space, Needlet frame etc[13, 14]. Since these blind algorithms require the number of foreground components to be lesser than the number of spectral channels available, it is necessary to divide the sky with similar foreground properties into different regions and apply the algorithm separately on those regions to reduce the number of foreground components. These existing variations try to divide the data into different clusters with similar foregrounds based on heuristic arguments and our current understanding of the properties of the foregrounds. We provide a spectral data based approach extending upon this data driven foreground clustering approach[15], which uses the signature of the foregrounds available in the data. Since the data is clustered only based on the spectra, different regions of the sky can still be in the same cluster based on their foreground properties. To the best of our knowledge, this is the first attempt at using *unsupervised* machine learning for foreground component separation.

In this chapter, first, we review the GILC method (Section 3.1). We then explain our algorithm for creating the new skymaps using unsupervised machine learning Section 3.2. We then present our results Section 3.3.

### 3.1 Generalized ILC

We want to separate out the cosmological signal of interest from the different frequency maps provided by the Planck collaboration, by subtracting out the different foreground signals. The Generalized ILC is used to separate out any signal with a known spectrum.

The observed temperature data, in different pixels (positions) and frequency channels can be written as,

$$T(\nu, p) = S(\nu)A(p) + F(\nu, p) + N(\nu, p) \quad (3.1)$$

Where,

- $S(\nu)$  is the spectrum of the signal we are looking for. This is essentially the unit conversion which let's us perform ILC to extract any signal of a known spectrum. For eg, in the case of CMB, in  $K_{cmb}$  units, it is equal to one. ie, the CMB temperature is independent of the frequency and is only a function of the pixel  $p$ .

- $A(p)$  is the position dependent signal (sky-map) we are looking for.
- $F(\nu, p)$  is the sum of all the foreground components in a particular pixel. Since, different foregrounds with different spectra contribute in each pixel, this is dependent on both the frequency and the pixel position in the sky.
- $N(\nu, p)$  is the noise which is in general dependent on both the frequency channel and the pixel position.

Now, assuming that the signal is a linear combination of the frequency maps. We can write,

$$s_{sig}(p) = \sum_{\nu} w(\nu)T(\nu, p) \quad (3.2)$$

Since, in the absence of foregrounds and noise, we need to recover back the signal (ie,  $s_{sig}(p) = A(p)$ ); We get the following constraint,

$$\sum_{\nu} w(\nu)S(\nu) = 1 \quad (3.3)$$

If we knew the true signal  $s_{true}(p)$ , we could find the weights ( $w(\nu)$ ) assigned to the different frequencies by minimising a cost function,

$$C = \sum_p (s_{true}(p) - s_{sig}(p))^2 \quad (3.4)$$

$$C = \sum_p \sum_{\nu} (F(\nu, p) + N(\nu, p)) \quad (3.5)$$

Since, we don't know the true value of the signal, let us consider an alternate cost function,

$$C' = \sum_p (s_{sig}(p))^2 \quad (3.6)$$

$$C' = \sum_p (s_{true}(p))^2 + C - \sum_p s_{true}(p) \sum_{\nu} (F(\nu, p) + N(\nu, p)) \quad (3.7)$$

Since the first term is a additive constant, which is irrelevant to the minimization problem. We see that  $C = C'$  if, the third term is zero (ie, The foregrounds and noise are not correlated with the signal). Though the noise is in general not correlated with the signal, we can't say the same about the foregrounds. This leads to a bias, where the ILC also will remove the part of the signal correlated with the foregrounds. This results in a solution which has lesser power than the actual signal.[15] It has also been previously studied that, this ILC Bias is significant only for  $l = 2$ , and is negligible for all the higher multipoles. It can however, start affecting the higher multipoles if we increase the number of partitions to a high number.[15] By minimising the cost, with respect to the constraint Eq. (3.3), We get the following equations for the weight,

$$w(i) = \frac{\sum_j D_{ij}^{-1}}{\sum_{ij} D_{ij}^{-1}} \quad (3.8)$$

where,

$$D_{ij} \equiv \sum_p T(i, p)T(j, p) \quad (3.9)$$

and where the sum over the pixels ( $\sum_p$ ) is done for pixels belonging to a single cluster.

### 3.1.1 Combining information from maps with different resolutions

Different resolution maps, has different information. In order to effectively use out algorithm, We need to find a way to combine these information effectively.[15] Since the noise properties depend on the resolution of the map used, We need to rebeam all the maps to a common resolution. But, it is not straightforward to see which resolution map is useful for performing the ILC. Rebeaming all the maps to a lower resolution means, we loose all the large  $\ell$  signals. Whereas, rebeaming all the maps to a higher resolution means, the noise in the lower resolution gets boosted significantly, and these noise dominated maps will contribute less to the final solution, effectively loosing any information contained in these maps.

We would effectively like a technique, where all the channels with sufficient signal to noise ratio, provide information. We first rebeam all the maps to gaussian beams with different resolutions. We then adopt the following technique.

- Rebeam all maps to lowest resolution map and apply the ILC algorithm on the maps. This solution will have the lowest foregrounds at that resolution since it uses the information from the maximum number of channels. Let's label the ILC solutions by index  $a$ , i.e.  $s^a(p)$ , with  $a = 1$  corresponding to the lowest resolution, and  $a > b$  implying resolution of map  $a$  is higher than map  $b$ . We will denote the corresponding (Gaussian) beams with  $b_\ell^a$ .
- We, leave the low resolution channels and perform ILC after rebeaming to the next higher resolution.
- Repeat the above step until not enough frequency channels remain to get a viable ILC solution
- Combine the solution in harmonic space, to get the final solution  $s^f(p)$ , or in harmonic space  $a_{\ell m}^f$ , with the resolution corresponding to the highest resolution channel with beam  $b_\ell^n$ , as follows:

$$a_{\ell m}^f = b_\ell^n \left[ a_{\ell m}^1 + (1 - b_\ell^1) \left( a_{\ell m}^2 + (1 - b_\ell^2) \left( a_{\ell m}^3 + \dots + (1 - b_\ell^{n-1}) \frac{a_{\ell m}^n}{b_\ell^n} \right) \right) \right]. \quad (3.10)$$

In the last term in nested brackets in the above equation, we correct the highest resolution solution with its beam and then use the resulting  $a_{\ell m}$  to fill in only the information not present in the next lower resolution map, hence the factor of  $(1 - b_\ell^{n-1})$ . We keep repeating this process to fill-in the information missing from the previous iterations.

## 3.2 New tSZ maps with Unsupervised Machine Learning

There have been attempts to classify the foregrounds in the data using supervised learning by using simulations to generate the training set[16]. This method involves using the simulations to generate the training data for different classes based on the different foreground models used in the simulation. We then expect the machine learning algorithm to *learn* from these simulations and help us classify the real sky into regions based on it's foreground properties. This method suffers from being model dependent, similar to the parametric models of component separation as the algorithm tends to learn only based on the foreground models fed during the simulation. In order to avoid this, we use unsupervised machine learning which gives us the advantage of clustering the data in a model independent manner. In unsupervised machine learning, we take the hard clustering approach, where each data belongs only to a single class. In order to smooth the boundaries of the partitions like the current ILC algorithms, we repeat the hard clustering multiple times with different seeds. Since various seeds produce different partitions, and there is no a-priori reason to consider one partition over the other, we consider all of them with equal probability. We see that this approach essentially smoothens the boundary and no additional smoothing across the cluster boundaries are necessary.

### 3.2.1 'k-means clustering' method

k-means clustering partitions  $n$  data points into  $k$  clusters, by associating each point to the nearest centroid, which serves as a representation of that cluster. By minimising the inertia, we partition the space into k-regions, which serve as the partition for our clusters to be used in ILC. The initial positions are initialized using k-means++ algorithm in order to avoid the local minimas, which the random initializations can get

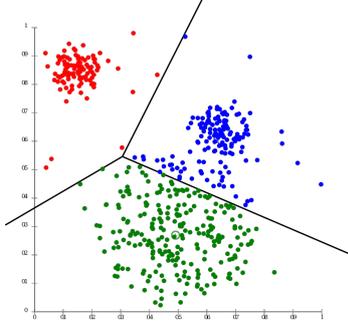


Figure 3.1: In this k-means clustering example, we can see that the points are classified into different clusters based on their distance to the nearest cluster point. This is an example of unsupervised machine learning because, the classifications were not given to the algorithm in advance for it to learn

into. The algorithm spreads out the initial cluster centres as much as possible. The data is rescaled with quartile scaling, to prevent the outliers from influencing the scaling (Refer3.2.2). k-means clustering was implemented using `scikit-learn`[17]

### Choice of Variables for k-means clustering

Let the frequency maps in  $K_{CMB}$  units be denoted by,  $T_i$ , where  $i \in \{30, 44, 70, 100, 143, 217, 353, 545, 857\}$ Hz. We obtain the frequency maps from Planck collaboration's second data release. We use both the high frequency and low frequency maps together, and the difference in resolution of these maps, is taken care of by the methodology explained before (Section 3.1.1)

Since the essential idea is to cluster the pixels based on their foreground properties. We capture the foregrounds by subtracting subtracting the spectra of both our signals of interest from our frequency maps. We first subtract the CMB.  $A_i = T_i - T_{100Hz}$ . Then we subtract out the tSZ, by changing to the corresponding  $y$  units. The maps we get are mostly dominated by foregrounds. Apart from clustering these foreground maps separately (labeled as *raw k-means*), we divide out the amplitudes of these *raw maps* to make different measures, which capture how fast the foreground is increasing or decreasing with frequency similar to the measure defined in the single component foreground clustering approach previously developed[15] (labelled as *kmeans with m*),

$$\begin{aligned}
 y_1 &= A_{857} - 24.371A_{143} & y_2 &= A_{857} - 7.199A_{217} & y_3 &= A_{545} - 14.836A_{143} \\
 y_4 &= A_{545} - 4.3826A_{217} & y_5 &= A_{353} - 8.215A_{143} & y_6 &= A_{353} - 2.4267A_{217} \\
 y_7 &= A_{30} - 1.7339A_{70} & y_8 &= A_{44} - 1.541A_{70} & y_9 &= A_{217} + 3.671A_{44} \\
 y_{10} &= A_{217} + 5.657A_{70} & y_{11} &= A_{353} + 13.73A_{70} & y_{12} &= A_{30} - 1.125A_{44}
 \end{aligned}
 \tag{3.11}$$

Apart from clustering these  $y$ -maps separately. We divide out the amplitudes to make different measures, which capture how fast the foreground is increasing or decreasing with frequency,

$$\begin{aligned}
 m_1 &= y_1/y_3 & m_2 &= y_2/y_4 & m_3 &= y_3/y_5 \\
 m_4 &= y_4/y_6 & m_5 &= y_7/y_{12} & m_6 &= y_{12}/y_5 \\
 m_7 &= y_7/y_5 & m_8 &= y_7/y_8 & m_9 &= y_8/y_5
 \end{aligned}
 \tag{3.12}$$

By visually inspecting the maps, we choose the 3 measures.  $m_1, m_3, m_7$ , and perform k-means clustering (Appendix A) We also masked the galactic center along with the point sources and clustered the masked regions separately.

### 3.2.2 Preprocessing

Before we do machine learning, we need to pre-process the data to remove outliers and heavily contaminated regions. For this, we first mask the sky and separate the masked and the un-masked regions. We then

perform machine learning on these two regions separately. Since all our cost functions are metric based, we need to scale all the different frequency maps into the same range. This makes sure that all the information in the different frequency maps are used to the same extent.

While there are many ways to perform this scaling, we specifically use quartile scaling, which is robust to the presence of outliers. Quartile scaling, scales using the percentile values. In our case, we scale the region between 15-percentile and 85-percentile to a range of 0 to 1. This makes sure that all the frequency maps are prioritized equally.

### Masks

In order to mask the regions which will be dominated by foregrounds, we create masks which cover different fractions of the sky. We use the Linearized Iterative Least squares method[10] to fit the spectra to a foreground model and mask 14% of the sky, based on the  $\chi^2$  value.[18] While calculating the power spectrum,

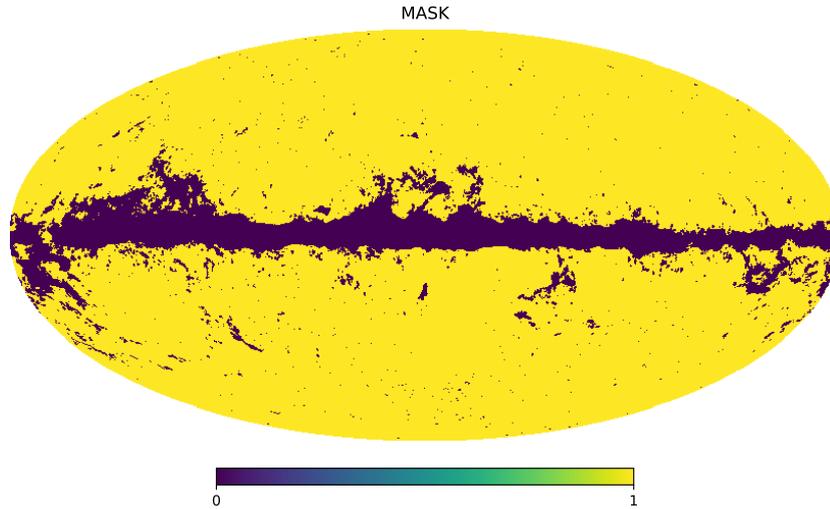


Figure 3.2: A mask where 14% of the sky is covered. We cluster the masked region and the unmasked region separately, since the foreground dominant masked region may affect the clustering by the presence of outliers.

we use a stricter condition on the masks (60% of the sky), revealing a lower fraction of the sky. We apodize the masks by replacing the 1s in the mask by  $1 - \exp(-9\theta^2/(2\theta_{ap}^2))$  for  $\theta < \theta_{ap} = 30'$ , the apodization angle, where  $\theta$  is the distance from the nearest masked pixel.

### 3.2.3 Dimensionality Reduction

We use non-linear dimensionality reduction algorithms in order to reduce the 12 dimensional input data (CMB and tSZ subtracted maps) to a lower dimension in order to apply the k-means algorithm on it. Unlike the previous method, we use neural networks to determine the parameters which encode the maximum information about the data. We use two non-linear dimensionality reduction algorithms.

The two algorithms used are,

- Auto Encoders
- Self Organising Maps

### 3.2.4 Auto Encoders

Auto Encoder is a neural network where, the output dimensionality is the same as input dimensionality. With a hidden layer having a smaller dimension. We train the network by use the same data for both input

and output. The idea is to make the network learn to reduce the dimensionality to a lower dimension and reconstruct the input data from that (known as encoder and decoder respectively).

We use a denoising auto encoder, made from an fully connected neural network, using  $L^2$ -norm as the cost. There are two intermediate layers for both the encoder and the decoder containing seven and three neurons each. Once the network has been trained we can use the encoder part for dimensionality reduction. And then the k-means algorithm is used in this lower dimensional space to cluster the pixels. Auto encoders were implemented using both `TensorFlow` and `PyTorch` to test out different architectures[19, 20]. We vary the learning rate of the neural network by decaying the learning rate, exponentially. The rate of decay is also considered a hyper-parameter. The hyper-parameters were tuned by performing a random search trying to minimise the cost as much as possible.

### 3.2.5 Self Organising Maps

Self organising map reduces the input data to two dimensions by fitting a discrete two dimensional manifold on the data and then use k-means algorithm on this manifold. Self Organising Maps was implemented using the `somoclu` Library[21]. We used a  $200 \times 200$  grid for testing the performance on FFP6 data, and a  $300 \times 300$  grid for the clustering using the actual Planck Data.

## 3.3 Results

After the maps are generated using the above algorithms, We compute the power spectrum using `PolSpice`[22, 23]. The result of using the various algorithms FFP6 simulations is shown in Fig. 3.4. We see that our

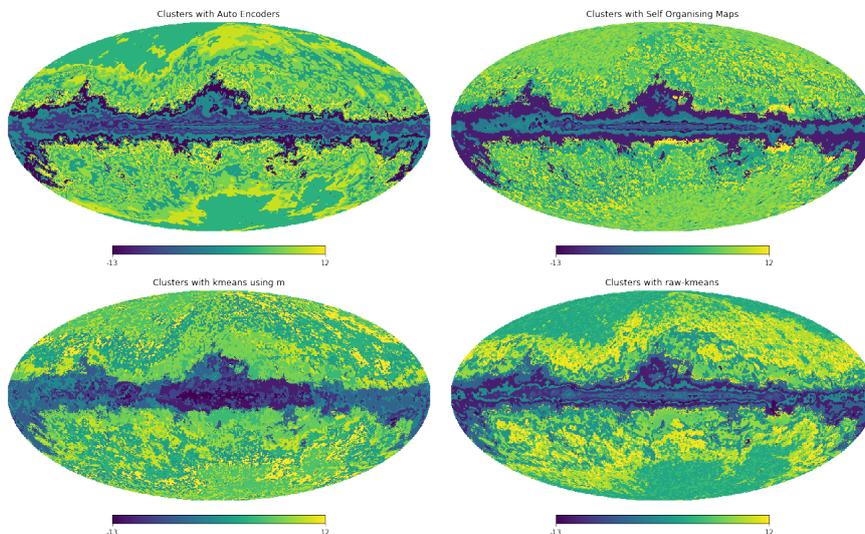


Figure 3.3: A single instance Clustering of the Sky based on foregrounds using various methods (negative values indicate masked regions)

clustering algorithm is better than the one dimensional version[15]. We hope to compare it with the existing algorithms using FFP6 simulations. We also see that the neural network method like self organising map with a  $200 \times 200$  grid performs, very close to the k-means algorithm where the measures are chosen by hand. It is interesting to note that *raw-kmeans* performs the same as a self organising map. Seeing that the k-means algorithm using the m-values as input, performs the best out of the various algorithms explored, We use that on the frequency maps made available by the Planck collaboration. From Fig. 3.5, We can see that the new machine learning based method performs better than the existing model based method[10]. Further analysis is necessary to understand the differences between the two maps.

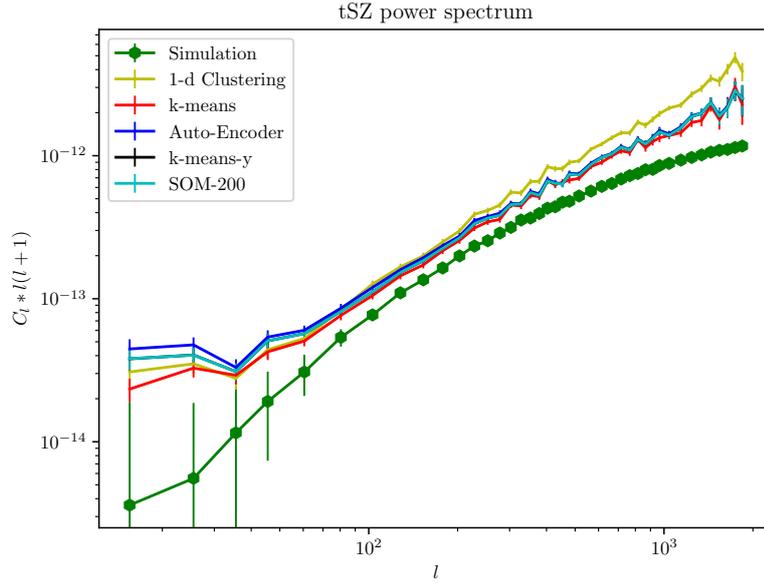


Figure 3.4: Comparison of the tSZ power spectrum for various methods. In this figure, k-means refers to power spectrum of maps generated using k-means on  $m$ -values. k-means-y refers to directly clustering using the foreground maps. SOM-200 refers to a self organising map with a grid size of  $200 \times 200$ . We see that all the other machine learning algorithms perform similarly, and better than the one parameter clustering [15]. We can also see that k-means clustering using  $m$ -values seem to perform better than the methods which involve directly using the foreground frequency maps as input

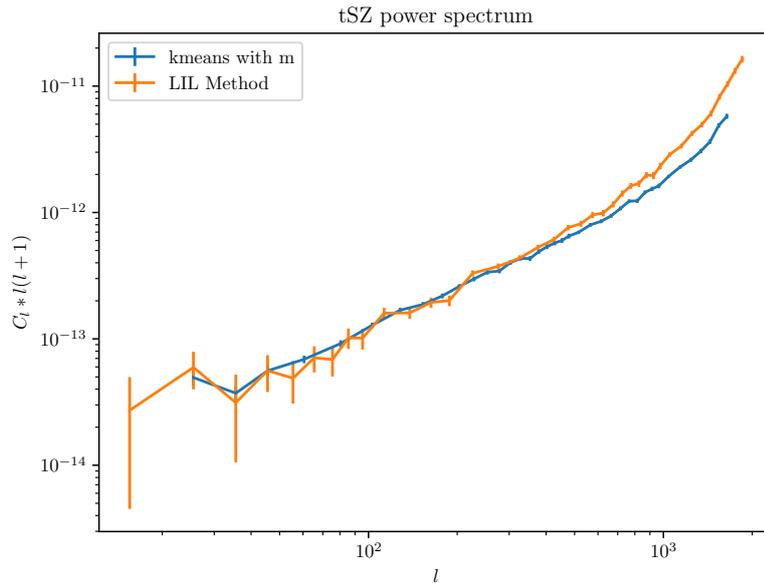


Figure 3.5: Comparison of the tSZ power spectrum on Planck data using Linearized Iterative Least-squares (LIL) method and k-means clustering using  $m$ -values as inputs. Considering that the large  $\ell$  exponential increase in power spectra is because of the noise, we see that the machine learning based approach performs better

## Chapter 4

# Thermal-SZ Weak Lensing Cross Correlation

Cross-correlation is one of the most powerful probes available to extract the mass distribution of diffuse components in the universe. Since, cross-correlation has the advantage of being immune to systematic effects. We attempt to utilize our new tSZ maps to extract information from the diffuse components. We compute the cross-correlation while working in the configuration space. This is because, since the galaxy sky surveys cover a small portion of the sky, the error associated with using a pseudo- $C_\ell$  method by deconvolving the mask is very high. One of the advantages of working in the configuration space is that the mask is directly taken care of, since we compute the correlation functions only in the small patches of the sky available to us. The disadvantage is the computational cost for computing the correlation functions in massive data sets.

This chapter is divided into the following parts, In the first section describes the Lensing survey datasets used and the systematics involved in computing the correlation functions (Section 4.1), We then describe the results (Section 4.2) of our attempts to compute the correlation function. We compute the correlation functions for both the Planck SkyMaps and the skymaps obtained by us in the previous section. We compare the results with the cross-correlation between RCSLenS and Planck Skymaps[1]. We also provide the first attempt in computing the cross-correlation between the tSZ skymaps and the KiDS Survey. In the final section, We review the theory of Halo Models which we plan to do next, in an attempt to interpret the results (Section 4.3).

## 4.1 Calculation from Data

### 4.1.1 Observational Data and Sampling

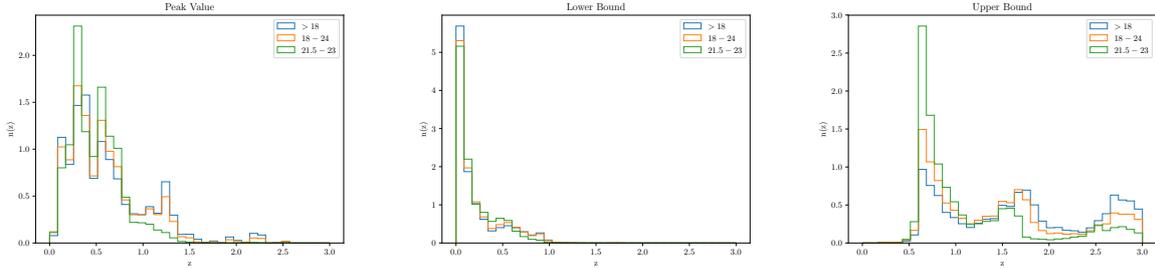
#### RCSLenS Survey

RCSLenS is the largest public multi-band survey, which is suitable for weak gravitational lensing measurements[24] It's parent survey, RCS2[25] is sub-arcsecond multiband imaging survey in the  $g,r,i,z$  bands. It covers a total area of  $785 \text{ deg}^2$ , to a depth of 24.3 mag in the r-band (for a point source at  $7\sigma$ ), carried out with a MegaCAM camera mounted on the Canada France Hawaii Telescope (CFHT). This area is divided into 14 square patches, from hereon referred to as *RCSLenS Fields*. The largest of the fields being  $100 \text{ deg}^2$  and the smallest being  $36 \text{ deg}^2$ . Each square represents a mosaic, which consists of multiple pointings of the  $1 \text{ deg}^2$  camera field of view. RCSLenS is reanalyzes the data with a dedicated weak-lensing pipeline.

The RCSLenS team used the methodologies developed by the CHFTLenS team, which performed data reduction with THELI[26]. , Bayesian Model fitting for shear measurement using Lensfit[27], gaussianised photometry for accurate photometric redshifts[28, 29], Robust field selection based on systematic error analyses[30]. More details can be found in the original paper[24].

In order to filter out only galaxies from our data. We use the following rules,

- We remove entries with  $\text{MASK} > 1$
- We only select entries with  $\text{weight} > 0$  and  $\text{fitclass} = 0$
- We perform a magnitude cut of  $\text{mag}_r > 18$  which includes all the points in the survey.

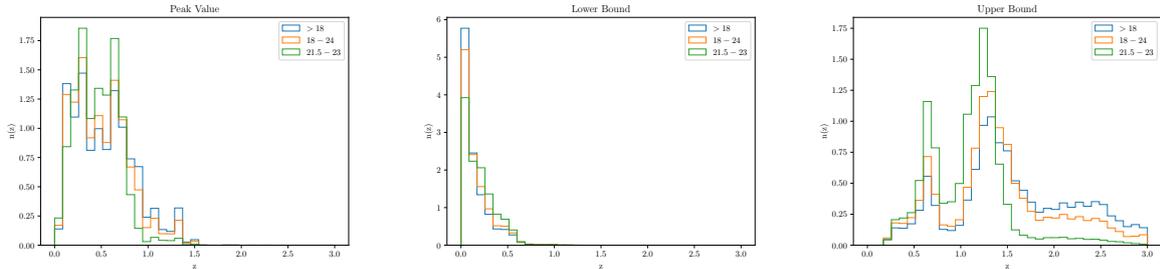


(a)  $n(z)$  using peak of the posterior redshift probability distribution (b)  $n(z)$  using lower bound with a 95% confidence interval (c)  $n(z)$  using upper bound with a 95% confidence interval

Figure 4.1: In this figure, We show the normalized  $n(z)$  for different r-band magnitude ( $\text{mag}_r$ ) cuts for the RCSLenS. The posterior redshift probability distribution were found using an BPZ estimate[28]. In our work we work with  $\text{mag}_r > 18$ , Since it doesn't have an impact on the SNR[1].

## KiDS Survey

We use the shear measurements from the Kilo Degree Survey's KiDS-450 dataset.[31, 32, 33] KiDS is an ongoing ESO optical survey which will eventually cover  $1350\text{deg}^2$  of the sky. Currently, the dataset covers of  $449.7\text{deg}^2$  of the sky, in 5 different patches. It is carried out using the OmegaCAM CCD mosaic camera mounted at the Cassegrain focus of the VLT Survey Telescope. The data processing consists of ASTRO-Wise[34, 35] for data reduction from individual exposures in multiple colours to photometry, and THELI[26, 36, 37] for lensing specific data reduction of  $r$ -band images. The shears calculated using Bayesian model fitting using *lensfit*[27], and the photometric redshifts are obtained from PSF-matched photometry and calibrated using external overlapping spectroscopic surveys[28, 32]



(a)  $n(z)$  using peak of the posterior redshift probability distribution (b)  $n(z)$  using lower bound with a 95% confidence interval (c)  $n(z)$  using upper bound with a 95% confidence interval

Figure 4.2: In this figure, We show the normalized  $n(z)$  for different r-band magnitude ( $\text{mag}_r$ ) cuts for the KiDS survey. The posterior redshift probability distribution were found using an BPZ estimate[28]. Here too we work with  $\text{mag}_r > 18$ . We note that the KiDS' galaxies are farther away than the RCSLenS galaxies (Fig. 4.1)

## tSZ Skymaps

For the tSZ skymaps, We use the maps generated by us as explained in the previous chapter and also the skymaps provided in the Planck 2015 Public Data Release[38]. Since tSZ skymaps are all skymaps, We use

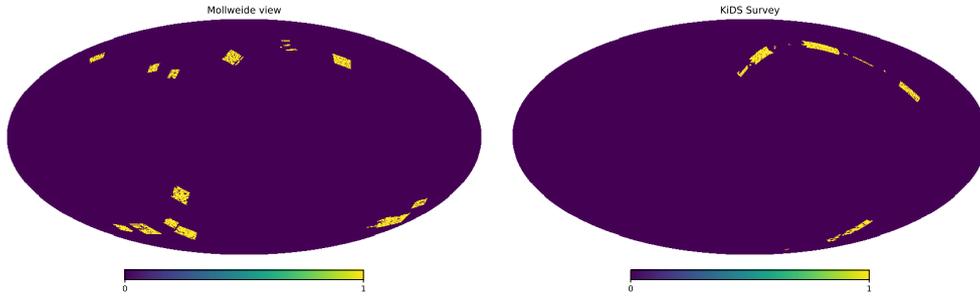


Figure 4.3: A visual representation of the RCSLenS and KiDS fields, in galactic coordinates, using Mollweide projection.

the entire skymap to compute the correlation function in order to provide a large correlation area around the RCSLenS fields, which reduces the statistical noise.

### Sampling

We compute the correlation function separately for each of the 14 RCSLenS fields or 5 KiDS fields. We then create jackknife samples of these results and then compute the mean and the standard deviation of these samples.

#### 4.1.2 Two Point Correlation Functions

Now, In order to compute the two-point correlation functions we work in the configuration space. For  $y - \gamma_T$ , we compute the two point correlation function as,

$$\xi^{y-\gamma_T}(\theta) = \frac{\sum_{ij} y^i e_t^{ij} w^j \Delta_{ij}(\theta)}{\sum_{ij} w^j \Delta_{ij}(\theta)} \quad (4.1)$$

where,  $y^i$  is the  $y$ -value from the tSZ maps in pixel  $i$ . And  $e^{ij}$  is the tangential ellipticity of the galaxy  $j$  in the catalogue with respect to pixel  $i$ . The tangential ellipticity is corrected for both multiplicative and additive bias.  $\Delta_{ij}(\theta)$  imposes our binning scheme. It is one if the angular separation between  $i$  and  $j$  is  $\theta$  and zero otherwise, and  $w^j$  is the *lensfit* weight (For definition see Reference[39]).

#### 4.1.3 Bias Correction

During the measurement of the ellipticities there exists calibration corrections to account for biases in our measurement. These are modelled by a multiplicative term and an additive term such that,

$$g_i^{obs} = (1 + m)g_i^{true} + c \quad (4.2)$$

Estimates of these biases from image simulations are given to us as part of the catalogue and we need to correct for them before computing the correlation functions. The RCSLenS catalogue contains two additive biases, Detector Bias and Noise Bias, whereas the KiDS catalogue contains only a multiplicative bias which needs to be corrected for. More information can be found in the respective catalogue information.

#### 4.1.4 Systematic Tests

In order to test for systematics in the data, We perform 2 systematic tests.

- **We calculate  $\langle y\gamma_x \rangle$ :** We can compute this by rotating the sources by  $45^\circ$  (ie,  $e_{1,new} = e_{2,old}$ ) and  $e_{2,new} = -e_{1,old}$ . We saw in a previous section how this will always be zero for spherically symmetric mass distributions Section 2.1.

- **We randomise the catalogue and find the correlation once again:** Since the catalogues are randomized (un-correlated), The correlation functions will become zero. Any deviations from zero, will show the existence of biases in our methodology.

in the absence of systematic errors, in our calculation. Both of these tests should give us values close to zero.

### 4.1.5 Effect of Masks

Since, we are computing cross-correlations, the effect of point sources and foregrounds which are uncorrelated with the signal should not have an effect on our computations. We verify this by computing the cross-correlation with various sky masks provided by the Planck Collaboration[38]. These masks are used to avoid point sources and various other foreground contaminations in our data. We find that these masks does not have an effect on the cross-correlation computed by us.

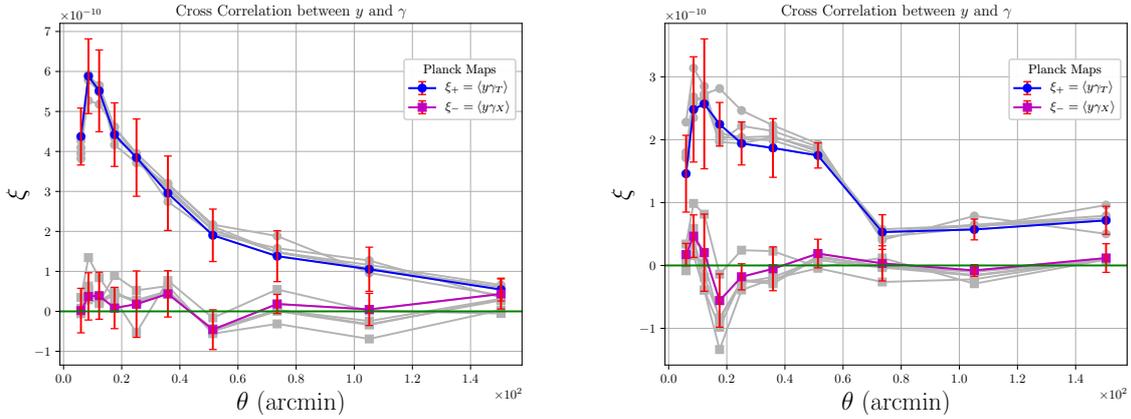


Figure 4.4: We see that the masks does not play a role in the computation of cross-correlations since the foregrounds and point sources are uncorrelated with the diffuse component signal we are looking for. (a) RCSLenS Catalogue (b) KiDS Catalogue

## 4.2 Results

Now, computing the cross-correlation using the Planck sky maps and our own sky maps; we get the results presented in Figs. 4.5 and 4.6.

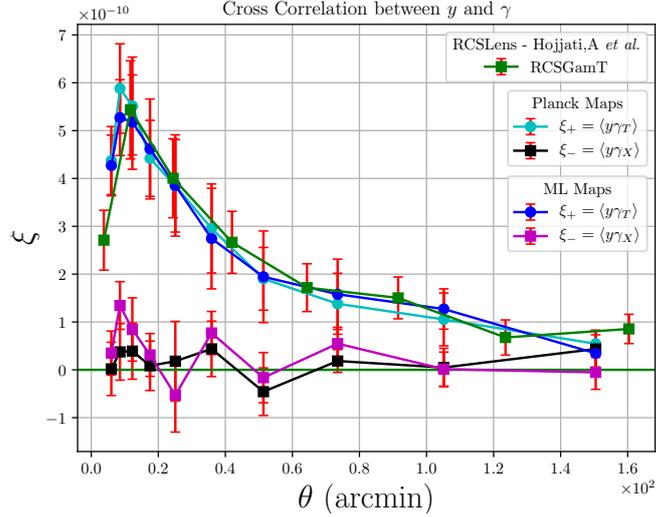


Figure 4.5: Cross Correlation between tSZ and Weak Lensing maps for RCSLenS fields. The two sets of data (Connected Lines) represent the *Planck skymaps* and the skymaps we generated using unsupervised machine learning techniques (ML Maps). Comparing it to the existing results,[1] our peak value seems to be smaller. We would like to point out that since the correlations seem to agree with different skymaps, The difference could be only in systematics and uncorrelated with the signal.

Since, as shown in Fig. 4.5, we see that the deviation between the two skymaps, don't manifest when computing the cross-correlation functions. This is indicative of the fact that the results can be a systematic deviation or noise uncorrelated with the signal. We would also like to point out the difference between the KiDS cross-correlation (Fig. 4.6) and RCSLenS cross-correlation such as a smaller peak value of the correlation and a sharper fall, which could be a artifact of the fact that the KiDS' galaxies are farther away than the RCSLenS galaxies (Figs. 4.1 and 4.2)

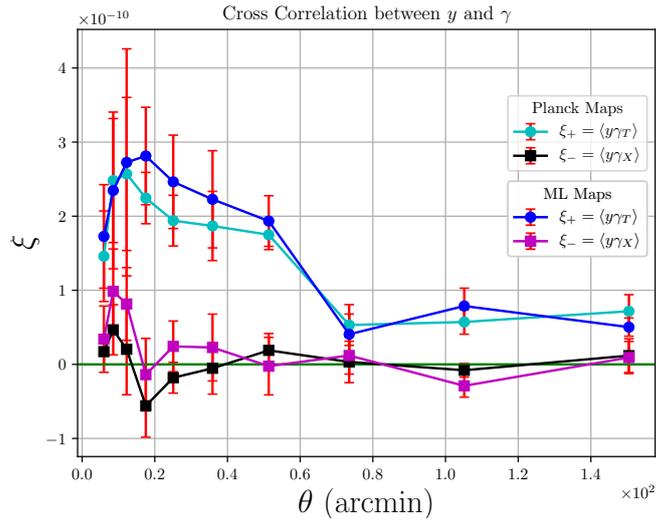


Figure 4.6: Cross Correlation between tSZ and Weak Lensing maps for KiDS fields. The Two sets of data represent the *Planck skymaps* and the Skymaps we generated using unsupervised machine learning techniques (ML Maps). We would like to point out that since the correlations seem to agree with different skymaps, The difference could be only in systematics and uncorrelated with the signal. We would like to point out that the peak seems to be smaller and broader than the RCSLenS fields. This could be because of the fact that, the Galaxies part of the KiDS survey are closer than the RCSLenS survey

### 4.3 Comparison with theory

We like to compare our observations with theoretical predictions based on Halo models, closely following the method developed by[2]. Currently we are concentrating on computing the real space cross correlation  $\xi = \langle \gamma_T - y \rangle$ , By using,

$$\xi^{y-\gamma_T}(\theta) = \int \frac{d^2\vec{l}}{2\pi^2} C_l^{y-k} \cos(2(\phi - \psi)) \exp(i\theta \cos(\phi - \psi)) \quad (4.3)$$

Where,  $\phi$  is the Polar angle with respect to the coordinate system and  $\psi$  is the angle between  $\vec{l}$  and the coordinate.

In order to compute the  $y - k$  cross correlation power spectra found in Eq. (4.3), we use the 1-halo term as defined in[40],

$$C_l^{y-k,1h} = \int_0^{z_{\max}} dz \frac{dV}{dz d\Omega} \int_{M_{\min}}^{M_{\max}} dM \frac{dn}{dM} y_l(M, z) k_l(M, z) \quad (4.4)$$

For the halo mass function, We use the form suggested by Sheth and Tormen[41] For the convergence profile in fourier space, We use

$$k_l = \frac{W^k(z)}{\xi^2(z)} \frac{1}{\rho_m} 4\pi \int_0^{r_{vir}} dr r^2 \frac{\sin(lr/\xi)}{lr/\xi} \rho(r; M, z) \quad (4.5)$$

And for the fourier transform of the projected gas pressure,

$$y_l = \frac{4\pi r_s}{l_s^2} \frac{\sigma_T}{m_e c^2} \int dx x^2 \frac{\sin(lx/\xi)}{lx/\xi} P_e(x; M, z) \quad (4.6)$$

For electron pressure, We use the *universal pressure profile* and the NFW model. We plan to use the best fit parameters for the Pressure profile from the Planck Collaboration and then compare the halo model predictions from both the Planck and WMAP-7yr cosmologies. To look for any non-gravitational feedback, we plan to use the method developed by[42], applied to tSZ  $C_{ls}$ .

# Chapter 5

## Conclusion

To summarize, we used a model independent algorithm using unsupervised machine learning for the problem of component separation. The algorithm was developed independent of Planck Collaboration's pipelines. We saw that the various machine learning algorithms used for component separation improves upon the existing foreground clustering algorithm, whose performance was on par with the Planck Collaboration's pipelines[15]. We saw that, using unsupervised machine learning, we perform better than the model based methods too[10]. We hope to improve upon the dimensionality reduction algorithms, to eliminate the need for making a choice of variables for performing the clustering. This would help in extracting the tSZ skymaps from the frequency maps in a truly model independent basis. We would also like to point out that the GILC method is robust for extracting any signal with a known spectra, and therefore our method is useful in extracting any signal in a model independent basis. We hope that, using an independent algorithm to produce maps with different residuals, would be useful in testing for the effect of foregrounds and any biases in the existing algorithms, for the estimates of the cosmological parameters.

In the second part of the work, we computed cross-correlation of tSZ with weak lensing surveys to study diffuse baryons in halos. We saw that the different maps generated do not have an effect on the cross-correlation, showing that it is a systematic deviation which is uncorrelated with the signal. We also performed the first ever computation of cross-correlation between tSZ skymaps and weak lensing shear from the KiDS survey, which detects hot baryons. We are currently attempting to make sense of this by theoretically modelling the diffuse baryons in halos. We hope to get better constraints on halo astrophysics and cosmology, and also better understand the systematic deviations from the existing Planck skymaps.

# Appendix A

## m-maps

Looking at the m-maps we can say that,  $m_5$  and  $m_8$  are very noisy.  $m_1, m_2, m_4$  have similar information and  $m_6, m_7, m_9$  have similar information. And  $m_9$  is also more noisier than  $m_7$ . Therefore, We choose  $m_1, m_3$  and  $m_7$ .

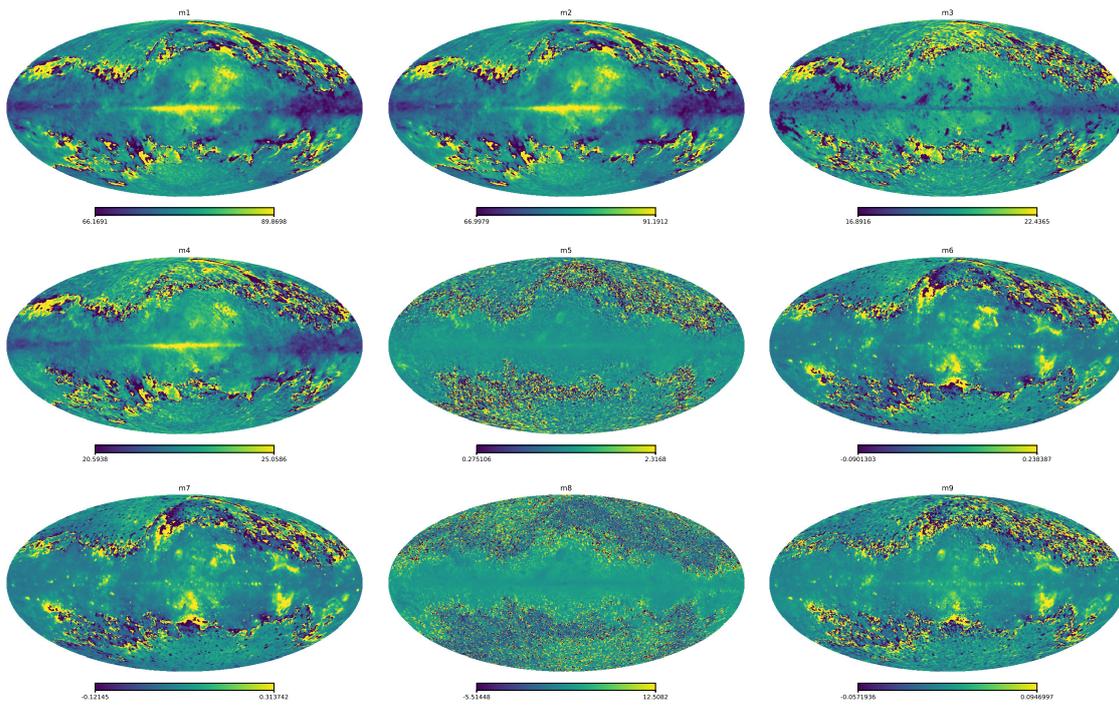
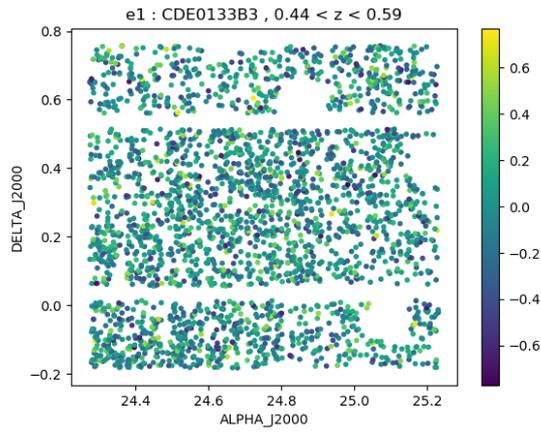


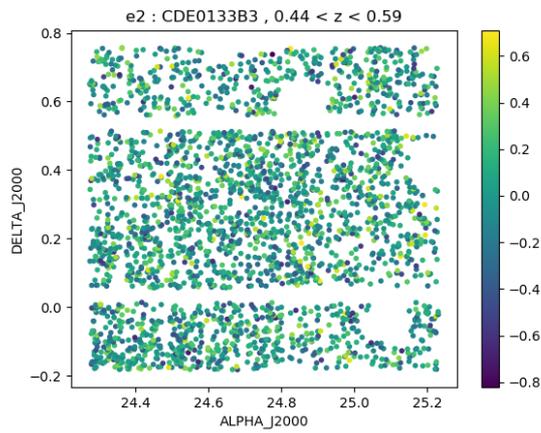
Figure A.1: The m-maps for planck data (these maps are plotted only from 5 to 95 percentile to avoid the outliers from scaling the plot)

# Appendix B

## A single RCSLenS Field



(a) A heatmap of ellipticity  $e_1$



(b) A heatmap of ellipticity  $e_2$

Figure B.1: A single RCSLenS field, for a particular  $z$  bin. We could see the masking of stellar halos and other instrumental artifacts in these. Factors like these, also contribute to the reason why Pseudo- $C_\ell$  are bad for working with galaxy surveys, since they reduce the skyfraction even more

# Bibliography

- [1] A. Hojjati et al., *Cross-correlating Planck tSZ with RCSLenS weak lensing: Implications for cosmology and AGN feedback*, *Mon. Not. Roy. Astron. Soc.* **471** (2017) 1565 [1608.07581].
- [2] Y.-Z. Ma, L. Van Waerbeke, G. Hinshaw, A. Hojjati, D. Scott and J. Zuntz, *Probing the diffuse baryon distribution with the lensing-tSZ cross-correlation*, *JCAP* **1509** (2015) 046 [1404.4808].
- [3] M. Fukugita and P. J. E. Peebles, *The Cosmic energy inventory*, *Astrophys. J.* **616** (2004) 643 [astro-ph/0406095].
- [4] S. Dodelson, *Gravitational Lensing*. Cambridge University Press, 2017, 10.1017/9781316424254.
- [5] PLANCK collaboration, P. A. R. Ade et al., *Planck 2013 results. IX. HFI spectral response*, *Astron. Astrophys.* **571** (2014) A9 [1303.5070].
- [6] K. Hornik, *Approximation capabilities of multilayer feedforward networks*, *Neural Networks* **4** (1991) 251 .
- [7] S. M. Omohundrol and S. M. Omohundro, *Five balltree construction algorithms*, 1989.
- [8] M. Jarvis, G. Bernstein and B. Jain, *The skewness of the aperture mass statistic*, *Mon. Not. Roy. Astron. Soc.* **352** (2004) 338 [astro-ph/0307393].
- [9] H. K. Eriksen et al., *CMB component separation by parameter estimation*, *Astrophys. J.* **641** (2006) 665 [astro-ph/0508268].
- [10] R. Khatri, *Linearized iterative least-squares (LIL): a parameter-fitting algorithm for component separation in multifrequency cosmic microwave background experiments such as Planck*, *Mon. Not. Roy. Astron. Soc.* **451** (2015) 3321 [1410.7396].
- [11] H. K. Eriksen, A. J. Banday, K. M. Gorski and P. B. Lilje, *Foreground removal by an internal linear combination method: Limitations and implications*, *Astrophys. J.* **612** (2004) 633 [astro-ph/0403098].
- [12] M. Remazeilles, J. Delabrouille and J.-F. Cardoso, *Foreground component separation with generalised ILC*, *Mon. Not. Roy. Astron. Soc.* **418** (2011) 467 [1103.1166].
- [13] J.-F. Cardoso, M. Martin, J. Delabrouille, M. Betoule and G. Patanchon, *Component separation with flexible models. Application to the separation of astrophysical emissions*, 0803.1814.
- [14] D. Marinucci, D. Pietrobon, A. Balbi, P. Baldi, P. Cabella, G. Kerkycharian et al., *Spherical Needlets for CMB Data Analysis*, *Mon. Not. Roy. Astron. Soc.* **383** (2008) 539 [0707.0844].
- [15] R. Khatri, *Data driven foreground clustering approach to component separation in multifrequency CMB experiments: A new Planck CMB map*, *JCAP* **1902** (2019) 039 [1808.05224].
- [16] H. U. Norgaard-Nielsen and H. E. Jorgensen, *Foreground removal from CMB temperature maps using an MLP neural network*, *Astrophys. Space Sci.* **318** (2008) 195 [0809.2914].
- [17] F. Pedregosa et al., *Scikit-learn: Machine Learning in Python*, *J. Machine Learning Res.* **12** (2011) 2825 [1201.0490].

- [18] R. Khatri, *An alternative validation strategy for the Planck cluster catalogue and y-distortion maps*, *Astron. Astrophys.* **592** (2016) A48 [1505.00778].
- [19] M. Abadi et al., *TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems*, 1603.04467.
- [20] A. Paszke, S. Gross et al., *Automatic differentiation in PyTorch*, in *NIPS Autodiff Workshop*, 2017.
- [21] P. Wittek, S. Gao, I. Lim and L. Zhao, *somoclu: An efficient parallel library for self-organizing maps*, *Journal of Statistical Software, Articles* **78** (2017) 1.
- [22] G. Chon, A. Challinor, S. Prunet, E. Hivon and I. Szapudi, *Fast estimation of polarization power spectra using correlation functions*, *Mon. Not. Roy. Astron. Soc.* **350** (2004) 914 [astro-ph/0303414].
- [23] I. Szapudi, S. Prunet and S. Colombi, *Fast clustering analysis of inhomogeneous megapixel cmb maps*, astro-ph/0107383.
- [24] H. Hildebrandt et al., *RCSLenS: The Red Cluster Sequence Lensing Survey*, *Mon. Not. Roy. Astron. Soc.* **463** (2016) 635 [1603.07722].
- [25] D. G. Gilbank, M. D. Gladders, H. K. C. Yee and B. C. Hsieh, *The Red-sequence Cluster Survey-2 (RCS-2): Survey Details and Photometric Catalog Construction*, *Aj* **141** (2011) 94 [1012.3470].
- [26] T. Erben, H. Hildebrandt, L. Miller, L. van Waerbeke, C. Heymans, H. Hoekstra et al., *CFHTLenS: the Canada–France–Hawaii Telescope Lensing Survey – imaging data and catalogue products*, *Monthly Notices of the Royal Astronomical Society* **433** (2013) 2545 [http://oup.prod.sis.lan/mnras/article-pdf/433/3/2545/13761746/stt928.pdf].
- [27] L. Miller, C. Heymans, T. D. Kitching, L. van Waerbeke, T. Erben, H. Hildebrandt et al., *Bayesian galaxy shape measurement for weak lensing surveys – III. Application to the Canada–France–Hawaii Telescope Lensing Survey*, *Monthly Notices of the Royal Astronomical Society* **429** (2013) 2858 [http://oup.prod.sis.lan/mnras/article-pdf/429/4/2858/3268972/sts454.pdf].
- [28] H. Hildebrandt, T. Erben, K. Kuijken, L. van Waerbeke, C. Heymans, J. Coupon et al., *CFHTLenS: improving the quality of photometric redshifts with precision photometry\**, *Monthly Notices of the Royal Astronomical Society* **421** (2012) 2355 [http://oup.prod.sis.lan/mnras/article-pdf/421/3/2355/5838743/mnras0421-2355.pdf].
- [29] N. Benitez, *Bayesian photometric redshift estimation*, *The Astrophysical Journal* **536** (2000) 571.
- [30] C. Heymans, L. van Waerbeke, L. Miller, T. Erben, H. Hildebrandt, H. Hoekstra et al., *CFHTLenS: the Canada–France–Hawaii Telescope Lensing Survey*, *Monthly Notices of the Royal Astronomical Society* **427** (2012) 146 [http://oup.prod.sis.lan/mnras/article-pdf/427/1/146/18231420/427-1-146.pdf].
- [31] K. Kuijken et al., *Gravitational Lensing Analysis of the Kilo Degree Survey*, *Mon. Not. Roy. Astron. Soc.* **454** (2015) 3500 [1507.00738].
- [32] H. Hildebrandt et al., *KiDS-450: Cosmological parameter constraints from tomographic weak gravitational lensing*, *Mon. Not. Roy. Astron. Soc.* **465** (2017) 1454 [1606.05338].
- [33] I. Fenech Conti, R. Herbonnet, H. Hoekstra, J. Merten, L. Miller and M. Viola, *Calibration of weak-lensing shear in the Kilo-Degree Survey*, *Mon. Not. Roy. Astron. Soc.* **467** (2017) 1627 [1606.05337].
- [34] E. A. Valentijn et al., *Astro-WISE: Chaining to the Universe*, *ASP Conf. Ser.* **376** (2006) 491 [astro-ph/0702189].
- [35] K. Begeman, A. N. Belikov, D. R. Boxhoorn and E. A. Valentijn, *The astro-wise datacentric information system*, *Experimental Astronomy* **35** (2013) 1.
- [36] M. Schirmer, *THELI: CONVENIENT REDUCTION OF OPTICAL, NEAR-INFRARED, AND MID-INFRARED IMAGING DATA*, *The Astrophysical Journal Supplement Series* **209** (2013) 21.

- [37] T. Erben, M. Schirmer, J. P. Dietrich, O. Cordes, L. Habertzettl, M. Hettterscheidt et al., *Gabods: The garching-bonn deep survey*, *Astronomische Nachrichten* **326** (2005) 432 [<https://onlinelibrary.wiley.com/doi/pdf/10.1002/asna.200510396>].
- [38] PLANCK collaboration, N. Aghanim et al., *Planck 2015 results. XXII. A map of the thermal Sunyaev-Zeldovich effect*, *Astron. Astrophys.* **594** (2016) A22 [1502.01596].
- [39] L. Miller, C. Heymans, T. D. Kitching, L. van Waerbeke, T. Erben, H. Hildebrandt et al., *Bayesian galaxy shape measurement for weak lensing surveys – III. Application to the Canada–France–Hawaii Telescope Lensing Survey*, *Monthly Notices of the Royal Astronomical Society* **429** (2013) 2858 [<http://oup.prod.sis.lan/mnras/article-pdf/429/4/2858/3268972/sts454.pdf>].
- [40] A. Cooray and R. K. Sheth, *Halo Models of Large Scale Structure*, *Phys. Rept.* **372** (2002) 1 [astro-ph/0206508].
- [41] R. K. Sheth and G. Tormen, *An Excursion Set Model of Hierarchical Clustering : Ellipsoidal Collapse and the Moving Barrier*, *Mon. Not. Roy. Astron. Soc.* **329** (2002) 61 [astro-ph/0105113].
- [42] A. Iqbal, S. Majumdar, B. B. Nath, S. Ettori, D. Eckert and M. A. Malik, *Excess entropy and energy feedback from within cluster cores up to  $r_{200}$* , *Mon. Not. Roy. Astron. Soc.* **472** (2017) 713 [1703.00028].