2022

# Towards Precision Measurements Of The Optical Depth To Reionization Using 21 Cm Data And Machine Learning

Tashalee S. Billings
*University of Pennsylvania*

# Towards Precision Measurements Of The Optical Depth To Reionization Using 21 Cm Data And Machine Learning

## Abstract

The Epoch of Reionization (EoR) was a phase transition from a neutral state to an ionized state where the first generation of luminous objects were able to heat and ionize the surrounding predominantly neutral hydrogen gas. Detection of brightnesstemperature fluctuations from the redshifted hyperfine 21 cm line of neutral hydrogen would provide a direct three-dimensional probe of astrophysics and cosmology during this period. Another important property of reionization is the redshift of its midpoint, when half the hydrogen in the intergalactic medium (IGM) was ionized. This quantity is often estimated by using the cosmic microwave background (CMB) optical depth, tau . Since the optical depth is obtained by integrating along the line of sight, it provides just one number to characterize reionization. As a result, this can be converted into a constraint for the midpoint under the assumption of a parametric form for the ionization history. This is also a probe of the EoR.

Upcoming measurements of the high-redshift 21cm signal from the EoR are a promising probe of the astrophysics of the first galaxies and of cosmological parameters. In particular, the optical depth tau to the last scattering surface of the CMB should be tightly constrained by direct measurements of the neutral hydrogen state at high redshift. A robust measurement of $\tau$ from 21cm data would help eliminate it as a nuisance parameter from CMB estimates of cosmological parameters. Previous proposals for extracting tau from future 21cm datasets have typically used the 21cm power spectra generated by semi-numerical models to reconstruct the reionization history. I present in this thesis a different approach which uses convolution neural networks (CNNs) trained on mock images of the 21cm EoR signal to extract tau. I constructed a CNN that improves upon on previously proposed architectures, and perform an automated hyperparameter optimization. I showed that well-trained CNNs are able to accurately predict tau, even when removing Fourier modes that are expected to be corrupted by bright foreground contamination of the 21cm signal.

I then began answering a slightly different question that involved raining three different Bayesian models using mock images of ionized fields of hydrogen to extract the ionization fraction of hydrogen by only looks at one redshift to infer the ionization fraction of each simulated image. I showed that for a simple fully Bayesian network it is possible to successfully produces predicted values that are closely aligned with the true values and the model was tuned to find the ``best" generalized model architecture for this particular problem.

## Degree Type

Dissertation

## Degree Name

Doctor of Philosophy (PhD)

## Graduate Group

Physics & Astronomy

## First Advisor

James E. Aguirre

## Subject Categories

Astrophysics and Astronomy

# TOWARDS PRECISION MEASUREMENTS OF THE OPTICAL DEPTH TO REIONIZATION USING 21 CM DATA AND MACHINE LEARNING

Tashalee Billings

A DISSERTATION

in

Physics and Astronomy

Presented to the Faculties of the University of Pennsylvania

in

Partial Fulfillment of the Requirements for the

Degree of Doctor of Philosophy

2022

Supervisor of Dissertation

James E. Aguirre, Associate Professor of Physics and Astronomy

Graduate Group Chairperson

Ravi Sheth, Professor of Physics and Astronomy

Dissertation Committee

Gary Bernstein, Reese W. Flower Professor of Astronomy and Astrophysics
Cullen Blake, Associate Professor of Physics and Astronomy
Masao Sako, Professor and Undergraduate Chair of Physics and Astronomy
Evelyn Thomson, Professor of Physics and Astronomy

# ACKNOWLEDGEMENT

James, thank you so much (from the bottom of my heart) for accepting me into your group and really taking the time to hold my hand through the beginning. I knew nothing about analysis and had zero experience with any programming language. You taught me how to think and problem solve. You may not remember how long of a way I've come, but I certainly do. Working with you has allowed me to develop the confidence in my ability to learn difficult subjects. After leaving your group, I'm confident I can learn anything if I really try. I realized more than ever that this was the best thing and I will eventually use it to help other people in my community.

It has been a privilege to work alongside other individuals within the HERA collaboration. Thank you so much Paul La Plante. He is the computer whisperer and somehow makes ever single thing work. Zac Martinot, thank you for taking the time to explain things to me. Saul Kohn, you're a great friend and I definitely look up to you! Thank you Jonathan Pober and Adrian Liu for your perspective on machine learning.

Thank you to the Graduate School for awarding me with the Fontaine Fellowship during my first year here at Penn. Thank you to the Department of Physics and Astronomy for awarding me with the teaching Fellowship during my second year and the School of Arts and Sciences for supporting my final year with a Dissertation Completion Fellowship. All three of these fellowships were very much needed and allowed me to have the freedom to explore.

Thank you to the Department of Physics and Astronomy of the University of Pennsylvania for giving me an opportunity to contribute to thee scientific community and introducing me to such amazing people. I've learned so much and I'm incredibly proud of myself. I've made some lifelong friends through the program and just living in Philadelphia.

Thank you David Spergel. That summer internship in your group made me realize that I am Ivy League material and that I should apply to all of the top PhD programs. Kalayu Belay, thank you so much for being an amazing mentor to me throughout my undergraduate career. You nurtured me throughout the years and helped me get those high grades that allowed me to be qualified for PhD programs like Penn. Thank you to all of my professors at Florida A&M University and here at Penn.

I want to thank my family! Mom, I know you wanted me to be a medical doctor so incredibly badly, and Dad, you wanted me to be a business owner like you because that is the American dream, but I decided to do something a little different. Thank you for abandoning the typical immigrant mentality and giving me your blessings to be something other than a medical doctor or lawyer. Still a "Dr." though! Gramsy and Gramps I love you with every fiber in my body (more than Mom and Dad but that's no secret haha). Thank you for your prayers over me and my life. Aunt Andrea, I love you I love you! You're my big sister and my emotional support system. Aunt B and Uncle Roy a.k.a my Philly "Parents". I love you and thank you so much for being there for me while my parents are living their best life in a warmer climate. You have both been a support system and a constant reminder that I still need to fulfill my duties as an immigrant child and achieve they best of the best haha. Thank you for all the yummy Jamaican food that Mom could not bring to me. My (not so baby) baby brothers, I love you guys even though we don't get to see each other often and you HARDLY text. It's like pulling teeth with you two! Anyway, I'm the first born and therefore thee leader of us and I want you to know you can do whatever you want. Mom and Dad had no idea how to raise me and I turned out pretty good. You will do better than me.

# ABSTRACT

TOWARDS PRECISION MEASUREMENTS OF THE OPTICAL DEPTH TO REIONIZATION USING 21 CM DATA AND MACHINE LEARNING

Tashalee Billings

James E. Aguirre

The Epoch of Reionization (EoR) was a phase transition from a neutral state to an ionized state where the first generation of luminous objects were able to heat and ionize the surrounding predominantly neutral hydrogen gas. Detection of brightness temperature fluctuations from the redshifted hyperfine 21 cm line of neutral hydrogen would provide a direct three-dimensional probe of astrophysics and cosmology during this period. Another important property of reionization is the redshift of its midpoint, when half the hydrogen in the intergalactic medium (IGM) was ionized. This quantity is often estimated by using the cosmic microwave background (CMB) optical depth, $\tau$ . Since the optical depth is obtained by integrating along the line of sight, it provides just one number to characterize reionization. As a result, this can be converted into a constraint for the midpoint under the assumption of a parametric form for the ionization history. This is also a probe of the EoR.

Upcoming measurements of the high-redshift 21 cm signal from the EoR are a promising probe of the astrophysics of the first galaxies and of cosmological parameters. In particular, the optical depth $\tau$ to the last scattering surface of the CMB should be tightly constrained by direct measurements of the neutral hydrogen state at high redshift. A robust measurement of $\tau$ from 21 cm data would help eliminate it as a nuisance parameter from CMB estimates of cosmological parameters. Previous proposals for extracting $\tau$ from future 21 cm datasets have typically used the 21 cm power spectra generated by semi-numerical models to reconstruct the reionization history. I present in this thesis a different approach which uses convolution neural networks (CNNs) trained on mock images of the 21 cm EoR signal to extract $\tau$. I constructed a CNN that improves upon on previously proposed architectures, and perform an automated hyperparameter optimization. I showed that well-trained CNNs are able to accurately predict $\tau$, even when removing Fourier modes that are expected to be corrupted by bright foreground contamination of the 21 cm signal.

I then began answering a slightly different question that involved raining three different Bayesian models using mock images of ionized fields of hydrogen to extract the ionization fraction of hydrogen by only looks at one redshift to infer the ionization fraction of each simulated image. I showed that for a simple fully Bayesian network it is possible to successfully produces predicted values that are closely aligned with the true values and the model was tuned to find the "best" generalized model architecture for this particular problem.

# TABLE OF CONTENTS

# LIST OF TABLES

# LIST OF ILLUSTRATIONS

# CHAPTER 1

# COSMIC HISTORY

## 1.1. A Brief History of The Universe

Shortly after the Big Bang, The Universe quickly expanded into a hot dense opaque primordial soup of quarks, leptons, gluons and extremely energetic photons. Tiny random fluctuations in the values of the electric and magnetic fields of the elementary particles represent the electromagnetic force are carried by photons; W and Z fields carry the weak force; and gluon fields which carry the strong force. This random change in the in the energy state of a point in space is called a quantum (vacuum) fluctuations. Density anisotropies formed by these hugely inflated quantum fluctuations, ionized mostly hydrogen and helium; in addition to that deuterium, lithium and beryllium. These ionized atoms filled The Universe as a hot plasma. Prior to this, the unbound and densely packed electrons were exceptionally good at absorbing and re-emitting light. Blackbody photons, thermal electromagnetic radiation within thermodynamic equilibrium with its environment, were continuously scattered throughout this plasma.

About 380,000 years after the Big Bang, the number of photons with energies above the 13.6 eV threshold required to ionize neutral hydrogen (astronomers refer to neutral hydrogen as HI, and ionized hydrogen [i.e. protons] as HII) became outnumbered by the number of baryons, and HII were able to recapture electrons without immediately being ionized. This critical period is known as recombination. The plasma was able to neutralize and The Universe underwent another cosmic transition as it adiabatically expanded from optically thick to optically thin as free electrons were captured allowing light to travel unimpeded, in straight lines for the first time. Some of the unimpeded photons that existed at the time of the 'last scattering', redshifted by the expansion of The Universe, are observed today as the Cosmic Microwave Background (CMB). The Universe we see today is structured, complicated and diverse in stars and galaxies, dark matter and dark energy, but very little

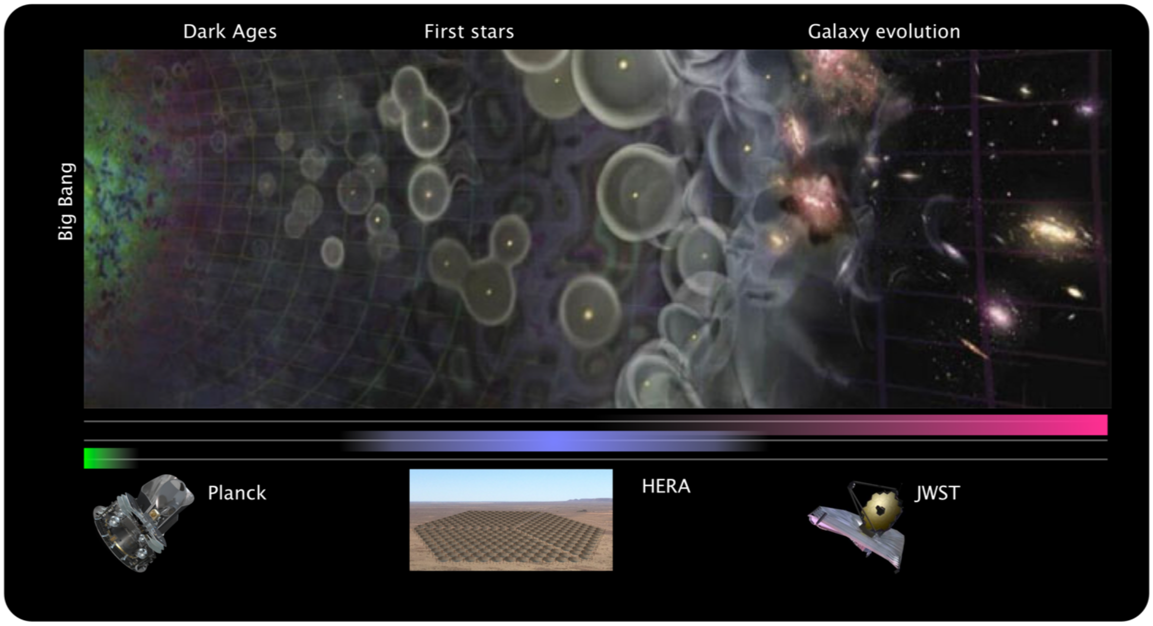


Figure 1.1: After the Big Bang, The Universe was a very hot soup of fundamental particles. The plasma quickly cooled over several thousands of years and we see the as the CMB. After The Universe became neutral, it became unobservable across much of the electromagnetic spectrum and any short wavelength radiation that might have been emitted was quickly absorbed by the atomic gas. The Universe then transitioned into a long interval known as the Dark Ages. Slowly, the gravitational collapse of overdense regions, led to the formation of more pronounced structure in the neutral medium, and eventually the first stars, galaxies and quasars started to form ending with the formation of the first generation of galaxies. These objects started emitting ultraviolet radiation that carve out ionized regions around them. After a sufficient number of ionizing sources have formed, the ionized fraction of the gas in The Universe rapidly increases until hydrogen becomes fully ionized. This period, during which the cosmic gas went from neutral to ionized, is known as The Universe's Epoch of Reionization. This figure was taken from http://www.reionization.org.

neutral gas. At the same time, observations of the CMB and standard cosmological models, Λ-CDM( standing for Dark Energy and Cold Dark Matter) find agreement with the story told above. However, there also exists a large observational gap: how and when did The Universe transition from its neutral state at recombination, to its ionized and structured state we see today? The formation of cosmological structure begins with the primordial density anisotropies of the hot plasma. The dark matter that permeate The Universe should have traced those perturbations, and gravitationally accreted into those regions, increasing the overdensities. Eventually, these overdense regions above some threshold density collapsed into halos; matter structures that permeates and surrounds individual galaxies, as well as groups and clusters of galaxies and are supported by their own gravitational potential. The similarly pervasive HI field should have traced the dark matter overdensities also. This gravity-dominated period represents an epoch of relatively simple physics directly driven by Λ-CDM and is known as the "Dark Ages". At that time no luminous structures existed, and the only photons that existed were from the slowly fading CMB. Within the early dark matter halos, enough HI accreted to a large enough density to fuse, igniting the first stellar cores. These first stars, commonly referred to as Population-3 or (PopIII) stars (stars formed very early with little to no elements heavier than helium), were the first source of ultraviolet (UV) and X-ray photons capable of ionizing HI since recombination. They were also extremely massive and therefore shortlived and their supernovae likely provided the seeds for the first galaxies composed of PopII stars. The PopIII era is also sometimes referred to as "Cosmic Dawn", and represents the birth of astrophysics in our Universe.

## 1.2. Cosmic Dawn

With the origin of galaxies, the HI region surrounding haloes began to be reionized, forming "bubbles" of HII regions. As more luminous structures formed, UV photon production increased, and the reionization rate overcame recombination. The Cosmic Dawn of our universe is one of the last unexplored frontiers in cosmic history. This history is summarized in Figure 1.1, starting with the Big Bang on the left. The early population of gravitationally condensed material produced sufficiently energetic flux to reionize the intergalactic medium

(IGM) from its previous neutral state in a period called the Epoch of Reionization. This period is the rapid transition from separated large reionized “bubbles” to the merged reionized state and to structures that began to resemble the inhabitant of our current universe. The structure of the IGM contains a plethora of information about the underlying astrophysical and cosmological phenomena governing this cosmic evolution.

The evolution of the cosmic structures depend on the constituents of those first galaxies with Population III stars but also the local and average cosmic density, the relative velocities of baryons and dark matter, and the sizes and clustering of the first galaxies to form. It also depends on stellar remnants, X-ray binaries, and early supermassive black holes. UV and X-ray luminosities and spectra also affect the thermal and ionization states of the IGM. The wealth of unexplored physics during the Cosmic Dawn, culminating in the Epoch of Reionization (EoR), led to the most recent US National Academies astronomy decadal survey entitled New Worlds, New Horizons to highlight it as one of the top three priority science objectives for the decade. Exploring the interplay of galaxies and large-scale structure during the EoR requires complementary observational approaches. Measurements of the photons permeating The Universe after becoming transparent to its own radiation by the recombination of the protons and electrons about 400,000 years after the Big Bang by telescopes like COBE, WMAP and Planck provide initial conditions for structure formation.

### 1.3. Processes

Because The Universe has many high energy photons from quasars and large amounts of hydrogen atoms, both the absorption and emission of photons occurs frequently. It is also important to mention that because The Universe is expanding the photons travel towards us “stretching" or has a wavelength that has increased, $\lambda$, and lowers the energies of the photons. This phenomena is called redshifting. Neutral hydrogen atoms in their ground state will interact with redshifted photon to a wavelength of 1216 Å when it reaches them. The rest of the light will keep travelling towards us.

### 1.3.1. Thomson Scattering

Thomson scattering, different from Rayleigh scattering (scattering from a harmonically bound electron), is the classical scattering of electromagnetic waves by free electrons. The electron experiences a force due to the incident electric field:

$$\vec{E} = \hat{\epsilon} E_0 e^{i\vec{k}\cdot\vec{r} - i\omega t} \tag{1.1}$$

where $\hat{\epsilon}$ represents the polarization direction and $\hat{\epsilon} \cdot \vec{k} = 0$. The equation of motion for the charged particle is then,

$$m\ddot{\vec{r}} = -eE \tag{1.2}$$

Recall that in the dipole approximation, or for a non-relativistic particle accelerated by a force, the power emitted per unit solid angle is

$$\frac{dP}{d\Omega} = \frac{e^2}{16\pi^2 c^3 \epsilon_0} \langle a^2 \rangle \sin^2 \theta \tag{1.3}$$

where $< a^2 >$ represents the time-averaged squared acceleration. The leading order acceleration is due to the plane wave electric field with polarization $\hat{\boldsymbol{\epsilon}}_0$, wave vector $\boldsymbol{k}_0$, and Newton's equation of motion law gives the acceleration directly and the time averaged squared acceleration is:

$$< a^2 >= \frac{e^2}{2m^2} |E_0|^2 \tag{1.4}$$

Pluging 1.4 into 1.3 yields, upon rearranging,

The term in the brackets is known as the classical electron radius. Recall that the time-averaged Poynting vector, directional energy flux (the energy transfer per unit area per unit time) of an electromagnetic field, is $I \equiv \epsilon c \frac{|E_0|^2}{2}$.

The differential scattering cross section is defined by:

$$\frac{d\sigma}{d\Omega} = \left(\frac{e^2}{4\pi^2 mc^2\epsilon_0}\right)^2 \frac{\epsilon_0 c|E_0|^2}{2} \sin^2\theta \tag{1.5}$$

The differential scattering cross section is defined by:

$$\frac{d\sigma}{d\Omega} = \frac{\frac{dP}{d\Omega}}{I} \tag{1.6}$$

and is the area of the wavefront which delivers the same power as is scattered into a given solid angle $d\Omega$. The total cross section is obtained by integrating over the differential solid angle:

$$\sigma = 2\pi r_c^2 \int_0^\pi (d\theta \mathrm{sin}\theta)\mathrm{sin}^2\theta = 2\frac{4\pi}{3}r_c^2 \tag{1.7}$$

This is known as the classical Thomson cross section. When the incident energy of the photon, $\hbar\omega$, becomes comparable to the rest mass of the electron quantum mechanical effects become important. This is the case of Compton scattering.

Classical Thomson Scattering of CMB photons by the ionized particles constrains the integrated column of ionized gas and kinetic Sunyaev-Zel'dovich measurements constrain the duration of the patchy phase of cosmic structures. But even with these measurements, the detailed evolution of the IGM is only loosely constrained. Lyman alpha absorption features in quasar and gamma-ray burst spectra give ionization constraints at the tail end of reionization z $< 7$, but these features saturate at low neutral fractions $\overline{x}_{HI} = 10^{-4}$, where $\overline{x}_{HI}$ is the average fraction of hydrogen in its neutral state.

# CHAPTER 2

# 21 CM COSMOLOGY

## 2.1. Brief Introduction

The 21 cm line from gas during the first billion years after the Big Bang redshifts to radio frequencies 30-200 MHz making it a prime target for a new generation of radio interferometers currently being built. Instruments, such as Murchison Widefield Array (MWA), the LOw Frequency ARray (LOFAR), and the Precision Array to Probe the Epoch of Reionization (PAPER), sought to detect the radio fluctuations in the redshifted 21 cm background arising from variations in the amount of neutral hydrogen. Next generation instruments will be able to go further in making more detailed maps of the ionized regions during reionization and measure properties of hydrogen out to redshift z = 30. These observations constrain the properties of the intergalactic medium and by extension the cumulative impact of light from all galaxies. Combining this with direct observations of the sources themselves, they provide a powerful tool for learning about the first generation of stars and galaxies. This probe will also provide information about active galactic nuclei (AGN), such as quasars, by observing the ionized bubbles surrounding individual AGN.

The quasar and gamma-ray burst shines within a certain spectrum. Gas around the quasar both emits and absorbs photons. With the presence of neutral hydrogen near the quasar, the emitted flux is depleted for certain wavelengths, indicating the absorption by the neutral hydrogen. It is known that at this location the photon of wavelength 1216 Å is absorbed. Its wavelength is stretched by the expansion of the Universe from what it was initially at the quasar, and, if it had continued to travel to us, it would have been stretched more from the initial 1216 angstroms wavelength it had at the absorber. When the emitted photon is absorbed we see the dip in flux at the wavelength corresponding to the 1216 angstrom photon if it had reached us. As we can calculate how the universe is expanding, we can tell where the photons were absorbed in relation to us. In other words, an absorption map can

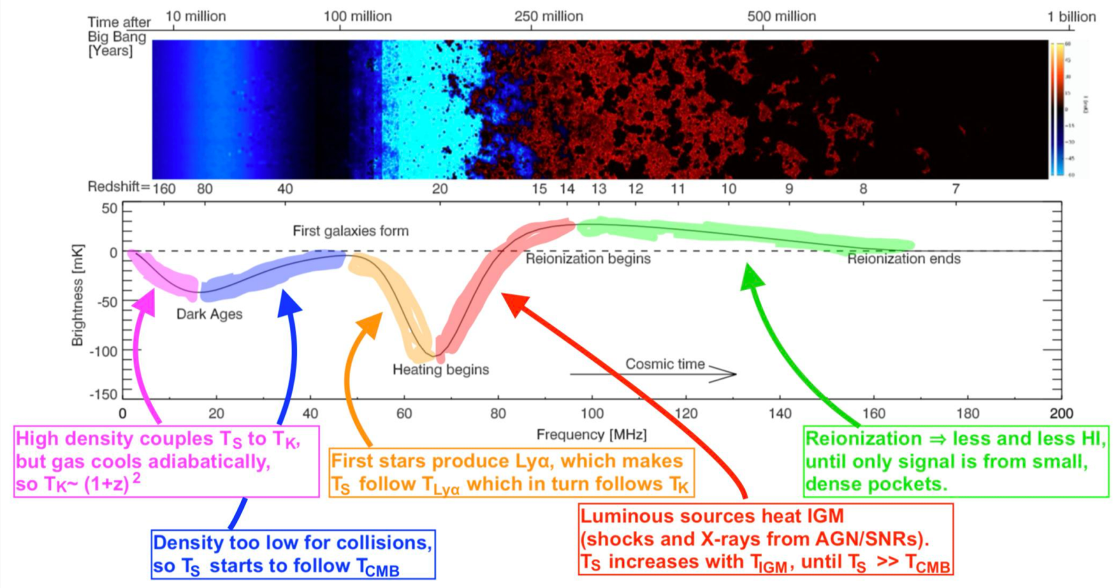


Figure 2.1: *The time evolution of the 21cm signal.* (**Top Plot**) Time evolution of fluctuations in the 21 cm brightness from just before the first stars formed through to the end of the Epoch of Reionization. Coloration indicates the strength of the 21cm brightness signal as it evolves through two absorption phases (purple and blue), separated by a period (black) where the excitation temperature of the 21cm hydrogen transition decouples from the temperature of the hydrogen gas, before it transitions to an emission (red) state and finally disappears (black) owing to the ionization of the hydrogen gas. (**Bottom Plot**) The expected evolution of the sky-averaged 21cm brightness signal from the Dark Ages at around redshift $r = 200$ to the end of reionization, sometime before redshift $z = 6$. The solid curve indicates the 21cm signal and the dashed curve indicates the brightness temperature of zero $T_b = 0$. The frequency structure within this redshift range is driven by several physical processes, including the formation of the first galaxies and the heating and ionization of the hydrogen gas. There is considerable uncertainty in the exact form of this signal, arising from the unknown properties of the first galaxies. Figure from Pritchard and Loeb (2012)

.

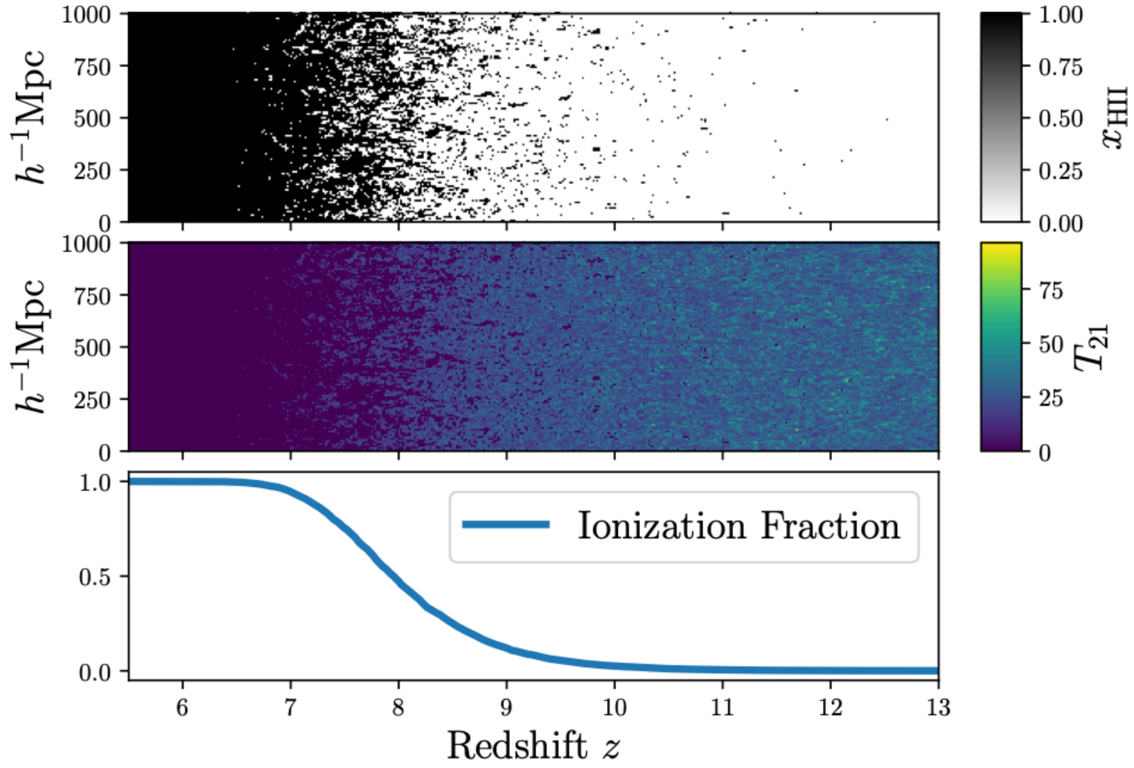


Figure 2.2: In order to determine the thing we want which is Tau, to first order knowing information about the ionization faction of hydrogen gets us pretty far. The most interesting thing in this image is the bottom Ionation Fraction plot. If the temperature field map is known it can be used to try and infer the ionization field and get the ionization fraction as a function of redshift. This approach gets us really close to getting Tau by making the connection between the observed (or simulated) data that we have which is in the form of temperature fields and the value we want which is the optical depth. Use the neutral fraction to calculate the number density of electron. The optical depth to the epoch of reionization as a function of (redshift $z$) can be computed by integrating $n_e * \sigma_T dl$, the electron density times the Thomson cross section along the proper length. Since the optical depth of this transition is small at all relevant redshifts, it yields a differential brightness temperature ($\delta T_b$). This figure was taken from Liu et al. (2016)

be used to plot the positions of region of intervening hydrogen between us and the quasar.

In addition to learning about galaxies and reionization, 21cm observations have the potential to inform us about fundamental physics too. Part of the 21cm signal traces the density field giving information about neutrino masses and the initial conditions from the early epoch of cosmic inflation in the form of the power spectrum. In addition, the spin temperature fluctuations driven by astrophysics also contribute to the signal. Understanding the astrophysical effects such as exploiting the effect of redshift space distortions, which also produce 21 cm fluctuations but directly trace the density field is important in order to understanding the cosmology. In the long term, 21 cm cosmology may allow precision measurements of cosmological parameters by opening up large volumes of the Universe to observation.

## 2.2. Basic 21 cm Physics

### 2.2.1. Hyperfine Splitting

Hydrogen, the most abundant atom in the Universe, is a useful tracer of the local properties of the surrounding gas. The 21 cm line of hydrogen arises from the hyperfine splitting of the 1S ground state of the atom due to the interaction of the magnetic moments of the proton and the electron. This splitting leads to two distinct energy levels separated by $\Delta E = 5.9 \times 10^{-6}$ eV, which corresponds to a wavelength of about 21.1 cm and a frequency of about 1420 MHz. This frequency is one of the most precisely known quantities in astrophysics and has been used as a probe of astrophysics since it was first detected . Radio telescopes look for emission by warm hydrogen gas within galaxies. Since the line is narrow it can be used as a probe of the velocity distribution of gas within surrounding galaxies and trace galactic dynamics.

Other atomic species do show hyperfine transitions, 2.3, that may be useful in probing cosmology. The 8.7 GHz hyperfine transition of 3He+, which could prove to be a probe of Helium reionization with the 92 cm deuterium analogue of the 21 cm line. The abundance of deuterium and 3He is much lower compared to neutral hydrogen making it more difficult to take advantage of these transitions. In a cosmological contexts the 21 cm line has been

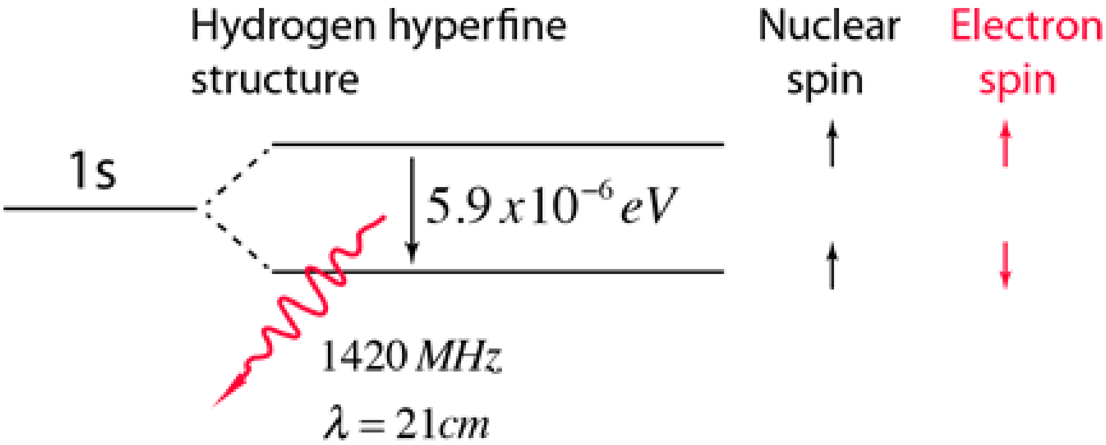


Figure 2.3: Hydrogen in its lowest energy state will absorb 1420 MHz and the observation of 1420 MHz in emission implies a prior excitation to the upper state by slightly split the interaction between the electron spin and the nuclear spin. The splitting is known as hyperfine structure. This image was taken from http://hyperphysics.phy-astr.gsu.edu/hbase/quantum/h21.html.

used as a probe of gas along the line of sight to some background radio source and depends upon the radiative transfer through gas along the line of sight. Since the relevant photon frequencies $\nu$ are much smaller than the peak frequency of the CMB blackbody this allows us to relate the intensity $I_\nu$ to a brightness temperature T by the relation,

$$I_\nu = 2k_B T \frac{\nu^2}{c^2} \tag{2.1}$$

where c is the speed of light and $k_B$ is Boltzmann's constant. We will also make use of the standard definition of the optical depth to reionization,

$$\tau = \sigma_T \int dz \overline{n}_e \frac{dl}{dz} \tag{2.2}$$

where $\overline{n}_e$ is the number density of electrons, $\sigma_T$ is the Thomson cross section along proper length. I will go into more detail and further expand this equation later.

In order to fully express the optical depth to reionization, the average number density of electrons must be approximated. Below is the analytical representation of the,

$$\overline{n}_e = \overline{x_{HII} n_H} + \overline{x_{HeII} n_{He}} + \overline{x_{HeIII} n_{He}} \tag{2.3}$$

The average number density of electrons is the sum of the average product of the ionization fraction ($x_{HII}$,$x_{HeII}$,$x_{HeIII}$) of the atom with its corresponding number density.

The excitation temperature of the 21 cm line is known as the spin temperature $T_S$ and is defined through the ratio between the number densities $n_i$ of hydrogen atoms in the two hyperfine levels. The label for the number density with a subscript 0 is for the 1S singlet and subscript 1 is the 1S triplet levels.

Recall that the singlet or a triplet can form when one electron is excited to a higher energy level. In an excited singlet state, the electron is promoted to it's highest energy state in the same spin orientation as it was in the ground state (paired). In a triplet excited stated, the electron that is excited has the same spin orientation (parallel) to the other unpaired electron.

The singlet, doublet and triplet states are derived using the equation for multiplicity, $2S+1$, where S is the total spin angular momentum (sum of all the electron spins). Individual spins are denoted as spin up ($s = +\frac{1}{2}$) or spin down ($s = -\frac{1}{2}$). The $S$ for the excited singlet state is

$$2\left[\left(+\frac{1}{2}\right) + \left(-\frac{1}{2}\right)\right] + 1 = 2(0) + 1 = 1$$

therefore making the center orbital a singlet state. The spin multiplicity for the excited triplet state is

$$2\left[\left(+\frac{1}{2}\right) + \left(+\frac{1}{2}\right)\right] + 1 = 2(1) + 1 = 3,$$

as expected.

In an atom the difference between a molecule in the ground and excited state is that the electrons is diamagnetic in the ground state and paramagnetic in the triplet state. This difference in spin state makes the transition from singlet to triplet (or triplet to singlet) more improbable than the singlet-to-singlet transitions. This singlet to triplet (or reverse) transition involves a change in electronic state. For this reason, the lifetime of the triplet state is longer than the singlet state by approximately 104 seconds. The radiation that induced the transition from ground to excited triplet state has a low probability of occurring, thus their absorption bands are less intense than singlet-singlet state absorption. The excited triplet state can be populated from the excited singlet state of certain molecules which results in phosphorescence.

$$\frac{n_1}{n_0} = \left(\frac{g_1}{g_0}\right) exp\left( - \frac{hc}{k\lambda_{21cm}}/T_S\right), \tag{2.4}$$

$$\frac{hc}{k_B\lambda_{21cm}} = 0.068 \text{ K}. \tag{2.5}$$

where $\left(\frac{g_1}{g_0}\right) = 3$ is the ratio of the statistical degeneracy factors of the two atomic energy levels. The degeneracy factor is the number of electron states in which have have same energy level. Mathematically these states are eigenstates of the system's Hamiltonian with the same eigenvalue (energy level).

### 2.2.2. Spin Temperature

Collisional excitation of the 21 cm hyperfine transition is not strong enough to thermalize it in warm neutral interstellar gas, which was shown Liszt (2001) by simultaneously solving the equations of ionization and collisional equilibrium. Coupling of the 21 cm excitation temperature and local gas motions may be established by the Lyman alpha radiation field. This is only true if strong Galactic Lyman alpha radiation permeates the interstellar gas. The Lyman alpha radiation tends to impart to the gas its own characteristic temperature, which is determined by the range of gas motions that occur on the spatial scale of the Lyman alpha scattering. In general, the calculation of neutral atomic hydrogen spin temperatures is a difficult problem as is any interpretation of neutral atomic hydrogen spin temperature measurements. For example, some calculations make use of the assumption that the collisional cross-sections are independent of velocity; the actual velocity dependence leads to a non-thermal distribution for the hyperfine occupation Hirata and Sigurdson (2007). This effect can lead to a suppression of the 21 cm signal at the level of 5%, which although small is still important from the perspective of using the 21 cm signal from the dark ages for precision cosmology.

There are three processes that determine the spin temperature:

- Absorption/emission of 21 cm photons from/to the radio background, primarily the CMB.
- Collisions with other hydrogen atoms and with electrons.
- Resonant scattering of Lyman alpha photons that cause a spin flip via an intermediate excited state. The rate of these processes is fast compared to the de-excitation time of the line, so that to a very good approximation.

The rate of these processes are significantly fast compared to the emission time of the transition line. To a good approximation, the spin temperature is given by the equilibrium balance of these effects. In this limit, the spin temperature can be expressed as,

$$T_S^{-1} = \frac{T_\gamma^{-1} + x_\alpha T_\alpha^{-1} + x_c T_K^{-1}}{1 + x_\alpha + x_c} \tag{2.6}$$

where $T_\gamma$ is the temperature of the surrounding radio photons. This is typically set by the CMB so that $T_\gamma = T_{CMB}$. $T_\alpha$ is the color temperature of the $Ly_\alpha$ radiation field at the $Ly_\alpha$ frequency. This field is closely coupled to the gas kinetic temperature $T_K$ by recoil during repeated scattering. Recall that temperature is related to color. This is because hot things radiate light. The temperature of the object affects the color of the light that is radiated.

$x_c$, $x_\alpha$ are the coupling coefficients due to atomic collisions and scattering of $Ly_\alpha$ photons, respectively. The spin temperature becomes strongly coupled to the gas temperature when

$x_{tot} \equiv x_c + x_\alpha \geq 1$ and relaxes to $T_\gamma$ when $x_{tot} \ll 1$. There are two forms of radio background sources that are important for the 21 cm line as an astrophysics probe Venkatesan (2000). Firstly, the use of CMB as a radio background source, $T_R$, such that $T_R = T_{CMB}$ and the 21 cm feature is seen as a spectral distortion to the CMB blackbody at appropriate radio frequencies since fluctuations in the CMB temperature are small $\delta T_{CMB} \approx 10^{-5}$ the CMB is effectively a source of uniform brightness. The spectral distortion forms a diffuse background that can be studied across the whole sky in a similar way to CMB anisotropies. Observations at different frequencies probe different spherical shells of the observable Universe, so that 3D brightness temperature maps can be constructed.

The second form uses a radio loud point source, for example a radio loud quasar, as the background. In this case, the source will always be much brighter than the weak emission from diffuse hydrogen gas, $T_R \gg T_S$, so that the gas is seen in absorption against the source instead. The appearance of lines from regions of neutral gas at different distances to the source leads to a series of absorption lines or a "forest" of lines known as the "21 cm forest" in analogy to the $Ly_\alpha$ forest. The high brightness of this particular background source allows the 21 cm forest to be studied with high frequency resolution so probing small scale structures to approximately kpc in the IGM. For statistical soundness, many lines of sight to different radio sources are required, making the discovery of high redshift radio sources a global scientific priority. Note that many of the quantities with unit temperature are not true thermodynamic temperatures. For instance $T_R$ and $\delta T_b$ are measures of a radio intensity. $T_S$ measures the relative occupation numbers of the two hyperfine levels. $T_\alpha$ is a color temperature describing the photon distribution in the vicinity of the $Ly_\alpha$ transition. Only the CMB blackbody temperature $T_{CMB}$ and $T_K$ are true thermodynamic temperatures.

### 2.2.3. The Optical Depth to Reionization

The reionization of the universe by the first generations of stars is described by the model developed in Haiman and Loeb (1997) and in Venkatesan (2000) they argued that the optical depth to reionization can be used as a probe of cosmological and astrophysical parameters.

The fraction of baryons in collapsed dark matter halos uses the Press-Schechter formulation; of these baryons, a fraction of them cool and form stars in a Scalo initial mass function. A fraction of the generated ionizing photons is assumed to escape from the host object and propagate isotropically into the IGM. One can then solve for the size of the ionized regions associated with each such star-forming cloud and when integrated over all haloes, yields at each redshift the average ionization fraction of the universe, given by the filling factor of ionized hydrogen by volume. Assuming a homogeneous IGM, the ionized region created by each source can be taken to be spherical with some radius. Reionization is defined to occur when filling factor of ionized hydrogen by volume has a value of 1. The total optical depth for electron scattering, $\tau_{reion}$, to the reionization redshift $z_{reion}$, is given by integrating the product of the electron density, the ionization fraction, and the Thomson cross section along the line-of-sight from the present to $z_{reion}$. The cosmology enters $\tau_{reion}$ through the first two terms of the integrand, and also through the path length of the photons last scattered at $z_{reion}$.

With this definition, the optical depth to reionization can also be expressed as,

$$\tau_{reion} = \int dz[1 - \exp(-E_{10}/k_B T_S)]\sigma_0 \phi(\nu) \frac{n_H}{4} \tag{2.7}$$

where $n_H$ is the number density of hydrogen, and we have denoted the 21 cm cross-section as,

$$\sigma(\nu) = \frac{3c^2 A_{10}}{8\pi\nu^2}\phi(\nu) \tag{2.8}$$

where $A_{10} = 2.85 \times 10^{-15}$ s$^{-1}$ is the spontaneous decay rate of the spin-flip transition. The line profile is normalised such that $\int \phi(\nu) d\nu = 1$. To evaluate Equation 2.8 we need to find the column length as a function of frequency $l(\nu)$ to determine the range of frequencies $d\nu$ over the path $ds$ that correspond to a fixed observed frequency $\nu_{obs}$. This can be done in one of two ways: (1) by relating the path length to the cosmological expansion,

$$dl = -c\frac{dz}{(1+z)}H(z) \tag{2.9}$$

where the redshifting of light to relate the observed and emitted frequencies is represented by $\nu_{obs} = \nu_{em}/(1+z)$. (2) assuming a linear velocity profile locally where $\nu = (d\nu/dz)z$, known as the Sobolev approximation and by using the Doppler law $\nu_{obs} = \nu_{em}(1 - v/c)$. Since the latter case describes the well known Hubble law in the absence of peculiar velocities these two approaches give identical results for the optical depth. The latter picture brings out the effect of peculiar velocities that modify the local velocity-frequency conversion. The optical depth of this transition is small at all relevant redshifts, yielding a differential brightness temperature

$$\begin{aligned}
\delta T_b &= \frac{T_s - T_R}{1+z}(1 - e^{-\tau_\nu}) \\
&= \frac{T_s - T_R}{1+z}\tau \\
&= 27 H x_{HI}(1+\delta_b)\frac{\Omega_b h^2}{0.023}\left(\frac{0.15}{\Omega_m h^2}\frac{1+z}{10}\right)^{\frac{1}{2}} \\
&= x\left(\frac{T_S - T_R}{T_s}\right)\left[\frac{\partial_r \nu_r}{(1+z)H(z)}\right]
\end{aligned} \tag{2.10}$$

## 2.3. Collisional Coupling

Collisions between different particles may induce spin-flips in a hydrogen atom and dominate the coupling in the early Universe where the gas density is high. There are three main channels available:

- Collisions between two hydrogen atoms.
- Collisions with a hydrogen and electron.
- Collisions with a hydrogen and proton.

$$x_i \equiv \frac{C_{10}}{A_{10}}\frac{T}{T_\gamma} \tag{2.11}$$

where $C_{10}$ is the collisional excitation rate, $x_i$ is the specific rate coefficient for spin de-excitation by collisions with species i (in units of $cm^3 s^{-1}$).

The total collisional coupling coefficient can be written as the sum of the hydrogen-hydrogen, hydrogen-electron, and hydrogen-proton coupling:

$$\begin{aligned} x_i &= x_{HH} + x_{eH} + x_{pH} \\ &= \frac{T}{A_{10} T_\gamma} \left[ \kappa_{1-0}^{HH}(T_k) n_H + \kappa_{1-0}^{eH}(T_k) n_e + \kappa_{1-0}^{pH}(T_k) n_p \right] \end{aligned} \tag{2.12}$$

where $\kappa_{1-0}^{HH}$ is the collisions rate between two hydrogen atoms, $\kappa_{1-0}^{eH}$ is the scattering rate between electrons and hydrogen atoms, and $\kappa_{1-0}^{pH}$ is the scattering rate between protons and hydrogen atoms. The collisional rates require a quantum mechanical calculation. Values for $\kappa_{1-0}^{HH}$ have been determined as a function of thermodynamic temperatures, $T_k$. The scattering rate between electrons and hydrogen atoms $\kappa_{1-0}^{eH}$ was considered.

The spin temperature of neutral hydrogen, which determines the optical depth to reionization and brightness of the 21 cm line, is determined by the competition between radiative and collisional processes. Furlanetto and Furlanetto (2007) examines the role of proton hydrogen atom collisions in setting the spin temperature by using fully quantum mechanical calculations of the relevant cross sections, which allows for accurate results over the entire temperature range of $1 - 10^4$ K. For large thermodynamic temperatures the proton hydrogen atom rate coefficient exceeds that for hydrogen hydrogen collisions by about a factor of two. However, at low thermodynamic temperatures ($T_K \leq 5$ K) proton hydrogen atom collisions become several thousand times more efficient than hydrogen hydrogen and even more important than electrons and hydrogen collisions.

In the high-redshift intergalactic medium, the dominant collisions are typically those between hydrogen atoms. However, collisions with electrons couple much more efficiently to the spin state of hydrogen than collisions with other hydrogen atoms and is therefore more important once the ionized fraction exceeds 1%. The authors of Furlanetto and Furlanetto (2007) compute the rate at which electron hydrogen collisions changes the hydrogen spin. Previous calculations included only S-wave scattering and ignored resonances near the n

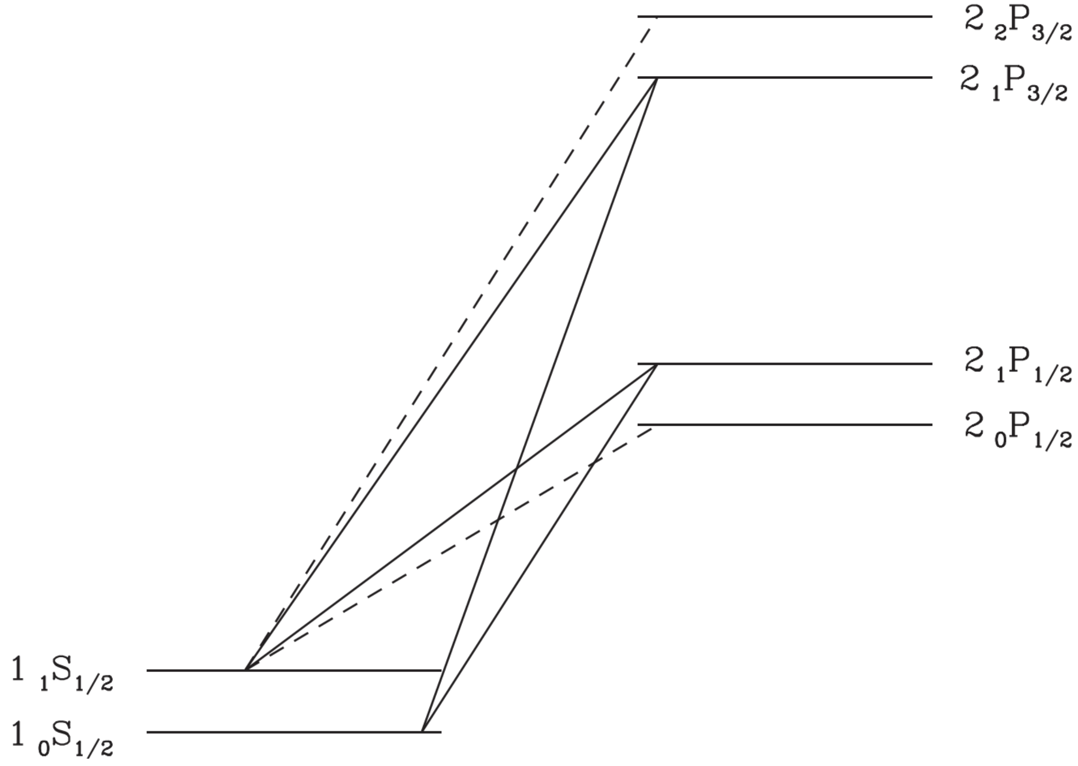


Figure 2.4: Hyperfine structure of the hydrogen atom 1S and 2P levels and the transitions relevant for the Wouthuysen-Field effect. Solid line transitions allow spin flips, while dashed transitions are allowed but do not contribute to spin flips.

= 2 threshold. Results, including all partial wave terms through the F-wave, for the de-excitation rate at thermodynamic temperatures $T_K \leq 1.5 \times 10^4$ K. It is important to note that beyond this point, excitation to $n \geq 2$ hydrogen levels becomes significant and accurate electron hydrogen collision rates at higher temperatures are not necessary, because collisional excitation in this regime inevitably produces Lyman alpha photons that dominate the spin exchange when $T_K$ is very large even in the absence of radiative sources.

## 2.4. Other Coupling Effects

This section will provide a sense of some of the subtleties that go into determining the strength of the Lyman alpha coupling by other coupling effects. These effects can modify the 21 cm signal at the 10% level, which will be important as observations begin to detect

21 cm fluctuations. For most of the redshifts that are probed, collisional coupling of the 21 cm line is inefficient. Once star formation begins, resonant scattering of Lyman alpha photons provides a second channel for coupling. This process is generally known as the Wouthuysen-Field effect and is illustrated in Figure 2.4. This figure depicts the hyperfine structure of the hydrogen 1S and 2P levels. Suppose that hydrogen is initially in the hyperfine singlet state, absorption of a Lyman alpha photon will excite the atom into either of the central 2P hyperfine states. From here emission of a Lyman alpha photon will relax the atom to either of the two ground state hyperfine levels. If relaxation takes the atom to the triplet state then a spin-flip has occurred. In other words, resonant scattering of Lyman alpha photons can produce a spin-flip.

The first ultraviolet sources in the Universe are expected to have coupled the neutral hydrogen atom spin temperature to the gas kinetic (thermal) temperature via scattering in the Lyman alpha resonance. Recall that by establishing an HI spin temperature different from the temperature of the CMB, the Wouthuysen-Field effect should allow observations of HI during the Epoch of Reionization in the redshifted 21 cm hyperfine line. The authors of the papers Hirata (2006) and Higgins and Meiksin (2009) investigates four mechanisms that can affect the strength of the Wouthuysen Field effect that were not previously considered:

1. Photons redshifting into the HI Lyman resonances may excite an hydrogen atom and result in a radiative cascade terminating in two-photon $2s_{1/2} \rightarrow 1s_{1/2}$ emission, rather than always degrading to Lyman alpha as usually assumed.

2. The fine structure of the Lyman alpha resonance alters the photon frequency distribution and leads to a suppression of the scattering rate.

3. The spin flip scatterings change the frequency of the photon and cause the photon spectrum to relax not to the kinetic temperature of the gas but to a temperature between the kinetic and spin temperatures, effectively reducing the strength of the Wouthuysen Field coupling.

4. Near the line center, a photon can change its frequency by several times the line width in a single scattering event, thus potentially invalidating the usual calculation of the Lyman alpha spectral distortion based on the diffusion approximation.

It has been shown that the first item suppresses the Wouthuysen-Field coupling strength by a factor of up to approximately 2, while [2] and [3] are important only at low kinetic temperatures. Effect [4] is said to have a less than 3% effect for kinetic temperatures greater than 2K. In particular if the intergalactic medium prior to reionization was efficiently heated by X-rays, only effect [1] is important.

The physics of the Wouthuysen-Field effect is more subtle and the coupling expression can be written as

$$x_\alpha = \frac{4P_\alpha}{27A_{10}}\frac{T}{T_\gamma} \tag{2.13}$$

where $P_\alpha$ is the scattering rate of Lyman alpha photons. Here we have related the scattering rate between the two hyperfine levels to $P_\alpha$ using the relation $P_{01} = \frac{4P_\alpha}{27}$, which results from the atomic physics of the hyperfine lines and assumes that the radiation field is constant across them. The rate at which Lyman alpha photons scatter from a hydrogen atom is given by the following equation,

$$P_\alpha = 4\pi\chi_\alpha \int d\nu J_\nu(\nu)\phi_\alpha(\nu) \tag{2.14}$$

where $\sigma_\nu \equiv \chi_\alpha\phi_\alpha(\nu)$ is the local absorption cross section, $\chi_\alpha \equiv (\pi e^2/m_e c)f_\alpha$ is the oscillation strength of the Lyman alpha transition, $\phi_\alpha(\nu)$ is the Lyman alpha absorption profile, and $J_\nu(\nu))$ is the angle-averaged specific intensity of the background radiation field (by number).

$$S_\alpha \equiv Rdx\frac{\phi_\alpha\alpha(x)J_\nu(x)}{J_\infty} \tag{2.15}$$

with $J_\infty$ is defined as the flux away from the absorption feature. This is used as a correction factor of order unity to describe the detailed structure of the photon distribution in the neighborhood of the Lyman alpha resonance. Equation 2.15 can be used to calculate the critical flux required to produce $x_\alpha = S_\alpha$. The critical flux can also be expressed in terms of the number of Lyman alpha photons per hydrogen atom. In general, this condition is easy to satisfy once star formation begins. The above description of the physics couples the spin temperature to the color temperature of the radiation field, which is a measure of the shape of the radiation field as a function of frequency in the neighbourhood of the Lyman alpha line defined by,

$$\frac{h}{k_B T_c} = -\frac{d \log n_\nu}{d\nu} \tag{2.16}$$

where $n_\nu = c^2 J_\nu / 2\nu^2$ is defined as the photon occupation number. Typically, $T_C = T_K$ because in most cases of interest the optical depth to Lyman alpha scattering is very large leading to a large number of scatterings of Lyman alpha photons that bring the radiation field and the gas into local equilibrium for frequencies near the line center. This relation occurs through the process of scattering Lyman alpha photons in the neighbourhood of the Lyman alpha resonance, which leads to a distinct feature in the frequency distribution of photons. This is defined as the "flow" of photons in frequency. Redshifting with the cosmic expansion leads to a flow of photons from high to low frequency at a fixed rate. As photons flow into the Lyman alpha resonance they may scatter to larger or smaller frequencies. Since the cross-section is symmetric, the net flow rate should in theory be preserved however, each time a Lyman alpha photon scatters from a hydrogen atom it will lose a fraction of its energy $\frac{h\nu}{m_p c^2}$ due to the recoil of the atom. This loss of energy increases the flow to lower energy and leads to a deficit of photons close to line center. This feature develops scattering redistributes photons leading to an asymmetry about the line and this asymmetry is precisely what is required to bring the distribution into local thermal equilibrium with $T_C \approx T_K$.

The shape of this feature determines $S_\alpha$ and, since recoils source an absorption feature, ensures $S_\alpha \leq 1$. At low temperatures, recoils have more of an effect and the suppression of the Wouthuysen Field effect is more prominent. If the IGM is "warm" then this suppression is then negligible. The processes whereby the distribution of photons is changed by spin-exchanges was not addressed because it complicates the determination of $T_S$ and $T_C$ considerably since they must then be iterated to find a self-consistent solution for the level populations and photon populations. However, the effect of spin-flips on the photon distribution is small, less than 10%.

From an astrophysical perspective, HERA will primarily be interested in photons redshifting into the Lyman alpha resonance from frequencies below the Lyman beta resonance. In addition to this, Lyman alpha photons can be produced by atomic cascades from photons redshifting into higher Lyman series resonances. For large values the conversion is approximately 30%. These photons are inserted into the Lyman alpha line instead of being redshifted from outside of the line. This effect changes their contribution to the Wouthuysen Field coupling since the photon distribution is now completely one-sided. Other processes apply to the redistribution of these photons that can lead to an amplification of the Lyman alpha flux.

## 2.5. Global 21 cm Signature

### 2.5.1. Spin Temperature Evolution

An important feature of the brightness temperature, $T_b$, is that its dependence on each of these quantities such as temperature of the surrounding radio photons, saturates at some point. For example, once the Lyman alpha flux is high enough the spin and kinetic gas temperatures become tightly coupled and further variation in $J_\alpha$ becomes irrelevant to the details of the signal. This leads to separate regimes where variation in only one of the variables dominating fluctuations in the signal. The most important phases of the global 21-cm signature are driven by the evolution of the spin temperature. Within a standard cosmological framework ($\Lambda$CDM Cosmology) these phases are summarized as follows,

- **Recombination** ($z \approxeq 1060$): The gas kinetically decouples from the CMB. The Universe then becomes neutral leaving free electrons and protons of the order of $10^{-4}$ charges per neutral atomic hydrogen.

- ($130 \lesssim z \lesssim 1060$): The number density of CMB photons is now much larger than the baryonic number density, the Compton scattering of CMB photons with the residual population of free electrons is efficient in keeping the gas in thermal equilibrium with the CMB. During this phase the gas temperature is written as $T_g = T_{CMB}(z) \approx (1+z)$. This high density gas to collisional coupling ($x_C \gg 1$) such that $T_S = T_g$. Since all the temperatures are the same, the fluctuations of the brightness temperature $T_b$ are negligible and as a result no 21-cm signal will be detected.

- ($40 \lesssim z \lesssim 130$): The gas is thermally decoupled from the CMB at $z \lesssim 130$. In this phase the gas adiabatically cools so that $T_g \approx (1+z)^2$. Again, the density of the gas is high enough to still make collisional coupling efficient ($x_C > 1$) so that $T_S = T_g$. Since the gas temperature is now colder than the CMB an early 21-cm signal in absorption is predicted.

- ($z^* \lesssim z \lesssim 40$): The density of the gas decreases and the collisional coupling is no longer efficient in keeping $T_S = T_g$ ($x_C < 1$). During this phase $T_S \approx T_{CMB}(z)$ and therefore a second period in history without a 21 cm signal is expected.

- ($z^\alpha \lesssim z \lesssim z^*$): The first stars and quasars emit both Lyman alpha photons and X-rays concluding the Dark Ages. In general the requirement for the Lyman alpha coupling $x_{alpha}$ is less than that for heating the gas above $T_R$. During this period the gas is still cooling adiabatically and the Lyman alpha couples $T_S$ to $T_\alpha$ ($x_\alpha > 1$). $T_\alpha \approx T_g$ due to the recoil of the Lyman alpha photons in the Wouthuysen-Field effect. We therefore expect a regime where the spin temperature is coupled to cold gas so that $T_S \approx T_g < T_{CMB}(z)$. A second period with a 21 cm signal in absorption is predicted and could be the one detected by telescopes.

- ($z^h \lesssim z \lesssim z^\alpha$): Heating has now become significant. The gas temperature overtakes the CMB one and the 21 cm signal in emission. The signal will die after the full reionization of the Universe because $x_{HI} \equiv 0$.

- ($z \lesssim z^r$): After reionization, any remaining 21 cm signal originates primarily from collapsed neutral hydrogen or damped Lyman alpha systems.

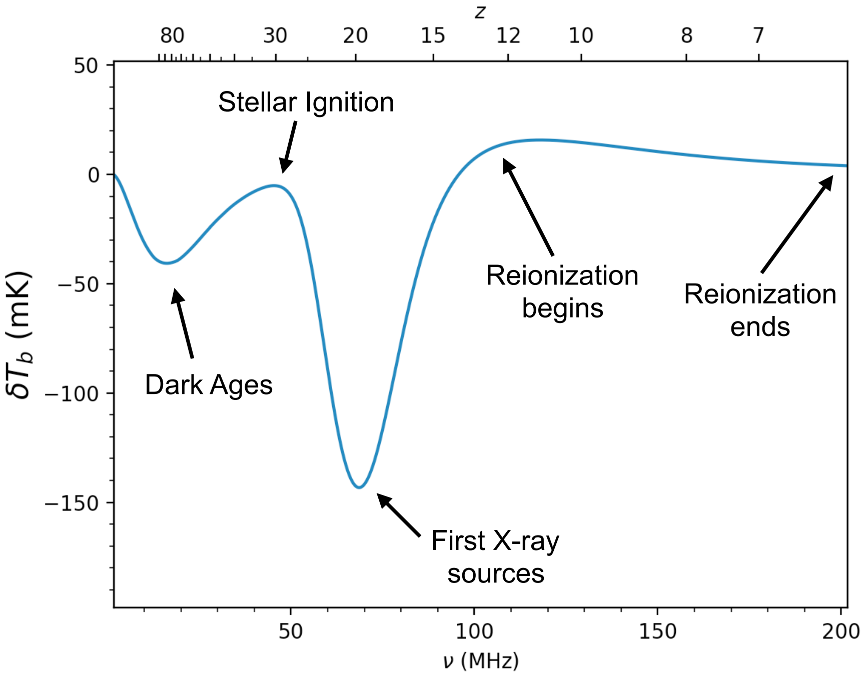


Figure 2.5: The history of the EoR global signal, with important turning-points indicated. This figure was produced using the fiducial model of Mirocha et al. (2012).

It is important to note that most of these epochs are not precisely defined resulting in considerable overlap between them. In fact, our ignorance of early sources is such that we can not definitively be sure of the sequence of events.

The largest uncertainty lies in the ordering of $z^{\alpha}$ and $z^{h}$. Although the authors in Nusser (2005) ignored Lyman alpha coupling and that an X-ray background may generate Lyman alpha photons they explored the possibility that $z^{h} > z^{\alpha}$, such that X-ray preheating allows collisional coupling to be important prior to the Lyman alpha flux becomes significant. Simulations of the very first miniature quasar also probe this regime and show that the first luminous X-ray sources could have had a greater impact on their surrounding environment. While these authors looked at the case where the production of Lyman alpha photons was inefficient, the case where heating is much more efficient can be considered.

### 2.5.2. Evolution of Global Signal

In calculating the 21 cm signal it helps to treat the IGM as a two phase medium. During the first phase, the IGM is composed of a single mostly neutral phase left over after recombination characterised by a gas temperature $T_g$ and a small fraction of free electrons, $x_e$. This is the phase that is expected to generate the observed 21 cm signal. Once galaxy formation begins, energetic UV photons ionize the surrounding HII regions because UV photons have a very short mean free path in a neutral medium leading to the ionized HII regions. The ionized HII bubbles can be treated as a second phase in the IGM characterised by a volume filling fraction $x_i$, the volume-averaged ionized fraction of hydrogen, provided that the free electron fraction is small. $x_i$ is then approximately the mean ionization fraction. These bubbles are fully ionized and the temperature inside the bubbles is fixed at $T_{HII} = 10^4 K$ determining the collisional recombination rate inside these bubbles. Since the photons that redshift into the Lyman alpha resonance initially have long mean free paths, the Lyman alpha flux, $J_\alpha$, may be treated as being the same in both phase one and two although in practice there is no 21 cm signal from these fully ionized bubbles. Technically, it is only the Lyman alpha flux in the mostly neutral phase that matters. To determine the 21 cm signal at a given redshift, the following four quantities must be calculated $x_i$, $x_e$, $T_g$, and $J_\alpha$. First accounting for adiabatic cooling of the gas due to the cosmic expansion and for other sources of heating/cooling. Then, consider the volume filling fraction $x_i$ and the ionization of the neutral IGM, $x_e$.

Since the ionization rate is a balance between ionizations and recombinations, the rate of change in $x_i$ and $x_e$ look the same, however, the main distinction lies in the manner in which recombination is treated. For fully ionized bubbles, recombinations occur in those dense clumps of material capable of self-shielding against ionizing radiation. These overdense regions will have a locally enhanced recombination rate, making it important to account for the inhomogeneous distribution of matter through the clumping factor. Since recombinations will occur on the edge of these neutral clumps. Secondary photons produced by the recombinations will likely be absorbed inside the clumps rather than in the mean IGM. Recombinations in the neutral IGM will occur at close to mean density in gas with temperature $T_g$. This two phase approximation will eventually break down if $x_e = 1$, indicating that most of the IGM has been ionized and that there is no clear distinction between ionized bubbles and a neutral IGM. In most models, $x_e$ remains small until the end of reionization making this a reasonable approximation.

### 2.5.3. Growth of HII Regions

The first thing to consider is the ionization rate per hydrogen atom which is a function fraction of ionizing photons, the number of ionizing photons per baryon produced in stars, and the star formation rate density as a function of redshift. The star formation rate is modeled as tracking the collapse of matter which is still poorly known observationally.

The model for $x_i$ is motivated by HII regions (ionized hydrogen) expanding into neutral hydrogen by calculating the mass function and determining a minimum mass for collapse by requiring that the appropriate cooling temperature of atomic hydrogen to be greater than $10^4$ K. This decreases this minimum galaxy mass of molecular hydrogen cooling of to about 300 K, will allow star formation to occur at earlier times resulting in a shift in the ionization features over redshift. The sources of ionizing photons in the early Universe are primarily from galaxies. However, the properties of these galaxies are currently poorly constrained. Observations from the Hubble Space Telescope provide some of the best constraints on early galaxy formation.

Faint galaxies are identified as being at high redshift using a Lyman alpha drop-out technique where a naturally occurring break in the galaxy spectrum at the Lyman alpha wavelength is seen in different color filters as a galaxy is redshifted. This technique states that very high redshifts, galaxies illuminate intervening clouds of neutral gas which produce absorption lines in the UV spectrum. Strong absorption by the intervening hydrogen occurs when the observed red shifted photons of wavelength is shorter the Lyman alpha line and is even stronger as the observed wavelength approaches the red shifted observed Lyman limit, 91.2

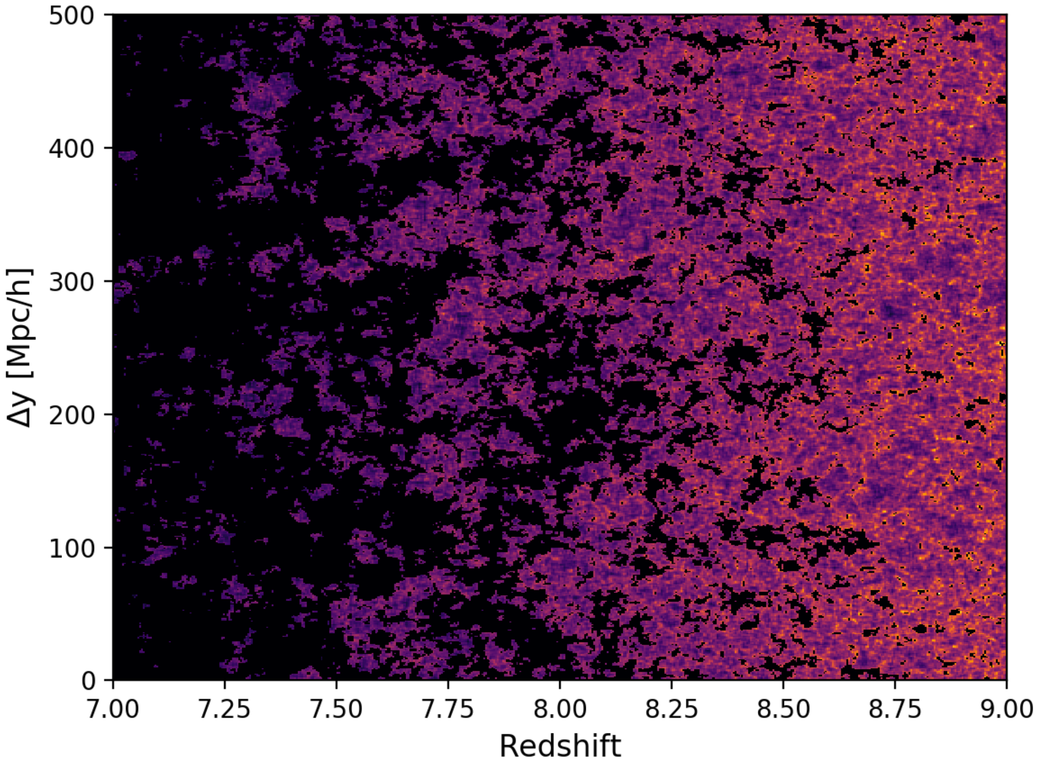


Figure 2.6: The 21 cm brightness as a function of redshift. This figure was taken from Kohn (2018).

nm. The galaxy is observed using two broad band filters for a galaxy. The galaxy is visible in one filter but not in the other filter. The Lyman alpha line is used to gain greater precision in the measurements. Galaxies at redshifts up to $z \approx 10$ have provided information on the sources of reionization.

The recombination rate is important at late times once a significant fraction of the volume has already been ionized. At this stage, dense clumps within an ionized bubble can act as sinks of ionizing photons slowing or even stalling further expansion of the bubble. The degree to which gas resides in these dense clumps is an important uncertainty in modelling reionization. Hydrodynamic effects such as the evaporation of gas from a halo as a result of photoionization heating can significantly modify the clumping factor. A more simple model for the clumping factor assumes that the Universe will be fully ionized up to some critical overdensity. The probability distribution can be modeled analytically starting from a consideration of behaviour of low density voids and accounting for Gaussian initial conditions. To accurately capture factor the clumping one should self consistently perform a full hydrodynamical simulation of reionization, since thermal feedback can modify the gas density distribution. Critical density can account for the patchy nature of reionization, which proceeds via the expansion and overlap of ionized bubbles. The size of a bubbles will then become limited if the mean free path of ionized photons becomes shorter than the size of the bubble. With this information the average clumping factor over the distribution of bubble sizes is determined.

There are limitations on the existing surveys due to their small sky coverage, which makes it unclear whether those galaxies seen are properly representative, and limitations to the frequency coverage. Even at the optical frequencies at which the galaxies are seen do not correspond to the UV photons that ionize the IGM. Limitations on the understanding of the mass distribution of the emitting stars introduces an uncertainty in the number of ionizing photons per baryon is emitted by galaxies. There is also considerable uncertainty in the fraction of ionizing photons that escape the host galaxy to ionize the IGM.

### 2.5.4. Heating and Reionization

Integrate gas temperature, $T_g$, using a specific heating mechanisms is used to determine the heating rate. At high redshifts, the dominant mechanism is Compton heating of the gas arising from the scattering of CMB photons from the small residual free electron fraction. Since these free electrons scatter from the surrounding baryons, this transfers energy from the CMB to the gas. Compton heating couples $T_g$ to $T_\gamma$ at some redshifts, but becomes ineffective below a redshift of about 150. It could serve as the initial conditions before star formation begins.

At redshifts below 150, the growth of non-linear structures leads to other possible sources of heat. Shocks associated with large scale structure occur as gas separates from the Hubble flow and undergo turnaround before collapsing onto a central overdensity. After the turnaround, different fluid elements may cross and shock due to the differential accelerations and could provide considerable heating of the gas at late times.

Another source of heating is the scattering of Lyman alpha photons off hydrogen atoms, which leads to a recoil of the nucleus that depletes energy from the photon. It was initially believed that this would provide a strong source of heating sufficient to prevent the possibility of seeing the 21 cm signal in absorption. Early calculations showed that by the time the scattering rate needed for Lyman alpha photons to couple the spin and gas temperatures was reached, the gas would have been heated well above the CMB temperature. However, these estimates did not account for the way the distribution of Lyman alpha photon energies was changed by scattering. This spectral distortion is a part of the photons coming into equilibrium with the gas and serves to greatly reduce the heating rate. While Lyman alpha heating can be important, it typically requires very large Lyman alpha fluxes and so it is most relevant at late times and may be insufficient to heat the gas to the CMB temperature alone.

The most important source of energy injection into the IGM is likely from X-ray heating of the gas. While shock heating dominates the thermal balance in the present day Universe. For sensible source populations, Lyman alpha heating is mostly negligible compared to X-ray heating. Since X-ray photons have a long mean free path, they are able to heat the gas far from the source, and can be produced in large quantities once compact objects are formed. The comoving mean free path of an X-ray with energy E is,

$$\lambda_X \approx 4.9\bar{x}_{HI}^{-1/3}\left(\frac{1+z}{15}\right)^{-2}\left(\frac{E}{300eV}\right)^{3} Mpc \tag{2.17}$$

The Universe will be optically thick over a Hubble length to all photons with energy below,

$$E \approx 2\left[\frac{1+z}{15}\right]\frac{1}{2}x_{HI}^{-1/3}keV \tag{2.18}$$

The $E^{-3}$ dependence of the cross-section means that heating is dominated by soft X-rays, which fluctuate on small scales. There will also be a uniform component to the heating from harder X-rays through photo-ionization of HI and HeI. This generates energetic photo-electrons, which dissipate their energy into heating, secondary ionizations, and atomic excitation. This energy can be divided into heating, ionization, and excitation by inserting a factor that is defined as the fraction of energy converted at a specific frequency. This allows us to calculate the contribution of X-rays to both the heating and the partial ionization of the IGM. The division of the X-ray energy depends on both the X-ray energy and the free electron fraction and can be calculated by using Monte-Carlo methods.

The total X-ray luminosity per unit star formation rate is consistent with that observed in starburst galaxies at the present epoch. Since then improved observations have revised this number resulting in a better separation of the contribution from LMXB and HMXB. While the data is fairly patchy it shows considerable scatter X-ray emissivity of 0.2 which seems to be a better fit to other data in the local universe. Extrapolating observations from the present day to high redshift is highly uncertain. In particular, the metallicity evolution of galaxies with redshift is likely to impact the ratio of black holes to neutron stars that form the compact object in the HMXB and with it the efficiency of X-ray production.

Other sources of X-rays are inverse Compton scattering of CMB photons from the energetic electrons in supernova remnants. Estimates of the luminosity of such sources is again highly uncertain, but of a similar order of magnitude as from HMXB. The X-ray luminosity from supernovae remnants is expected to track the star formation rate. Finally, miniquasars accretion onto black holes with intermediate masses $10^6$ solar mass can produce significant levels of X-rays. Since the early formation of black holes depends sensitively on the source of seed black holes and their subsequent merger history there is again considerable uncertainty. The evolution of miniquasars could be considerably highly complex but for simplicity assuming that miniquasars track the star formation rate is sufficient.

The total X-ray luminosity at high redshift is constrained by observations of the present day unresolved soft X-ray background (SXRB). An early population of X-ray sources would produce hard X-rays that would redshift to lower energies contributing to this background. Since there will be faint X-ray sources at lower redshift that also contribute to this background, the SXRB can be used to place a conservative upper limit on the amount of X-ray production at early times. This rules out complete reionization by X-rays but allows for heating.

Since heating requires considerably less energy than ionization, X-ray emissivity is still relatively unconstrained by the CMB polarisation anisotropies on the optical depth for electron scattering. Constraining this parameter will mark a step forward in the understanding of the thermal history of the IGM and the population of X-ray sources at high redshifts.

## 2.6. Direct Measurements of HI

Throughout the Dark Ages and the EoR, neutral hydrogen, HI, shone weakly at radio wavelengths. As stated section 2.2.1 the neural hydrogen atom is capable of a hyperfine transition between its spin energy levels. Observations of HI during the Dark Ages and the EoR would directly constrain the ionization history of the Universe, allowing us to probe the evolution of the density fields at high redshifts, the properties of the first stars, galaxies and black holes, and map the structure of the IGM as it evolves as a function of redshift. It would also represent the furthest baryons ever detected, and the largest volume of the Universe ever surveyed. This Section reviews the nature of the 21 cm line during the EoR, and two parallel endeavours to detect it, using the monopolar signal (power averaged over the sky), and the anisotropic signal (power as a function of spatial scale). The 21 cm photons generated through the hyperfine transition are highly unpolarized. Primordial magnetic fields could induce circular polarization via Zeeman splitting, but such an effect would be 3 to 4 orders of magnitude smaller than the already faint total intensity signal itself (Hirata et al., 2018).

Mitigation of foregrounds is essential for accessing the EoR observation. This fact was recognized by Madau et al. (1997) in one of the first in-depth studies of the promises and challenges of 21 cm tomography. The authors suggest that fitting the smooth synchrotron spectra may have been sufficiently accurate to subtract the foregrounds from the total signal. There were few low-frequency instruments powerful and well-characterized enough to attempt an EoR detection, and precise observations of low-frequency foregrounds did not exist. The first work to concentrate solely on the foreground challenge was Di Matteo et al. (2002) and they concluded that the challenge was surmountable with accurate and precise multi-frequency fitting.

Parsons et al. (2012) realized that from their numerical simulated visibilities dominated by smooth synchrotron foregrounds could be Fourier transformed along their frequency axis, and mapped into a narrow region of Fourier space. This is because the sinusoidal structure of the fringe term in the visibility equation, coupled with smooth sky emission and a smooth evolution of the beam term, produced a visibility that could be described with a relatively low number of Fourier modes. However, the inherent spectral structure of 21 cm emission caused its power to scatter to high $k_{\|}$. This allowed per-baseline access to a statistical power spectrum measurement of the EoR.

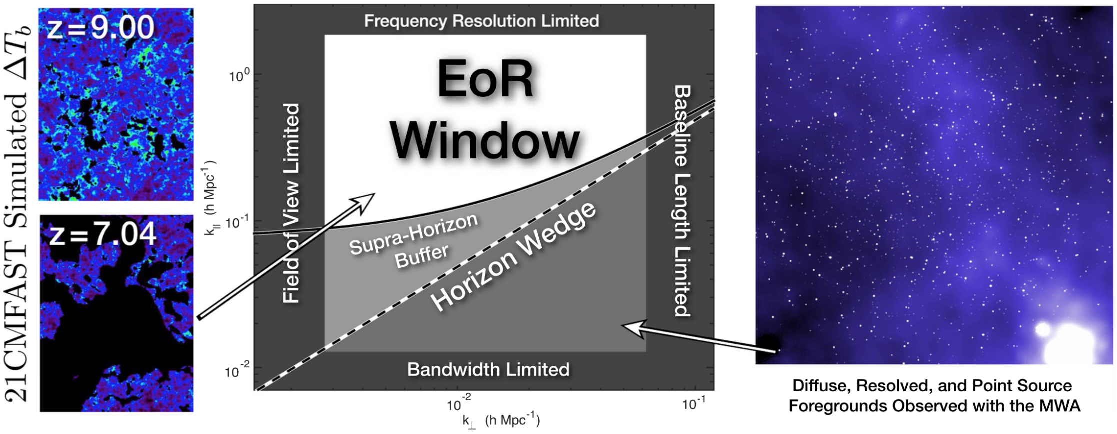


Figure 2.7: An illustration of the foreground wedge and the EoR window. Bright synchrotron foregrounds (such as those imaged by the MWA, on the right) are localized in $(k_\perp, k_\|)$ to a wedge-shaped region, while spectrally-structured 21 cm power (simulations from Mesinger et al. (2010) on the left) exists throughout Fourier space – crucially, outside of the wedge, in an EoR window. Pober et al. (2013) identified the requirement of a buffer region where instrumental effects could spill power just beyond the wedge region. Figure taken from DeBoer et al. (2017).

# CHAPTER 3

# OPTICAL DEPTH TO THE LAST SCATTERING SURFACE

## 3.1. Motivation

Using only five parameters ($\Omega_b h^2$, $\Omega_c h^2$, $\theta_*$, $A_s$, $n_s$), the description of the history and contents of the universe can be accomplished. A variety of precision measurements cosmological probes of the CMB have measured the values of these parameters to perfect precision. In addition to these parameters, the optical depth to the last scattering surface $\tau$ is constrained using CMB data. However, the constraints on $\tau$ are poor compared to the other parameters: $\tau$ has an uncertainty upwards of 10% as reported in the *Planck* analysis, whereas the other parameters have been determined to 1% uncertainty or better Planck Collaboration et al. (2020). Furthermore, $\tau$ is partially degenerate with several other parameters, such as $A_s$, and thus impacts the overall uncertainty of all other parameters. In Liu et al. (2016) they argue that this degeneracy can be broken with the aid of 21 cm data. This is done by modeling the underlying astrophysics of reionization and precisely describing the various quantities (both astrophysical and cosmological) that are needed for such modeling. Measuring the value of $\tau$ with higher precision from large-scale polarization data in CMB measurements is still several years away and will suffer from irreducible cosmic variance.

Placing tighter constraints on the value of $\tau$ using 21 cm measurements and secondary anisotropies in CMB data, focusing particularly on upcoming data from the Hydrogen Epoch of Reionization Array (HERA) and the Simons Observatory (SO) is the general goal. The value of $\tau$ is sensitive to the global ionization history of the universe and measurements of Cosmic Dawn (CD) from $z > 12$. Using the hydrogen 21 cm line, the Epoch of Reionization (EoR; $12 \geq z \geq 6$) can provide constraints on the value of $\tau$ independent of CMB data. Combining measurements of the 21 cm signal from HERA and the kinetic Sunyaev–Zel'dovich (kSZ) effect from SO have the ability to even more tightly constrain the global ionization history, and thus $\tau$. However, the optimal method for extracting $\tau$ from these different data sets is not obvious, and instrumental effects on the data leave artifacts that can pollute the cosmological information. Recent work has shown a significant amount of information in the two- and three-point correlation statistics of these fields.

Machine learning techniques featuring artificial neural networks (ANNs) provide a method for robustly and reliably inferring scientific conclusions from both real data from telescopes and simulated data from cosmological simulations. Leveraging these techniques to infer the value of $\tau$ from 21 cm data from HERA and kSZ data from SO is possible. These techniques provide a complementary approach to the multi-point statistics proposed above. Importantly, ANN-based techniques generally do not require the explicit definition of an estimator for a parameter, and instead may find unforeseen or unexpected correlations in the data.

## 3.2. Background Summary

When deriving cosmological constraints from the CMB, the five "canonical" parameters fit to the data are: (1) baryon density $\Omega_{\rm b}h^2$, (2) cold dark matter density $\Omega_{\rm c}h^2$, (3) acoustic scale angle $\theta_*$, (4) initial amplitude of scalar fluctuations $A_s$, and (5) the spectral index of scalar fluctuations $n_s$. From this set of parameters, it is possible to derive the value of other quantities such as the Hubble parameter $H_0$, the amplitude of matter fluctuations $\sigma_8$, and the age of the universe. In addition to these five parameters, it is necessary to fit an additional "nuisance" parameter—the optical depth to the last scattering surface $\tau$—due to the degeneracy of this quantity with other parameters. As stated previously, when analyzing the overall amplitude of the temperature power spectrum, the quantities $A_s$ and $e^{-2\tau}$ are degenerate. However, unlike the other parameters, the value for $\tau$ is driven primarily by astrophysics occurring since the last scattering surface, rather than fundamental physics. As CMB experiments seek to move beyond the cosmology and constrain additional parameters such as the sum of the neutrino masses $\Sigma m_\nu$, the number of effective relativistic species $N_{\rm eff}$, and the running of the spectral index $dn_s/d\ln k$, the uncertainty in $\tau$ begins to drive the accuracy with which these extensions to the model can be measured. For example, SO predicts $\sigma(\Sigma m_\nu) = 22$ meV (baseline) if $\Sigma m_\nu$ is the *only* extension to the fitting Ade et al. (2019). Even then, the errors are limited by uncertainty on $\tau$.

There are ways of constraining $\tau$ directly from CMB measurements of the power spectrum of temperature fluctuations ($TT$) and divergence-like polarization fluctuations ($EE$) directly. $\tau$ measurements from $TT$ have nearly reached the cosmic variance limit from *Planck* Planck Collaboration et al. (2020), while the cosmic variance limit using $EE$ measurements is still some years off, to be obtained by CLASS or a future satellite mission such as PICO Hanany et al. (2019). In contrast, inferring $\tau$ from other methods can be done with data in the near future, and provide an independent method of verifying the value of $\tau$ measured from CMB data. As mentioned above, two data sets capable of providing independent measurements of $\tau$ are 21 cm observations and the kSZ effect.

A feature of both 21 cm and kSZ measurements is that they can directly measure the ionization process which contributes to $\tau$, notably the component from $12 \geq z \geq 6$. However, the problem of estimating $\tau$ reliably from these data is a non-trivial one. Quantitatively, $\tau$ can be expressed as the optical depth of a photon passing through an ionized intergalactic medium (IGM) along a given line-of-sight $\hat{\mathbf{n}}$:

$$\tau(\hat{\mathbf{n}}) = \sigma_T \int_0^{z_{\rm rec}} n_e(\hat{\mathbf{n}}, z) \frac{dl}{dz} dz, \tag{3.1}$$

where $\sigma_T$ is the Thomson cross-section of the electron, $n_e(\hat{\mathbf{n}}, z)$ is the local electron number density at a given redshift $z$, and $z_{\rm rec}$ is the redshift of recombination. When considering the spatially averaged value $\langle\tau\rangle = \int \tau(\hat{\mathbf{n}}) d\Omega/4\pi$, the electron number density can be derived from $x_i$, the average ionization fraction at a redshift $z$ (where $x_i = 0$ denotes a neutral IGM, and $x_i = 1$ is ionized). Given this simplification, inferring the value of $x_i$ from 21 cm or kSZ data is complicated by the relatively broad range of models of early galaxy formation. For example, these various models predict different expected values for star formation and X-ray production, which affect the ionization level and temperature of the low-density gas in the intergalactic medium (IGM). Given these uncertainties, the possible values of $\tau$ allowed by theoretical models of reionization are quite broad, and the connection between specific parameters and $\tau$ is complicated.

$$\overline{n}_e = \overline{x_{HII} n_H} + \overline{x_{HeII} n_{He}} + \overline{x_{HIII} n_{He}} \tag{3.2}$$

$$= \overline{x_{HII} n_b} + \frac{1}{4}\overline{x_{HIII} n_b} Y_p^{BBN} \tag{3.3}$$

where $n_H$, $n_{He}$, and $n_b = n_H + n_{He}$ are the hydrogen, helium, and baryon number densities, respectively. The ionization fractions (defined to be between 0 and 1) are given by $x_{HII}$, $x_{HeII}$, and $x_{HeIII}$, referring to singly ionized hydrogen, singly ionized helium, and doubly ionized helium, respectively. The helium fraction $Y_p^{BBN}$ is defined as $\frac{4n_{He}}{n_b}$. The electron number density depends on the cosmology. This is because HERA actually measures hydrogen and helium fraction and the amount of primordial helium determines the amount of helium measured and therefore measures the amount of electrons bound to helium atoms instead of hydrogen atoms.

The averaged baryon density can be easily related to cosmological parameters via this equation,

$$n_b = \frac{3H^2\Omega_b}{8\pi G\mu m_p}(1+z)^3 \tag{3.4}$$

this is actually what HERA measures where $\Omega_b$ is the normalized baryon density, G is the gravitational constant, $m_p$ is the mass of the proton, and $\mu$ is the mean molecular weight.

Although HERA does not actually measure the cosmological parameters well directly (it is difficult to look at temperature maps to determine the baryon faction of the universe), these parameters are well determined by other experiments. HERA instead will focus on measuring the ionization history which is independent of the cosmology.

### 3.2.1. Simulation

In a standard parameter estimation process certain parameters $\theta_r$ are assumed to describe the reionization process (and in addition parameters $\theta_c$ which describe the cosmology). It is then possible to construct cosmological simulations of various types ranging from simple

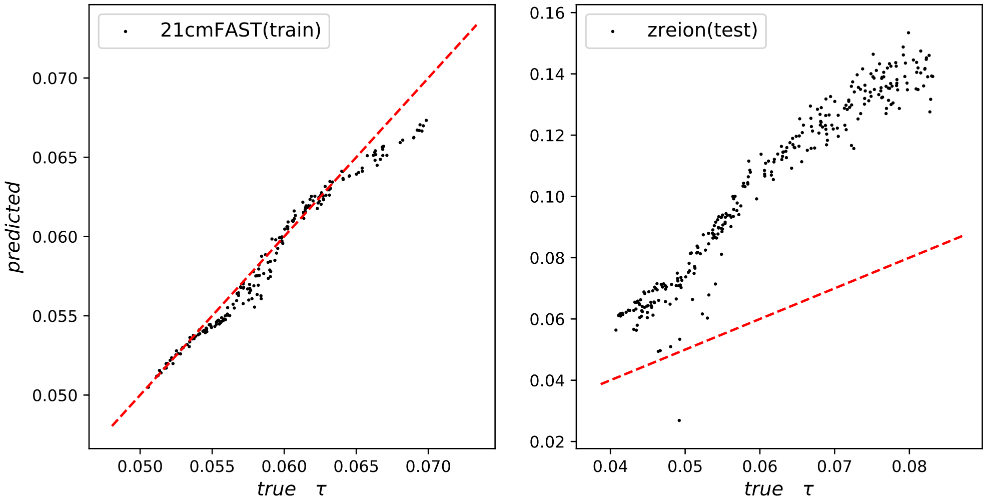


Figure 3.1: A comparison of evaluating a CNN trained using input data from 21cmFAST (left) and then evaluating using data from `zreion` (right). In both plots, we show the predicted value of the network versus the true value. As can be seen, there is some small bias that changes as a value of the true value of $\tau$, though across the entire range the results seem relatively unbiased. Conversely, for the `zreion` data, the results are significantly biased. This result has implications for scientifically relevant quantities like $x_i(z)$ and, by extension, $\tau$. Minimize the error in inferred parameters, and understand how the uncertainty in the 21 cm semi-numeric model affects the ultimate error on $\tau$ is necessary.

semi-numeric approaches to full radiation-hydrodynamic simulations which have a reionization history $x_i(z)$associated with them, and from which $\tau$ can be derived. $\tau$ is not an *input* parameter to these simulations and consequently, it is difficult to write the standard likelihood analysis $\mathcal{L}(d|\tau)$, data given optical depth. Previous attempts to quantify the ability of $\tau$ measured from 21 cm observations to affect uncertainties in other CMB parameters, such as Liu et al. (2016), was to infer the underlying parameters of a semi-numeric 21cmFAST simulation Mesinger et al. (2010) which best fit the measured 21 cm power spectra, and then use that to reconstruct $x_i(z)$and $\tau$. This approach is limited by the degree to which the simulation parameters adequately capture the physics producing $x_i(z)$. It is also worth noting that many different cosmic reionization scenarios can map onto the same $\tau$.

### 3.2.2. Temperature Maps

At a deeper level, having some method of inferring the underlying ionization field, independently of the details of the reionization process, then the calculation of $\tau$ should be trivial. While the power spectrum is a statistical method that does contains information about the features, a direct examination of the 21 cm temperature field $\delta T_{21}$ might be more productive in detecting features that reliably indicate the ionization state:

$$\delta T_{21}(\mathbf{x}, z) = \delta T_0(z)[1 - x_i(\mathbf{x}, z)][1 + \delta_m(\mathbf{x}, z)] \left(1 - \frac{T_\gamma}{T_s(\mathbf{x}, z)}\right), \tag{3.5}$$

where $\delta T_0(z)$ is a prefactor that depends only on redshift and cosmology, $\delta_m$ is the local matter fluctuation, $T_\gamma$ is the temperature of the CMB, and $T_S$ is the spin temperature of the IGM. The kSZ effect appears as a secondary temperature fluctuation in the CMB temperature $\Delta T_{\mathrm{kSZ}}$, and also contains information about the ionization state of the gas:

$$\frac{\Delta T_{\mathrm{kSZ}}(\hat{\mathbf{n}})}{T_{\mathrm{CMB}}} = -\frac{\sigma_T}{c} \int \mathrm{d}l \, n_e(l) e^{-\tau(l)} \mathbf{v} \cdot \hat{\mathbf{n}}, \tag{3.6}$$

where $\mathbf{v}$ is the peculiar velocity of the gas in the IGM. For both observables, it is generally expected that ionization fluctuations dominate over other factors such as the spin temperature, density fluctuations, and peculiar velocities for much of reionization, and the ionization fluctuations are strongly non-Gaussian. Hence, an approach which could largely isolate characteristic of the ionization field would provide to be a path that directly estimates $\tau$.

The question then becomes regardless of the underlying physical processes, does a generic features exist and is their a method that can be found to identify them. The flexibility of artificial neural networks (ANNs) for identifying features in image processing might provide a natural method for maximizing the inference ability given some measurement.

### 3.3. Current Progress

In 2020 I submitted a first author research paper, "*Extracting the Optical Depth to Reionization $\tau$ from 21 cm Data Using Machine Learning Techniques*", illustrating the success of training two different CNNs on simulated data. Through the use of model parameter optimization, I was able to find the optimal model architectures for each of the two data sets, and demonstrated that they accurately predicted $\tau$ values and performed well over the full range of input data. I also showed that I was able to provide constraints on the optical depth with a fractional error of 3.06% or better, which makes this approach competitive with observations based on the theoretical best determinations using CMB only. Due to the fact that instruments capable of providing such a constraint are many years away, using 21 cm measurements may be able to provide a constraint on a shorter time line. Machine learning techniques such as that outlined in my paper are most powerful in conjunction with more traditional analysis, providing additional cross-checks of results inferred by other means.

While the actual noise of 21 cm instruments will be quite complicated, I was able to gain some insight into the robustness of my analysis method by simply adding noise with mean value zero, white Gaussian noise, to the test image data and re-running the predictions. In my paper I was able to demonstrate the ability of my network to predict the optical depth at these different noise levels. I examined the linear relationship between the true optical depth values and the prediction by plotting them. If the predicted optical depth matches the true values then I would expect to see a straight line with a slope of one going through the origin. My predictions did indeed follow the one-to-one line closely, except for the highest noise level which showed a noticeable bias. However, even in this case, the clear correlation between $\tau_{true}$ and $\tau_{pred}$ still remains, giving confidence that a network properly trained using the actual noise properties of the instrument would still be able to make accurate predictions.

I also did two types of error analysis to assess the accuracy with which my machine learning-based approach is able to determine the value of the optical depth to reionization. The first method empirically derived the error bars from the training data, and are not "proper" error bars in either the Bayesian or Frequentist sense (how one chooses to view the data and model parameters). The other method uses an approximate Bayesian view: the data is a fixed set but the model parameters are treated as random variables. These are preliminary and were not treated as a proper forecast of the potential accuracy of future 21 cm experiments.

It is important to note that both networks were trained on 21 cm data generated using a particular set of cosmological parameters. To provide an estimate of the kinds of errors which would occur in this analysis if the underlying cosmologies were wrong, we generated new test data using a different choice of parameters. The most notable difference between these cosmologies ($\tau$ aside) is $\Omega_m$ (which includes both baryonic matter and dark matter), which differs by  10%. I then used the networks trained on data using one set of cosmological parameters to make predictions on the other. From previous work, it has been determined that there is a weak dependence of the 21 cm power spectrum on cosmology (e.g., (Kern et al., 2017)) and indeed we find that the dependence of $\tau$ on cosmological parameters is weak.

When using data from 21 cm measurements to constrain $\tau$, the uncertainties associated with the predicted value are important for understanding how competitive the results are with the value inferred from other methods. A significant portion of my work focused on providing estimates of the error bars associated with an inferred value of $\tau$. I also approached estimating $\tau$ in such a way that the uncertainty can be quantified as a straightforward part of the parameter estimation process. Other techniques such as Markov Chain Monte Carlo (MCMC), a probabilistic way of estimating a value by sampling from a distribution of values, approach involves deriving the Bayesian posterior, a way to summarize what we know about uncertain quantities like the model weights and optical depth.

I used Bayesian Neural Networks (BNNs) as a means of producing well-informed error bars. A BNN is a stochastic, randomly determined, network that attempts to estimate from the training data, both the mean and standard deviation of each model parameter using Bayesian inference. This is a method of statistically concluding information in which Bayes' theorem is used to update the probability of a hypothesis occurring. Bayes' theorem describes the probability of an event, based on prior knowledge of conditions that might be related to the event. This method assumes that the training data is fixed, and the model parameters are random variables that capture conditionally dependent and conditionally independent relationships between random variables. This approach allows BNNs to address the interdependence of the prior distributions of the parameters internal to the BNN, which while formally calculable is essentially impossible in practice.

Instead of employing other sampling methods that uses the variational inference to learn distributions which approximate the exact summary parameter of how uncertain we are, I will exploit the power of optimized ML libraries such as TensorFlow Probability . This allow users to implement this Bayesian inference method by using the *Flipout* estimator, which approximates model parameters and draws from their distribution during training and testing. This estimator is typically paired with an approximation of the gradient of the loss function like negative log likelihood or evidence lower bound (ELBO).

### 3.4. Future Work

Actual noise in 21 cm instruments are extremely difficult to replicate. Including these realistic noise realizations as part of the data used for forecasting purposes in ML training data should have many of these features present in them. This will be done in such a way that the level of the noise can be varied. Implementing such an approach allows for the implementation of varying amounts of noise to be applied to data that was unseen by the model during the training process resulting in the final output parameter uncertainty. This approach also allows for understanding how the relative level of signal and noise affect the overall uncertainty budget necessary to complement and perhaps improve upon current methods of measuring $\tau$. Building a more precise noise model will allow further assess the impact of foreground contamination from the wedge and other instrumental noise by applying these effects to the output of the 21 cm temperature maps before computing any summary statistic such as the mean, standard deviation, power spectrum, and so on.

### 3.5. This Thesis

This thesis is divided into five parts. Part I is devoted to introducing the scientific background. Part II details the structure of the interferometer telescope and its limitations. In Part III I provide a brief introduction to Machine Learning, I go through important terminologies, and how to use explores the building and the use of Convolution Neural Network Models. Part IV elaborates on the mathematical concepts used to build the simulated data used in this these and in my papers. Finally, in Part V I present findings from my published paper and work in progress for publication.

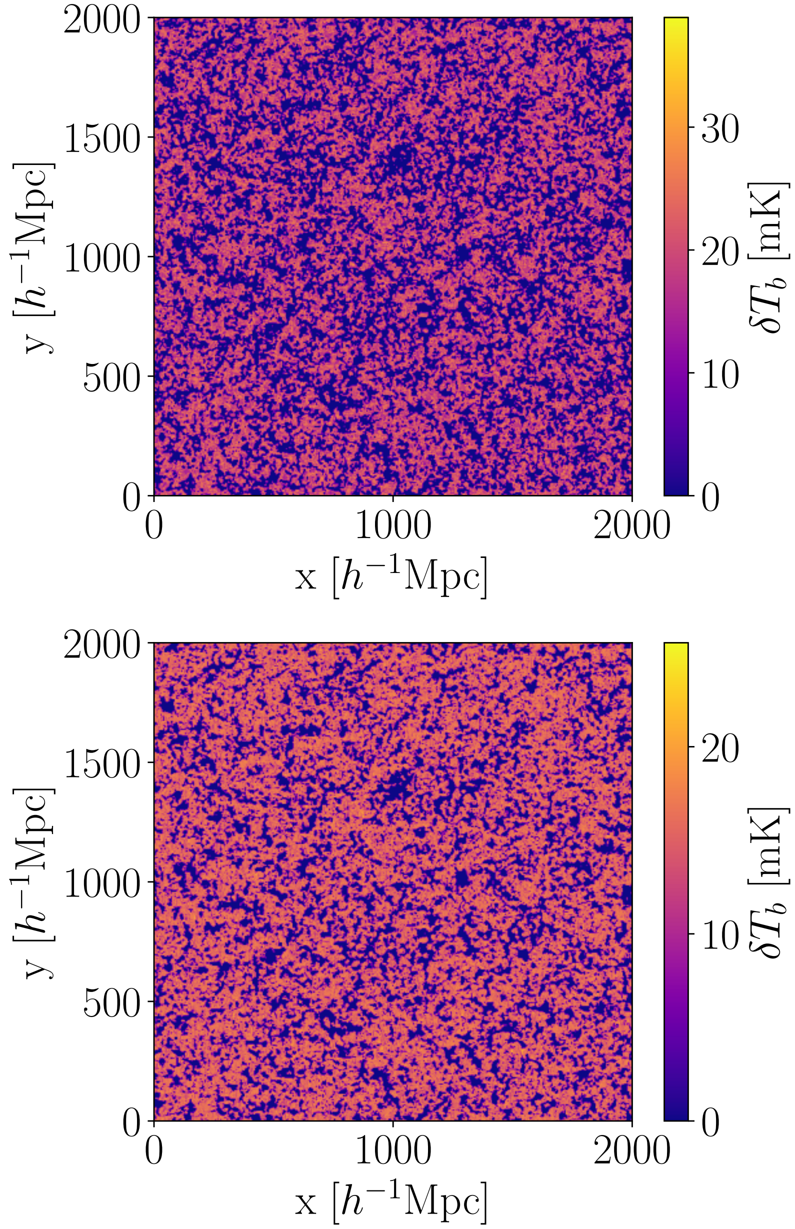


Figure 3.2: A visualization of the 21 cm field generated by `zreion` (top) and 21CMFAST (bottom) at comparable redshift values with similar values for $x_i(z)$. As can be seen, although there are differences in the shape of features, such as the dark-blue "bubbles" of ionized gas, overall the structures in the field are comparable.

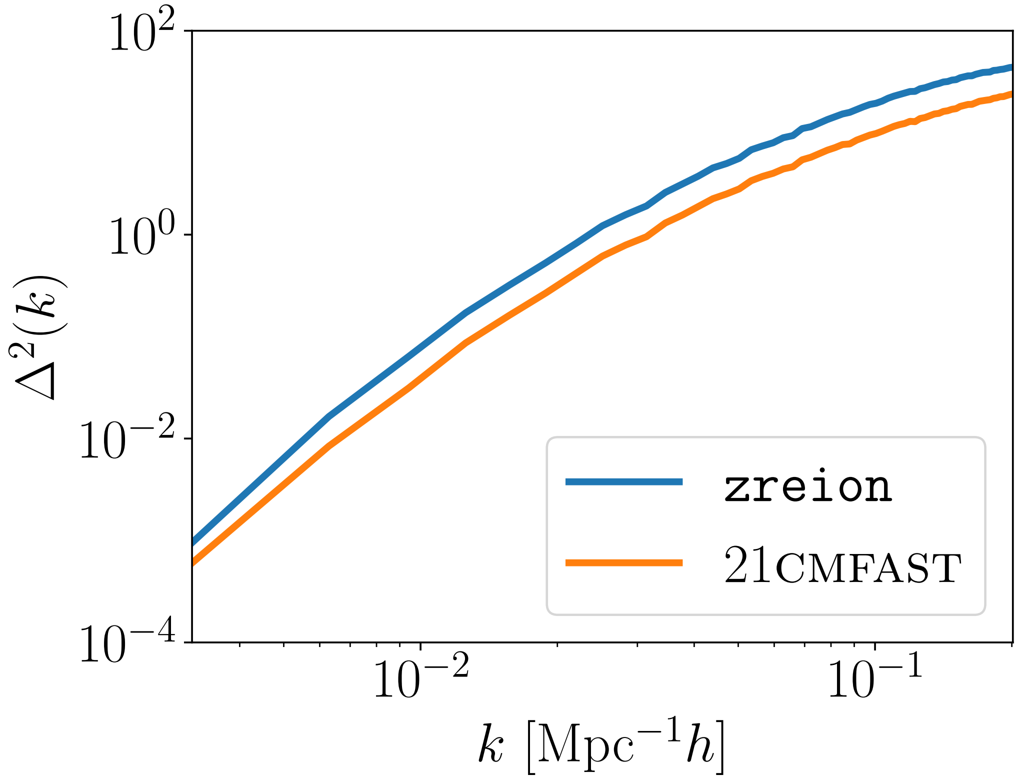


Figure 3.3: The one-point statistics (like pixel value distributions) and two - point statistics (like the dimensionless power spectrum $\Delta^2(k)$, above) of two different simulations are comparable. Nevertheless, there are differences, such as the overall amplitude of the power spectrum (though not the shape). Despite the apparent agreement, these differences are noticeable through image-based machine learning methods, and can bias the inferred parameter values, as shown in Figure 3.1. A significant effort goes toward ensuring that these model differences are accounted for in the estimators chosen and understanding their ultimate impact on the uncertainty of $\tau$.

# CHAPTER 4

# RADIO INTERFEROMETER

## 4.1. Analysis of Interferometer Response

In this section, I will introduce a simplified analysis of interferometry and other important concepts. The instantaneous response of a radio interferometer to a point source can be analyzed by considering the signal paths in the plane containing the electrical centers of the two interferometer antennas and the source under observation, Figure 4.1. For an extended observation, it is necessary to take account of the rotation of the Earth and consider the geometric situation in three dimensions. The two-dimensional geometry is a good approximation for short-duration observations and facilitates visualization of the response pattern.

Consider the geometric situation shown in Figure 4.1, where the antenna spacing is east to west. The two antennas are separated by a distance $D$, the baseline, and observe the same far field cosmic source of the interferometer. The source is sufficiently distant that the incident wavefront can be considered to be a plane over the distance $D$. The source will be assumed to have infinitesimal angular dimensions. The receivers will be assumed to have narrow bandpass filters that pass only signal components very close to $\nu$. The signal voltages are multiplied and then time-averaged, which has the effect of filtering out high frequencies. The wavefront from the source in a particular direction, reaches the right antenna at some time,

$$\tau_g = \frac{D}{c} \sin\theta \tag{4.1}$$

before it reaches the left antenna $\tau_g$ is called the geometric delay, and c is the speed of light. In terms of the frequency $\nu$, the output of the multiplier is proportional to

$$\begin{aligned} F &= 2\sin(2\pi\nu t)\sin(2\pi\nu(t-\tau_g)) \\ &= 2[\sin^2(2\pi\nu t)\cos(2\pi\nu\tau_g) - \sin(2\pi\nu t)\cos(2\pi\nu t)\sin(2\pi\nu\tau_g)] \end{aligned} \tag{4.2}$$

The center frequency of the receivers is generally in the range of tens of megahertz to hundreds of gigahertz. As the Earth rotates, the most rapid rate of variation of $\theta$ is equal to the Earth's rotational velocity, which is of the order of $10^{-4}$ rad $s^{-1}$. $D$ cannot be more than $10^7$ m for terrestrial baselines; therefore the rate of variation of $\nu\tau_g$ is smaller than $\nu t$ by at least six orders of magnitude. For an averaging period $T \gg 1/\nu$, the average value of $\sin^2(2\pi\nu t) = \frac{1}{2}$.

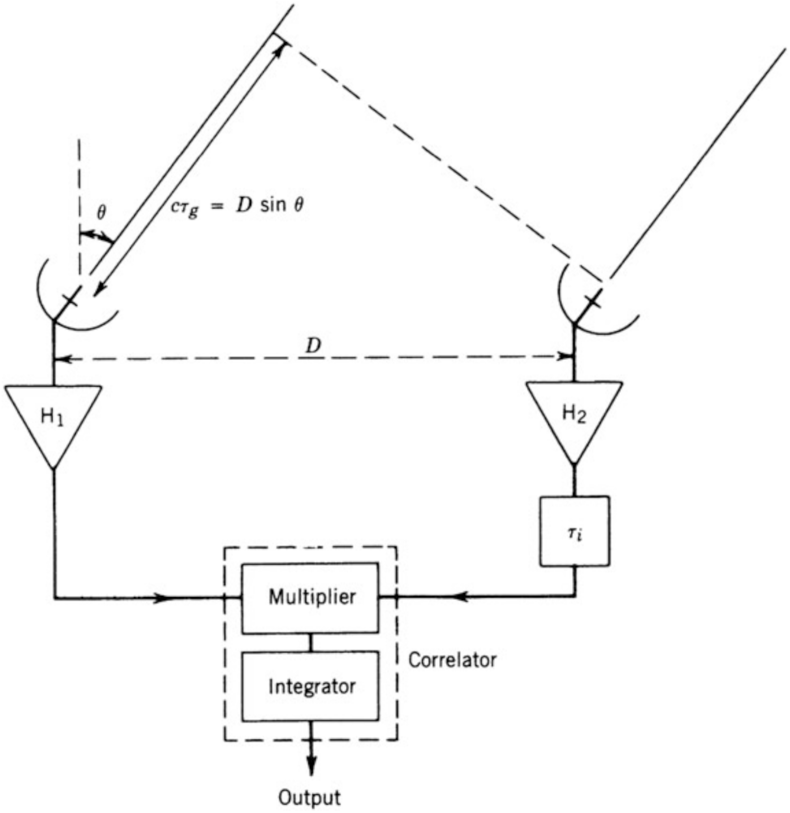


Figure 4.1: This is a simplified depiction of an interferometer showing bandpass amplifiers H1 and H2, the geometric time delay $\tau_g$ of the incident approximate plane, the instrumental time delay $\tau_i$, and the correlator consisting of a multiplier and an integrator. This image was taken from Taylor et al. (1999).

Figure 4.1 shows a general type of interferometer with the amplifiers H1 and H2, the multiplier, and an integrator with respect to time. An instrumental time delay $\tau_i$ is inserted into one arm. Assuming a point source, each antenna delivers the same signal voltage $V(t)$ to the correlator, and that one voltage lags the other by a time delay $\tau = \tau_g - \tau_i$, as determined by the baseline $D$ and the source direction $\theta$. The integrator within the correlator has a time constant $2T$ meaning that it sums the output from the multiplier for $2T$ seconds and then resets to zero after the sum is recorded. The output of the correlator represents a physical quantity with the dimensions of voltage squared and may be a voltage, a current, or a coded set of logic levels.

### 4.1.1. Types of Interferometer Arrays

#### Source Tracking Array

In general it is easy to deriving the relationship between intensity and visibility in a coordinate-free form and then show how the choice of a coordinate system results in an expression in the familiar form of the Fourier transform. In the case where the antennas track the source under observation, which is common in most situations but not for HERA, the phase reference position, sometimes also known as the phase-tracking center, becomes the center of the field to be imaged. For one polarization, an element of the source of solid angle at some position contributes a component of power at each of the two antennas.

The integration of the antenna response to the element solid angle over the source assumes that the source is spatially incoherent. In other words, the radiated waveforms from different elements solid angle are uncorrelated. This assumption is justified for essentially all cosmic radio sources. For a given antenna collecting area pointing in some direction at the sky in which the beam is pointed a normalized reception pattern is introduced. The output of the correlator can be expressed in terms of a fringe pattern corresponding to that for a hypothetical point source in the direction of the source, which is the phase reference position. The modulus and phase of the output of the correlator are equal to the amplitude and phase of the fringes. The phase is measured relative to the fringe phase for the hypothetical source. the output of the correlator has the dimensions of flux density ($\mathrm{W\,m^{-2}\,Hz^{-1}}$), which is consistent with its Fourier transform relationship with power.

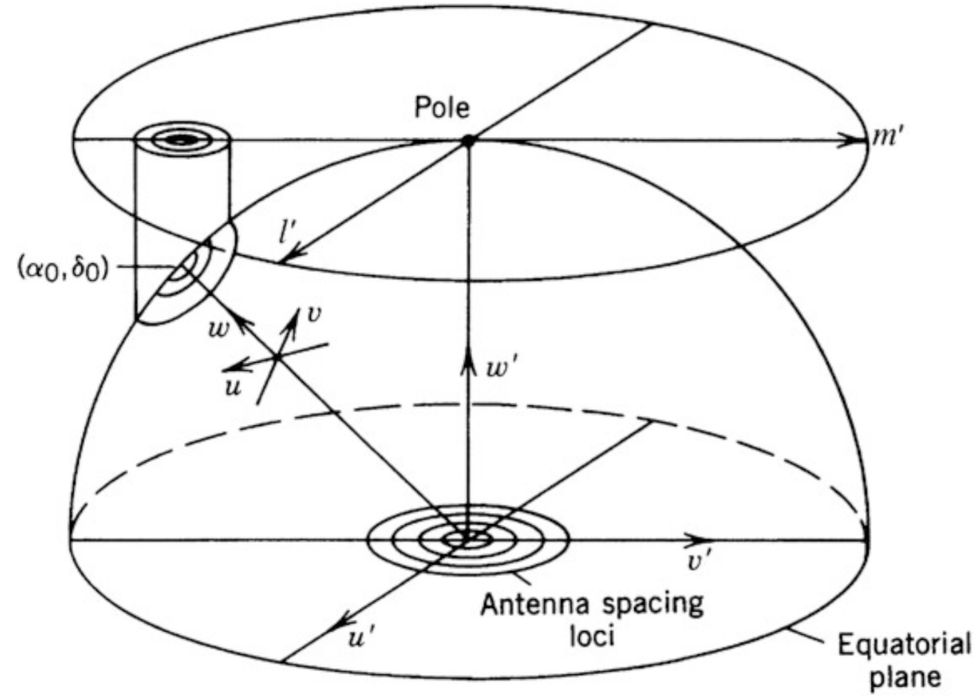


Figure 4.2: The $(u', v', w')$ coordinate system for an east–west array. The $(u', v')$ plane is the equatorial plane and the antenna spacing vectors trace out arcs of concentric circles as the Earth rotates. Note that the directions of the $u'$ and $v'$ axes are chosen so that the $v'$ axis lies in the plane containing the pole, the observer, and the point under observation $(\alpha_0, \delta_0)$. In Fourier transformation from the $(u', v')$ to the $(l', m')$ planes, the celestial hemisphere is imaged as a projection onto the tangent plane at the pole. The $(u, v, w)$ coordinates for observation in the direction $(\alpha_0, \delta_0)$ are also shown.This images was taken from a series of lectures, Taylor et al. (1999).

**East–West Linear Arrays**

HERA resembles the case of arrays with east–west spacing only. First, rotate the $(u', v', w')$, the quantities measured in the rotated system, coordinate system about the u axis until the w axis points toward the pole. The $(u', v')$ axes lie in a plane parallel to the Earth's equator. The east–west antenna spacing contain components in this plane only causing $w' = 0$, and as the Earth rotates, the spacing vectors sweep out circles concentric with the $(u', v')$ origin. The visibility equation and inverse visibility equation respectively can be written as,

$$V(u', v', w' = \mathbf{0}) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} A_N(l', m')I(l', m')e^{-2\pi(u'l'+v'm')}\frac{dl'dm'}{\sqrt{1-(l')^2-(m')^2}} \tag{4.3}$$

$$\frac{A_N(l', m')I(l', m')}{\sqrt{1-(l')^2-(m')^2}} = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} V(u', v', w' = \mathbf{0})e^{+2\pi(u'l'+v'm')}du'dv' \tag{4.4}$$

where I$(l', m')$ is the derived intensity distribution and $A_N(l', m')$ is the geometric mean of the beam patterns of the two antennas. Notice that this visibility space is $(u', v', w')$ and the Fourier Transform of it is the image space $(l', m', n')$.

In this imaging, the hemisphere is projected onto the tangent plane at the pole, as shown in Figure 4.2. In practice an image may be confined to a small area within the antenna beams. In the vicinity of such an area, centered at right ascension and declination $(\alpha_0, \delta_0)$, angular distances in the image are compressed by a factor $sin\delta_0$ in the $m_0$ dimension. Also, in imaging the $(\alpha_0, \delta_0)$ vicinity, it is convenient if the origin of the angular position variables is shifted to $(\alpha_0, \delta_0)$.

It is clear from Figure 4.2 that if all the measurements lie in the $(u', v')$ plane, then the values of v in the (u, v) plane become portrayed as having less depth or distance for directions close to the celestial equator. Obtaining two-dimensional resolution in such directions requires components of antenna spacing parallel to the Earth's axis. With the exception of short duration observation times, the effect of the Earth's rotation is to distribute the measurements in (u, v, w) space so that they no longer lie in a plane. Then the restriction of the synthesized field is acceptable in some case however, in other cases it may be necessary to image the entire beam to avoid source confusion, and several techniques are possible based for the following approaches:

- Write visibility equation in the form of a three-dimensional Fourier transform. The resulting intensity distribution is then taken from the surface of a unit sphere in (l, m, n) space.

- Large images can be constructed as mosaics of smaller ones that individually comply with the field restriction for two-dimensional transformation. The centers of the individual images must be taken at tangent points on the same unit sphere referred to in 1.

- Since in most terrestrial arrays the antennas are mounted on an approximately plane area of ground, measurements taken over a short time interval lie close to a plane in (u, v, w) space. It is therefore possible to analyze an observation lasting several hours as a series of short duration images, which are subsequently combined after adjustment of the coordinate scales.

## 4.2. Applications of Interferometry

Radio interferometers and synthesis arrays are used to make measurements of the fine angular detail in the radio emission from the sky. The angular resolution of a single radio antenna is insufficient for many astronomical purposes. Practical considerations limit the resolution to a few tens of arcseconds. For example, the beam width of a 100 meter diameter antenna at 7 mm wavelength is approximately 1700. The minimum angular separation of two sources that can be distinguished by a telescope depends on the wavelength of the light being observed and the diameter of the telescope. This angle is called the diffraction limit. The diffraction limit of large telescopes with diameter of approximately 8 m is about 0.01500, but the angular resolution achievable from the ground by conventional techniques is limited to about 0.500 by turbulence in the troposphere. It is also very important to be able

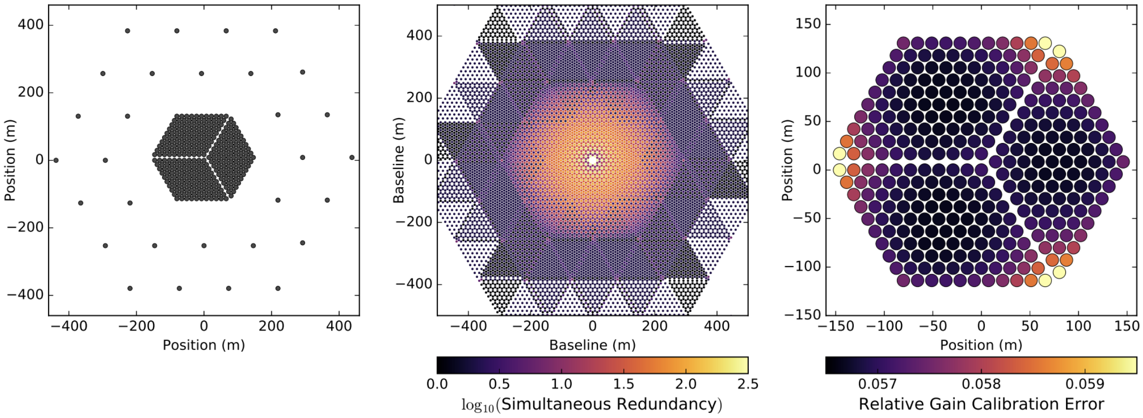


Figure 4.3: HERA's elements are divided between a 320-element, hexagonally-packed core and 30 outriggers (left). This produces instantaneous uv coverage at triple the element packing out to 250$\lambda$ at 150 MHz, supressing grating lobes in the synthesized beam (middle). All 350 elements can be redundantly calibrated, yielding calibration errors that are a small fraction of the residual noise per antenna (right). This figure was taken from DeBoer et al. (2017).

to measure parameters such as intensity, polarization, and frequency spectrum with similar angular resolution in both the radio and optical domains. The mathematical expression for half power beam width is $\text{Half power Beam width} = 70\frac{\lambda}{D}$ where $\lambda$ is the incident wavelength and D is the diameter of the beam. Trivially we can express the full power beam width as $FNBW\,2\left(70\frac{\lambda}{D}\right) \;= 140\frac{\lambda}{D}$, which is twice the half power beam width.

An advantage in the radio domain is that the phase variations induced by the Earth's neutral atmosphere are less severe than at shorter wavelengths. Future technology will provide even higher resolution at infrared and optical wavelengths from observatories above the Earth's atmosphere. Radio waves will remain important in astronomy since they reveal objects that do not radiate in other parts of the spectrum, and they are able to pass through galactic dust clouds that obscure the view in the optical range.

## 4.3. Astrophysical Radiation

The target signal of EoR experiments has a brightness temperature of order about 10 mK. At the 50-200MHz frequencies foreground radiation from the Milky Way, extragalactic sources and man made interference, with total brightness temperature of about 104 times greater

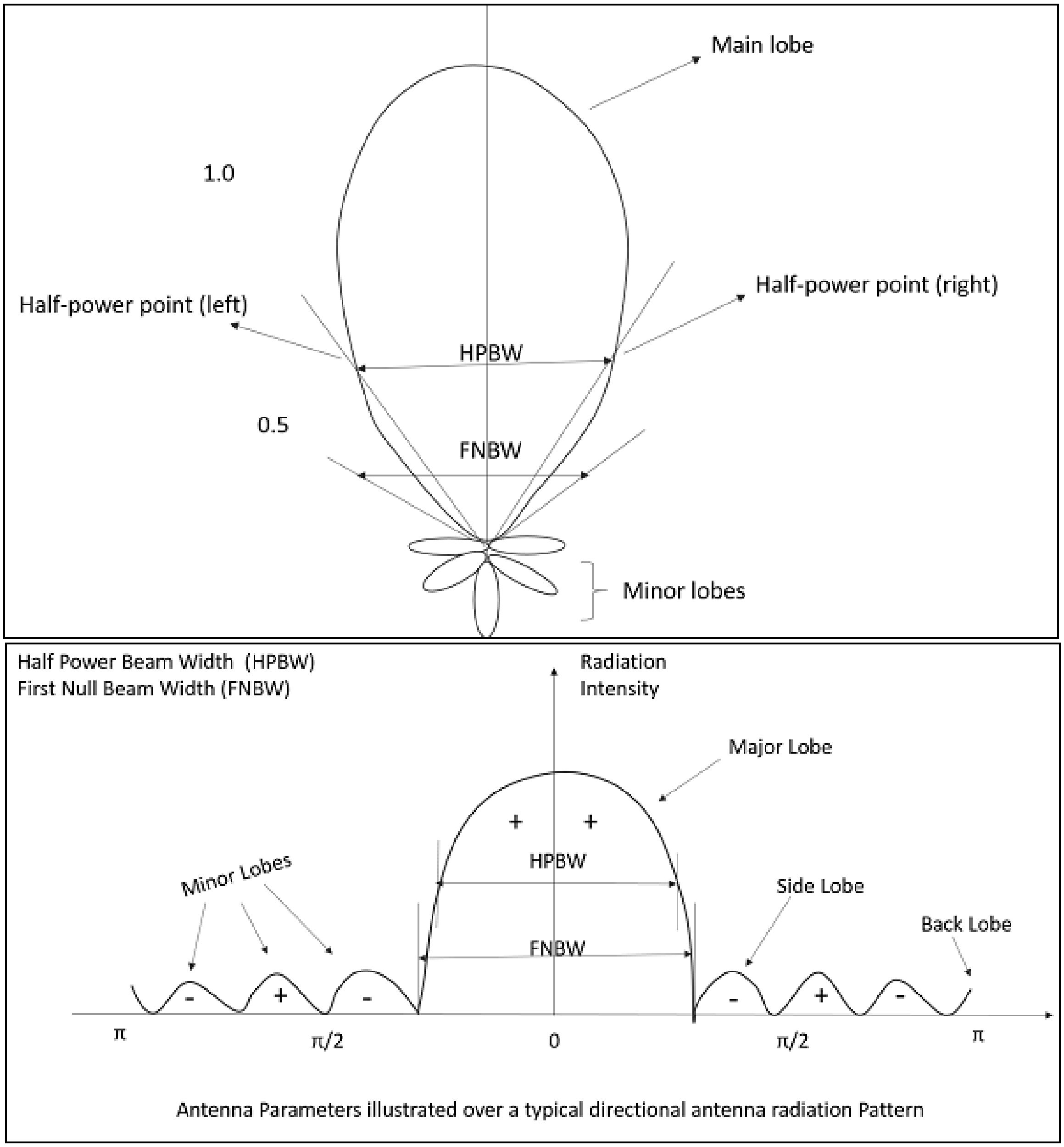


Figure 4.4: **TOP**: These images show the radiation pattern of an antenna, known as beam width. In the radiation pattern of an antenna, the main lobe is the main beam of the antenna where maximum and constant energy radiated by the antenna flows. Beam width is the aperture angle from where most of the power is radiated. The two main considerations of this beam width are Half Power Beam Width (HPBW) and First Null Beam Width (FNBW). **BOTTOM**: This image shows the half power beam width and first null beam width, marked in a radiation pattern along with minor and major lobes. This illustration was taken from https://www.tutorialspoint.com/antenna_theory/antenna_theory_beam_width.htm.

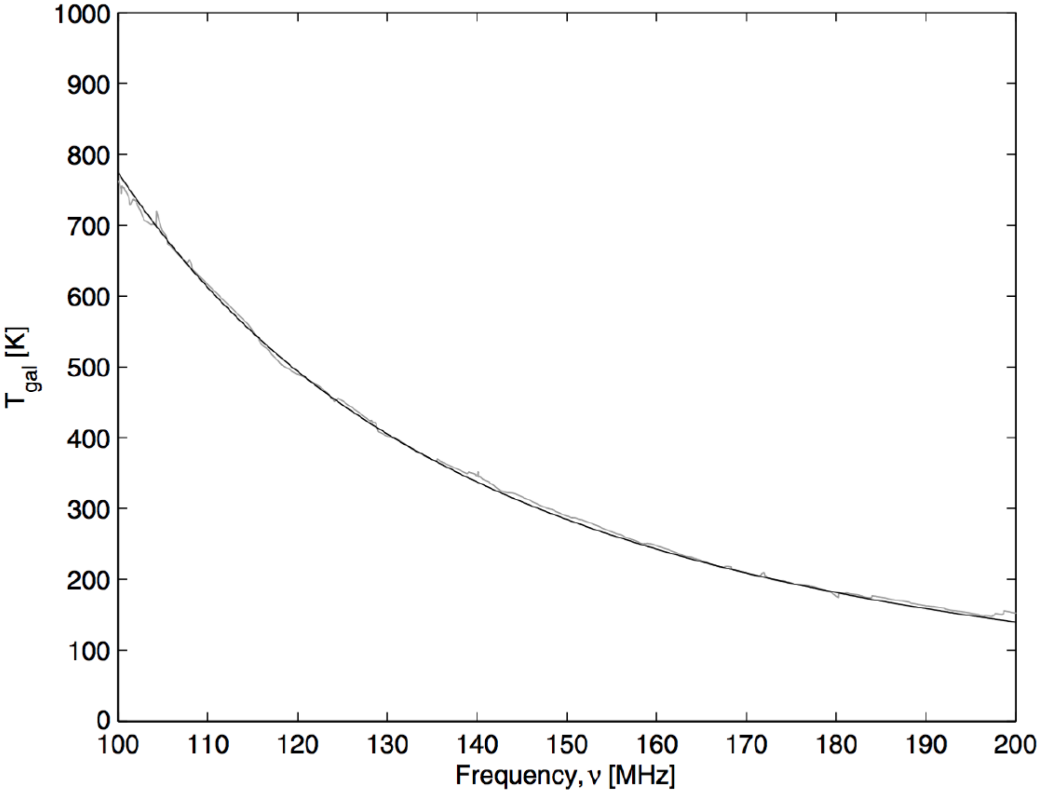


Figure 4.5: The low-frequency synchrotron spectrum of sky after removal of RFI and subtraction of $T_{cmb} = 2.725$. Their measurement is shown in light grey, overlaid with a well-fit power law of spectral index $\beta = 2.52 \pm 0.04$. This figure was taken from Rogers and Bowman (2008).

than the target signal, is the challenge to overcome. In this Chapter, I outline the current understanding of polarized and unpolarized foregrounds, and the implications for the dynamic range required for an EoR detection.

### 4.3.1. Synchrotron

Any accelerating charged particle will radiate light. For any low-frequency radio foregrounds, the radiation we are most interested in is the one emitted by electrons accelerated by Galactic magnetic fields. The radiation of a charged particle at non-relativistic velocities accelerated by magnetic field is described as the cyclotron radiation. The emission spectrum of this radiation is determined by gyration frequency about the magnetic field lines. At relativistic velocities, the spectrum becomes more complicated, and is referred to as synchrotron radiation. The equation of motion of a relativistic charged particle of mass m, charge q and velocity v can be written as,

$$\frac{d(\gamma m\vec{v})}{dt} = \frac{q}{c}\vec{v} \times \vec{B} \tag{4.5}$$

$$\frac{d(\gamma mc^2)}{dt} = q\vec{v} \cdot \vec{E} \tag{4.6}$$

where $\gamma$ is the Lorentz factor. Decomposing the velocity into components perpendicular and parallel to $\vec{B}$. Both $v_{||}$ and $v_{\perp}$ are constant proving that motion along the magnetic field lines is helical, with gyration frequency. Since the electrons follow a helical path, an observer will only see components of the emission when the motion is parallel to the line-of-sight. The observed radiation will be emitted along path $\delta s$ with radius of curvature r and angle $\delta\theta$, such that $\delta\theta = 2/\gamma$ and $\delta s = 2r/\gamma$. The frequency spectrum of synchrotron radiation is intimately tied to the helical geometry of the problem and is innately smooth as a function of frequency; it can be described as a power law. An example observation of low-frequency Galactic synchrotron is shown in Figure 4.5. The observed sky spectrum has a brightness temperature $T(\nu) \approx \nu^{-\beta}$, where spectral index is defined as $\beta = 2.52 \pm 0.04$. The HI EoR field has a complex structure in both the image plane and along the line-of-sight. The superposition of emission lines from this field will lead to unsmooth, structured emission per unit frequency. In the presence of synchrotron foregrounds being many orders of magnitude brighter than the 21 cm emission, this difference in spectral behavior is the single most important distinguishing feature between foregrounds and the EoR.

### 4.3.2. Stokes parameters

It is extremely unlikely that we will observe radiation from electrons spiralling along magnetic field lines that are exactly parallel or perpendicular to our line-of-sight. Instead, some elliptically polarized component of the radiation is observed. The Stokes parameters are quantities used to describe the polarization state of electromagnetic waves. We can describe a monochromatic electromagnetic wave propagating towards the observer as the real part of

$$\vec{E} = \vec{E}_0 e^{-i\omega t} = \begin{pmatrix} E_1 e^{-i\phi_1} \\ E_2 e^{-i\phi_2} \end{pmatrix} \tag{4.7}$$

The real part of the vector above can be mapped to the principle axes of an ellipse at position angles $\Psi$ and $\chi$ to the Cartesian plane. These two angles, $E_1$, and $E_2$, can be used to define the Stokes parameters for monochromatic waves. The values of $E_1$, $E_2$, $\phi_1$ and $\phi_2$ cannot be precisely measured. Instead, radiometers measure the average sum of squares of the components of $\vec{E}(t)$. These can be used to define the Stokes parameters for quasi-monochromatic waves:

$$I \equiv \langle E_1 E_1^* \rangle + \langle E_2 E_2^* \rangle = \langle E_1^2 + E_2^2 \rangle \tag{4.8}$$

$$Q \equiv \langle E_1 E_1^* \rangle - \langle E_2 E_2^* \rangle = \langle E_1^2 - E_2^2 \rangle \tag{4.9}$$

$$U \equiv \langle E_1 E_2^* \rangle + \langle E_2 E_1^* \rangle = \langle 2 E_1 E_2 cos(\phi_1 - \phi_2) \rangle \tag{4.10}$$

$$V \equiv -i(\langle E_1 E_2^* \rangle - \langle E_2 E_1^* \rangle) = \langle 2 E_1 E_2 sin(\phi_1 - \phi_2) \rangle \tag{4.11}$$

The interpretation of these parameters is as follows:

- Stokes I measures the total intensity of the electric field.
- Stokes Q and U measure the orientation of the electric field relative to, in this case, the x-axis of the Cartesian plane, where Q measures the projection parallel to the x-axis and U measures the projection at 45 degrees to the x-axis. They are clearly linked, as $\tan(y) = U = Q$.
- Stokes V is the circularity parameter, measuring the ratio of the principle axes of the ellipse.

| Instrument | Collecting Area<br>($m^2$) | Foreground<br>Avoidance | Foreground<br>Modeling |
|---|---|---|---|
| PAPER | 1,188 | $0.77\sigma$ | $3.04\sigma$ |
| MWA | 3,584 | $0.31\sigma$ | $1.63\sigma$ |
| LOFAR NL Core | 35,762 | $0.38\sigma$ | $5.36\sigma$ |
| HERA-350 | 53,878 | $23.34\sigma$ | $90.97\sigma$ |
| SKA1 Low Core | 416,595 | $13.4\sigma$ | $109.90\sigma$ |

Table 4.1: Comparing telescope sensitivities as a function of redshift to models of the spherically averaged 21 cm EoR power spectrum with 50% ionization at z = 9.5, characterized by the dimensionless power spectrum parameter with 1080 hours observation, integrated over a $\Delta z$ of 0.8.

For monochromatic waves, $I^2 \geq Q^2 + U^2 + V^2$. In practice, observed radiation fields are rarely purely elliptically polarized instead they are a superposition of electric fields, each with its own polarization state. The radiation field is completely unpolarized if Q = U = V = 0. It it common to refer to the "polarization fraction" or "degree of polarization" of observed radiation, p.

$$P \equiv \frac{\sqrt{Q^2 + U^2 + V^2}}{I} \tag{4.12}$$

### 4.3.3. Foreground Radiation

The diffuse Galactic radio emission at meter wavelengths is significant because this emission is the dominant contaminating foreground for the study of the redshifted 21 cm line from the EoR (Shaver et al. (1999), Furlanetto et al. (2006)). About 75% of the sky brightness is due to diffuse emission from our Galaxy, most of which is caused by synchrotron radiation. Current diffuse radio emission telescopes at the meter wavelength are not adequate to estimating these contaminating effects on the EoR signal. Knowledge of both their intensity and polarization structure for extracting this cosmological signal from the data on the angular scale of 1-30 arcmin will be essential. For the foreground avoidance approach, several design optimizations allow HERA to achieve significantly higher sensitivities than LOFAR and comparable sensitivities to SKA, despite its modest collecting area.

Galactic foreground lacks significant fine-scale structure at small angular scales and at low frequencies from 315 to 380 MHz and the diffuse synchrotron emission from the Galaxy is generally very smooth and is mostly resolved out by the interferometer. Extrapolating to lower limits of frequencies and angular scales for both the spectral and the spatial properties is not possible due to the fact that the diffuse emission can vary across the sky. The mean spectral index of the synchrotron emission at high Galactic latitude has been quite well constrained to be $\beta = 2.5 \pm 0.1$ in the 100-200 MHz range, (Bernardi et al. (2009), Basu et al. (2019), Ghosh et al. (2012)), the relevant frequencies for the EoR.

## 4.4. Measuring the EoR

The optical depth lets us see through the entire universe back almost to the period of recombination. An observation of an area of sky over 13 billion light-years provides an average signal that is both weak over a cosmic volume subtends the entire sky. Initial measurements of the EoR measure a statistical power spectrum of the signal over the sky since the nature of the reionization process should have a specific spatial signature. The main goal is to measure a range of spatial scales on the sky, rather than the actual image the signal directly. Imaging does still remain an ultimate goal to fully understand the process, however it will likely need a greater understanding of the signal characteristics and the systematics to achieve which is a more difficult goal.

As stated in Section 4.3, between us and the EoR is a much brighter signal due to diffuse Galactic synchrotron radiation, supernova remnants and extragalactic radio sources. All of these foreground signals are smooth spectrum sources whereas the expected spectrum of the EoR is expected to be rough since it is made up of nonionized regions which are randomly distributed over a wide range of redshifts. Knowledge of this fact allows us to try and isolate foregrounds from the EoR which is about 5 orders of magnitude brighter than the 21 cm signal.

PAPER's, HERA's predecessor, lack of imaging support and its uneven uv sampling leave it with limited diagnostic capability of direction-dependent systematics such as polarization leakage from Faraday-rotated emission. HERA emphasizes the proven approaches of redundant calibration and delay filtering, while simultaneously increasing the extent and density of uv sampling for high-fidelity imaging.

### 4.4.1. Wedge

HERA's design informs us on how smooth-spectrum foregrounds interact with instrument chromaticity to produce the characteristic "wedge" of foreground leakage in Fourier space outside the region where the 21 cm signal dominates, the "EOR window". The wedge is a consequence of the chromatic response of the interferometer. The boundary between the wedge and the EoR window is determined by the separation between antennas, signal reflections within antennas, and the angular response of the antenna beam. To the limits of current sensitivity, foreground emission is absent outside of the wedge and can only appear there through instrumental leakage.

The power spectrum measurement provides the spatial correlations across the sky, characterized by the magnitude of the wavenumber, **k**. Although the full magnitude of the **k**-vector is used, it is more informative to split them into two components (perpendicular and parallel), $\mathbf{k} = \mathbf{k}_\perp + k_{||}\hat{z}$ where $|\mathbf{k}_\perp| \equiv \mathbf{k}_\perp = 2\pi b/(\lambda X)$ is determined by the antenna baseline, b, and $k_{||} = 2\pi/(YB)$ by the bandwidth, B. X and Y are some cosmological parameters relating angular size and spectral frequency to cosmic volumes respectively or relating wavenumber to physical volume at a given redshift. $\mathbf{k}_\perp$ corresponds to the plane of the sky and $k_{||}$ to the line-of-sight. This allows us to split the chromatic response of the interferometer visibility measurement from the instrument bandpass and isolate a phase space where smooth-spectrum foreground sources contaminate the signal of interest from where they don't. The $\mathbf{k}_\perp$ components are directly proportional to the baselines and $k_{||}$ are proportional to the Fourier transform of the frequency response. The Fourier transform of a frequency spectrum is a delay spectrum. The cosmic evolution limits the largest bandwidth which determines the smallest $k_{||}$ to about 10 MHz. For larger bandwidths the evolution of the Universe begins to impact the result. For HERA, wavenumbers are dominated by the bandwidth and not the baseline.

The unit of the spatial wavenumber **k** is the inverse of length, in this case the relevant length scale is megaparsecs (Mpc). The Hubble parameter, $H_0$, is further normalized by a factor h, where $H_0 = 100h(\mathrm{km}/s)/\mathrm{Mpc}$, such that the wavenumbers are expressed in units of h Mpc$^{-1}$. At the redshifts of interest, X has a value of about 160 Mpc deg$^{-1}$ and Y has a value of about 16 Mpc MHz$^{-1}$. The contaminated phase space is conveniently a wedge-shaped region in $\mathbf{k}_\perp$ - $k_\|$ space such that the ratio of the time-delay across a given baseline (b/c) and the delay associated with a given bandwidth (1/B) is less than some parameter determined by the details of the system, which we denote as $1/\beta$. Substituting $\mathbf{k}_\perp$ and $k_\|$ in for b and B in this ratio, this wedge is bounded by,

$$k_\| \leq \beta \frac{X\lambda}{Yc}\mathbf{k}_\perp + \frac{S}{Y} \tag{4.13}$$

where an offset S accounts for effects related to the combined spectral smoothness of the foregrounds and the antenna response. The techniques to measure the power spectrum may be broken down into two principle techniques, delay-space and map-making. I will elaborate on these two techniques in the following two sections.

### 4.4.2. Delay Spectrum

The response of an interferometer measures the power in the fourier modes of the sky within its beam and we can see that it is a natural instrument to use for the measurement of the EoR spatial power spectrum. The delay-spectrum approach leverages the interferometer measurement to optimize sensitivity to the desired modes while rejecting modes contaminated by the foreground power in the wedge. The delay-spectrum approach does not combine baselines before squaring and calculating the power spectrum, which contrasts to the map-making techniques. The sky power spectrum $P(\mathbf{k})$ is linearly proportional to the Fourier transform along the frequency axis (the delay-transform) of an interferometer baseline visibility, V:

$$P(\mathbf{k}) = \frac{X^2Y}{4k_B^2}\left[\frac{V^2}{\Omega_b B\lambda^4}\right] \tag{4.14}$$

$\Omega_b$ is the integrated beam response, B is the effective bandwidth, $\lambda$ is the observation wavelength, and $k_B$ is Boltzmann's constant. The terms in square brackets are instrumental terms, as opposed to the constants and cosmological parameters out front.

For a fixed number of elements using the delay-spectrum approach, hexagonal packed redundant arrays provide about an order of magnitude improvement over imaging arrays. Since the baselines go as order $N^2$, this corresponds to using about 1/3 of the number of elements to yield the same performance. Although a filled hexagonal packed array provides excellent imaging, the resolution could be limited unless outriggers are included. The HERA array design incorporates both hexagonal packed redundant arrays and outriggers.

### 4.4.3. Map Making

In contrast to the delay-spectrum approach, mapmaking approaches combine baselines to build information before squaring and calculating the power spectrum. The image domain is a natural place to combine information from partially-coherent pairs of visibility measurements. First make an image cube of the sky as a function of angular position and frequency, take the spatial Fourier transform, square and bin the cube, then subtract the dominant foreground power and take the transform to determine the EoR power spectrum. The data compression step keeping track of statistical properties of the maps including complex frequency and position dependent point spread functions and noise covariance matrices, but can be computationally challenging. Existing approaches have had to make a number of approximations, including that point spread functions do not vary over the field of view and that noise is not correlated.

One of the benefits of mapmaking is that it not as vulnerable to polarization leakage as the delay-spectrum approach since only polarization mismodeling can cause leakage from Stokes Q or U to I. Another advantage is that sky images can be interesting both for dealing with measurement systematics and for accessing the non-Gaussian observational signatures (ionized bubble structures) that are missed by the power spectrum statistics. Although the direct subtraction of bright foregrounds expands the EoR window, accessing these high-order statistics or the modes of the power spectrum inside the wedge requires extreme precision in calibration and forward-modeling of bright foreground models through instrument systematics. For this reason, this approach to measure the EoR and initially characterize the power spectrum is not implemented. The delay-spectrum technique is used instead while researcher developing tools for other approaches.

# CHAPTER 5

# TELESCOPE SUMMARY

## 5.1. Telescopes

### 5.1.1. The Hydrogen Epoch of Reionization Array

The Hydrogen Epoch of Reionization Array (HERA) uses the unique properties of the 21 cm line of neutral hydrogen to probe the Epoch of Reionization and the preceding heating epoch, roughly 0.3 to 1 Gyr after the Big Bang. These epochs represent the frontier in studies of cosmic structure formation, during which emission from the first stars and black holes heated and reionized the Universe. By directly observing the time evolution of these fluctuations in the large scale structure of 21 cm emission, HERA provides an unique tool to study the cosmological and astrophysical processes that governed the formation of the first galaxies and black holes, and how they heated and ionized the primordial intergalactic medium (IGM). With the Precision Array to Probe the Epoch of Reionization (PAPER) and the Murchison Widefield Array (MWA), characterization of the strong continuum foregrounds masking the cosmological 21 cm signal is possible. Techniques for overcoming foreground systematics in power spectrum measurements were also developed. Through these analysis efforts, The HERA collaboration has developed a robust pipelines for estimating the 21 cm power spectrum and setting upper limits on 21 cm emission during reionization.

Continuation of these efforts uncovered errors in estimating signal loss in the PAPER power-spectrum pipeline. Experience with PAPER and the MWA culminated in the design of HERA, a new instrument optimized to deliver both the sensitivity for precision constraints of the 21 cm power spectrum and a spectrally smooth instrumental response necessary for foreground mitigation.

#### Scientific Background: Precision Constraints on Reionization

HERA's primary science goal is to change the understanding of the first stars, galaxies, and black holes, and their role in driving reionization. Through power spectral measurements of the 21 cm line of hydrogen in the IGM, HERA will be able to directly constrain the topology and evolution of reionization, opening a window into the astrophysics of the first luminous objects and their environments. The spectral nature of 21 cm cosmology means that the signal at each observing frequency can be associated with an emission time to determine both the time evolution and three-dimensional spatial structure of ionization in the IGM. This 3D structure has information about the clustering properties of galaxies, allowing us to distinguish between models, regardless of whether or not they predict the same average ionized fraction. The new telescope is optimized for 3D power-spectral measurements and with support for theoretical modeling efforts, the HERA program will advance our understanding of early galaxy formation and cosmic reionization. HERA builds on the advances of first-generation 21 cm EoR experiments (PAPER; Parsons et al. (2010)), the Murchison Widefield Array (MWA; Bowman et al. (2013) Tingay et al. (2013)). However, current experiments cannot expect more than marginal detections of the EoR signal.

$$\Delta^2(k) \equiv k^3 \frac{P(k)}{2\pi^2} \tag{5.1}$$

The expected performance of HERA relative to current and planned telescopes to detect the peak of reionization (as well as the total collecting area) is shown in Table 5.1. The sensitivity calculations done were performed with `21cm Sense1` (Pober et al. (2013),Pober et al. (2014)). For the foreground avoidance approach, several design optimizations allow HERA to achieve significantly higher sensitivities than LOw Frequency ARray (LOFAR) and comparable sensitivities to SKA, despite its modest collecting area. The primary driver is HERA's compact configuration. The 21 cm signal is a diffuse background, with most of its power concentrated on large scales. Therefore, most of an instrument's sensitivity to the EoR comes from short baselines. Since HERA a filled out to ≈ 300 m, for a fixed collecting area, one fundamentally cannot build an array with more short baselines. Within about a 150 m radius from the center, LOFAR has only 11 stations, amounting to just over 8000 $m^2$ of collecting. Within this radius, the SKA is nearly filled, with aboout 80% the collecting area of HERA; however, the SKA underperforms in the foreground avoidance schema, where long baselines lose more modes of the power spectrum to foreground contamination Parsons et al. (2012).

**Scientific Background: Secondary Scientific Objectives**

By advancing our understanding of reionization astrophysics, HERA will improve CMB constraints on fundamental cosmological parameters by removing the optical depth to reionization. HERA measurements will be able to break the degeneracy between the constraints on $\tau$ and the sum of the neutrino masses, which has been identified as a potential problem for CMB lensing experiments. This would represent an $\approx 5\sigma$ cosmological detection of the neutrino masses even under the most pessimistic assumptions allowed by neutrino oscillation experiments, making HERA key to understanding neutrino physics. HERA's estimate of $\tau$ would also break the degeneracy between $\tau$ and the amplitude of matter fluctuations that arises when using only CMB data. HERA effectively reduces error bars on $\sigma_8$ by more than a factor of three Liu et al. (2016).

In addition to measuring the power spectrum, there is the potential for HERA to directly image the IGM during reionization over the 1440 deg$^2$ stripe that transits overhead, which is comparable to future WFIRST large area near-IR surveys. After 100 hours on a single field (achievable in 200 nights). HERA has the ability to reach a surface brightness sensitivity of 50 $\mu$Jy/beam (synthesized beam FWHM about 240) compared to the brightness temperature fluctuations of up to 400 $\mu$Jy/beam in typical reionization models. At this sensitivity alone makes HERA more than capable of detecting the brightest structures at z = 8 with SNR > 10. Additionally, the design of HERA places it in a unique position to directly explore calibration techniques while retaining a high quality point spread functions for imaging and identifying foregrounds.

Prior to reionization, the 21 cm signal is a sensitive probe of the first luminous sources and IGM heating mechanisms. First stars are expected to form at a redshift of about 25 - 30 and their imprint on the 21 cm signal is expected to be sensitive to the halo mass where they are formed. The IGM is then expected to be heated by either the first generation X–ray

| Instrument Design Specification | Observational Performance |
|---|---|
| Element Diameter: 14 m | Field of View: 9° |
| Minimum Baseline: 14.6 m | Largest Scale: 7.8° |
| Maximum Core Baseline: 292 m | Core Synthesized Beam: $25'$ |
| Maximum Outrigger Baseline: 876 m | Outrigger Synthesized Beam: $11'$ |
| EOR Frequency Band: 100–200 MHz | Redshift Range: $6.1 < z < 13.2$ |
| Extended Frequency Range: 50–250 MHz | Redshift Range: 4.7 < z < 27.4 |
| Frequency Resolution: 97.8 kHz | LoS Comoving Resolution: 1.7 Mpc (at z = 8.5) |
| Survey Area: 1440 $\text{deg}^2$ | Comoving Survey Volume: $\approx$ 150 $\text{Gpc}^3$ |
| Tsys: $100 + 120(\nu/150MHz)^{-2.55}$ K | Sensitivity after 100 hrs: 50 $\mu$Jy $\text{beam}^{-1}$ |

Table 5.1: HERA-350 design parameters and their observational consequences. Angular scales computed at 150 MHz.

binaries or by the hot interstellar medium produced by the first supernovae. Dark matter annihilation could have left an imprint in the 21 cm signal.

**System Design**

As described in the previous sections, the critical insights from the first generation 21 cm EOR experiments have been applied to define the requirements for HERA. It is designed to ensure that foregrounds remain bounded within the wedge while delivering the sensitivity for high-significance detection of the 21 cm reionization power spectrum with established foreground filtering techniques Pober et al. (2014). In this section, I summarize key features of the HERA design (see Table 5.1) and system architecture directly inherited from the PAPER and MWA experiments. HERA began by reusing the analog, digital, and real-time processing systems deployed for PAPER-128. This allows for immediate observing with the new elements with a well-characterized system. As HERA develops, the architecture is incrementally upgraded to improve performance and add features while simultaneously addressing issues of modularity and scalability. As with PAPER, HERA proceeds in stages of development, with annual observing campaigns driving a cycle of development, testing, system integration, calibration, and analysis. This cycle ensures that HERA's instrument is always growing, that systematics are being found and eliminated at the earliest build-out stages, that data analysis pipelines are tested and debugged while data volumes are smaller, and that HERA is always producing high quality science.

## 5.2. Other Telescopes

HERA is not the only instrument researching the EoR and how to detect it. Several low-frequency interferometers around the world are contributing to the understanding of the EoR and the difficulties of observing it. Below I briefly describe a few of the leaders of the field – but it is not an exhaustive list.

### The LOw-Frequency ARray (LOFAR)

The LOFAR is an interferometer made-up of “stations”, the core of which are arranged in a random scatter in the Netherlands. LOFAR baselines extend across Europe with stations in Ireland, Sweden, Germany, Poland and other locations. These extremely long baselines can provide LOFAR with exquisite imaging capabilities. LOFAR observations will allow detection and quantification of the Cosmic Dawn and EoR over wide range in angular scales and redshifts. Such measurement will help answer the main questions surrounding the earliest phases of the formation of the Universe: The nature of the first objects that ended the Dark Ages, ushering in the Cosmic Dawn and the reionization of the high-redshift IGM. The experiment seeks to answer the following questions: What is the relative role of galaxies and active galactic nucleus, of UV-radiation and X-rays? When did the EoR start and how did it percolate through the intergalactic medium? Did the low or high density regions re-ionized first? What are the detectable imprints that the re-ionization process left on the 21-cm signal and is it possible to learn from 21-cm measurements about the matter density fluctuations on the conditions prior to the EoR? Is it possible to learn about the formation of supermassive black holes and the duration of their active phases?

### Murchison Widefield Array (MWA)

MWA, based in Murchison Radio-astronomy Observatory in Western Australia, was deployed, it was one of the first generation of telescopes aiming to detect the EoR power spectrum. It was designed as a general-purpose array with emphasis on imaging capability and brightness sensitivity on the angular scales relevant to the EoR. In the years following the initial design and deployment, it has become increasingly apparent that the dynamic range requirements such as accuracy of calibration, accuracy of sky model etc of the EoR detection experiment are just as challenging as the sensitivity requirements.

Aside from raw sensitivity, the capability of a radio interferometer is usually constrained by its angular resolution and by the spatial scales that are measured given the antenna layout. MWA layout was a hybrid configuration having baselines ranging in length from approximately 7 to 2800 metres, with many tiles in the core region. The synthesised beam size of the Phase I array is approximately 2 arcmin at 150 MHz. Improving the angular resolution of the MWA for Phase II was identified for its potential to impact a broad range of science areas. In addition to the ability to better resolve objects and structures within sources, improved angular resolution directly impacts the classical and sidelobe confusion in continuum images.

**The Square Kilometre Array (SKA)**

The SKA will be one of the premier instrument to study radiation at the 21 cm wavelengths from the cosmos, and in particular hydrogen, the most abundant element in the universe. SKA will probe the dawn of galaxy formation. The focus of this program is the first luminous objects in the Universe and their formation. At a redshift of around 1100, the Universe became largely neutral as protons and electrons combined to form the first hydrogen atoms and the photons that we now see as the CMB began free streaming across the Universe. Today the Universe is largely ionized. The epoch of reionization of the Universe is dated to $z \approx 6$ to 12 from observations.

The original focus of the SKA was observations of the 21 cm HI line from galaxies, and such observations remain a significant focus of the SKA Key Science Program. Neutral hydrogen is the raw material from which stars form. The peak of the star formation rate in the Universe occurred at redshifts between about 1 to 2 (e.g. Madau et al. (1996), Adelberger (2000), Hopkins and Beacom (2006)). The SKA will be able to probe the evolution of neutral hydrogen to this crucial point in the assembly of galaxies. A third motivation for astronomy is that it provide tests of theories of fundamental physics such as observations of gravitational lensing by the Sun provided some of the key early support for the Einsteins General Theory of Relativity, which now finds widespread use, such as in corrections applied to timing signals from the Global Positioning System satellites.

SKA will be built on two sites: the Karoo Radio Quiet Zone, currently occupied by HERA, will be the central location of the "high–mid band" (350 MHz – 14 GHz; "SKA-Mid") observatory and the Murchinson Radio-astronomy Observatory, currently occupied by the MWA, will be the central location of the "low band" observatory (50 – 350 MHz; "SKA-Low"). SKA-Low will consist of over 100,000 receiving elements in an imaging configuration. It will be the most powerful low-frequency radio telescope ever created, and be used not only for EoR science but a host of low-frequency science objectives.

# CHAPTER 6

# MACHINE LEARNING MOTIVATION

## 6.1. Introduction

Upcoming observations of the EoR are expected to be primarily sensitive to astrophysical parameters related to properties of the first stars and galaxies, rather than cosmological parameters such as those inferred from measurements of the cosmic microwave background (CMB). One exception to this is $\tau$, the optical depth to the CMB. Radio interferometry telescopes such as the Hydrogen Epoch of Reionization Array (HERA ), the Low Frequency Array (LOFAR ), and the Square Kilometre Array (SKA ) aim to map the thermal distributions and ionization state of neutral hydrogen in the IGM throughout Cosmic Dawn, and will be the only direct probes of the formation of the first generations of stars, galaxies, and stellar-mass black holes. Full tomographic 3D images generated by these instruments can be useful to learn information about these sources that precipitated reionization.

More imminently, statistical measurements using the 21 cm power spectrum can provide insight about the EoR. Thus far, measurements of the 21 cm power spectrum upper limits have been established at different Fourier wavenumbers and redshift values Paciga et al. (2013); Beardsley et al. (2016); Patil et al. (2017); Kolopanis et al. (2019). However, direct extraction of astrophysical and cosmological parameters from the power spectrum or from images is challenging due to large levels of contamination from bright foreground emission which are typically several orders of magnitude larger than the target signal. This limitation makes simple imaging of the sky using traditional techniques impossible. The main difficulty in detecting the faint signal from the EoR is to separate it from various types of foreground emissions such as galactic synchrotron radiation and extragalactic point sources Di Matteo et al. (2004); Jelić et al. (2008). Another common approach proposed for extracting information about the EoR from observations is to compute the power spectrum using Fourier modes that are uncontaminated by these foregrounds. The downside to any power-spectrum based approach that is its insensitive to any non-Gaussian information and therefore does not leverage all of the information present in the images. The power spectrum does not capture the full information present in the field because the 21 cm field is highly non-Gaussian during the EoR Majumdar et al. (2018); Shimabukuro and Semelin (2017).

Using measurements of the 21 cm signal, it may be possible to infer key properties of the EoR, such as the timing and duration of reionization. Although these properties are interesting in their own right, they also provide important information about the cosmological parameter $\tau$, which measures the optical depth to the CMB. To date $\tau$ has been measured by the Planck collaboration, which has provided important constraints on its value. However, the value of $\tau$ still has some of the largest relative uncertainty of the cosmological parameters, which impacts the uncertainty of other parameters such as the density of dark matter $\Omega_c$ and clustering of matter $\sigma_8$. Liu et al. (2016) proposed using measurements of the 21 cm power spectrum as a way to provide tighter constraints than is currently feasible from CMB measurements alone, which can lead to improved uncertainties of other parameters. The authors jointly constrained $\tau$ and other cosmological parameters using semi-numeric simulations of reionization, leading to a fractional uncertainty several times better than current CMB-based constraints. Thus, data analysis techniques that provide constraints on $\tau$ are promising ways forward for measuring cosmological parameters more accurately.

An alternative approach to computing power spectra is to use supervised machine learning techniques on simulated image cubes of the EoR by using two-dimensional convolution neural networks to perform regression on astrophysical and cosmological parameter values, and ultimately predict on new images not previously seen by the network and predict the desired reionization values. This image processing approach allows for the extraction of non-Gaussian information present in the maps. In this section I discuss our approach to extracting $\tau$ using convolution neural networks (CNNs). Machine learning techniques have been exploited in a variety of fields to explore different scientific questions. (See the review of Gu et al. (2015) and references therein for examples.)

For example, the authors of Hortúa et al. (2020) use Bayesian Neural Networks (BNNs) to predict the posterior distribution of the cosmological parameters directly from the CMB temperature and polarization maps.In the context of 21 cm data, Gillet et al. (2019) used CNNs to extract semi-analytic model parameters related to astrophysics from 21CMFAST simulations. La Plante and Ntampaka (2019) applied CNNs to simulated images of the EoR, and were able to successfully infer the duration of reionization to a high degree of accuracy. There have also been several recent studies where cosmological or astrophysical parameters are inferred by applying machine learning techniques to simulated 21 cm data Kwon et al. (2020); Villanueva-Domingo and Villaescusa-Navarro (2021). In this chapter, I explain how I build upon the approach of La Plante and Ntampaka (2019), and vary the reionization history to include changes in the midpoint and duration of reionization. I also predict directly on $\tau$, rather than inferring the reionization meta-parameters. This approach allows us to compare more directly with the uncertainty on $\tau$ related to other methods.

## 6.2. ML Modeling vs Statistical Modeling

### 6.2.1. Statistics to Machine Learning

"*Machine Learning is a branch of study in which a model can learn automatically from the experiences based on data without exclusively being modeled like in statistical models...*"

"*Statistics is the branch of mathematics dealing with the collection, analysis, interpretation, presentation, and organization of numerical data...*"

Statistics are mainly classified into two subbranches. Descriptive statistics, these are used to summarize data, such as the mean, standard deviation for continuous data types or frequency and percentage are useful for categorical data. Inferential statistics, which is often a subset of the data points called a sample, and conclusions about the entire population will be drawn, which is known as inferential statistics. Inferences are drawn using hypothesis testing, the estimation of numerical characteristics, the correlation of relationships within data, and so on.

Machine learning is broadly classified into three categories but nonetheless, based on the situation, these categories can be combined to achieve the desired results for particular applications:

- **Supervised learning**: Machines learn the relationship between predictor variables and target variables. This is commonly applied to classification problems (categorical selection) or regression problems (numerical selection).
- **Unsupervised learning**: Algorithms learn by themselves without any supervision or without any target variable provided. This is commonly applied to problems where dimensionality reduction is needed or clustering of categories of data.
- **Reinforcement learning**: Machine or agent learn its behavior based on feedback from the environment. Agent takes a series of decisive actions without supervision and, in the end, a reward will be given, either +1 or -1. Based on the final payoff/reward, the agent reevaluates its paths. Reinforcement learning problems are closer to the artificial intelligence methodology rather than frequently used machine learning algorithms.

In the case of my project, I used supervised learning techniques to solve a regression problem. In other words, in order for my machine learning algorithm to extract the numerical value, optical depth to Reionization, it must be trained using predictor variables and target variables.

In machine learning, a small sample size of 30 observations would be enough to update the weights at the end of each iteration. Machine learning models can be effectively parallelized. Effectively parallelized algorithm is an algorithm that can execute several instructions simultaneously on different processing devices and then combine all the individual outputs to produce the final result. These algorithms are made to work on multiple machines in which model weights are broadcast across the machines.

In statistical modeling, samples are drawn from the population and the model will be fitted

| Statistical Modeling | Machine Learning Modeling |
|---|---|
| Formalization relationships between variables in the form of mathematical equations and rules. | Algorithm that can learn from the data without relying on rule-based programming. |
| Assumes the shape of the model curve prior to perform model fitting on the data. | Does not assume underlying shape, as machine learning algorithms learns complex patterns automatically based on the provided data. |
| Statistical model predicts the output with accuracy of a particular percent and having a particular percent confidence about it. | Machine learning just predicts the output with accuracy of a particular percent. |
| Various diagnostics of parameters are performed, like p-value, power, confidence interval, significance level. | Does not perform any statistical diagnostic significance tests. |
| Data will be split into training (approximately 70 percent) and testing (approximately 30 percent) data. Model developed on training data and tested on testing data. | Data will be split into training (approximately 60 percent), validation (approximately 20 percent), and testing (approximately 20 percent) data. Models developed on training and hyperparameters are tuned on validation data and finally get evaluated against test data. |
| Models can be developed on a single dataset called training data, as diagnostics are performed at both overall accuracy and individual variable level. | Due to lack of diagnostics on variables, machine learning algorithms need to be trained on two datasets, called training and validation data, to ensure two-point validation. |
| Statistical modeling is mostly used for research purposes. | Machine learning is very apt for implementation in a production environment. |
| From the school of statistics and mathematics. | From the school of computer science. |

Table 6.1: This table highlights the differences in methodologies.

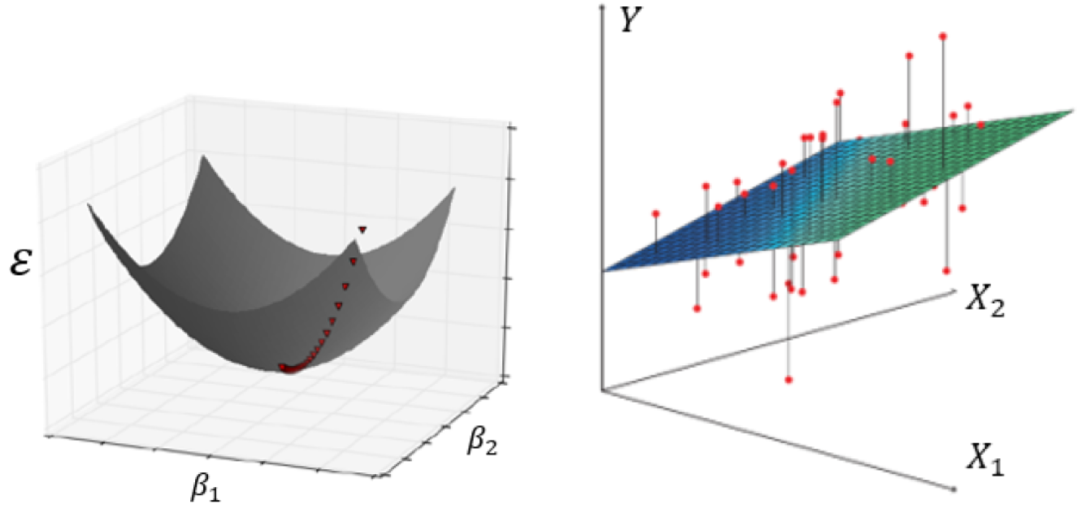


Figure 6.1: **Left**: Minimization of the Error function $\epsilon$ as a function of the weights, $(\beta_1, \beta_2)$. **Right**: Fitting a linear plane to the sample data (red).

on sampled data. These models are parametric in nature, which means a model will have parameters on which diagnostics are performed to check the validity of the model.

However, in machine learning models are non-parametric and do not have any parameters or model curve assumptions. The models are design to learn by themselves based on the data provided and come up with complex and intricate functions rather than predefined function fitting like statistical models. Multicollinearity, more than two predictor variables in a multiple regression model are highly linearly related, checks are required to be performed in statistical modeling. Whereas in machine learning weights automatically get adjusted to compensate the multicollinearity problem. Weights get adjusted with the goal of minimizing the error, $\epsilon$, while optimizing while simultaneously optimizing the weights, $(\beta_1, \beta_2)$.

$$\hat{Y} = \beta_1 X_1 + \beta_2 X_2 \tag{6.1}$$

$$\epsilon = (Y - \hat{Y})^2 = (Y - \beta_1 X_1 - \beta_2 X_2)^2 \tag{6.2}$$

Consider the following simplified example, where the data is described by two predictor variables $(X_1, X_2)$ or independent variables to get a is the model predicted target value, $\hat{Y}$. The statistical model can be represented by Equation 6.1 and the machine learning model attempts to minimize the error in Equation 6.2. In the Figure 6.1 the left plot shows that although the weights can take on multiple values, there is a specific set that will minimize the error term. As you increase the number of weights one can imagine that the terrain would be more complicated than what is expressed here. The right plot show how statistical modeling attempts to fit a linear plane to sample data.

## 6.3. Bias-Variance Tradeoff Derivation

This section will detail the mathematical explanation of the the Bias - Variance Tradeoff and is based on both the video lecture and lecture notes on Bias - Variance Tradeoff. This concept was important when tuning the model in my paper, Billings et al. (2021), to get more favorable predictions. Every model has bias, variance, and noise error components. Together they represent the generalization error. Bias and variance are inversely related to each other. This means that attempts to reduce one component will increase the other. The ideal model will have a balance of both low bias and low variance. In general, errors from the bias component come from assumptions in the learning algorithm. The algorithm essentially misses the relevant features and relations between the feature vector and target outputs. This causes the model to underfit. Errors from the variance component come from sensitivity to changes in training data resulting in large change to the fit of the model and can cause overfitting.

The goal of this section is to decompose the generalization error into three meaningfully terms of a machine learning classifier. First start with some data points $(\mathbf{x}_n, y_n)$ where $\mathbf{x}_n$ are $n$ different feature vectors and $y_n$ are n different labels. These n identically independent distributed (i.i.d) data points are randomly selected from some unknown probability distribution, $P(X, Y)$. Also, for a given feature vector $\mathbf{x}$ there may not exist a unique label, $y$. In other words, for a given feature vector $\mathbf{x}$ there exists a distribution of labels that can be determined by calculating the expectation value. Once a set of points have been randomly selected those n data points, $(\mathbf{x}_n, y_n)$, form a training data D $= \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_n, y_n)\}$. The data D are also i.i.d. and are drawn from some distribution P(X,Y). The data pair $(\mathbf{x}_n, y_n)$ and D are independent from each other.

As mentioned in the paragraph above, for a give feature vector $\mathbf{x}$ the expected label can calculated. The conditional expectation value of the label given some feature vector is,

$$\bar{y}(\mathbf{x}) = E_{y|\mathbf{x}}\left[Y\right] = \int_{y} y \ \mathrm{Pr}(y|\mathbf{x})\partial y. \tag{6.3}$$

where $\bar{y}(\mathbf{x})$ is a function that produces a single mean label value provided you know that feature vector, $\mathbf{x}$. This is the best you can do. $\mathrm{Pr}(y|\mathbf{x})$ is the conditional probability of a label provided you know the feature vector. $y$ is the random variable label that takes on some value. Together, $y\,\mathrm{Pr}(y|\mathbf{x})$, you get the weighted average of a $y$ taking on some value. Then, integrate out all possible label values to get the mean label as a function of the feature vector, $\bar{y}(\mathbf{x})$. Later Equation 6.3 will be used in the decomposition of the generalized error.

The training set, $D$, that was drawn from the probability distribution, $P(X, Y)$, is used to train some machine learning algorithm, $\mathcal{A}$. The algorithm, $\mathcal{A}$, that is trained on some training data, $D$, is called the hypothesis $h_D$ or classifier and is defined as,

$$h_D = \mathcal{A}(D) \tag{6.4}$$

$D$ is a random variable so therefore $h_D$ is a random variable and I can calculate the expected value of some algorithm $\mathcal{A}$ trained on the data $D$ is equivalent to the expected value of the hypothesis and can be expressed as,

$$\bar{h} = E_{D\sim P^n}\left[h_D\right] = \int_{D} h_D \,\mathrm{Pr}(D)\partial D \tag{6.5}$$

where $\bar{h}$ is the expected hypothesis trained on a data $D$ where $D$ is a random variable that was sampled from $P^n$. $\mathrm{Pr}(D)$ is the probability of selecting that particular training set $D$.

$h_D$ is the classifier that was trained. Together I get the weighted average of the hypothesis integrated over all possible data $D$. This means that $\bar{h}$ is a constant and is completely independent of the data because all possible data have been integrated out.

This hypothesis can be used to develop an analytical representation of the generalization error of this equation by first selecting a loss function. I chose the residual square loss function because it has a more simple and clean expression to demonstrate. Given $h_D$, the expected test error as a function of the classifier is,

$$E_{(\mathbf{x},y)\sim P}\left[(h_D(\mathbf{x})-y)^2\right] = \int\limits_x\int\limits_y (h_D(\mathbf{x})-y)^2\,\mathrm{Pr}(\mathbf{x},y)\partial y\partial\mathbf{x} \tag{6.6}$$

For some test data points, $(\mathbf{x},y)$, that are randomly selected from a probability distribution P, the expected value of the loss function is presented above. The loss function is expressed as the difference between the hypothesis, $h_D$, trained data D as a function of the feature vector, $\mathbf{x}$, that was randomly selected from the probability distribution from the label, y, that was randomly selected from the probability distribution. $\mathrm{Pr}(\mathbf{x},y)$ is the probability of drawing the data pair $(\mathbf{x},y)$ which are the test points. $(h_D(\mathbf{x})-y)^2$ is the predicted label for some classifier or algorithm trained on training data D minus the actual label that exists within the distribution P, squared. This is also known as the square of the residual. Together I get the weighted average of residual squared. Integrate over all possible test feature vectors and test label pairs to get the expected test error.

Finally, the expected test error can be written as a function of $\mathcal{A}(D)$ instead of $h_D$ as,

$$\underset{D\sim P^n}{E_{(\mathbf{x},y)\sim P}}\left[(h_D(\mathbf{x})-y)^2\right] = \int_D\int_{\mathbf{x}}\int_y (h_D(\mathbf{x})-y)^2\,\mathrm{P}(\mathbf{x},y)\mathrm{P}(D)\partial\mathbf{x}\partial y\partial D \tag{6.7}$$

This is the final expression for the generalization error, the expected value of the test error gives the algorithm, $\mathcal{A}(D)$, where D are the training data points and the $(\mathbf{x},y)$ pairs are the test data points. Again, given that $(\mathbf{x},y)$ was randomly sampled from the probability distribution P(X,Y) and the training data D was sampled from $P^n$, now the expected value of the test error of some loss function of our choosing can be calculated. As a reminder, I chose the square of the residual because it will give a simple and clean expression of each error term. The generalization error is the product of the residual squared with the probability of selecting the test data $(\mathbf{x}, y)$ with the probability of selecting that training data D, integrated over the the entire test and training data. Expand the left side of the equation in Equation 6.7. Subtract and add $\bar{h}(\mathbf{x})$ and expand the square.

$$\begin{aligned} E_{\mathbf{x},y,D}\left[[h_D(\mathbf{x}) - y]^2\right] &= E_{\mathbf{x},y,D}\left[\left[\left(h_D(\mathbf{x}) - \bar{h}(\mathbf{x})\right) + \left(\bar{h}(\mathbf{x}) - y\right)\right]^2\right] \\ &= E_{\mathbf{x},D}\left[(h_D(\mathbf{x}) - \bar{h}(\mathbf{x}))^2\right] \\ &+ 2\ E_{\mathbf{x},y,D}\left[\left(h_D(\mathbf{x}) - \bar{h}(\mathbf{x})\right)\left(\bar{h}(\mathbf{x}) - y\right)\right] \\ &+ E_{\mathbf{x},y}\left[\left(\bar{h}(\mathbf{x}) - y\right)^2\right] \end{aligned} \tag{6.8}$$

The first term in Equation 6.8 is the variance. This is because the square of the difference between the classifier, $h_D(\mathbf{x})$, trained on data, D, as a function of the feature vector, $\mathbf{x}$, from the mean of the classifier $\bar{h}(\mathbf{x})$ is the definition of the variance.

The middle term is zero because $E_{\mathbf{x},y,D}$ can be expressed as the

$$E_{\mathbf{x},y}\left[E_D[h_D(\mathbf{x}) - \bar{h}(\mathbf{x})]\left(\bar{h}(\mathbf{x}) - y\right)\right]$$

such that the following is true,

$$E_{\mathbf{x},y,D}\left[\left(h_D(\mathbf{x}) - \bar{h}(\mathbf{x})\right)\left(\bar{h}(\mathbf{x}) - y\right)\right] = E_{\mathbf{x},y}\left[E_D\left[h_D(\mathbf{x}) - \bar{h}(\mathbf{x})\right]\left(\bar{h}(\mathbf{x}) - y\right)\right] \tag{6.9}$$

Lets focus on the first product $E_D[h_D(\mathbf{x}) - \bar{h}(\mathbf{x})]$. Since the expectation value is a linear operator, applying the linear distribution property to get $E_D[h_D(\mathbf{x})] - E_D[\bar{h}(\mathbf{x})]$ is simple. Recall, $\bar{h}(\mathbf{x})$ is the expected hypothesis/classifier as a function of feature vector $\mathbf{x}$ so the expected value of an expected value is simply that value, a constant, $\bar{h}$. This is true because $\bar{h}$ is not a function of the training data, D, then $\bar{h}$ is completely independent of the training data. So you cannot take the expectation value of some function that is trained on data D if it is not dependent on D. However, $h_D$ is the classifier that was trained on data D. The expected value of this is exactly $\bar{h}$. Therefore,

$$\begin{aligned} E_D[h_D(\mathbf{x})] - E_D[\bar{h}(\mathbf{x})] &= \bar{h}(\mathbf{x}) - \bar{h}(\mathbf{x}) \\ &= 0 \end{aligned}$$

Concluding that,

$$E_{\mathbf{x},y}\left[E_D\left[h_D(\mathbf{x}) - \bar{h}(\mathbf{x})\right]\left(\bar{h}(\mathbf{x}) - y\right)\right] = 0$$

Rewrite the original expression as,

$$E_{\mathbf{x},y,D}\left[(h_D(\mathbf{x}) - y)^2\right] = \underbrace{E_{\mathbf{x},D}\left[\left(h_D(\mathbf{x}) - \bar{h}(\mathbf{x})\right)^2\right]}_{\text{Variance}} + E_{\mathbf{x},y}\left[\left(\bar{h}(\mathbf{x}) - y\right)^2\right]$$

The second term can be expressed using the same tick as before where by subtracting and adding $\bar{y}(\mathbf{x})$ and expand the square.

$$\begin{aligned} E_{\mathbf{x},y}\left[\left(\bar{h}(\mathbf{x}) - y\right)^2\right] &= E_{\mathbf{x},y}\left[\left(\bar{h}(\mathbf{x}) - \bar{y}(\mathbf{x})\right) + (\bar{y}(\mathbf{x}) - y)^2\right] \\ &= E_{\mathbf{x},y}\left[(\bar{y}(\mathbf{x}) - y)^2\right] \\ &+ E_{\mathbf{x}}\left[\left(\bar{h}(\mathbf{x}) - \bar{y}(\mathbf{x})\right)^2\right] \\ &+ 2\ E_{\mathbf{x},y}\left[\left(\bar{h}(\mathbf{x}) - \bar{y}(\mathbf{x})\right)(\bar{y}(\mathbf{x}) - y)\right] \end{aligned} \tag{6.10}$$

The first term is the noise because of the difference between the average label as a function of the feature vector and the label. The second term is the square of the bias because given the model average $\bar{h}(\mathbf{x})$, it is independent of the training data D, when you subtract the mean label as a function of the feature vector, $\bar{y}(\mathbf{x})$, it tells you information about the model being bias towards something not in the training data. The more data you introduce will not convince it otherwise. Note, the bias error term is specifically squared because the predictions can be biased either above or below the true value.

The third term is zero using the same argument as before in Equation 6.9. It is clear that, $E_{\mathbf{x},y}\left[\left(\bar{h}(\mathbf{x})-\bar{y}(\mathbf{x})\right)\left(\bar{y}(\mathbf{x})-y\right)\right]$. Given that a function dependent on the feature vector $\mathbf{x}$ and the label y, the expectation value is equivalent to given a function dependent on the feature vector $\mathbf{x}$, the expectation value of the function for a label y provided you know the feature vector $\mathbf{x}$.

$$E_{\mathbf{x},y}\left[\left(\bar{h}(\mathbf{x})-\bar{y}(\mathbf{x})\right)\left(\bar{y}(\mathbf{x})-y\right)\right]=E_{\mathbf{x}}\left[E_{y|\mathbf{x}}\left[\bar{y}(\mathbf{x})-y\right]\left(\bar{h}(\mathbf{x})-\bar{y}(\mathbf{x})\right)\right]$$

The first product, $E_{y|\mathbf{x}}\left[\bar{y}(\mathbf{x})-y\right]$, can be simplified by using the linear distributive property of the expectation value. $\bar{y}(\mathbf{x})$ is the mean label as a function of the feature vector. It is not a function of y so the $E_{y|\mathbf{x}}[\bar{y}(\mathbf{x})]$ is $\bar{y}(\mathbf{x})$. The expected value of the label, y is $\bar{y}(\mathbf{x})$ and as a result,

$$\begin{aligned}E_{y|\mathbf{x}}[\bar{y}(\mathbf{x})]-E_{y|\mathbf{x}}[y]&=\bar{y}(\mathbf{x})-\bar{y}(\mathbf{x})\\&=0\end{aligned}$$

The final expression for the generalization error is

$$\begin{aligned}\underbrace{E_{\mathbf{x},y,D}\left[\left(h_D(\mathbf{x})-y\right)^2\right]}_{\text{Generalization Error}}&=\underbrace{E_{\mathbf{x}}\left[\left(\bar{h}(\mathbf{x})-\bar{y}(\mathbf{x})\right)^2\right]}_{\text{Bias}^2}\\&\quad+\underbrace{E_{\mathbf{x},D}\left[\left(h_D(\mathbf{x})-\bar{h}(\mathbf{x})\right)^2\right]}_{\text{Variance}}\\&\quad+\underbrace{E_{\mathbf{x},y}\left[\left(\bar{y}(\mathbf{x})-y\right)^2\right]}_{\text{Noise}}\end{aligned}\tag{6.11}$$

The total error is the sum of the square of the bias, the variance, and the noise. Bias is inherent to the model. If the classifier were trained on infinite training data, the classifier would still be biased towards a particular solution or features not in the data. Variance describes the change in the classifier if you change the training data. Instead of looking at a single large accurate classifier, the average variance of a model from many smaller data tells you how certain you are about the model. The classifier not generalized but instead it is overly specialized. Finally, noise measures ambiguity due to the data distribution and feature representation.

## 6.4. Epistemic and Aleatoric Uncertainty

The goal of uncertainty estimation in machine learning is to assign a level of confidence to a model's output. By definition, the true uncertainty correlates with a model's performance. While a correct prediction can have high uncertainty, model performance degrades on average where uncertainty increases. Uncertainty quantification is an important challenge for applications of deep learning within systems. Uncertainty that can vary from sample to sample within a data domain is called heteroscedastic and are input data-dependent uncertainty . Though it is frequently neglected, heteroscedasticity uncertainty in deep learning can be modeled from two sources: epistemic uncertainty and aleatoric uncertainty. Epistemic uncertainty describes uncertainty in the model parameters and has been addressed by [Tellini (2015), Pearce et al. (2018), Pawlowski et al. (2017)].

Aleatoric uncertainty is the uncertainty arising from the natural stochasticity of observations. Noisy measurements of the underlying process lead to high aleatoric uncertainty in the data or by additional measurements. Aleatoric uncertainty is irreducible with additional training or even when more data is provided. When it comes to measurement errors, it is called the homoscedastic uncertainty because it is constant for all samples. Finally, the uncertainties of predictions of multiple values are often correlated, so it is important to account for the full multivariate uncertainty.

Epistemic uncertainty or model uncertainty, represents uncertainty in the neural network model parameters. Like aleatoric uncertainty, epistemic uncertainty can vary from measurement to measurement and is a concern for neural networks given their many free parameters, and can be very large for data that is significantly different from the training set. Epistemic uncertainty is due to limited data and knowledge. In other words, high epistemic uncertainty arises in regions where there are few or no observations for training. This is because too many plausible model parameters can be suggested for explaining the underlying ground truth phenomenon. The solution to this is given enough training samples, epistemic uncertainty will decrease. Epistemic uncertainty can arise in areas where there are fewer samples for training.

Numerous approaches for Bayesian inference have been developed that allow for the estimation of this uncertainty. The easiest and most practical approach is to use the dropout Monte Carlo method. The benefit of this is it trades off accuracy for speed and convenience. If epistemic uncertainty can be estimated from N samples, the total predictive covariance estimate should be calculated by

$$\Sigma_{\text{pred}} = \Sigma_{\text{epistemic}} + \Sigma_{\text{aleatoric}} \tag{6.12}$$

a full derivation of the covariance of the variational predictive distribution and the variance of the variational predictive distribution is presented in Hortúa et al. (2020). This is important because neural network can be used to learn the correlations between the the targets and produce estimates of their uncertainties.

This formulation can be used at test time as long as epistemic uncertainty for the training data is small relative to the aleatoric uncertainty. However, if epistemic uncertainty is significantly large for data in the training set after training, the total predictive covariance estimate predicted directly from the neural network will incorporate this uncertainty into the prediction, making it challenging to separate. It has been shown Russell and Reale (2019) that it is possible to calculate the epistemic uncertainty covariances for the training data set and tune the model covariance prediction to predict the residual. However, this tuning proves to be less stable and best results are generally achieved by training until the epistemic uncertainty had become small for the training set.

# CHAPTER 7

# NETWORK ARCHITECTURE

## 7.1. What is a Convolution Filter

The bases of convolution neural networks is a vector/matrix of some shape is received as input and is multiplied with a matrix to produce an output to which a bias vector is usually added before passing the result through a nonlinearity layer. This is applicable to inputs of any form such as sound clips or images. Data stored in the form of multi-dimensional arrays, or feature one or more axes for which ordering matters (e.g., width and height axes for an image, time axis for a sound clip) or if one axis is called the channel axis, is used to access different views of the data (e.g., the red, green and blue channels of a color image, or the left). Regardless of the input dimensionality, their representation can always be flattened into a vector before it gets transformation by the convolution filter.

These properties are not exploited when the transformation is applied. All the axes are treated in the same way and the topological information is not taken into account while still taking advantage of the implicit structure of the data and in many cases preserving it. This is where discrete convolutions come into play. A discrete convolution is a linear transformation that preserves this notion of ordering. It is sparse (only a few input units contribute to a given output unit) and reuses parameters (the same weights are applied to multiple locations in the input). Figure 7.1 provides an example of a simple discrete convolution. For simplicity a single $5 \times 5$ input matrix or feature map is drawn. It is not uncommon to have multiple feature maps stacked one onto another. In fact, for the work presented in this thesis the input feature maps were 30 maps stacked onto each other representing the ordered redshift evolution of the data. Taking a $3 \times 3$ kernel of some unknown values that are not predetermined, slide the kernel across the input feature map. At each location, the product between each element of the kernel and the input element it overlaps is computed and the results are summed up to obtain the output in the current

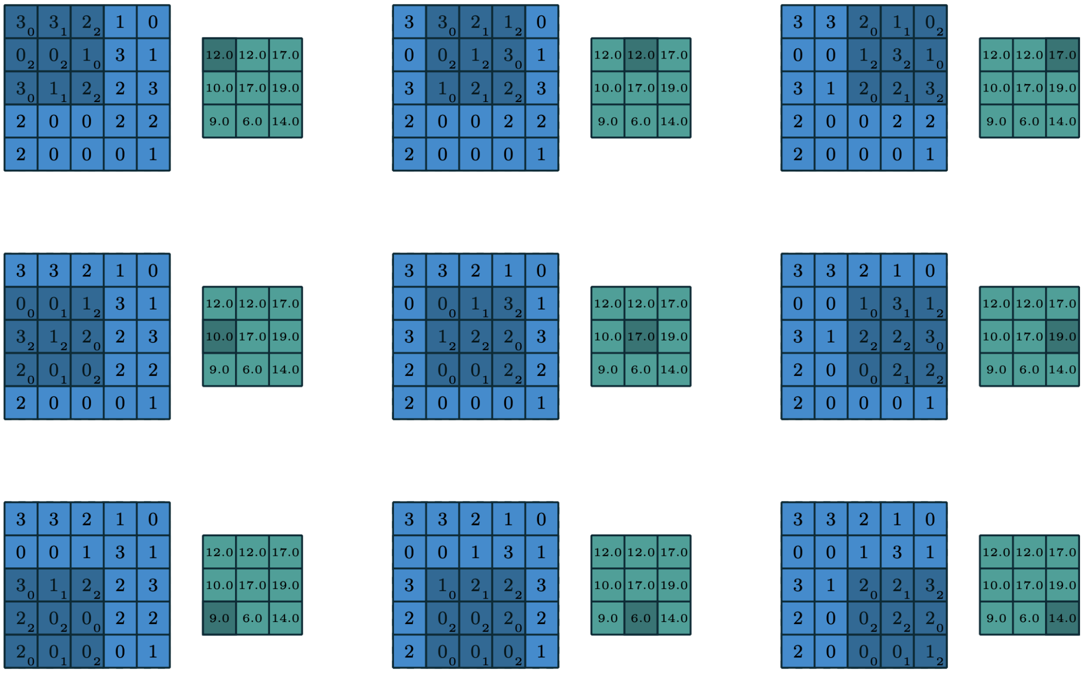


Figure 7.1: Computing the output values of a discrete convolution. This cartoon was taken from Dumoulin and Visin (2016)

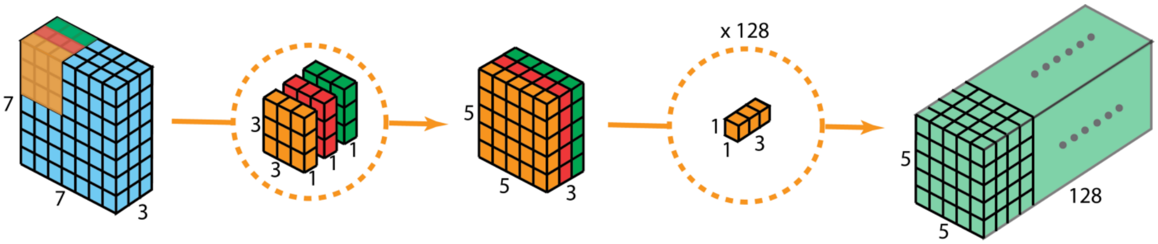


Figure 7.2: The author in Dumoulin and Visin (2016) showed depthwise separable 2D convolution which first processes each channel separately and then applies inter-channel convolutions

location. The procedure can be repeated using different kernels to form as many output feature maps as desired. The final outputs of this procedure are called output feature maps. If there are multiple input feature maps (having a 3rd ordered axis like redshift), the kernel will have to be 3-dimensional or, equivalently each one of the feature maps will be convolved with a distinct kernel and the resulting feature maps will be summed up element wise to produce the output feature map.

The following properties affect the output feature map size, $o_j$, of a convolution layer along axis j:

- $i_j$ : input size along axis j.
- $k_j$ : kernel size along axis j.
- $s_j$ : stride (distance between two consecutive positions of the kernel) along axis j.
- $p_j$ : zero padding (number of zeros concatenated at the beginning and at the end of an axis) along axis j.

## 7.2. How Convolution Filters Work

The remarkable thing about convolution neural networks (CNN) is its ability to automatically learn a large number of convolution filters components in parallel specific to an input training data under the constraints of a specific predictive modeling problem. The results are highly specific features that can be detected anywhere on input images. CNNs are specialized types of neural network model designed for working with one, two, or three-dimensional image or sound data. Central to the convolutional neural network is the layer performs an operation called a “convolution”, which that gives the network its name.

In the context of a convolutional neural network, a convolution is a linear operation that involves the multiplication of a set of weights with an array of input. This two-dimensional array of weights is formally called a filter or a kernel. The filter is typically smaller than the input data and the type of multiplication applied between a filter-sized patch of the input data and the filter is a dot product. Recall that the dot product is the element-wise multiplication between the filter-sized patch of the input and filter, which is then summed, always resulting in a single value often referred to as the “scalar product”.

Using a filter that is significantly smaller than the input data array is intentional as it allows the same filter to be multiplied by the input array multiple times at different points on the input. Specifically, the filter is applied systematically to each overlapping part or filter-sized patch of the input data, left to right, top to bottom. This is why the convolution filter is so powerful. The application of a small filter systematically allows for the extraction of localized features within an input data array. Then the application of that filter systematically across the entire input image allows the filter an opportunity to discover that feature anywhere in the image. This is commonly referred to as translation invariance, the general interest in whether the feature is present rather than where it was present.

A filter must always have the same number of channels, often referred to as “depth”, as the input data. If an input image has 3 channels (depth of 3), then a filter applied to that image must also have 3 channels (depth of 3). For example, a $3 \times 3$ filter would in fact be $3 \times 3 \times 3$ or [3, 3, 3] for rows, columns, and depth.

This means that if a convolutional layer has $f$ number of filters, these $f$ filters are not just two-dimensional for the two-dimensional image input, but are also three-dimensional, having specific filter weights for each of the three channels. The output from multiplying the filter with the input array a single time results in a single value. However, as the filter is applied multiple times to the input array, the result is a two-dimensional array of output values that represent a filtering of the input called a “feature map”. Which means that the depth of the output of applying the convolutional layer with $f$ filters is $f$ for the $f$ feature maps created. Once the feature map is created it can pass each value in the feature map through a nonlinearity activation function.

## 7.3. How to train a CNN

### 7.3.1. Activation Function

A neural network without an activation function is essentially a linear regression model. In other words, every neuron will only be performing a linear transformation on the inputs using the weights and biases. Although linear transformations make the neural network simpler and using a non-linearity activation function introduces an additional step at each layer during the forward propagation, this network would be less powerful and will not be able to learn the complex patterns from the data. Therefore, it is important to apply a non-linear transformation to the inputs of the neuron and this non-linearity in the network is introduced by an activation function.

The input data is fed to the input layer and then the neurons perform a linear transformation on these inputs using the weights and biases. Then after that the activation function introduces some non-linearity to the network. Figure 7.3 details a few common activation functions. Starting from top to bottom, left to right: binary step, rectified linear unit, softmax, leaky rectified linear unit, sigmoid, parameterized rectified linear unit, and tanh. In my research I used the rectified linear unit activation. Below is a simple linear model,

$$\text{output} = \sum_{i=1}^{N}(\text{weights} \times \text{input}) + \text{bias} \tag{7.1}$$

Where N is the number of neurons connected to a particular node, weights is a trainable parameters updated during the backprapogation process. Weights are used to connect each

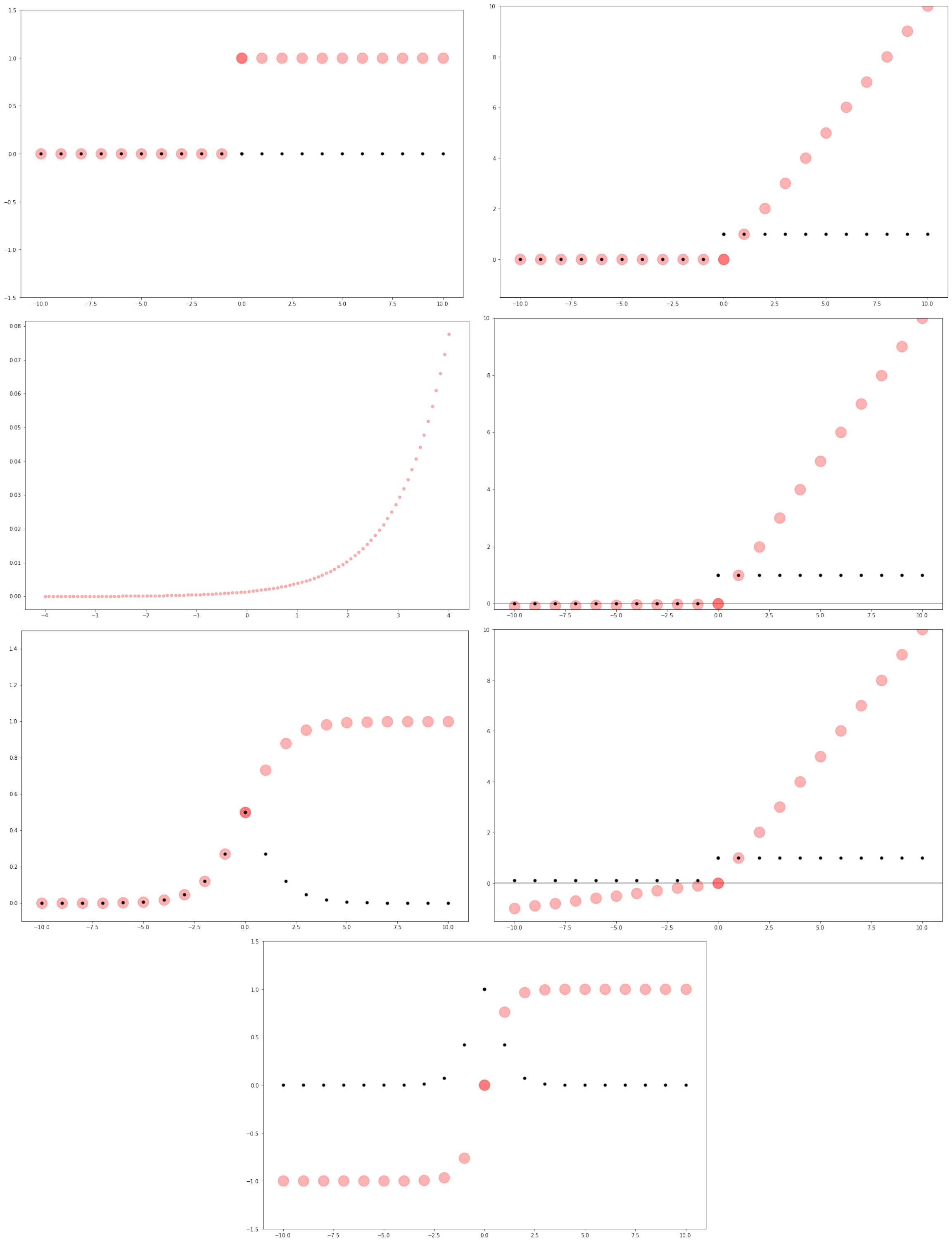


Figure 7.3: The red large circles represent the activation function. The gradient of the activation function is represented by the black small dots.

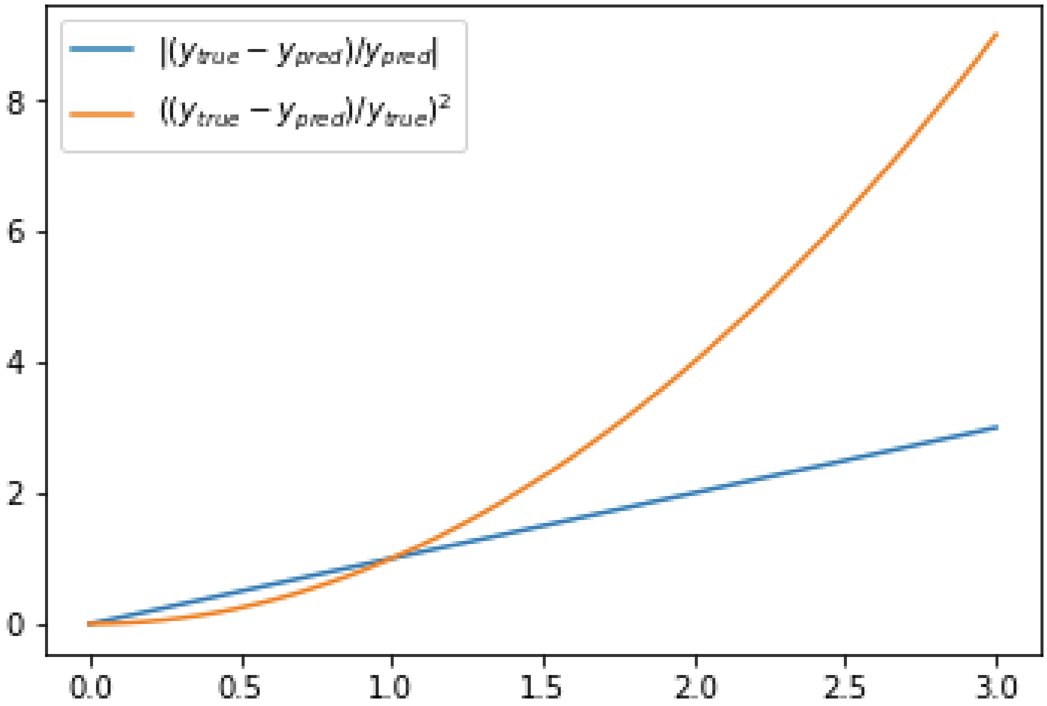


Figure 7.4: The absolute fractional error penalizes the network more than the square fractional error when errors are less than 1, but when for errors larger than 1, the square fractional error penalizes the model far more.

neurons in one layer to the every neurons in the next layer. They also determine the strength of the connection of the neurons. Weights near zero mean changing this input will not change the output. The bias node is another trainable parameters updated during the backpropagation process. The bias means how far off my predictions are from real values. A low Bias suggests more assumptions about the form of the target function, whereas high bias suggests less assumptions about the form of the target function.

### 7.3.2. Loss Function

A deep learning neural network learns to map a set of inputs to a set of outputs from training data and the choice of loss function must match the framing of the specific predictive modeling problem, such as classification or regression. In addition, the configuration of the output layer must also be appropriate for the chosen loss function. As part of the optimization algorithm, the error for the current state of the model must be estimated repeatedly. This requires the choice of an error function (general form of the loss function, Equation 6.2) that can be used to estimate the loss of the model so that the weights can be updated to reduce the loss on the next evaluation.

It is impossible to calculate the perfect weights for a neural network because there are too many unknowns. Instead, the problem of learning is instead a search or optimization problem and an algorithm is used to navigate the parameter space of all possible sets of weights the model may use in order to make good enough predictions. A neural network model is commonly trained using the stochastic gradient descent optimization algorithm or some combination of its extension and weights are updated using the backpropagation of error algorithm. The optimization algorithm seeks to change the weights so that the next evaluation reduces the error function, or in other words, the optimization algorithm is navigating down the gradient (or slope) of error function in a complex parameter space of all possible sets of weights.

The goal is to maximize or minimize the error function; search for a candidate solution that has the highest or lowest score respectively. For neural networks the goal is to minimize the error (minimize the loss function). The loss function reduces all the various good and bad aspects of a highly complex system down to a or a couple of scalar values, which allows candidate solutions to be ranked and compared. Selecting the proper error function of the model can be a very challenging problem as the function must capture the properties of the problem.

For my project I initially used the “mean squared error" function (MSE) but later realized that for my particular input data I needed a loss function that would severely penalizes the model the more wrong it was. Instead of using the absolute size of the error, $MSE = (y_{\text{true}} - y_{\text{pred}})^2$, it is more beneficial to use the fractional error instead. For example, I know the optical depth to reionization is a small value and the error could come out to be $0.01 \pm 0.01$ (100% error) and count equally in the loss function with and error $0.1 \pm 0.01$ (10% error). However, if the loss function were fractional, then every value counts in the

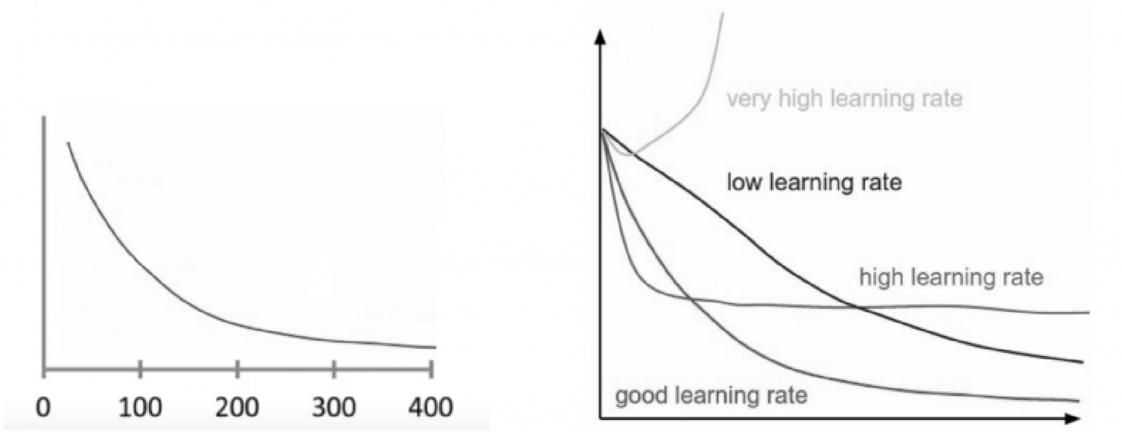


Figure 7.5: Models with different learning rates.

loss function as its fractional error, so you try to enforce that small values are correct to the same percent error as large values.

The fractional MSE, $MSE_{\text{frac}}$, can be written as,

$$MSE_{\text{frac}} = \left( \frac{y_{\text{pred}} - y_{\text{true}}}{y_{\text{true}}} \right)^2 \tag{7.2}$$

### 7.3.3. Backpropagation

**Learning Rate**

As previously stated, deep learning models are trained using some kind of optimization algorithm. The learning rate is a hyperparameter, a parameter that is not learned but preset prior to the training process, that controls how much to change the model in response to the estimated error each time the model weights are updated. The goal it to minimize the loss by optimizing the weights by “moving” through the space of all possible parameters. This space is highly complex and could resemble rough/structured terrain with mountains and valleys. Choosing the learning rate is very challenging as a value that is too small may result in a long training process that could potentially not converge. In other words, if the step size taken towards the minimum loss are too small the model could get stuck in a local or false minimum. Whereas a learning rate that is too large may result in learning a sub-optimal set of weights too fast or an unstable training process. This is equivalent to taking too large of a step potentially jumping past the minimum loss.

The learning rate may be the most important hyperparameter when configuring your neural network. Therefore, it is good practice when optimizing your model to first adjust the learning rate to see the most pronounced changes in the performance of the model. It is vital to know how to investigate the effects of the learning rate on model performance and to build an intuition about the dynamics of the learning rate on model behavior. Examination of the loss as a function of epoch can prove to be a good indicator of how well or poorly the learning rate was selected.

### Gradient Descent

The evaluation of how close a fit a machine learning model estimates the target function can be calculated a number of different ways. To start, the model gets initialized with some random values for weight that are less than 1. After that the loss function is calculated again with a goal of reducing it by modify the parameters by using the Gradient descent algorithm over the given data. The loss function involves evaluating the coefficients in the machine learning model by calculating a prediction for the model for each training instance in the dataset and comparing the predictions to the actual output values and calculating a sum or average error. The loss is calculated for a machine learning algorithm over the entire training dataset for each coefficient and for each iteration of the gradient descent algorithm. One iteration of the algorithm is called one batch and this form of gradient descent is referred to as batch gradient descent. Batch gradient descent is the most common form of gradient descent described in machine learning.

To understand the mathematical representation for run the gradient descent algorithm on some loss function I will go through a simple example by rewriting the loss function has two parameters to update, $\omega$ (weight) and $b$ (bias), consideration of the impact each parameter has on the final prediction can be retrieved by using partial derivatives and the chain rule.

By calculating the partial derivatives of the loss function with respect to each parameter and store the results in a gradient, I can learn how each parameter impacts the final prediction.

Given a simple MSE loss function:

$$L(\omega, b) = \frac{1}{N}\sum_{i=1}^{n}(y_i - (\omega x_i + b))^2 \tag{7.3}$$

$y_i$ is a vector of true values and $x_i$ is the vector of input values. The gradient can be calculated as:

$$L'(\omega, b) = \begin{bmatrix} \frac{dL}{d\omega} \\ \frac{dL}{db} \end{bmatrix} = \begin{bmatrix} \frac{1}{N}\sum -2x_i(y_i - (\omega x_i + b)) \\ \frac{1}{N}\sum -2(y_i - (\omega x_i + b)) \end{bmatrix} \tag{7.4}$$

To solve for the gradient, I must iterate through the entire data set and compute the partial derivatives. This new gradient tells us the slope of our loss function at our current position (current parameter values) and the direction I should move to update our parameters. The size of this update is controlled by the learning rate.

An advantages of the gradient descent algorithm is that it produces a stable error gradient and a stable convergence. One major disadvantages is the algorithm is computationally expensive on large data because too many gradient descent updates are required and each gradient descent step is too expensive to calculate for each iteration. Computing the the first order gradient will require $O(n\omega)$ operations (n = sample size, $\omega$ = number of weights). In this case, methods that approximate the derivative based on smaller subsets of the data are more attractive, such as stochastic gradient descent.

**Stochastic Gradient Descent**

Although, using the whole dataset is really useful for getting to the minima in a less noisy and less random manner, but the problem arises when our datasets gets big. By contrast, stochastic gradient descent (SGD) updates the parameters for each training example one by one and is a much faster algorithm than batch gradient descent. The word ‘stochastic’ means a system or a process that is linked with a random probability. As a result, in SGD, a few samples are selected randomly instead of the whole data set like in gradient descent for each iteration.

The first step of the procedure requires that the order of the training dataset is randomized. This is to mix up the order that updates are made to the coefficients. Because the coefficients are updated after every training instance, the updates will be noisy, and so will the corresponding loss function. By mixing up the order for the updates to the coefficients, it harnesses this random walk and avoids it getting distracted or stuck. The update procedure for the coefficients is the same as that above, except the loss is not summed over all training patterns, but instead calculated for one training pattern.

The learning can be much faster with stochastic gradient descent for very large training data sets and often only need a small number of passes through the dataset to reach a good or good enough set of coefficients (1-10 passes through the dataset).

**Momentum**

SGD has trouble navigating areas where the surface curves much more steeply in one dimension than in another. In these scenarios, SGD oscillates across the slopes of the surface area while only making "hesitant" progress along the bottom towards the local minima. Momentum is a method that helps accelerate SGD in the relevant direction and simultaneously dampens the oscillations. It does this by adding a fraction $\gamma$ of the update vector of the past time step to the current update vector:

$$\begin{aligned} v_t &= \gamma v_{t-1} + \eta \nabla_\theta J(\theta) \\ \theta &= \theta - v_t \end{aligned} \tag{7.5}$$

Think of pushing a ball down a hill. The ball accumulates momentum as it rolls downhill, becoming faster and faster on the way until it reaches its terminal velocity if there is resistance, $\gamma < 1$. The same thing happens to the parameter updates: The momentum term increases for dimensions whose gradients point in the same directions and reduces updates for dimensions whose gradients change directions. As a result, the model gain faster convergence and reduced oscillation.

### Adaptive Gradient Algorithm (AdaGrad)

Adagrad is an optimization algorithm for gradient-based optimization that adapts the learning rate to the parameters, performing smaller updates or low learning rates, for parameters associated with frequently occurring features, and larger updates or high learning rates, for parameters associated with infrequent features. For this reason, it is well-suited for dealing with sparse data. Adagrad is an improvement to the robustness of SGD and is often used for training large-scale neural nets, train GloVe word embeddings, as infrequent words require much larger updates than frequent ones.

Updates performed for all parameters $\theta$ were done at once as every parameter $\theta_i$ used the same learning rate $\eta$. Adagrad uses a different learning rate for every parameter $\theta_i$ at every time step t. The per-parameter update are then vectorize. $g_t$ denotes the gradient at time step t. $g_{t,i}$ is then the partial derivative of the objective function w.r.t. to the parameter $\theta_i$ at time step t:

$$g_{t,i} = \nabla_\theta J(\theta_{t,i}) \tag{7.6}$$

In its update rule, Adagrad modifies the general learning rate $\eta$ at each time step t for every parameter $\theta_i$ based on the past gradients that have been computed for $\theta_i$:

$$\begin{aligned} \theta_{t+1,i} &= \theta_{t,i} - \eta \cdot g_{t,i} \\ \theta_{t+1,i} &= \theta_{t,i} - \frac{\eta}{\sqrt{G_{t,ii} + \epsilon}} \cdot g_{t,i} \end{aligned} \tag{7.7}$$

$G_t$ contains the sum of the squares of the past gradients w.r.t. to all parameters $\theta$ along its diagonal, it becomes is easy to vectorize the implementation by performing matrix-vector product $\odot$ between $G_t$ and $g_t$:

$$\theta_{t+1} = \theta_t - \frac{\eta}{\sqrt{G_t + \epsilon}} \odot g_t \tag{7.8}$$

### Root Mean Square Propagation (RMSProp)

AdaGrad decays the learning rate very aggressively (as the denominator grows). As a result, after a while, the frequent parameters will start receiving very small updates because of the decayed learning rate. To avoid this the algorithm can not decay the denominator and prevent its rapid growth. RMSprop was developed stemming from the need to resolve Adagrad's radically diminishing learning rates. Everything is very similar to AdaGrad, except now I decay the denominator as well. RMSprop's first update vector is described as:

$$\begin{aligned} E[g^2]_t &= 0.9E[g^2]_{t-1} + 0.1g_t^2 \\ \theta_{t+1} &= \theta_t - \frac{\eta}{\sqrt{E[g^2]_t + \epsilon}} g_t \end{aligned} \tag{7.9}$$

RMSprop as well divides the learning rate by an exponentially decaying average of squared gradients. $\gamma$ should be set to 0.9, while a good default value for the learning rate $\eta$ can be 0.001.

### Adaptive Moment Estimation (Adam)

Adam, the algorithm used throughout my research, is another method that computes adaptive learning rates for each parameter. Adam is a replacement optimization algorithm for stochastic gradient descent for training deep learning models. Adam combines the best properties of the AdaGrad and RMSProp algorithms to provide an optimization algorithm that can handle sparse gradients on noisy problems. In addition to storing an exponentially decaying average of past squared gradients $v_t$ RMSprop, Adam also keeps an exponentially decaying average of past gradients $m_t$, similar to momentum. Whereas momentum can be seen as a ball running down a slope, Adam behaves like a heavy ball with friction, which thus prefers flat minima in the error surface. We compute the decaying averages of past and past squared gradients $m_t$ and $v_t$ respectively as follows:

$$\begin{aligned} m_t &= \beta_1 m_{t-1} + (1 - \beta_1) g_t \\ v_t &= \beta_2 v_{t-1} + (1 - \beta_2) g_t^2 \end{aligned} \tag{7.10}$$

$m_t$ and $v_t$ are estimates of the first moment (the mean) and the second moment (the variance) of the gradients respectively, hence the name of the method. As $m_t$ and $v_t$ are initialized as vectors of 0's, the authors of Adam observe that they are biased towards zero, especially during the initial time steps, and especially when the decay rates are small, ($\beta_1 \approx 0.9, \beta_2 \approx 0.999$) are close to 1.

They counteract these biases by computing bias-corrected first and second moment estimates:

$$\begin{aligned} \hat{m}_t &= \frac{m_t}{1 - \beta_1^t} \\ \hat{v}_t &= \frac{v_t}{1 - \beta_2^t} \end{aligned} \tag{7.11}$$

The Adam update rule says:

$$\theta_{t+1} = \theta_t - \frac{\eta}{\sqrt{\hat{v}_t} + \epsilon} \hat{m}_t \tag{7.12}$$

### 7.3.4. Training/Validation

The training dataset is the sample of data used to fit the model. This is the actual dataset that is used to train the model and find the most optimal weights and biases. The model sees and learns from this data. The validation dataset is the sample of data used to provide an unbiased evaluation of a model fit on the training dataset while tuning model hyperparameters. The evaluation becomes more biased as skill on the validation dataset is incorporated into the model configuration. The validation set is used to evaluate a given model or use this data to fine-tune the model hyperparameters. Hence, the model occasionally sees this data, but it never does "learn" from this dataset. The validation set results are often used to update higher level hyperparameters. So the validation set used in this way affects a model, but only indirectly during the "development" stage of the model. Recall from section 5.2.1, due to lack of diagnostics on variables, machine learning algorithms need to be trained on two data sets, called training and validation data, to ensure two-point validation.

### 7.3.5. Batches

Batch is a hyperparameter that defines the number of samples needed to update the model. Think of a batch of training data as iterating over small random samples of data. The data is sampled without replacement to ensure fast convergence Recht and Re (2012). At the end of the batch, the predictions are compared to the expected output variables and an error is calculated. From this error, the update algorithm is used to improve the model, e.g. move down along the error gradient.

- Batch Gradient Descent Size = Size of Training Set
- Stochastic Gradient Descent Size = 1
- Mini-Batch Gradient Descent = 1 < Batch Size < Size of Training Set

### 7.3.6. Epoch

One epoch means that each sample in the training dataset has had an opportunity to update the internal model parameters. An epoch is comprised of one or more batches. Imagine a for-loop over the number of epochs where each loop proceeds over the training dataset. Within this for-loop is another nested for-loop that iterates over each batch of samples, where one batch has the specified batch size number of samples.

### 7.3.7. K-Fold Cross Validation

Cross-validation is another way of ensuring robustness in the model at the expense of computation. In order to train a CNN model, one popular technique is to split the data into training and testing. The ratio of the split can be 90% training data and 10% testing data (referred to as 90/10), or 80/20.

One potential issue with this method is that it is possible to introduce variance in output values because the volume of input data may not be large enough. Such a situation might prevent the network from being able to extract key features, as well as seeming to be biased. An alternative approach is using the $k$-fold cross-validation method. The $k$-fold technique is a popular alternative because it generally results in a less biased model because it allows the user to estimate how well a given model is truly doing before actually testing it. It also ensures that every observation from the original data set has a chance to appear in training and test set. In my model I take the approach that no part of the data is over sampled. My goal in using this method initially was to understand the variance of my model from sample to sample. First, I split the entire data randomly into 10 folds where 80% of the data is used for training and the other 20% of the data is used for testing. The general rule of thumb is that a higher value of $k$ leads to a less biased model, but if the value is too large it may lead to a large variance, potentially causing overfitting. In general, the best practice is to use somewhere between 5 and 10 $k$-folds. Although $k$-folding is a potentially effective method to decrease the bias and variance, it was ultimately decided to use hyperparameter tuning to decrease the bias and variance more robustly.

In this work, I use $k$-fold cross validation to help demonstrate that the performance of the trained CNN model does not vary significantly between different partitioning of the input data. To accomplish this, I reserve a randomly selected 20% of the total pool of images as "test" data that is not used for training or validation. Then, I use six-fold validation within the training set. I divide the data up into six equally sized groups, and train ten different networks. Each network uses a different group in turn to serve as validation data in the training process. Throughout the work, results I show are for predictions made on the "test" data that was not used as training or validation data. I also use results from different folds as estimates of the bias term. For each fold, I perform a linear regression of the predicted versus true values of $\tau$. I then compute the slope and $y$-intercept of these lines. These can be interpreted as multiplicative and additive forms of bias, and for a well-trained model, the values should be 1 and 0, respectively. Deviations from these values can indicate that a particular CNN model is not well-suited for the problem at hand resulting in a model that does not generalize to data not present in the training set. To remedy this, the hyperparameters may require adjustment. Estimating the variance and noise terms is also important, though not at all quantified by fitting the slope and intercept of a linear

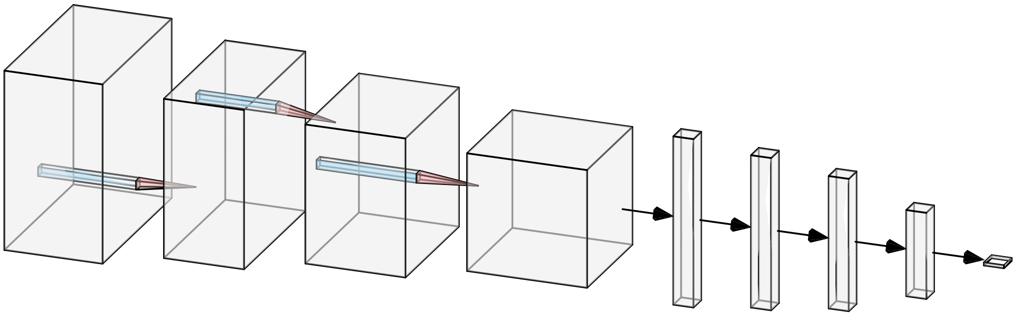


Figure 7.6: A visualization of a typical convolution neural network (CNN) architecture used in the analysis. Cubes represent convolutional layers and vertical bars fully connected layers. Note that other layers, such as max pooling or regularization layers, are not explicitly depicted in the figure. The specific model shown is the "Full Modes" CNN model (Table 7.1). The image was generated using the `NN-SVG` tool. The input images are $512 \times 512 \times 30$, and the output is a vector value of length one, corresponding to the optical depth $\tau$.

regression model. I discuss means by which these forms of error can be quantified.

## 7.4. Architecture Layers

Two-dimensional convolutional neural networks (CNNs) are deep learning algorithms that take in some input data, use an algorithm such as backpropagation to adjust weights and biases internal to the network, and then typically solve regression or classification problems. CNNs are among the most effective algorithms to use in order to understand image contents and their success has captured the attention in industry giants as well as academics.

Convolutions are a useful image processing tool because the convolution operation is able to extract features from locally correlated data points rather than globally which helps in learning about abstract features within the image. In other words, CNNs can capture the spatial dependencies within an image through the application of the convolution filters. The ability of CNNs to make use of information on multiple scales is due in part to CNNs typically having two components: hidden CNN layers, followed by dense layers.

The hidden CNN layers localize feature extractions and also perform a series of convolutions on the image. Then the output of the convolution layers are fed into the pooling layer where further feature extractions are performed. The second component involves the classification

| La Plante & Ntampaka | Full Modes | Cut Modes |
|---|---|---|
| 16 3x3 Conv2D filters | 16 3x3 Conv2D filters | 16 3x3 Conv2D filters |
| BatchNormalization | BatchNormalization | BatchNormalization |
| 2x2 MaxPooling2D | 2x2 MaxPooling2D | 2x2 MaxPooling2D |
| 32 3x3 Conv2D filters | 32 3x3 Conv2D filters | 32 3x3 Conv2D filters |
| BatchNormalization | BatchNormalization | BatchNormalization |
| 2x2 MaxPooling2D | 2x2 MaxPooling2D | 2x2 MaxPooling2D |
| 64 3x3 Conv2D filters | 64 3x3 Conv2D filters | 64 3x3 Conv2D filters |
| BatchNormalization | BatchNormalization | BatchNormalization |
| 2x2 MaxPooling2D | 2x2 MaxPooling2D | 2x2 MaxPooling2D |
| — | 256 3x3 Conv2D filters | 128 3x3 Conv2D filters |
| — | BatchNormalization | BatchNormalization |
| — | 2x2 MaxPooling2D | 2x2 MaxPooling2D |
| — | — | 128 3x3 Conv2D filters |
| — | — | BatchNormalization |
| — | — | 2x2 MaxPooling2D |
| GlobalAvgPooling2D | GlobalAvgPooling2D | GlobalAvgPooling2D |
| — | 20% Dropout | 20% Dropout |
| — | 350 neurons FC | 250 neurons FC |
| 20% Dropout | 20% Dropout | 20% Dropout |
| 200 neurons FC | 200 neurons FC | 200 neurons FC |
| 20% Dropout | 20% Dropout | 20% Dropout |
| 100 neurons FC | 100 neurons FC | 100 neurons FC |
| 20% Dropout | 20% Dropout | 20% Dropout |
| 20 neurons FC | 20 neurons FC | 20 neurons FC |
| Output neuron | Output neuron | Output neuron |

Table 7.1: A summary of the models used in my paper, Billings et al. (2021), with the number of parameters expressed using Keras, a high-level python deep learning library that uses a TensorFlow/Theano backend to do lower level calculations. The model on the far left was trained in La Plante and Ntampaka (2019), the center model was optimized using the full data without the foreground-contaminated $k$-modes removed, and the model on the right was optimized and trained on data with the foreground $k$-modes filtered out.

portion. These are the fully connected, or dense, layers, which serve as the classifier on top of the extracted features. When placing fully connected layers after the convolution layers, the model reduces the image into an easily processed form without losing key features and then assigns a probability for the object on the image being what the algorithm predicts it is. Figure 7.6 shows a visualization of a network used in this work. The CNN layers are shown as cubes, with skewers through the cube depicting the convolutions, and dense layers are columns. The diagram represents a number of CNN hidden layers, after which the transition is made to dense layers. The final output layer is a single neuron containing the value of $\tau$. In general, the CNN architectures used in this work begin with **two-dimensional convolution** layers with a stride of one followed by the **rectified linear unit activation** (ReLU) function. The purpose of the activation function is to become "active" and transfer data exiting one neuron to the next. This non-linearity is key to successful operation of the machine learning network. If the neuron is not activated, no information gets through. After that comes a **batch normalization layer** Ioffe and Szegedy (2015); Santurkar et al. (2018) and finally, a two-dimensional **max pooling layer** with a stride of 2. The batch normalization layer works to restrict the activation of each layer to strictly have zero mean and variance of one. This was once called "covariate shift" and if ignored it can be a problem because the behavior of machine learning algorithms can change when the input distribution changes from layer to layer. It is important to limit covariate shift by normalizing the activation neurons of each layer and as a result batch normalization transforms the inputs to be mean zero and variance of value one making them constant.

Pooling layers allow for down sampling important features within an image by summarizing the presence of features into patches of the feature map. This pooling produces a feature image with low resolution. **Average pooling** and **max pooling** summarize the average presence of a feature and the most activated presence of a feature respectively. In my networks, max pooling layers are repeated four times. Then, the two-dimensional **global average pooling** layer is used. After this, the network alternates using dropout layers and dense layers four times before it reaches the output layer, the predicted value of $\tau$ based on the input images. The **dropout** (20% dropout) layer is a regularization method that randomly ignores some number of neurons in some layer outputs during the training process Srivastava et al. (2014). This dropout is done according to the Bernoulli distribution.Including dropout layers makes the model more robust because it does model averaging with neural network and forces the neurons within the network to extract features. However, if there is too much dropout it can introduce noise into the training process and force other neurons to take on more or less responsibility than necessary. The **dense** or **fully-connected** layer, takes all the input features from different neurons and makes them connected to all the neurons in each layer. The resulting networks will have 358,211 and 354,239 trainable parameters respectively and 736 non-trainable parameters.

For defining, training, and predicting using my CNN architectures, I make use of Keras. Keras Chollet and others (2018) is a high-level wrapper of TensorFlow Abadi et al. (2016), a numerical library capable of constructing and training machine learning networks. It is important to point out that the weights and biases are considered “trainable parameters”, and are updated during the backpropagation process by some optimization algorithm. By default Keras does not update the weights of layers that work to make statistical corrections. For example, the batch normalization and dropout layers are not updated during the training process.

Grid search and manual search are the most widely used strategies for hyperparameter optimization. Randomly chosen trials are a more efficient method for hyperparameter optimization than trials on a grid (Bergstra and Bengio, 2012). To optimize the hyperparameter (architecture, learning rate, loss function) in my model, I use `Keras-Tuner`. This package does a random search through a user-provided list of hyperparameters and then determines the optimal combination of these parameters for the neural network. I use the random search method instead of the manual search because it is much faster. Table 12.1 lists the optimal hyperparameter that were determined by a random search scheme.

## 7.5. Activation Maximum Technique

When using image-based machine learning techniques such as CNNs, an interesting question is how to interpret the inner workings of the algorithm. One way to do this is to examine the effect on an input image of the different convolutional filters at each layer. Though the resulting "images" no longer represent information in the same space as the input images after the first input layer, they do contain information about which particular features of the map the CNN has learned to focus on. Alternatively, one can use the activation maximization technique (Erhan et al., 2009) to visualize the important features in the input map directly, rather than using a partially processed image like I saw before. In this approach, a specific neuron in a dense layer or convolution filter in a convolutional layer of a trained network is chosen. An initially random input image is gradually transformed into an image that maximizes the response of the chosen neuron or filter layer through gradient ascent. The resulting input images do not necessarily look like input images, but instead emphasize the features that are important for the machine to discriminate between different values or feature classes. Such an approach helps visualize which aspects of the the image are being used by the trained network to provide predictions, and complement other methods of visualizing CNN operations. To carry out the actual computation, one can make use of the [`keras-vis`] package . An example of this applied to the problem of $\tau$ extraction is shown in Figure 12.5.

# CHAPTER 8

# OPTIMIZATION TECHNIQUES

## 8.1. Hyperparameter Optimization Background

In the world of machine learning the goal of any supervised machine learning model is to minimize the loss function while still remaining general. To achieve this state, the optimal model parameters and hyperparameters must be selected. Model parameters, sometimes referred to as “weights and biases”, are variables whose best values are informed by the data. These variables are updated during the training process. By updating these parameters, the optimization algorithm indirectly assigns importance to certain features that minimize the loss. In contrast, model hyperparameters are variables associated with the model that are determined only once, prior to the training process. Put another way, these hyperparameters are not typically modified as part of the training process using the input data. At the same time, choosing appropriate values for these hyperparameters is essential for generating results that are acceptably accurate. Global optimization these hyperparameters is a challenging problem of finding an input that results in the minimum or maximum cost of a given objective function. Typically, the form of the objective function is complex and intractable to analyze and is often non-convex, nonlinear, high dimension, noisy, and computationally expensive to evaluate. When attempting to systematically infer which hyperparameter choices are best for a given application, there are four tuning strategies that can implemented: manual search, grid search, random search, and Bayesian search. As might be suggested by its name, manual search can be tedious but worth exploring if one wants to see the effects that each hyperparameter has on the model’s overall accuracy.

Most systematic hyperparameter optimization techniques require first constructing a “grid” of hyperparameters. The user specifies the hyperparameters the algorithm has the freedom to vary, along with an array of acceptable values they each can take on. The outer product of all such combinations forms a multi-dimensional grid of choices. I made use of the

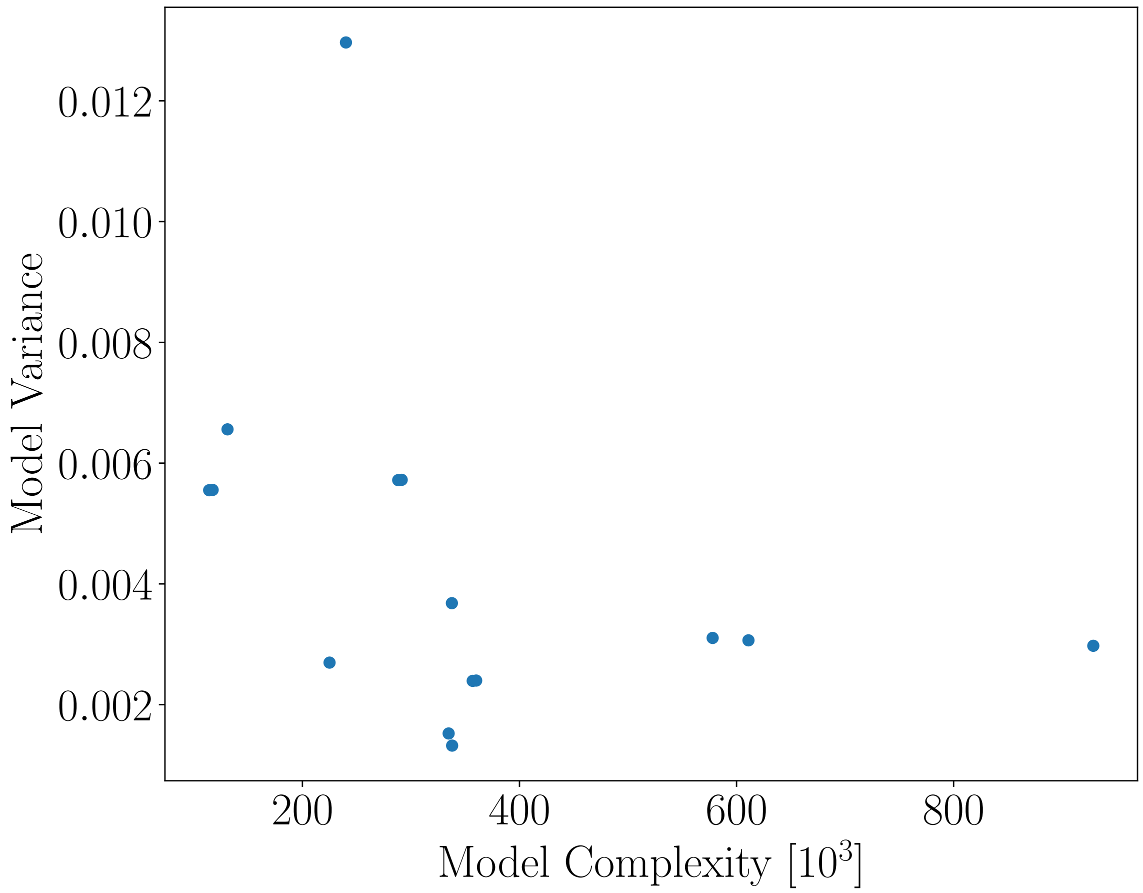


Figure 8.1: In order to select the best possible model from the parameter tuning, I evaluated 15 different models on 10 different test data and calculated the variance of the error for each of the 15 models. I picked the best model by prioritising first the model with the smallest variance and then the lowest complexity (number of trainable parameters). This method was applied for models trained on both the “Full” data (shown here) as well as the wedge filtered data (not shown).

`Keras-Tuner`package to perform the hyperparameter optimization. Specifically, I used the so-called random search method in my research, proposed by Bergstra and Bengio (2012).

### 8.1.1. Grid Search

In the grid search approach, a model is constructed for all possible combinations of hyperparameter values within the grid of hyperparameters values. This is the most basic hyperparameter tuning method. This algorithm explores fewer parameters over the iteration. Though this approach is exhaustive, it does not scale well with dimensionality (similar to traditional parameter fitting for statistical models). Instead, it is more beneficial to have an algorithm that can explore more of the parameter space in the same number of iterations. For this reason this search method is not used often because it is both computationally and time inefficient.

### 8.1.2. Random Search

The search method used throughout this work is random search (Bergstra and Bengio, 2012), which differs from a grid search in that the user no longer provide a discrete set of values to explore for each hyperparameter. Instead, a statistical distribution is provided for each hyperparameter from which values may be randomly sampled. This an extremely valuable strategy to explore because it is known that not all hyperparameters have the same weighted importance. Random search may solve the drawbacks of grid search, as it goes through only a fixed number of hyperparameter settings within the grid in a random fashion to find the best set hyperparameters.

### 8.1.3. Bayesian Search

Finally, the Bayesian methods differ from grid search or random search in that it uses past evaluation results to choose the next values to evaluate and determines the distribution of the parameter instead of some point result. All the hyperparameter combinations are chosen randomly in the random search algorithm. Choosing hyperparameters randomly helps to explore the hyperparameter space but does not guarantee absolute optimal hyperparameters. Instead of all combinations being random, it chooses first few randomly, then based on the performance on these hyperparameters it chooses the next best possible hyperparameters. Hence this algorithm takes into account the history of the hyperparameters which were tried. The iterations of choosing next set of hyperparameters based on history and evaluating performance continues till the tuner reaches optimal hyperparameters or exhausts maximum number of allowed trails.

Note that in addition to quantities used by the optimizer, such as the learning rate and the loss function, the architecture itself can also be varied: allow the number of convolution layers to vary, as well as the number of neurons in the dense layers. In addition to the hyperparameter which were varied, auxiliary parameters can be fixed as part of the optimization process. In general, those parameters may not have a significant impact on the overall performance of the network, and so holding them fixed while varying other quantities proves beneficial.

## 8.2. Model Pruning

Pruning is a data compression technique in machine learning and search algorithms that reduces the size of the model (number of weights and biases). There are different types of pruning methods in an experiment. In a neural network, every neuron is connected to the layer above it which mathematically means it has got to do a lot of multiplications with floating-point values, which takes time and computational power. The neurons can be ranked based on how much they influence the result and by doing so using L1 or L2 norm I can actually sort the neurons in accordance to their ranking.

Using iterative Pruning method, the accuracy will drop and the network is usually trained-pruned-trained-pruned iteratively to recover. If I prune too much at once, the network might be damaged so much it won't be able to recover. In practice, this should be used as an iterative process. Weight pruning sets individual weights in the weight matrix to zero. This corresponds to deleting connections from neuron to neuron to achieve sparsity of k%, rank the individual weights in weight matrix W according to their magnitude, and then set to zero the smallest k%. Finally, neuron pruning, which sets entire columns to zero in the weight matrix to zero, in effect deleting the corresponding output neuron. Here to achieve sparsity of k% I rank the columns of a weight matrix according to their L2-norm and delete the smallest k%.

Using `tensorflow_model_optimization` from the model optimization libraryallows pruning methods to be used. `tensorflow_model_optimization` uses magnitude-based weight pruning that gradually zeroes out model weights during the training process to achieve model sparsity. Sparse models are easier to compress, and the zeroed neurons are skipped during inference process. This pruning library however does not currently work for probabilistic models. A very clever workaround is to use L1 regularization methods to perform magnitude-based weight pruning that also gradually zeroes out model weights during the training process to achieve model sparsity.

## 8.3. L1 and L2 Regularization

A regression model that implements L1 norm for regularisation is called lasso regression. Lasso Regression (Least Absolute Shrinkage and Selection Operator) adds "absolute value of magnitude" of coefficient as penalty term to the loss function. A regression model that implements (squared) L2 norm for regularisation is called ridge regression. Ridge regression adds "squared magnitude" of coefficient as penalty term to the loss function.

The change in weights depends on the $\pm\lambda$ term or the $-2\lambda w$ term, which highlight the influence of the following:

- sign of current weight (L1, L2)
- magnitude of current weight (L2)
- doubling of the regularisation parameter (L2)

Notice that the weight updates using L1 are influenced by the first point, weight updates from L2 are influenced by all the three points.

$$\min_{\mathbf{w}} \frac{1}{n} \sum_{i=1}^{n} (\mathbf{x}_i^\top \mathbf{w} - y_i)^2 + \lambda ||\mathbf{w}||_1 \tag{8.1}$$

$$\min_{\mathbf{w}} \frac{1}{n} \sum_{i=1}^{n} (\mathbf{x}_i^\top \mathbf{w} - y_i)^2 + \lambda ||\mathbf{w}||_2^2 \tag{8.2}$$

The key difference between lasso, Eq. 8.1, and ridge, Eq. 8.2 is that Lasso shrinks the less important feature's coefficient to zero thus, removing some feature altogether. So, this works well for feature selection in case I have a huge number of features. As the weight goes to 0, the number of features are being reduced by reducing the variable importance. This in turn reduces the model complexity, making our model simpler and a simpler model can reduce the chances of overfitting.

# CHAPTER 9

# BAYESIAN STATISTICS

## 9.1. Motivation

Bayesian statistical inference is dependent on Bayes Theorem and is the process of deducing properties about a probability distribution from data. This technique begins with stating prior beliefs about the system being modelled. These beliefs are combined with data to constrain the details of the model. Then, when used to make a prediction, the model does not give one answer, but rather a distribution of likely answers, allowing us to assess confidence. This method of inference naively incorporates the idea of confidence, it performs well with sparse data, and the model, and results are highly interpretable.

## 9.2. Deterministic Network

Deterministic neural networks are prone to overfitting because these networks are also often incapable of correctly assessing the uncertainty in the training data and makes overly confident decisions about the correct prediction. This is addressed by introducing uncertainty in the model parameters of the network. Algorithms like Bayes by Backprop Blundell et al. (2015) introduce an efficient, principled and backpropagation-compatible algorithm for learning a probability distribution on the weights of a neural network. Recall, all weights in the neural networks are represented by probability distributions over possible values, rather than having a single fixed value. The learned representations and computations must therefore be robust under perturbation of the weights, but the amount of perturbation each weight exhibits is also learned in a way that explains variability in the training data. Thus, instead of training a single network, the method trains an ensemble of networks, where each network has its weights drawn from a shared probability distribution. Unlike other ensemble methods, this method typically only doubles the number of parameters yet trains an infinite ensemble using unbiased Monte Carlo estimates of the gradients Mohamed et al. (2019).

To put it plainly, neural networks can act as universal function approximaters, including for ultra-complex functions between inputs and outputs such as image and natural language processing. When gauging predictive uncertainty mere knowledge of the input-output mapping by a deterministic network is inadequate. One of the main limitations of deterministic network is that they are fundamentally a frequentist tool. This is part of the reason why, when there's not much data to work with, deterministic neural networks will often overfit to the data. Overfitting can occur one of two ways; (1) when there is too little training data, (2) lack of diverse data that result in unwanted extrapolations of the data that are obviously unfounded. This becomes obvious when trying to apply a deterministic neural network to data far outside the realm of what the network was trained on. The trendlines that far exceed the range of the original training data for regression tasks or a network will often have to choose one of it's categories to assign data to, even if that instance does not belong to any of them for a classification task. This is by definition overfitting.

In addition to having this overfitting problem to contend, ideally some kind of predictive uncertainty from the model that could reflect the confidence intervals of the model which is incredibly difficult to do with a deterministic neural network is desired.

## 9.3. Problem I and Potential Solution

It is important to note that for classification analysis, modifications to the softmax layer at the end of a neural network will not provide truly reliable confidence intervals Szegedy et al. (2013). The region in the input domain space corresponding to a particular class may be much larger than the space in that region occupied by training examples from that class. The result of this is that an image may lie within the region assigned to a class and be classified with a large peak in the softmax output, while still being far from the images that occur within the domain of the training set. In other words, the data that is far from the domain of the training data should never get a high confidence level, since the model cannot be sure about it as it has never seen it. The reason for this comes down to the kind of problem neural networks are try to solve.

Recall that from the training dataset $\mathcal{D} = \{(x_i, y_i)\}$, it is possible to calculate a likelihood function $p(\mathcal{D}|\mathbf{w}) = \prod_i p(y_i|\mathbf{x}_i, \mathbf{w}_i)$ over the data and $\mathbf{w}$. Maximizing this likelihood function gives the maximimum likelihood estimate (MLE) of $\mathbf{w}$ (maximizing the likelihood of the seen data given the network parameters $\boldsymbol{w}$). However, since MLE is trying to maximize the probability of the data itself, for large numbers of parameters the consequence can be overfitting and failure to generalize. A solution would be to calculate the maximum a posteriori (MAP) point estimates instead of calculating the MLE. This makes the model more robust and resistant by instead optimizing for a data distributions that makes the parameters more likely instead of optimizing for network parameters that make the observed data more likely. This means that instead of using L1 and L2 regularization, Section 8.3, for the MLE calculations, you can use Gaussian Priors and Laplace Priors respectively for MAP calculations.

The goal is to make accurate and precise predictions, estimate the confidence interval, and estimate the uncertainty (standard error) of the predictions however, this change in the optimization problem does not completely fix the problem of overfitting and unwanted extrapolation. Uncertainty should be highest away from the data and lower within the range of the training data (and vice versa for confidence). In addition, both MLE and MAP give point estimates of parameters. We instead want a full posterior distribution over the parameter space to make predictions that take weight uncertainties into account.

### 9.3.1. Bayesian inference

Both MLE and MAP are used to estimate parameters for a distribution. In MLE the goal is to choose parameters that maximize the conditional likelihood. The conditional data likelihood $P(\mathbf{y} \mid X, \theta)$ is the probability of the observed values $\mathbf{y} \in \mathbb{R}^n$ in the training data conditioned on the feature values $\mathbf{x}_i$. Note that $X = [\mathbf{x}_1, \ldots, \mathbf{x}_i, \ldots, \mathbf{x}_n] \in \mathbb{R}^{d \times n}$ and $\mathbf{y}_i \in \mathbb{R}^n$. We choose the parameters that maximize this function and we assume that the $y_i$'s are independent given the input features $\mathbf{x}_i$ and $\mathbf{w}$. So,

$$P(\mathbf{y} \mid X, \mathbf{w}) = \prod_{i=1}^{n} P(y_i \mid \mathbf{x}_i, \mathbf{w}) \tag{9.1}$$

$$\widehat{\mathbf{w}}_{MLE} = \underset{\mathbf{w}}{\operatorname{argmax}} \; P(D; \mathbf{w}) \tag{9.2}$$

In the MAP estimate we treat $\mathbf{w}$ as a random variable and can specify a prior belief distribution over it. It is common to use $\mathbf{w} \sim \mathcal{N}(\mathbf{0}, \sigma^2 I)$, Tran et al. (2020), as the Gaussian approximation for the prior. In order to determine the MAP estimate replace the likelihood in the MLE with the posterior distribution. Comparing the equation of MAP with MLE, it is clear that the only difference is that MAP includes a prior in the formula, which means that the likelihood is weighted by the prior in MAP. In other words, finding the most likely model parameters that maximize the posterior distribution.

$$P(\mathbf{y} \mid X, \mathbf{w}) \approx \prod_{i=1}^{n} P(y_i \mid \mathbf{x}_i, \mathbf{w}) P(\mathbf{x}_i) P(\mathbf{w}) \tag{9.3}$$

$$\widehat{\mathbf{w}}_{MAP} = \underset{\mathbf{w}}{\operatorname{argmax}} \log \; (P(\mathbf{y} \mid X, \mathbf{w}) P(\mathbf{x}_i) P(\mathbf{w})) \tag{9.4}$$

Note that MAP is only one way to get an estimator. There is much more information in $P(\mathbf{w} \mid D)$. A true Bayesian approach is to use the posterior predictive distribution directly to make prediction about the label $y^*$ of a test sample with features $X^*$. MLE, MAP and Bayesian inference are methods used to deduce properties of a probability distribution behind observed data. That being said, there's a big difference between MLE/MAP and Bayesian inference. MLE gives you the value which maximises the Likelihood $P(D|\mathbf{w})$. And MAP gives you the value which maximises the posterior probability $P(\mathbf{w}|D)$. As both methods give you a single fixed value, they're considered point estimators. On the other hand, Bayesian inference fully calculates the posterior probability distribution. Hence, the output is not a single value but a probability density function (when $\mathbf{w}$ is a continuous variable) or a probability mass function (when $\mathbf{w}$ is a discrete variable).

Bayesian models treat the parameters as a random variable and impose preliminary knowledge about the parameter through the prior. Inference in the Bayesian model amounts to conditioning on the data and computing the posterior $P(\mathbf{w}|D)$. This computation is intractable for models where the prior and likelihood take different functional forms (non-conjugates). In these cases, analytical closed form estimation of the marginal likelihood is also intractable. This has led to the usage of sampling methods to solve for such intractable distributions. Markov Chain Monte Carlo (MCMC) method is a sampling techniques which allow us to sample any unnormalized distribution. The idea of the MCMC algorithms is to construct and sample from a Markov chain whose stationary distribution is the same as the desired distribution and use those samples to compute expectations and integrals of required quantities using Monte Carlo integration technique. Gunapati et al. (2022) argues that although most applications of Bayesian Inference for parameter estimation and model selection in astrophysics involve the use of Monte Carlo techniques such as MCMC, they are time consuming and their convergence to posterior is difficult to determine. A remedy to this optimization problem is to introduce variational inference as an alternative to solving parameter estimation and model selection.

## 9.4. Problem II and Potential Solution

The posterior predictive distribution, Equation 9.6, in which the model parameters have been marginalized out is a solution. This is equivalent to averaging predictions from an ensemble of neural networks weighted by the posterior probabilities of their parameters **w**. As a result, this built-in model-averaging component to the model makes it noise resistant. Doing a full Bayesian inference in order to estimate the entire posterior distribution is possible by adjusting the beliefs about a distribution of data or evidence using Bayes rule on the seen data to estimate a full posterior distribution of the parameters.

$$p(\hat{y}|\hat{\boldsymbol{x}}) = \mathbb{E}_{p(\mathbf{w}|\mathcal{D})} p(\hat{y}|\hat{\boldsymbol{x}}, \mathbf{w}) = \int_{\mathbf{w}} p(\hat{y}|\hat{\boldsymbol{x}}, \mathbf{w}) p(\mathbf{w}|\mathcal{D}) d\mathbf{w} \tag{9.5}$$

$$p(\boldsymbol{w}|\mathcal{D}) = \frac{p(\mathcal{D}|\boldsymbol{w})p(\boldsymbol{w})}{p(\mathcal{D})} \tag{9.6}$$

where, $p(\boldsymbol{w}|\mathcal{D})$ is the posterior parameter distribution, $p(\boldsymbol{w})$ is our Prior parameter distribution, $p(\mathcal{D})$ is our evidence, and $p(\mathcal{D}|\boldsymbol{w})$ is the likelihood of the data. It is common to reframe Bayes rule as an approximation, $p(\boldsymbol{w}|\mathcal{D}) \sim \frac{p(\mathcal{D}|\boldsymbol{w})}{p(\mathcal{D})}$, as the likelihood of the data is unknown.

The obvious problem with this built-in model-averaging approach is that calculating an exact solution to this would be computationally expensive and the equations are not differentiable, which means approximating by backpropagation is out of the question. The exact computation of the posterior parameter distribution is also intractable. By extension, the exact computation of the posterior predictive distribution is also intractable. This implies that the differentiation with respect to the model weights is intractable which means that the backpropagation algorithm cannot be used to update the model during the training process.

In general, exact Bayesian inference on the weights of a neural network is intractable because the number of trainable parameters is significantly large and the functional form of a neural network does not lend itself to exact integration. Instead, an approximate inference on the weights can be made using Variational Bayesian approximation, discussed in the next section. While the number of trainable parameters within the hidden layer of a neural network can often be on the order of thousands, the number of weights in a Bayesian Neural Network (BNN) is easily two orders of magnitude larger, making the optimisation problem much larger in scale. Uncertainties associated with the weights within the hidden layers allows the expression of uncertainty about a particular observation and is complementary in that it captures uncertainty about which neural network is appropriate, leading to regularisation of the weights and model averaging. Weights with greater uncertainty introduce more variability into the decisions made by the network. As more data is observed and introduced to the network during the training process, the uncertainty can decrease, allowing the decisions made by the network to become more deterministic as the environment is better understood.

## 9.5. Inference of Probabilistic Models

In this section I will briefly introduce some variational inference techniques to deal with difficult-to-compute probability densities. Variational inference algorithm requires model specific computations to obtain the variational lower bound . The central idea behind variational inference is to solve an optimization problem by approximating the target probability density. The target probability density could be the Bayesian posterior or the likelihood from frequentist analysis. The first step is to propose a family of densities and then to find the member of that family, which is closest to the target probability density Kullback-Leibler divergence Kullback and Leibler (1951) to the conditional of interest. The fitted variational density then serves as a proxy for the exact conditional density.

Neural networks are commonly trained by MLE, the model parameters $\theta$ are estimated in such a way that the likelihood of the observations in $\mathbf{x}_i, y_i$ is maximized. In the Bayesian setting, I choose a prior on the weights $p(\mathbf{w})$ and a model which allows capturing the predictive probability of the model given $\mathbf{w}$. The aim then is to find the posterior distribution given the observed dataset using Bayes' theorem and can be written as Equation (9.3). Once this posterior is computed, the predictive probability distribution of $y^*$ for a new input $\mathbf{x}^* \equiv X^*$ can be obtained by integrating out the model parameters $\mathbf{w}$ resulting in the Equation (9.6). However, this posterior distribution, $P(\mathbf{w} \mid D)$ cannot be approximated analytically and variational inference methods are used to approximates the posterior distribution $P(\mathbf{w} \mid D)$ by a variational distribution $q(\mathbf{w} \mid \theta))$. This variational distribution is a well behaved functional space and depending on a set of variational parameters $\theta$. The objective can then be formalized as finding $\theta$ that makes q as close as possible to the true posterior. This strategy converts the problem of computing the posterior $p(\mathbf{w} \mid \mathbf{D})$ $p(z \mid x)$ into an optimization problem: minimize a divergence equation.

One form of variational inference minimizes the KullBack-Leibler (KL) divergence between the two distributions, $KL(q(\mathbf{w}|\theta)||p(\mathbf{w}|D))$. This function always produces to a positive value and is a measure of relative entropy. It is called a divergence because it is not symmetric, $KL(q(\mathbf{w}|\theta)||p(\mathbf{w}|D)) \neq KL(p(\mathbf{w}|D)||q(\mathbf{w}|\theta))$. The backward (or reverse) KL divergence and the forward KL divergence mean different things. The forward KL divergence means that the resulting approximation will be mode averaging the two distributions (positioned in between the two modes). In the reverse KL divergence means that the resulting approximation will be mode fitting of the two distributions (fitting one of the two modes). It is essentially the gap between the evidence and the Evidence Lower Bound.

The divergence can be expressed as,

$$KL(q(\mathbf{w}|\theta)||p(\mathbf{w}|D)) \equiv \int_{\mathbf{W}} q(\mathbf{w}|\theta)\frac{ln(q(\mathbf{w}|\theta))}{p(\mathbf{w}|D)}d\mathbf{w} \tag{9.7}$$

The problem here is that KL divergence is intractable and I need to find a way to avoid calculating the posterior distribution because its evidence is intractable. Decomposing the KL divergence provides a solution to the intractability problem.

$$\mathrm{KL}(q(\mathbf{w}|\theta)||p(\mathbf{w}|D)) = \log p(\mathbf{D}) + \mathbb{E}_{q(\mathbf{w},\theta)}\big[\log p(D,\mathbf{w}) - \log q(\mathbf{w},\theta)\big] \tag{9.8}$$

The first term on the right hand side of the equation is intractable and the left hand side of the equation is intractable. Rearranging this equation to put all intractable things on one side I can maximize the solvable quantity while simultaneously minimizing KL divergence between our variational distribution and the true posterior and maximizing the evidence.

It has been shown Graves (2011) that minimizing this KL divergence is equivalent to minimizing the following variational free energy function, $\mathfrak{F}$,

$$\mathfrak{F}(D,\theta) = \mathrm{KL}(q(\mathbf{w}|\theta)||p(\mathbf{w})) - \sum_{(\mathbf{x},\mathbf{y})\in D}\int_{\Omega} q(\mathbf{w}|\theta)\ln(q(\mathbf{y}|\mathbf{x},\mathbf{w}))d\mathbf{w} \tag{9.9}$$

Minimising the Kullback–Leibler divergence is equivalent to maximising the log Evidence Lower Bound Gal and Ghahramani (2015a).

The first term in the Variational free energy equation Equation (9.9) is the KL divergence

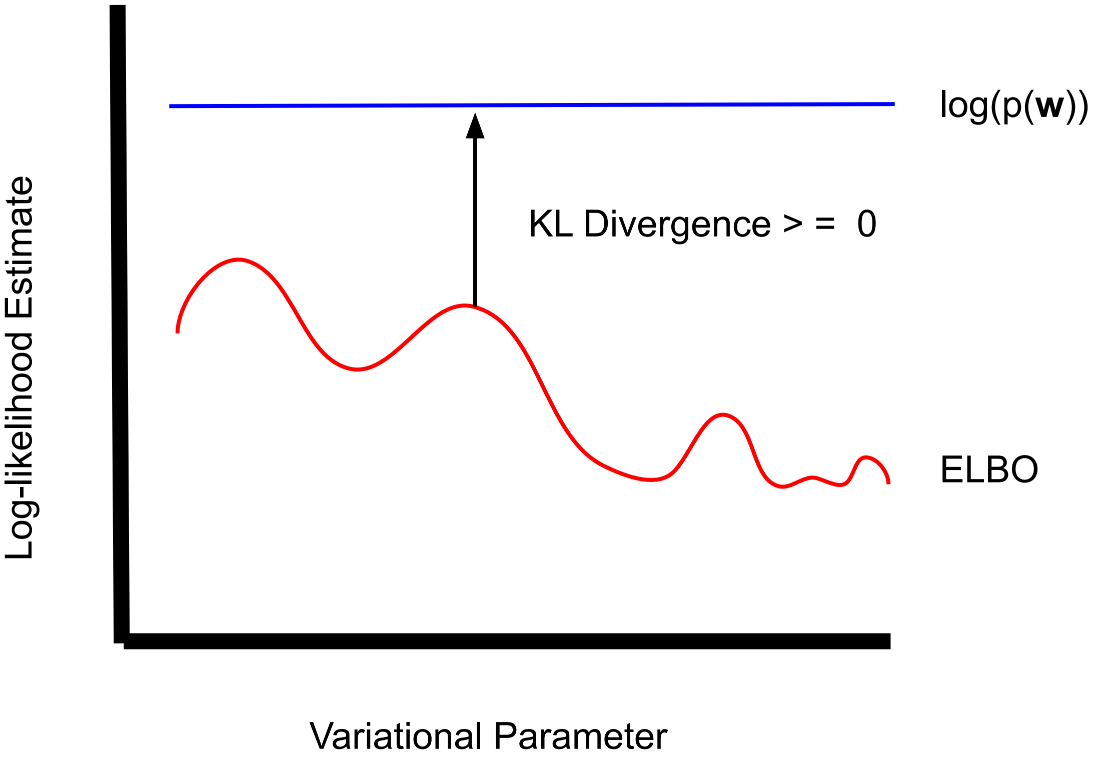


Figure 9.1: Illustration for the KL divergence gap between the marginal log-likelihood log p(D,w) for some data D and the corresponding ELBO for a single 1D-parameter variational distribution.

between the variational distribution and the prior that penalizes complexity priors, while the second term drives the variational distribution to place where the likelihood is high and the data is well explained. As a result, variational inference transforms Bayesian learning from an analytically intractable integral to a manageable optimization problem.

### 9.5.1. Gradient Estimate of ELBO

In this section is I will briefly talk about how the gradients of the ELBO can estimate by using two techniques: the score function estimator, REINFORCE, and the pathwise estimation reparametrization trick.

The goal is to get the gradient of the expectation. The gradient of an expectation cannot be approximated using Monte Carlo methods, instead the Leibniz's rule is used to move the gradient inside the integral. The problem is that most of the time the gradient of a density function is not itself a density function, since it may have negative values and could not integrate to one. When trying to get the gradient of the ELBO the same issue occurs since the gradient of the ELBO corresponds to the gradient of an expectation.

### Reparameterization

In Auto-Encoding Variational Bayes, Kingma and Welling (2013), the author presents an unbiased, differentiable, and scalable estimator for the Evidence lower bound in variational inference and the key idea behind this estimator is the reparameterization trick. The goal of this section is to explain why the reparameterization trick of the variational lower bound yields a simple differentiable unbiased estimator of thee ELBO equation. The problem is that the probability density (or mass) of the data point under the model given the sample is negative. In order to solve this problem alternative methods for generating samples from variational distribution are used. The essence of the parameterization trick involves letting the input sample be a deterministic continuous random variable with independent marginal probability, and is some vector-valued function parameterized by $\theta$. This reparameterization is useful because the the variational distribution expectation is written such that the Monte Carlo estimate of the expectation is differentiable with respect to some vector-valued function parameterized, $\theta$.

Imagine taking the gradient w.r.t. $\theta$ of the following expectation

$$\mathbb{E}_{p(w)}[f_\theta(w)] \tag{9.10}$$

where $p$ is a probability density (or mass). Provided some general function, $f_\theta(w)$, is differentiable, computing the gradient of the expectation value is relatively easy:

$$\begin{aligned}\nabla_\theta \mathbb{E}_{p(w)}[f_\theta(w)] = \nabla_\theta \left( \int_w p(w) f_\theta(w) dw \right) &= \int_w p(w)(\nabla_\theta f_\theta(w)) dw \\ &= \mathbb{E}_{p(w)}[\nabla_\theta f_\theta(w)] \end{aligned} \tag{9.11}$$

The equation above says that the gradient of the expectation is equal to the expectation of the gradient because the probability density function is independent of the parameter $\theta$. However, if the density function is in fact also parameterized by $\theta$, $p_\theta(w)$, the derivative of the expectation can be written as,

$$\begin{aligned}
\nabla_\theta \mathbb{E}_{p_\theta(\mathbf{w}^i)}[f_\theta(\mathbf{w}^i)] &= \nabla_\theta \left( \int_{\mathbf{w}^i} p_\theta(\mathbf{w}^i) f_\theta(\mathbf{w}^i) d\mathbf{w}^i \right) \\
&= \int_{\mathbf{w}^i} \nabla_\theta (p_\theta(\mathbf{w}^i) f_\theta(\mathbf{w}^i)) d\mathbf{w}^i \\
&= \int_{\mathbf{w}^i} (f_\theta(\mathbf{w}^i) \nabla_\theta p_\theta(\mathbf{w}^i)) d\mathbf{w}^i + \int_w (p_\theta(w) \nabla_\theta f_\theta(w)) d\mathbf{w}^i \\
&= \int_{\mathbf{w}^i} (f_\theta(\mathbf{w}^i) \nabla_\theta p_\theta(\mathbf{w}^i)) d\mathbf{w}^i + \mathbb{E}_{p_\theta(\mathbf{w}^i)}[\nabla_\theta f_\theta(\mathbf{w}^i)] \qquad (9.12)
\end{aligned}$$

The first term of the last equation is not guaranteed to be an expectation. Monte Carlo methods require that we can sample from $p_\theta(w)$, but not its gradient. This is not a problem if we have an analytic solution to $\nabla_\theta p_\theta(w)$, but this is not true in general. Applying the reparameterization trick goes as follows

$$\begin{aligned}
\epsilon &\approx p(\epsilon) \\
\mathbf{w} &= g_\theta(\epsilon, \mathbf{x}) \\
\mathbb{E}_{p_\theta(\mathbf{w}^i)}[f_\theta(\mathbf{w}^i)] &= \mathbb{E}_{p_\theta(\mathbf{w}^i)}[f_\theta(g_\theta(\epsilon, \mathbf{x}^i)] \\
\nabla_\theta \mathbb{E}_{p_\theta(\mathbf{w}^i)}[f_\theta(\mathbf{w}^i)] &= \nabla_\theta \mathbb{E}_{p_\theta(\mathbf{w}^i)}[f_\theta(g_\theta(\epsilon, \mathbf{x}^i)] \\
&= \mathbb{E}_{p_\theta}[\nabla_\theta f_\theta(g_\theta(\epsilon, \mathbf{x}^i)]
\end{aligned} \qquad (9.13)$$

provided that $g_\theta$ is differentiable, the reparameterization trick can be used to express a gradient of an expectation as an expectation of a gradient. The issue is that backproping would not compute an estimate of the derivative and without the reparameterization trick, there is no guarantee that sampling large numbers of $\mathbf{w}$ will help converge to the right estimate of derivative.

This is precisely the problem we have when estimating the derivative of the ELBO loss function. The authors in Kingma and Welling (2013) presents two estimators, one is an estimator that can be used when there is an analytic solution to the KL-divergence (the approximate from the true posterior) term in the ELBO and the other term (the variational lower bound on the marginal likelihood of data point) can be Monte Carlo gradient estimated. For the second type of estimator the KL-divergence term can then be interpreted as regularizing $\theta$, encouraging the approximate posterior to be close to the prior. This second version typically has less variance than the generic first estimator. Now that the full loss can compute through a sequence of differentiable operations, a gradient-based optimization technique can be used to maximize the ELBO.

### Bayesian Inference

Variational Bayes approximates the full posterior by attempting to minimize the Kullback-Leibler divergence between the true posterior and a predefined factorized distribution on the same variables. Minimizing this divergence is equivalent to maximizing the familiar variational objective function. Stated another way, this method is used to approximate a posterior distribution with a factorized set of distributions by maximizing a lower bound on the marginalized likelihood, Figure 9.1. This requires the ability to integrate a sum of terms in the log joint likelihood using this factorized distribution. As always, the closed form solution of the integrals is typically difficult to calculate. Also, the solution is only locally optimal when the loss function is not convex, which is usually the case. Most variational inference algorithms optimize the loss function by coordinate ascent, which repeatedly cycles through and optimizes with respect to each variational parameter, Paisley et al. (2012). Coordinate descent updates one parameter at a time, while gradient descent attempts to update all parameters at once.

Typically locally optimal value of each variational parameter have a closed-form solution, such as conjugate exponential models. The log of the joint likelihood results in a sum of terms. One particular issue that often arises in Variational Bayes is that not all expectations in this sum are in closed form. In this scenario a typical solution involves replacing the problematic function with another function of the same variables that is a point-wise lower bound.

Coordinate ascent mean-field variational inference,

- Initialize parameter randomly.
- **repeat**
  - **for** each local variational parameter **do**: Update variational parameter.
  - **end for loop**
  - Update the current estimate of the global variational parameters.
- **repeat until forever** until the ELBO converges.

### Stochastic Inference

Stochastic variational inference optimization algorithms follow noisy estimates of the gradient of the ELBO with a decreasing step size. Noisy estimates of a gradient are often computationally cheaper to compute than the true gradient. Such estimates can allow algorithms to escape shallow local optima of complex objective functions. In statistical estimation problems of the global parameters, the gradient can be written as a sum of terms for each data point and the fast noisy approximation can be computed by subsampling the data. With certain conditions on the step-size schedule, these algorithms provably converge to an optimum. The auther in Nowak (2007) gives an overview of stochastic optimization; Bottou Bottou (2004) gives an overview of its role in machine learning.

Algorithm of stochastic natural gradient ascent on the global variational parameters,

- Initialize parameter randomly.
- Set the step-size schedule appropriately.

- **repeat**
  - Sample a data point uniformly from the data set.
  - Compute its local variational parameter.
  - Compute intermediate global parameters as though is replicated N times.
  - Update the current estimate of the global variational parameters.
- **repeat until forever** until the ELBO converges.

## 9.6. Other Techniques

### 9.6.1. Markov chain Monte Carlo (MCMC)

**Law of Large Numbers**

The law of large numbers is a theorem from probability and statistics that says that the average result from repeating an experiment or trials multiple times will better approximate the true result or before making inferences about what the result means. As the sample size increases, the mean of the sample will move towards the population mean, the true underlying expected value. This is called regression to the mean or sometimes reversion to the mean. The observations in the sample from each trial must be independent. This means that when a trial is run in an identical manner the observations within the sample do not depend on the results of any other trial. In statistics, this expectation is called independent and identically distributed (iid). This is to ensure that the samples are indeed drawn from the same underlying population distribution.

The law of large numbers helps to understand why we cannot trust a single observation from an experiment in isolation. It may be very strangely unlikely to expect that a single result or the mean result from a small sample to represent the mean of the population distribution.

### Central Limit Theorem

The central limit theorem is the most important finding from probability and statistics and it states that if you have a population with mean $\mu$ and standard deviation $\sigma$ and take sufficiently large random samples from the population with replacement, then the distribution of the sample means will be approximately normally distributed. This **is** true regardless of whether the source population is normal or not, provided the sample size is sufficiently large (usually $n \geq 30$). If the population is normal, then the theorem holds true even for samples smaller than 30. This is incredible because this means that we can use the normal probability model to quantify uncertainty when making inferences about a population mean based on the sample mean.

As the sample size increases, the standard deviation of the sampling distribution becomes smaller because the square root of the sample size is in the denominator. In other words, the sampling distribution clusters more tightly around the mean as sample size increases. This also means that the sampling distribution more closely approximates the normal distribution, and the spread of that distribution tightens.

### Monte Carlo Simulation

Monte Carlo methods, Katzgraber (2009) and Walter and Barkema (2015), are a broad class of computational algorithms that rely on repeated random sampling a probability distribution to obtain numerical results. To put it simply, Monte Carlo simulations are just a way of estimating a fixed parameter by repeatedly generating random numbers. By taking the random numbers generated and doing some computation on them, Monte Carlo simulations provide an approximation of a parameter (posterior distribution) where calculating it directly is impossible or expensive. The purpose is to use randomness to solve problems that might be intractable. They are often used in physical and mathematical problems and are most useful when it is difficult or impossible to use other approaches. Monte Carlo methods are mainly used in three problem classes:

- Estimate density, gather samples to approximate the distribution of a target function.
- Approximate a quantity, such as the mean or variance of a distribution.
- Optimize a function, locate a sample that maximizes or minimizes the target function.

Monte Carlo methods are defined in terms of the way that samples are drawn or the way constraints are imposed upon the sampling process:

- Direct Sampling: Sampling the distribution directly without prior information.
- Importance Sampling: Sampling from a simpler approximation of the target distribution.
- Rejection Sampling: Sampling from a broader distribution and only considering samples within a region of the sampled distribution.

There are many problems where describing or estimating the probability distribution is relatively straightforward, but calculating some desired quantity is intractable and calculating an analytical solution cannot be done directly. This happens to be true for most practical probabilistic models. The quantity can instead be approximated either directly or indirectly via a computational simulation by using random sampling methods. Samples can be drawn randomly from the probability distribution and used to approximate the desired quantity. Monte Carlo random sampling methods were initially used around the time that the first computers and are used throughout all fields of science and engineering. The desired calculation is typically a sum of a discrete distribution or integral of a continuous distribution. The calculation may be intractable for many reasons, such as the large number of random variables, the stochastic nature of the domain, noise in the observations, the lack of observations, or due to the stochastic nature of the number of random variables.

It is important to note that drawing a sample from some distribution may be as simple as calculating the probability for a randomly selected event, or may be as complex as running a computational simulation, called a Monte Carlo simulation. Several samples are collected and used to approximate the desired quantity. From the law of large numbers from statistics, the more random trials that are performed, the more accurate the approximated quantity will become. Essentially, the number of samples provides control over the precision of the quantity that is being approximated, often limited by the computational complexity of drawing a sample. Additionally, due to the central limit theorem, the distribution of the samples will form a Normal distribution, the mean can be taken as the approximated quantity and the variance is used to provide a confidence interval for the quantity.

### Markov Chain

The second element to understanding the MCMC methods is the Markov chain portion. These are simply sequences of events that are probabilistically related to one another. Each event comes from a set of outcomes, and each outcome determines which outcome occurs next. This is according to a fixed set of probabilities.

An important feature of Markov chains is that they are memoryless. This means that all the information needed to predict the next event is available in the current state, and no new information comes from knowing the history of events. Markov chains proves that in the long run non-independent distribution of events may also conform or settle to patterns. In other words, interdependent events, if they are subject to fixed probabilities, conform to an average. Markov chains, which seem like an unreasonable way to model a random variable over a few periods, can be used to compute the long-run tendency of that variable if we understand the probabilities that govern its behavior.

### MCMC

MCMC methods allow us to estimate the shape of a posterior distribution where direct calculations are impossible. MCMC methods pick a random parameter value to consider. The simulation will continue to generate random values (the Monte Carlo part), but subject to some rule for determining what makes a good parameter value. The trick is, for a pair of parameter values, it is possible to compute which is a better parameter value, by computing how likely each value is to explain the data, given some prior. If a randomly generated parameter value is better than the last one, it is added to the chain of parameter values with a certain probability determined by how much better it is (the Markov chain part). MCMC methods begin by randomly sampling along the feature axis. Since the random samples are subject to fixed probabilities, they tend to converge after a period of time in the region of highest probability for the parameter we're interested in. MCMC sampling then yields a set of points which are samples from the posterior distribution.

Any statistics calculated on the set of samples generated by MCMC simulations is the best guess of that statistic on the true posterior distribution. MCMC methods can also be used to estimate the posterior distribution of more than one parameter. For n parameters, there exist regions of high probability in n-dimensional space where certain sets of parameter values better explain observed data.

### 9.6.2. Laplace Approximation

As said before, deriving a closed form solution of the posterior distribution can be difficult. Another approximation method that can be used is the Laplace. It states that a Gaussian distribution can be used for approximating the posterior if it is roughly symmetric and unimodal. One of the great benefits of working with the Gaussian distribution is that you only need two numbers to describe it, the mean and the variance. This approximation is often called a quadratic approximation because it uses a quadratic function for approximating the logarithm of the posterior density. This approximation uses the fact that the log of a Gaussian is a parabola - a quadratic function. Taking a Taylor series expansion of the log of the posterior density centered at the posterior mode gives you,

$$\log\, p(\theta|x) = \log\, p(\hat{\theta}|x) + \frac{1}{2}(\theta - \hat{\theta})^T \left[\frac{d^2}{d\theta^2}\log\, p(\theta|x)\right]_{\theta=\hat{\theta}} (\theta - \hat{\theta}) + \ ... \tag{9.14}$$

Finding the posterior mode is simple with a standard optimization procedure. This mode represents the center of the Gaussian that will be used to approximate the posterior. The posterior mode is also referred to maximum a-posteriori. The first term in the expansion above is a constant, but the second term provides an estimate of the curvature near the posterior's peak and is proportional to the log of a Gaussian density. This provides the variance of the Gaussian to approximate the posterior as:

$$p(\theta|x) \approx \mathrm{N}\big(\hat{\theta}, [I(\hat{\theta})]^{-1}\big) \tag{9.15}$$

where $I(\hat{\theta})$ is an estimate for $I(\theta)$, the observed information,

$$I(\theta) = -\frac{d^2}{d\theta^2}\log p(\theta|x). \tag{9.16}$$

Knowledge of the posterior mode and the curvature of the posterior density is enough to determine the approximation of the posterior distribution. It is important to note that since the second order derivative is being calculated, this can be computationally expensive. If there are n parameters, $n^2$ second order derivatives are calculated. Recall, that in the case of scalar functions the concept of derivative is very simple as there is only one variable whose value need to be changed and there is only one output for which we need to measure the change. The derivative of the multivariate functions is the gradient operator and the gradient of a multivariate function is a vector with each component proportional to the derivative of the function with respect to that component. The Jacobian operator is a generalization of the derivative operator to the vector-valued functions. A vector-valued function is a mapping from one space to another, hence, instead of having a scalar value of the function f, a mapping $[x1, x2,\ ..., xn] \longrightarrow [f1, f2,\ ..., fn]$ is true. The gradient and the Jacobian operator are the first order derivative of a multivariate function. To find the second order derivative of a multivariate function is called a Hessian matrix. Finally, the trace of the Hessian matrix is known as the Laplacian operator.

## 9.7. Loss Function

Sampling methods like Metropolis-Hastings use the variational inference method that learns a variational distribution $q(\mathbf{w}|\boldsymbol{\theta})$ which approximates the exact posterior distribution. Using optimized machine learning libraries, such as TensorFlow Probability, allow users to implement this Bayesian variational inference method using the *Flipout* estimator, which approximates model parameters and draws from their distribution during training and testing.

Minimizing the Kullback-Leibler divergence between variational distribution and the true posterior w.r.t. to variational parameter is important. The Kullback-Leibler divergence is not a true measure of distance because as the name (divergence) suggests it is not symmetric ($KL(p||p) \neq KL(q||p)$) and is always positive. It can be viewed as the gap between the evidence and the evidence lower bound and it describes the relative entropy, Figure 9.1.

The KL divergence between the variational distribution $q(\mathbf{w}|\boldsymbol{\theta})$ and the true posterior $p(\mathbf{w}|\mathcal{D})$ is defined as,

$$\begin{aligned}
\mathrm{KL}(q(\mathbf{w}|\boldsymbol{\theta}) \mid\mid p(\mathbf{w}|\mathcal{D})) &= \int \frac{q(\mathbf{w}|\boldsymbol{\theta}) \log q(\mathbf{w}|\boldsymbol{\theta})}{p(\mathbf{w}|\mathcal{D})} d\mathbf{w} \\
&= \mathbb{E}_{q(\omega)} \log \frac{q(\mathbf{w}|\boldsymbol{\theta})}{p(\mathbf{w}|\mathcal{D})}
\end{aligned}$$

Applying Bayes' theorem to the true posterior, $p(\mathbf{w}|\mathcal{D})$ to get,

$$\begin{aligned}
\mathrm{KL}(q(\mathbf{w}|\boldsymbol{\theta}) \mid\mid p(\mathbf{w}|\mathcal{D})) &= \mathbb{E}_{q(\omega)} \log \frac{q(\mathbf{w}|\boldsymbol{\theta})}{p(\mathcal{D}|\mathbf{w})p(\mathbf{w})} p(\mathcal{D}) \\
&= \mathbb{E}_{q(\omega)} \left[\log q(\mathbf{w}|\boldsymbol{\theta}) - \log p(\mathcal{D}|\mathbf{w}) - \log p(\mathbf{w}) + \log p(\mathcal{D})\right] \\
&= \mathbb{E}_{q(\omega)} \left[\log q(\mathbf{w}|\boldsymbol{\theta}) - \log p(\mathcal{D}|\mathbf{w}) - \log p(\mathbf{w})\right] + \log p(\mathcal{D}) \\
&= \mathrm{KL}(q(\mathbf{w}|\boldsymbol{\theta}) \mid\mid p(\mathbf{w})) - \mathbb{E}_{q(\omega)} \log p(\mathcal{D}|\mathbf{w}) + \log p(\mathcal{D})
\end{aligned}$$

using the fact that the log marginal likelihood $\log(p(\mathcal{D}))$ does not depend on the model,$\mathbf{w}$. The first two terms on the RHS are called the variational free energy $\mathcal{F}(\mathcal{D}, \boldsymbol{\theta})$. This equation can be written as,

$$\mathrm{KL}(q(\mathbf{w}|\boldsymbol{\theta}) \mid\mid p(\mathbf{w}|\mathcal{D})) = \mathcal{F}(\mathcal{D}, \boldsymbol{\theta}, \phi) + \log p(\mathcal{D})$$

In order to minimize $\mathrm{KL}(q(\mathbf{w}|\boldsymbol{\theta}) \mid\mid p(\mathbf{w}|\mathcal{D}))$ we really only need to minimize $\mathcal{F}(\mathcal{D}, \boldsymbol{\theta})$ w.r.t.

$\theta$ and $\phi$ as $p(\mathcal{D})$ does not depend on $\theta$. The negative variational free energy is also known as evidence lower bound $\mathcal{E}(\mathcal{D}, \boldsymbol{\theta})$ (ELBO).

$$\mathrm{KL}(q(\mathbf{w}|\boldsymbol{\theta}) \;||\; p(\mathbf{w}|\mathcal{D})) = -\mathcal{E}(\mathcal{D}, \boldsymbol{\theta}) + \log p(\mathcal{D})$$

Recall that the evidence which is a constant with respect to the variational parameter, can be defined as the log of the marginalized likelihood. ELBO also bounds the $\log p(\mathcal{D})$ because the Kullback-Leibler divergence is always non-negative.

This means that if the correct model weights and biases are selected, then you can expect that the marginal probability of the observed data to be high. A large likelihood is "evidence" that the right model for the data has been selected and that you are on the right track. The optimization of ELBO maximizes the probability of the observed data.

$$\mathcal{E}(\mathcal{D}, \boldsymbol{\theta}) = \log p(\mathcal{D}) - \mathrm{KL}(q(\mathbf{w}|\boldsymbol{\theta}) \;||\; p(\mathbf{w}|\mathcal{D}))$$
$$\mathcal{E}(\mathcal{D}, \boldsymbol{\theta}) \leq \log p(\mathcal{D})$$

Therefore, the KL divergence between the variational distribution $q(\mathbf{w}|\boldsymbol{\theta})$ and the true posterior $p(\mathbf{w}|\mathcal{D})$ is also minimized by maximizing the evidence lower bound.

This divergence estimator is typically paired with an approximation of the gradient of the loss function used to update the model. This algorithm is known as backpropagation. The problem with backpropagation is that it cannot flow through the nodes because they are random. The gradient of the expected value of the ELBO loss with respect to the variational parameter $\theta$ for a give probability density function is not guaranteed to be an expectation, only that the distribution can be sampled. A solution to this is to use a reparameterization trick, Section 9.5.1, to rewrite the expectation such that the Monte Carlo Estimate is differentiable. Now, ELBO can be expressed as the sum of the Kullback-Leibler

(KL) divergence and negative log likelihood function Kingma and Welling (2013):

$$\begin{aligned}\nabla_\theta \mathrm{ELBO}(\theta) &= \nabla_\theta[\mathbb{E}_{q(\mathbf{w})}[\log p_\theta(\mathcal{D},\mathbf{w}) - \log q(\mathbf{w}|\mathcal{D})]] \\ &= -\underbrace{\nabla_\theta[\mathrm{KL}[q(\mathbf{w}|\mathcal{D})||p(\mathbf{w})]]}_{\text{Compute Analytically}} + \underbrace{\nabla_\theta\left[\frac{1}{L}\sum_{i=1}^{L}(\log p(\mathcal{D}.|\mathbf{w})\right]}_{\text{Monte Carlo Estimate}}. \qquad (9.17)\end{aligned}$$

This expression is the approximation of the gradient with respect to $\theta$. In this expression, $p(\mathbf{w})$ is the prior distribution, $q(\mathbf{w}|\mathcal{D})$ is the posterior approximation, and $p(\mathcal{D}|\mathbf{w})$ is the likelihood distribution.

After performing stochastic forward passes through the model, it is possible that for a fixed input data set on which predictions are done, sample the posterior distributions and producing an output distribution of the model parameters (i.e., $\tau$ in this case) can be done. This gives a more rigorous way of encapsulating the error due to an inherent variability or stochasticity within the data and subjective uncertainty due to our imperfect knowledge of the best model parameters.

## 9.8. Monte Carlo Dropout

The authors in Gal and Ghahramani (2015b) have shown a connection between Dropout and approximate variational inference for the probabilistic deep Gaussian processes, allowing the neural network to be interpreted as an approximate Bayesian model. In other words a neural network with arbitrary depth and non-linearities, with dropout applied before every weight layer, is mathematically equivalent to an approximation to the probabilistic deep Gaussian process Damianou and Lawrence (2012) and is marginalised over its covariance function parameters. The dropout objective minimises the Kullback–Leibler divergence between an approximate distribution and the posterior of a deep Gaussian process. These results are applicable to any network architecture that makes use of dropout exactly as it appears and no simplifying assumptions are made on the use of dropout in the literature.

Let $\widehat{\mathbf{y}}_i$ be the output of a CNN model with L layers and a loss function (name of the loss we use). $\mathbf{W}_i$ represent the CNN's weight matrices with of dimensions $K_i X K_{i-1}$, and the bias vectors, $b_i$, of dimensions $K_i$ for each layer $i = 1, ..., L$. Recall that $n$ input and output data sets, $(X, Y)$ are denoted by $(x_i, y_i)$. During CNN optimisation an L2 regularisation term is often added and weighted by some weight decay $\lambda$, resulting in a minimisation objective loss function,

$$\mathcal{L}_{\text{dropout}} = \frac{1}{N} \sum_{i=1}^{n} E(\mathbf{y}_i, \widehat{\mathbf{y}}_i) + \lambda \sum_{i=1}^{L} (||\mathbf{W}_i||^2 + ||\mathbf{b}_i||^2) \tag{9.18}$$

With dropout, we sample binary variables for every input point and for every network unit (neuron) in each layer apart from the output layer. Each binary variable takes value 1 with probability $p_i$ for layer i. A neuron is dropped by setting its corresponding binary variable value to zero for a given input. It uses the same values in the backpropagating of the derivatives to the parameters.

Recall that since the posterior distribution $p(w|\mathcal{D})$ is intractable and using $q(w)$, variational distribution for the weights, a distribution over matrices whose columns are randomly set to zero, to approximate the intractable posterior. The variational distribution for the weights of the j-th layer is defined by is defined as,

$$\mathbf{W}_i = \mathbf{M}_i \cdot \text{diag}([\mathbf{z}_{i,j}]_{j=1}^{K_i}) \tag{9.19}$$

$$\mathbf{z}_{i,j} \sim \text{Bernoulli}(p_i) \tag{9.20}$$

for $i = 1, \ldots, L$, $j = 1, \ldots, K_i$ is the neural network's weight matrices of dimensions $(K_i \times K_{i-1})$ given some probabilities $p_i$ and matrices $M_i$ is a matrix of variational parameters to be optimise. The binary variable $z_{i,j} = 0$ corresponds to unit $j$ in layer $i - 1$ being dropped out as an input to layer $i$. The variational distribution $q(w)$ induces strong joint correlations over the rows of the matrices $W_i$ which correspond to the frequencies in the sparse spectrum GP approximation. Inserting this variational distribution, $W_i$, into the variational free energy function Equation 9.9, we obtain an unbiased estimator for the objective function,

$$-\mathcal{F} = \sum_{i=1}^{D} \ln(p(y_i|\mathbf{x}_i, \mathbf{w})) - \lambda \sum_{i=1}^{L} ||\mathbf{W}_i||^2 \tag{9.21}$$

where $\lambda$ is some positive constant and the weights of the network are sampled at each layer from $q(W_i)$. The first term corresponds to the likelihood that encourages the model parameter, w, to explain well the observed data, while the second term is a L2 regularization, weighted by the weight decay parameter $\lambda$. This parameter mimics the KL divergence term. Therefore, training a neural network using Dropout has the same effect as minimizing the KL divergence. Besides working similar to a Bayesian Neural Network, this scheme acts also as a regularization method which prevents over-fitting. After training the neural network, Dropout remains active during the predictive phase allowing us to perform inference and estimates of the uncertainties of the network.

## 9.9. Flipout Estimator

The authors in Adams et al. (2021) present the flipout estimator as an efficient method for decorrelating the gradients within a mini-batch by implicitly sampling pseudo-independent weight perturbations for each example. Flipout achieves the ideal 1/N variance reduction for fully connected networks, convolutional networks, and RNNs and significantly speedups in training neural networks with multiplicative Gaussian perturbations.

Libraries such as TensorFlow Probability allows users to implement the Bayesian inference method by using the *Flipout* estimator, which approximates model parameters and draws from their distribution during training and testing. This flipout estimator reshuffles the weights in a mini-batch to make them more independent of each other. In turn this reduces variance and requires fewer training epochs than the reparameterization method that is also available in TensorFlow Probability. The *Flipout* estimator is typically paired with an approximation of the gradient of the loss function. However, although *Flipout* needs fewer epochs it actually requires twice as many floating point operations as the reparameterization estimator because it is calculating the mean and variance of the model parameters. The

*Flipout* arguments permit separate specification of the surrogate posterior ($q(w|\mathcal{D})$), prior ($p(w)$), and divergence for both the kernel and bias distributions.

Upon building the model, these *Flipout* layers adds losses (accessible via the losses property) representing the divergences of kernel and/or bias surrogate posteriors and their respective priors. Recall that in order to fit a model using variational inference method, it is necessary to minimize the negative expected lower bound, which is the sum of the expected negative log likelihood and the KL divergences. These two terms are not on the same scale and need to be on the rescale. This is due to the design of Tensorflow and how it backpropagates the corrections of batch trained models. By averaging the log likelihood (that is, divide the sum by the number of samples in the batch) and dividing the sum of the divergences by the total number of samples in the dataset (NOT in the batch), puts the two terms on the same scale. This rescaling is extremely important when doing minibatch stochastic optimization. In Graves (2011) section 6, the author details noisy gradient estimates and emphasizes the need for rescaling.

## 9.10. Frequentist vs Bayesian Method

In summary, the frequentist method is the probability of an event is the limit of the relative frequency of an event after a large number of trials. This is the same as calculating the probability that the experiment would have the same outcomes if the experiment were replicate many times under the same conditions. This model only uses data from the current experiment when evaluating outcomes.

When applying the frequentist statistics, the p-value is the calculated probability of obtaining an effect at least as extreme as the one in the sample data, assuming the truth of the null hypothesis. For a small p-value this means that there is a small chance that the results could be completely random (the more statistically significant the results) whereas for a large p-value it means that the results have a high probability of being random and not due to anything done in the experiment. P-values are probability statements about the data sample not about the hypothesis itself.

In Bayesian statistics, the probability of an event is expressed as a degree of belief in that event. This method is different from the frequentist methodology in a number of ways. One of the big differences is that probability actually expresses the chance of an event happening. This method is simpler and more intuitive approach for testing. The Bayesian concept of probability is more conditional. It uses prior and posterior knowledge as well as current experiment data to predict outcomes. The Bayesian approach attempts to account for previous knowledge and data that could influence the end results.

# CHAPTER 10

# BACKPROPAGATION

## 10.1. Classical Backpropagation

Backpropagation (backpropagate the error) is an algorithm used to help calculate the gradient of the loss/cost function for a single training example or if backpropagation combined with a learning algorithm such as stochastic gradient descent, in which we compute the gradient for many training examples. Recall that the gradient descent is an algorithm that updates the weights and bias of the model . At the heart of backpropagation is an expression for the partial derivative of the loss function with respect to any weight w or bias b in the network. The expression tells us how quickly the cost changes when we change the weights and biases and how changes the overall behaviour of the network.

In the paper Rumelhart et al. (1986) they describes several neural networks where backpropagation works far faster than earlier approaches to learning, making it possible to use neural nets to solve problems which had previously been unsolvable. Today, the backpropagation algorithm is the workhorse of learning in neural networks.

Backpropagation is the fasted algorithm as opposed to a obvious way of estimating the change in the loss function with respect to a certain weight or bias term using

$$\frac{\partial \mathcal{L}}{\partial w^L_{jk}} \approx \frac{\mathcal{L}(w + \epsilon e_j) - \mathcal{L}(w)}{\epsilon} \tag{10.1}$$

where $\epsilon > 0$ is a small positive number, and $e_j$ is the unit vector in the $j^{\text{th}}$ node direction. This looks deceptively simple however, in practice this algorithm is computationally expensive. For a training dataset of size one million, this means for each distinct weight, $\mathcal{L}(w+\epsilon e_j) - \mathcal{L}(w)$ needs to be computed in order to compute partial derivative. That means that for a million weights the gradient of the loss function is computed a million different times, requiring a million forward passes through the network per training data input and

$\mathcal{L}(w)$ needs to be computed as well, so that's a total of a million and one passes through the network.

The amazing thing about the backpropagation algorithm is that it simultaneously computes all the partial derivatives using only one forward pass through the network, followed by one backward pass through the network. Therefore, the computational cost of the backward pass is about the same as the forward pass meaning the total cost of backpropagation is roughly the same as making just two forward passes through the network compared to the million and one forward passes.

### 10.1.1. Assumption about Loss Function

Remember, the goal of backpropagation is to compute the gradient of the loss function, $\mathcal{L}$, written as partial derivatives $\partial\mathcal{L}/\partial w$ and $\partial\mathcal{L}/\partial b$ of the cost function with respect to any weight (w) or bias (b) in the network.

Assume some quadratic loss function with simplified indices,

$$\mathcal{L} = \frac{1}{2n}\sum_x \|y(x) - a^L(x)\|^2, \tag{10.2}$$

where $a^L(x)$, it the vector of activations output from the last layer, L, for an input x; n is the total number of training examples; the sum is over individual training examples, x; $y = y(x)$ is the corresponding desired output; L denotes the number of layers in the network.

In order for backpropagation to work the first assumption needed is that the loss function can be written as an average $\mathcal{L} = \frac{1}{n}\sum_x \mathcal{L}_x$ over loss functions $\mathcal{L}_x$ for each training data, x. The loss of a single training data is $\mathcal{L}_x = \frac{1}{2}\|y - a^L\|^2$. This assumption is needed because backpropagation is computing the partial derivatives $\partial\mathcal{L}/\partial w$ and $\partial\mathcal{L}/\partial b$ for a single training data point. $\partial\mathcal{L}/\partial w$ and $\partial\mathcal{L}/\partial b$ are recovered by averaging over a fixed training data sets.

The second assumption made about the loss is that it can be written as a function of the outputs from the neural network making it a function of the output activations instead of simply a function of the desired output y,

$$\mathcal{L} = \frac{1}{2}\|y - a^L\|^2 = \frac{1}{2}\sum_j (y_j - a_j^L)^2, \tag{10.3}$$

Since the input data x is fixed, and as such the output y is also a fixed parameter, it's not something that can be modify by changing the weights and biases in any way. In other words, it is not something which the neural network learns. The input and desired output are merely parameters that helps define that loss function.

### 10.1.2. Fundamental Equations of Backpropagation

Backpropagation is about understanding how changing the weights and biases in a network changes the loss function by computing the partial derivatives $\partial\mathcal{L}/\partial w_{jk}^L$ and $\partial\mathcal{L}/\partial b_j^L$ to compute the some error $\delta_j^L$ that are related to the partial derivatives.

$$\delta_j^L \equiv \frac{\partial \mathcal{L}}{\partial z_j^L} \tag{10.4}$$

$\delta_j^L$ of neuron j denoteS the vector of errors associated with layer L. Backpropagation is a way of computing $\delta^L$ for every layer and the gradient of the loss function. All four equations below are a consequences of the chain rule.

**Error in the output layer, $\delta^L$**

Recall that the components of $\delta^L$ are given by,

$$\delta_j^L = \frac{\partial \mathcal{L}}{\partial z_j^L}$$

Applying the chain rule the equation above can re-expressed in terms of partial derivatives with respect to the output activations,

$$\delta_j^L = \sum_k \frac{\partial \mathcal{L}}{\partial a_k^L}\frac{\partial a_k^L}{\partial z_j^L}$$

where the sum is over all neurons k in the output layer. Of course, the output activation $a_k^L$ of the $k^{\text{th}}$ neuron depends only on the weighted input $z_j^L$ for the $j^{\text{th}}$ neuron when k=j. And so $\partial a_k^L / \partial z_j^L$ vanishes when $k \neq j$. As a result we can simplify the previous equation to,

$$\delta_j^L = \frac{\partial \mathcal{L}}{\partial a_j^L} \frac{\partial a_j^L}{\partial z_j^L}$$

We know that $a_j^L = \sigma(z_j^L)$. Finally, the components of $\delta^L$ are given by,

$$\delta_j^L = \frac{\partial \mathcal{L}}{\partial a_j^L} \sigma'(z_j^L) \tag{10.5}$$

The first term on the right, $\frac{\partial \mathcal{L}}{\partial a_j^L}$, measures how fast the loss is changing as a function of the $j^{\text{th}}$ output activation. If the loss does not depend much on a particular output neuron, j, then $\delta_j^L$ will be small. The second term on the right, $\sigma'(z_j^L)$, measures how fast the activation function is changing at $z_j^L$.

**Error in the next layer, $\delta^{L+1}$**

An equation for the error $\delta^l$ can be written in terms of the error in the next layer, $\delta^{L+1}$,

$$\begin{aligned} \delta_j^L &= \frac{\partial \mathcal{L}}{\partial z_j^L} \\ &= \sum_k \frac{\partial \mathcal{L}}{\partial z_k^{L+1}} \frac{\partial z_k^{L+1}}{\partial z_j^L} \\ &= \sum_k \frac{\partial z_k^{L+1}}{\partial z_j^L} \delta_k^{L+1} \end{aligned}$$

where in the last line we have interchanged the two terms on the right-hand side, and substituted the definition of $\delta_k^{L+1}$. To evaluate the first term on the last line, note that

$$z_k^{L+1} = \sum_j w_{kj}^{L+1} a_j^L + b_k^{L+1} = \sum_j w_{kj}^{L+1} \sigma(z_j^L) + b_k^{L+1}$$

and differentiating this equation gives,

$$\frac{\partial z_k^{L+1}}{\partial z_j^L} = w_{kj}^{L+1}\sigma'(z_j^L)$$

The final result is,

$$\delta_j^L = \sum_k w_{kj}^{L+1}\delta_k^{L+1}\sigma'(z_j^L) \tag{10.6}$$

where $w_{kj}^{L+1}$ is the transpose of the weight matrix for the (L+1)th layer. Application of this transposed matrix is equivalent to moving the error backward through the network and through the activation function, giving some sort of measure of the error, $\delta^L$, at the output of the $L^{\text{th}}$ layer.

Start by using Equation 10.5 to compute $\delta^L$, then apply Equation 10.6 to compute $\delta^{L-1}$, then Equation 10.6 again to compute $\delta^{L-2}$, and so on, all the way back through the network.

**Rate of change of the loss with respect to any bias**

An equation for the rate of change of the cost with respect to any bias in the network,

$$\frac{\partial \mathcal{L}}{\partial b_j^L} = \delta_j^L. \tag{10.7}$$

Equation 10.5 and Equation 10.6 show how to compute $\delta_j^L$.

**Rate of change of the loss with respect to any weight**

An equation for the rate of change of the loss with respect to any weight in the network,

$$\frac{\partial \mathcal{L}}{\partial w_{jk}^L} = a_k^{L-1}\delta_j^L. \tag{10.8}$$

This tells us how to compute the partial derivatives in terms of quantities already know, $\delta^L$ and $a^{L-1}$. The partial derivative equals the product of the activation of the neuron input to the weight w and the error of the neuron output from the weight w. If the activation of the neuron input to the weight is small then the gradient term also tends to be small. This means that the weight learns slowly and is no longer changing much during gradient descent. One consequence of this is that weight's output from low-activation neurons learn slowly. The output neuron can also be saturated resulting in either the weights no longer learning or are learning slowly. The same is true for the biases of output neuron.

To improve intuition about what the algorithm is doing, imagine that there is a small change $\Delta w_{jk}^L$ to some weight in the network, $w_{jk}^L$. That change in weight will cause a change in the output activation from the corresponding neuron. Then that will cause a change in all the activations in the next layer. Those changes will in turn cause changes in the next layer, and then the next, and so on all the way through to causing a change in the final layer, and then in the loss function,

$$\Delta \mathcal{L} \approx \frac{\partial \mathcal{L}}{\partial w_{jk}^L} \Delta w_{jk}^L. \tag{10.9}$$

This equation tracks how a small change in $w_{jk}^L$ propagates to cause a small change in $\mathcal{L}$. Calculation of the $\partial \mathcal{L} / \partial w_{jk}^L$ by remembering that if $\Delta w_{jk}^L$ is non-zero, then a small change $\Delta a_j^L$ in the activation of the $j^{\text{th}}$ neuron in the $L^{\text{th}}$ layer. This change is given by,

$$\Delta a_j^L \approx \frac{\partial a_j^L}{\partial w_{jk}^L} \Delta w_{jk}^L. \tag{10.10}$$

and the change is propagated to the next layer is given by,

$$\Delta a_q^{L+1} \approx \frac{\partial a_q^{L+1}}{\partial a_j^L} \frac{\partial a_j^L}{\partial w_{jk}^L} \Delta w_{jk}^L. \tag{10.11}$$

Then the change in the loss function, $\mathcal{L}$, due to changes in the activations along this particular path through the network a more generalized sum over all the possible paths between the weight and the final loss is given by,

$$\Delta\mathcal{L} \approx \sum_{mnp\ldots q} \frac{\partial\mathcal{L}}{\partial a_m^L}\frac{\partial a_m^L}{\partial a_n^{L-1}}\frac{\partial a_n^{L-1}}{\partial a_p^{L-2}}\cdots\frac{\partial a_q^{L+1}}{\partial a_j^L}\frac{\partial a_j^L}{\partial w_{jk}^L}\Delta w_{jk}^L, \tag{10.12}$$

| |
|---|
| $\delta_j^L = \frac{\partial\mathcal{L}}{\partial a_j^L}\sigma'(z_j^L)$ |
| $\delta_j^L = \sum_k w_{kj}^{L+1}\delta_k^{L+1}\sigma'(z_j^L)$ |
| $\frac{\partial\mathcal{L}}{\partial b_j^L} = \delta_j^L$ |
| $\frac{\partial\mathcal{L}}{\partial w_{jk}^L} = a_k^{L-1}\delta_j^L$ |

Table 10.1: The four fundamental equations of backpropagation.

### 10.1.3. Backpropagation Pseudocode

The backpropagation equations provide us with a way of computing the gradient of the cost function. Let's explicitly write this out in the form of an algorithm:

- Input x: Set the corresponding activation a1 for the input layer.
- For each training data x: Set the corresponding input activation as $a^{x,1}$, and perform the following steps
  - Feedforward: For each $l = 2, 3, \ldots, L$ compute, $a^{x,l} = \sigma(z^{x,l})$
  - Output error $\delta^{x,L}$: Compute the vector,
    $\delta^{x,L} = \nabla_a\mathcal{L}_x \odot \sigma'(z^{x,L})$
  - Backpropagate the error: For each l = L-1, L-2, . . . ,
    $\delta_j^{x,l} = \sum_k w_{kj}^{l+1}\delta_k^{x,l+1}\sigma'(z_j^{x,l})$
- Gradient descent: For each $l = L, L-1, \ldots, 2$ update the weights according to the rule $w^l \rightarrow w^l - \frac{\eta}{m}\sum_x \delta^{x,l}(a^{x,l-1})^T$, and the biases according to the rule $b^l \rightarrow b^l - \frac{\eta}{m}\sum_x \delta^{x,l}$.

As you can see, the error vectors $\delta^L$ is computed backwards, starting from the final layer.

The backward movement is a consequence of the fact that the cost is a function of the outputs from the network. To understand how the cost varies with earlier weights and biases we need to repeatedly apply the chain rule, working backward through the layers to obtain usable expressions.

# CHAPTER 11

# SIMULATION

## 11.1. Zreion Data Design

The simulated 21 cm data used in this paper was generated using the semi-numeric technique first developed in Battaglia et al. (2013). This model considers the redshift at which different region in the universe become highly ionized, such that the ionization fraction $x_i \sim 1$. This leads to defining a local "redshift of reionization" field $z_{\rm re}(\mathbf{x})$, with fractional fluctuations $\delta_z(\mathbf{x})$:

$$\delta_z(\mathbf{x}) = \frac{[z_{\rm re}(\mathbf{x}) + 1] - [\bar{z} + 1]}{\bar{z} + 1}, \tag{11.1}$$

where $\bar{z}$ is the mean redshift of reionization, chosen as an input to the model. The reionization field $\delta_z(\mathbf{x})$ is assumed to be a biased tracer of dark matter on large scales ($\geq 1\ h^{-1}$Mpc) with bias parameter $b_{zm}(k)$:

$$b_{zm}^2(k) \equiv \frac{\langle \delta_z^* \delta_z \rangle}{\langle \delta_m^* \delta_m \rangle} = \frac{P_{zz}(k)}{P_{mm}(k)}. \tag{11.2}$$

This bias parameter is a three-parameter function of spherical wavenumber $k$ and the result of relating the dark matter density and redshift fields:

$$b_{zm}(k) = \frac{b_0}{\left(1 + \frac{k}{k_0}\right)^{\alpha}}, \tag{11.3}$$

where $b_0$ is the bias amplitude, $k_0$ is the scale threshold, and $\alpha$ is an asymptotic exponent. We use a value of $b_0 = 1/\delta_c = 0.593$. Given this parameterization, we are able to vary the reionization history by changing the parameter $\bar{z}$ to modify the midpoint, and the parameters $k_0$ and $\alpha$ to adjust the duration.

The dark matter density field is generated at the mean redshift $\bar{z}$, then it is Fourier transformed into $k$-space. The bias function in Equation 11.3 is then used to generate $\delta_z(\mathbf{k})$ by

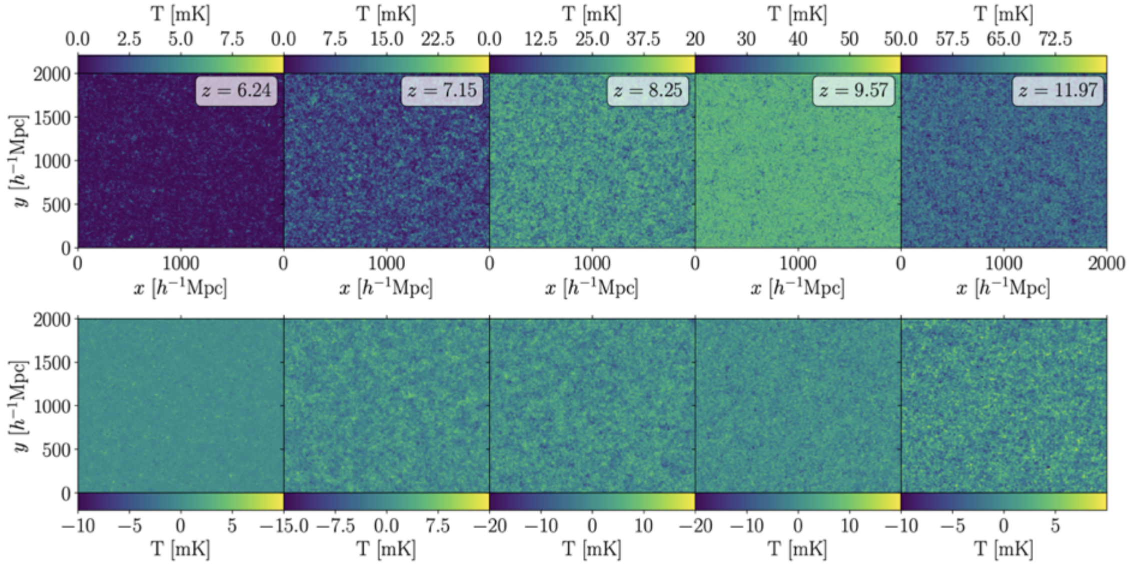


Figure 11.1: This is a visualization of the 21 cm images before (top) and after (bottom) the application of the foreground effects. The different columns are different redshift slices. The most dramatic change is that the zero-level is no longer the absence of the 21 cm signal, as in the top row, but is the mean value of the Fourier-transformed slab. This effect is due to the removal of the $k_{\|} = 0$ mode as part of the foreground effects, which ensures that the resulting inverse-Fourier transformed slab must have mean 0. Also note that the structures in the top panel are no longer in the corresponding places in the bottom panel, another effect of the application of the foreground wedge. Although this effect makes matching the locations of individual sources difficult, statistically, the fields seem to have similar properties.

simple mode-wise multiplication. An inverse Fourier transform is applied to this $k$-space field to arrive at $\delta_z(\mathbf{x})$. Then it is finally inverted using Equation 11.1 to get the field $z_{\mathrm{re}}(\mathbf{x})$, the reionization history for some volume. We can use the redshift of reionization field to calculate the local ionization field for some redshift $z$. Finally, we combine the local ionization field with the matter density field to compute the 21 cm signal:

$$\delta T_b \quad = 26(1+\delta_m)x_{\mathrm{HI}}\left(\tfrac{T_S - T_\gamma}{T_S}\right)\left(\tfrac{\Omega_b h^2}{0.022}\right) \times \left[\left(\tfrac{0.143}{\Omega_m h^2}\right)\left(\tfrac{1+z}{10}\right)\right]^{\frac{1}{2}} \text{ mK} \qquad (11.4)$$

where $x_{\mathrm{H_I}} = 1 - x_i$ is the neutral fraction field for a given point in the volume, $T_S$ is the spin temperature of the gas, and $T_\gamma$ is the temperature of the CMB at some redshift. Using this semi-analytic model of reionization, we can generate mock images of the 21 cm brightness temperature $\delta T_b$ at different redshift values in an efficient way by adjusting the parameters $\bar{z}$, $k_0$, and $\alpha$.

For this study, we generated 1000 realizations of the 21 cm field from a dark-matter density field generated from a single $N$-body simulation which tracked $2048^3$ particles in a cubic volume of 2 $h^{-1}$Gpc on a side using a P$^3$M algorithm described in Harnois-Deraps et al. (2012). To avoid presenting the same density structures in different 21 cm realizations (and thus potentially biasing the results), the line-of-sight direction and zero-indexed pixels were randomly chosen. Once the starting indices for each axis were chosen, the box was permuted using periodic boundary conditions. This approach helps mitigate repetition of the underlying density field for the purposes of generating snapshots. We then randomly sample the parameters $\bar{z}$, $\alpha$, and $k_0$ from a uniform range of values to generate a unique reionization field $z_{\mathrm{re}}(\mathbf{x})$.

In order to obtain the density field at $z = \bar{z}$, the two neighboring matter density fields are loaded into memory, and interpolated in scale factor $a$ for every point in the volume. This allows for the construction of an approximate density field for any desired redshift without having to run a new simulation. The simulation of that particular realization then proceeds as outlined above.

Note that this method does not take $\tau$ directly as an input, but given the field $z_{\mathrm{re}}(\mathbf{x})$, the value of $\tau$ can be computed from the simulated volume.

In general, the reionization histories produced by our parameter choices feature a late end of reionization, and tend to have a relatively broad duration of reionization. The corresponding values of $\tau$ range from $0.045 \leq \tau \leq 0.068$. This range covers the values of $\tau$ reported by the Planck 2015 and Planck 2018 cosmological parameters. Afterwards, 30 redshift values at constant intervals in co-moving distance between $6 \leq z \leq 12$ are chosen. Two-dimensional slices are generated at each redshift value, which serve as the input data for the CNN architecture. By treating these 30 input images as "color channels" in the input data, we are able to make full use of the tomographic data potentially available from observations.

To construct the 1000 reionization realizations we use as training and validation data, an $N$-body simulation is performed which generates the dark matter density field. After selecting a new combination of $\bar{z}$, $k_0$, $\alpha$, a matter density field at $z = \bar{z}$ is generated. In order to obtain the density field at an arbitrary redshift, the two neighboring matter density fields are loaded into memory, and interpolated in scale factor a for every point in the volume. This allows for the construction of an approximate density field for any desired redshift without having to run a new simulation. The bias relation in Equation 11.3 is applied to the matter density field in Fourier space to generate the field $\delta_z(\mathbf{x})$, which can then yields $z_{\mathrm{re}}(\mathbf{x})$ by applying an inverse Fourier transform along with Equation 11.1. Equation 11.4 yields the local 21 cm brightness temperature.

To understand the performance of the CNN in the presence of foreground contamination, we generate two versions of the same input data: one using just the data from the simulation, and another where the modes expected to be contaminated by foreground emission have been removed from the data. The power of foreground emission can be written as a function of the Fourier mode along the line of sight $k_{||}$ and in the plane of the sky $k_\perp$. The slope $m$ relating the two is a function of redshift, but is largely independent of instrument specifics. The boundary between the foreground contaminated and foreground free region is given by

Thyagarajan et al. (2015):

$$m(z) \equiv \frac{k_\parallel}{k_\perp} = \frac{\lambda(z) D_c(z) f_{21} H(z)}{c^2 (1+z)^2}, \tag{11.5}$$

where $\lambda(z) = \lambda_0(1+z)$ is the wavelength of the 21 cm radiation at the redshift of interest, $D_c$ is the co-moving distance to redshift $z$, $f_{21}$ is the rest-frame frequency of the 21 cm signal, and $H$ is the Hubble parameter. For the redshifts of interest, $m \sim 3$. We approximate the effects of ignoring contaminated foreground modes by setting all modes below the slope $m$ in Equation 11.5 to 0. For this "Cut" input data where the foreground-contaminated modes were removed, we extracted a slab of 50 pixels along the line of sight (so the full slab had dimensions of 2048 × 2048 × 50) and applied the wedge cut individually to each of the 30 input slabs of data. Afterwards, we selected the central slice from the input data to serve as a representative sampling of the slab.

As a point of comparison, we also generated "Full" input data which did not remove any $k$-modes, and the 30 input slices were merely averages over this same 50-pixel slab. In both cases, we also down-sampled from the native 2048 × 2048 pixel resolution within a slice to 512 × 512 pixels, in order to make the data more manageable for the machine learning application. Note that in practice the actual combination of different co-moving regions along the line of sight will be determined by the observing strategy of the instruments, though for the time being we use this approach as an approximation. Also note that this approach does not include other sources of observational uncertainty, such as thermal noise from the instrument or other systematic errors. We defer additional treatment of these issues to future work.

## 11.2. Other Simulators

### 11.2.1. `21cmFast` Data Design

`21cmFAST` is a powerful semi-numeric modeling tool designed to efficiently simulate the cosmological 21-cm signal from neutral hydrogen Mesinger et al. (2010). By varying the number of parameters relating to the amount of ionizing photons escaping from high-redshift galaxies (ionizing efficiency), the minimum virial temperature of halos producing ionizing photons (typically chosen to be $10^4$ K, which corresponds to a halo solar mass of $10^8$ at $z = 10$), the soft X-ray emissivity, and the X-ray energy threshold for self-absorption by the galaxy, independent realization of the initial Gaussian random field generate the 21-cm light-cones. This is done by randomly sampling the 4 astrophysical parameters stated above, uniformly over the ranges of possible values, Pober et al. (2014); Gillet et al. (2019).

### 11.2.2. `Toy Model`

The `ToyModel` is a simple simulation of ionized maps of the Universe. Each image is normalized from zero to one such that in the ideal case has no instrument features and noise. The ionization fraction at a particular redshift is trivially calculated by taking the averaging of the image resulting in the fractional number of pixels ionized. The simulation is constructed by starting with a filtered Gaussian random field and converting each pixel to standard normal image. Then I select a random ionization threshold such that while that threshold is not met, the pixels will not be labeled as ionized. However, if any of the pixels exceed this threshold, then they are labeled ionized. In short, the `ToyModel` is an nonphysically motivated smoothed Gaussian random field where all pixels above a particular threshold are labeled "ionized" smoothing filter of around sigma = 3 pixels.

In summary, the `ToyModel` is the only truly Gaussian field while `zreion` and `21cmfast` both start from a simulation of the cosmological density field, which is slightly non-Gaussian. `zreion` takes that density field, applies a spherical wavenumber dependent bias at the midpoint redshift, and then determines at what redshift a given biased density would ionize. This method involves matching simulations like `21cmfast`. `21cmfast` takes the density field and assigns a certain amount of star formation to each spot. There is no star formation if the density is such that the halo mass is below a cutoff value, and the amount of star formation proportional to the halo mass if it is above the cutoff.

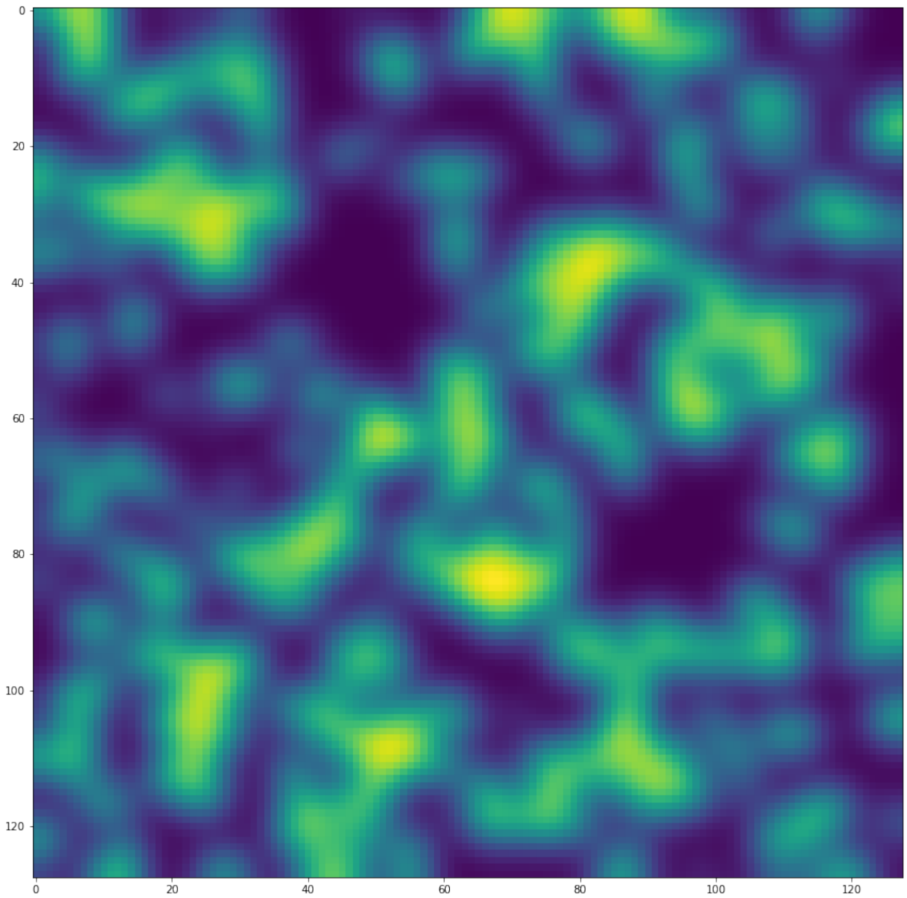


Figure 11.2: Gaussian Filter used in on the Toy model simulation to imitate instrumental effect on images.

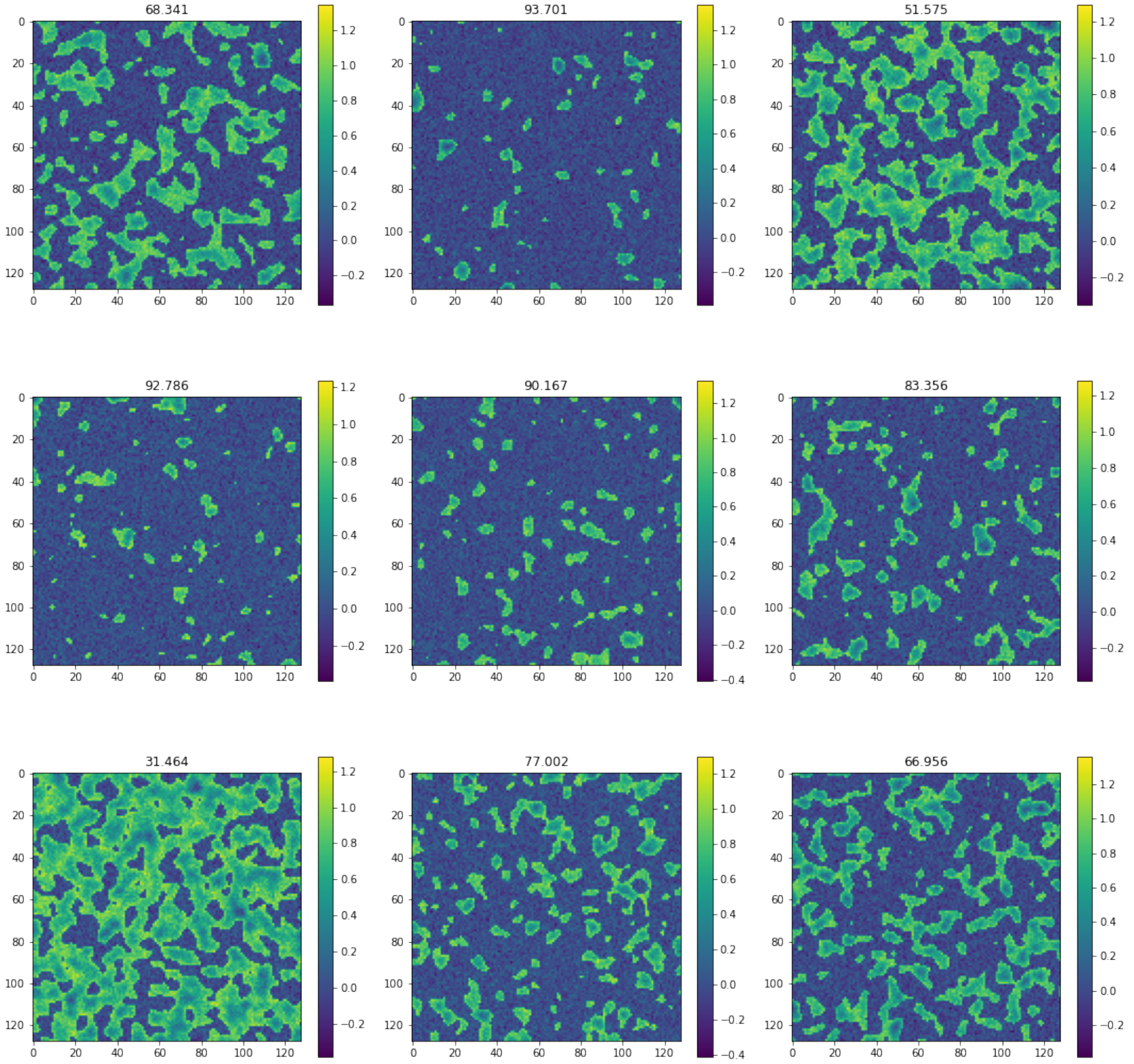


Figure 11.3: `ToyModel` simulations, no filter with noise. Ionization fractions are listed above each images.

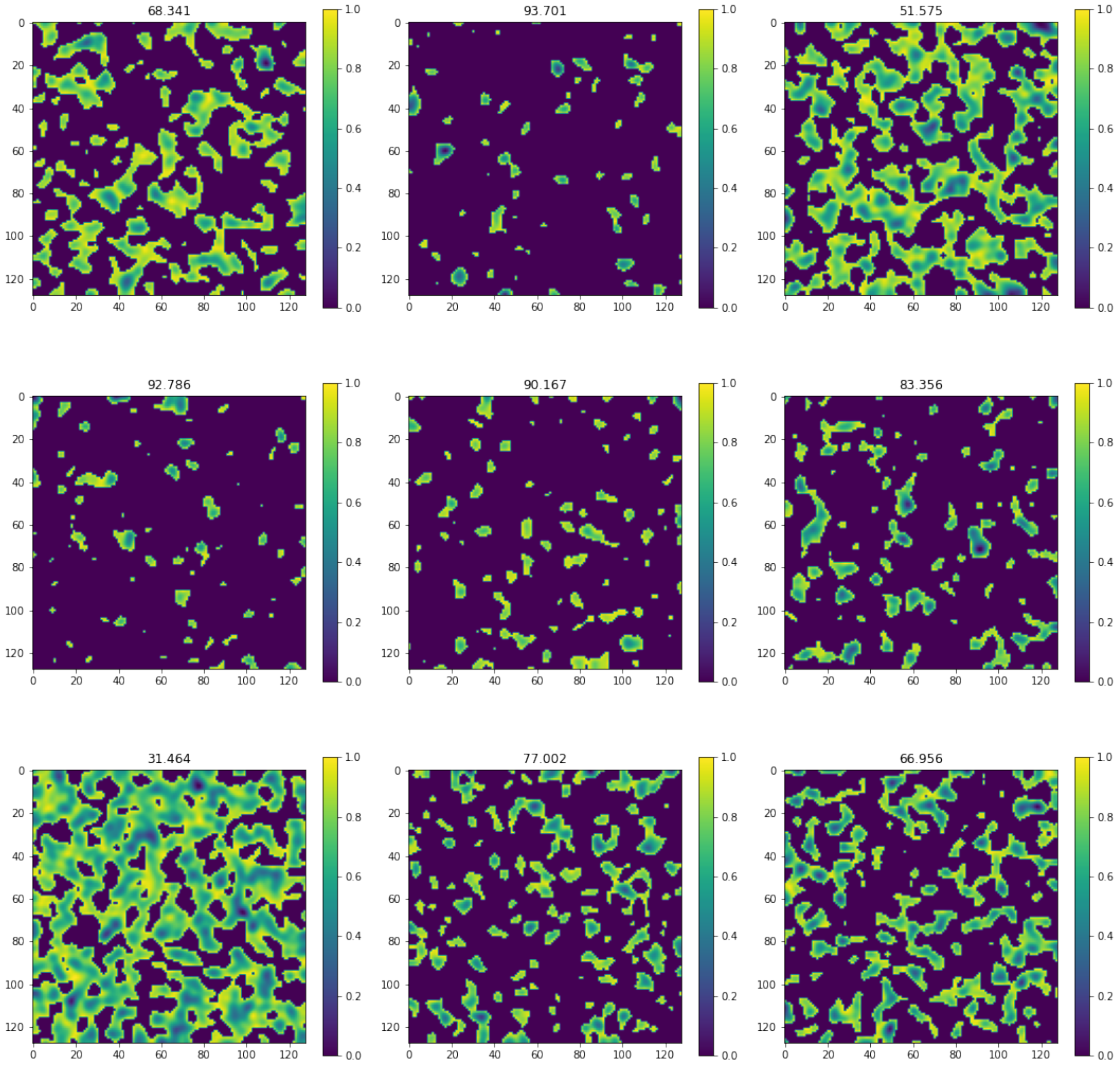


Figure 11.4: `ToyModel` simulations, no filter and no noise. Ionization fractions are listed above each images.

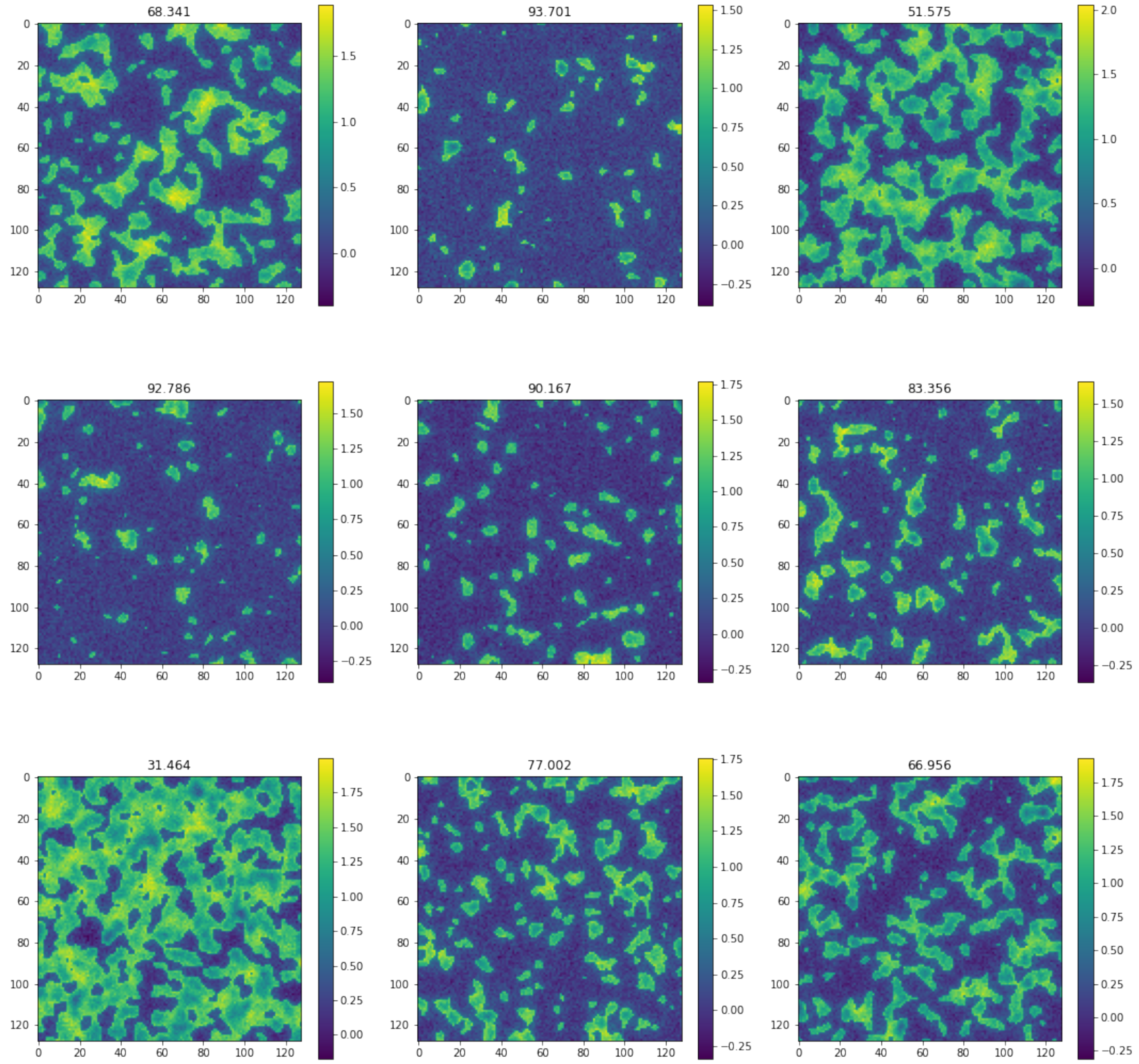


Figure 11.5: ToyModel simulations, filer plus noise. Ionization fractions are listed above each images.

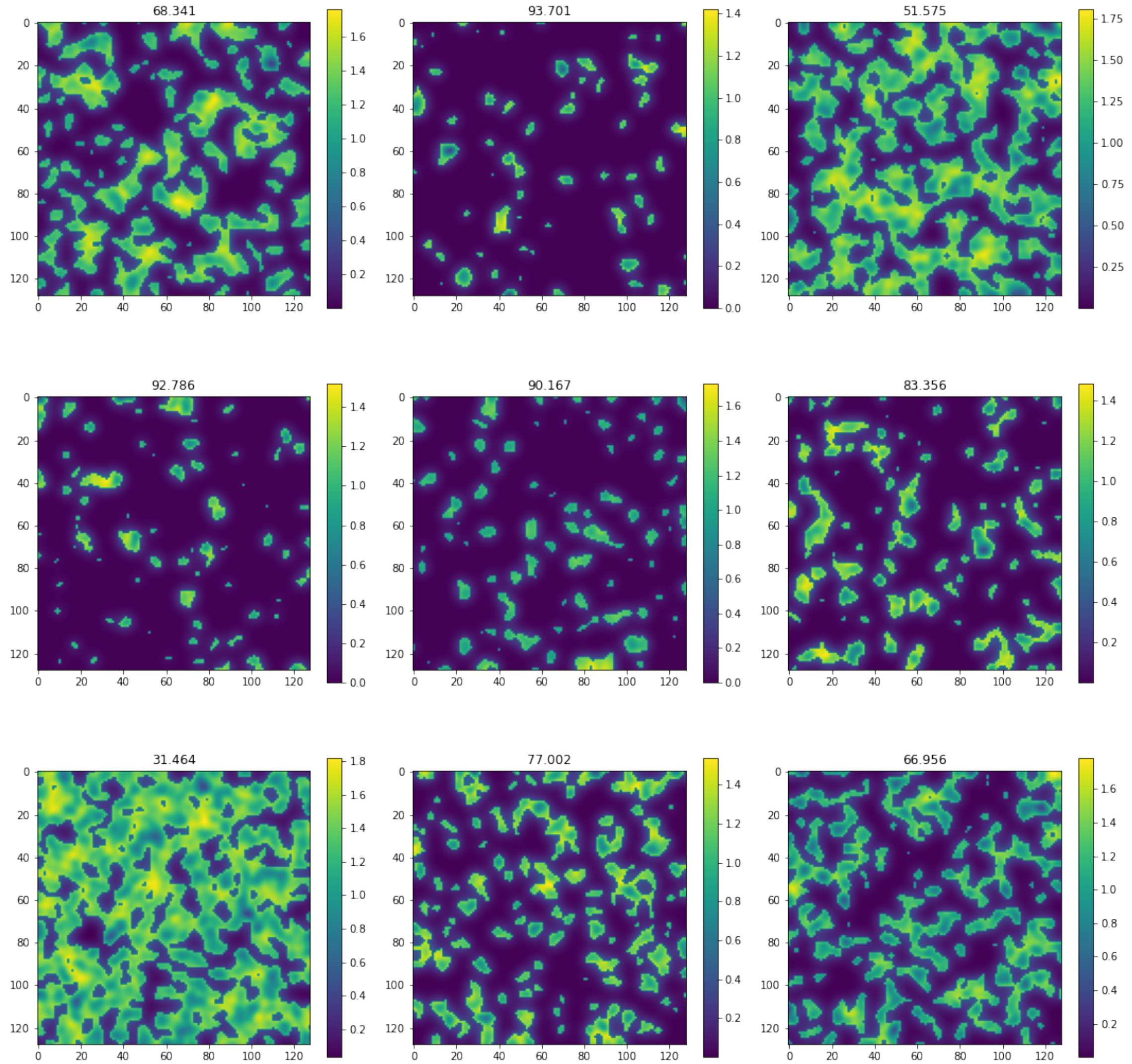


Figure 11.6: `ToyModel` simulations, filer plus noise. Ionization fractions are listed above each images.

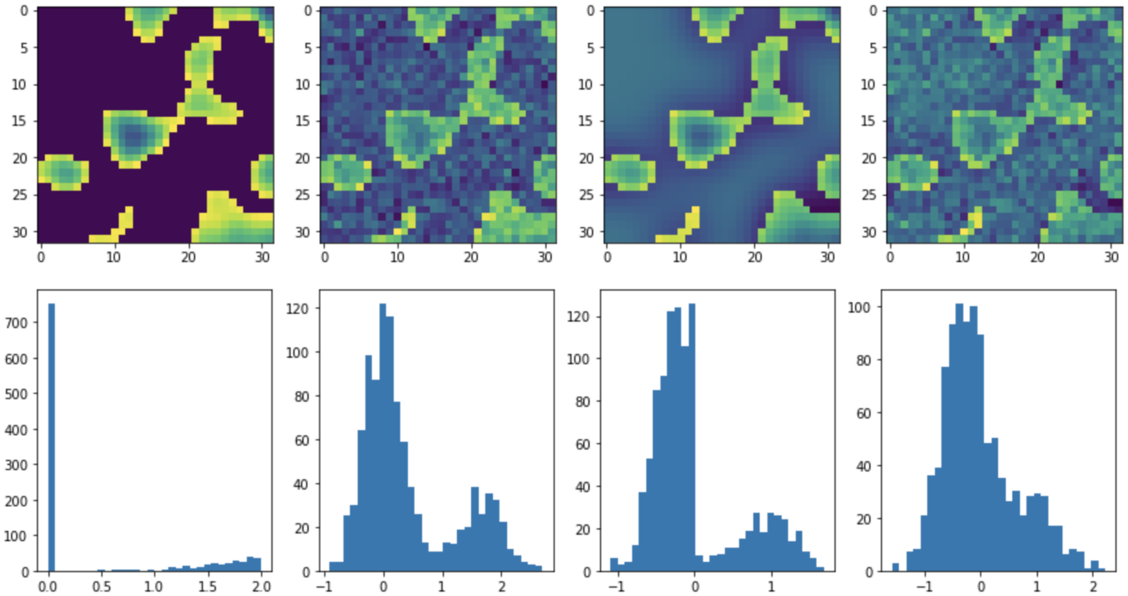


Figure 11.7: A visualization of the ToyModel as corruptions are added.

# CHAPTER 12

# EXTRACT OPTICAL DEPTH FROM SIMULATED DATA

## 12.1. Model Hyperparameter Optimization

To minimize the loss function while still remaining general, the optimal model parameters and hyperparameters must be selected. In addition to quantities used by the optimizer, such as the learning rate and the loss function, I also varied the architecture itself: I allowed the number of convolution layers to vary, as well as the number of neurons in the dense layers. In addition to these hyperparameter which were varied, the bottom half of Table 12.1 shows auxiliary parameters that were fixed as part of the optimization process. In general, these parameters did not have a significant impact on the overall performance of the network, and so I held them fixed while varying other quantities. Additionally, due to the relatively small value of $\tau$, I rescaled the labels for the testing and training datasets by a factor of 1000, which was then removed from predicted values. This led to faster convergence when training the networks, especially for loss functions such as MSE that do not normalize by the "true" input values.

Table 12.1 displays hyperparameters used as part of the grid search for the hyperparameter optimization, as well as parameters that were fixed. The top half of the table shows the hyperparameters that were allowed to vary. In particular, it includes various choices for the loss function. These were: relative square residual (RSR), mean square error (MSE), mean absolute error (MAE), mean absolute percentage error (MAPE), and mean squared logarithmic error (MSLE). Mathematically, RSR can be expressed as:

$$\mathrm{RSR} = \left( \frac{y_{\mathrm{true}} - y_{\mathrm{pred}}}{y_{\mathrm{true}}} \right)^2 \tag{12.1}$$

In order to evaluate the best overall network, I used two main criteria. The first one used was the value of the loss function: given a fully trained network, I examined the average value of the loss function across the validation data. There were several models that performed

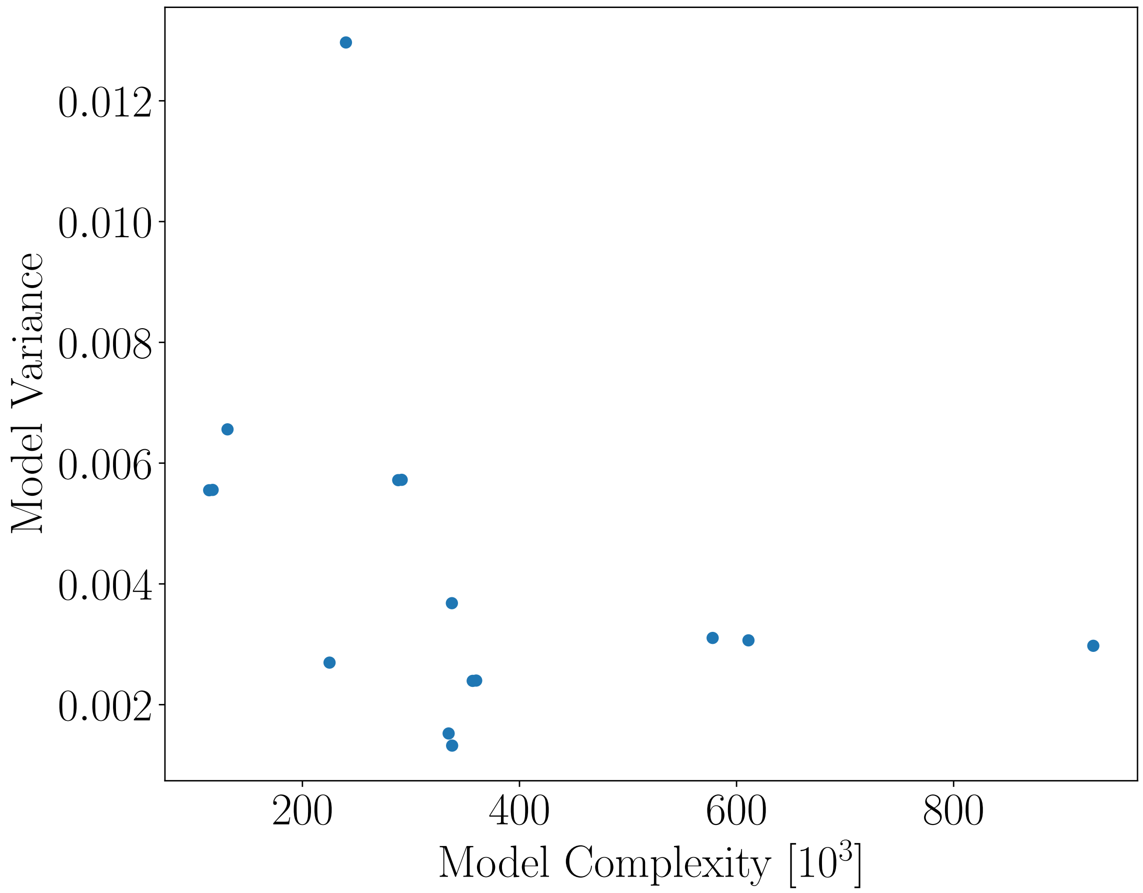


Figure 12.1: In order to select the best possible model from the parameter tuning, I evaluated 15 different models on 10 different test data and calculated the variance of the error for each of thee 15 models. I essentially picked the best model by prioritising first the model with the smallest variance and then the lowest complexity (number of trainable parameters). This method was applied for models trained on both the "Full" data (shown here) as well as the "Cut" data.

| Parameter | Values |
|---|---|
| Learning Rate | [0.1, 0.01, 0.001, 0.0001] |
| Loss Function | [RSR, MSE, MAE, MAPE, MSLE] |
| Number of Convolution Layers | [3, 4, 5] |
| Convolution Filter Size | [125, 256] |
| Dense Layer | [(200-350, 200, 100, 20)] |
| Batch Size | 32 |
| Optimizer | Adam |
| Activation Function | ReLU |
| Dropout Rate | 20% |
| Metrics | [loss, validation loss] |

Table 12.1: A summary of the parameters used in the final models. Parameters in the top half of the table were included in the hyperparameter optimization. The different loss functions and ultimate best options are explained further in Section 12.1. Parameters in the bottom half of the table were not included in the hyperparameter optimization, and fixed to the values shown.

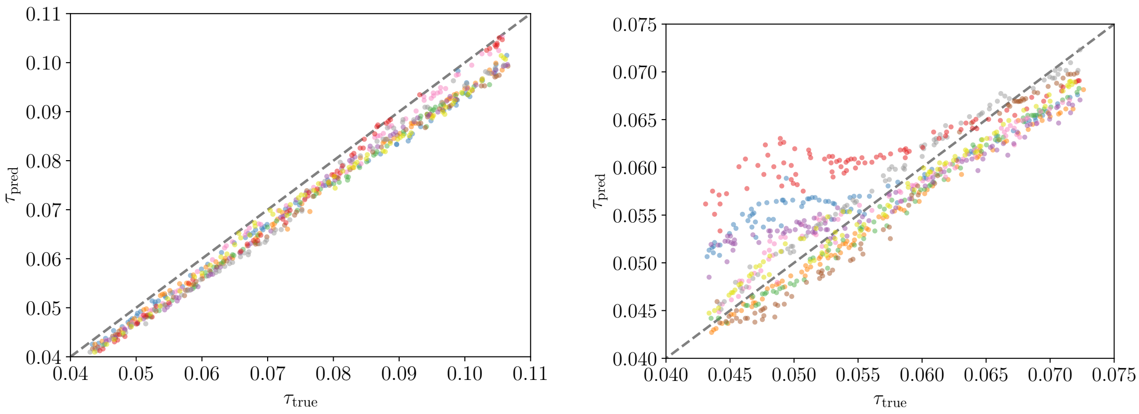


Figure 12.2: A visualization of the bias/variance trade-off for two different networks. These networks were small perturbations of the model used in La Plante and Ntampaka (2019). At left is a model with low variance but significant bias, and at right one with large heteroscedastic variance and high bias. As can be seen, the error in the predicted value is quite large, which suggests that additional complexity is needed to generate accurate prediction.

noticeably worse than the others. These poorly performing networks tended to be less complex, in the sense that they contained fewer trainable parameters. Above a particular number of parameters, many of the networks performed comparably in terms of the average loss value. This led to the second criterion used: the complexity of the network. I quantified the complexity by looking at the number of trainable parameters in the network. Thus, the "best" network chosen was the one that had the smallest number of trainable parameters while still performing well in terms of the average loss function. I performed the random search separately for the two different sets of input data, and found that slightly different network architectures yielded the best results.

Figure 12.1 shows model performance as a function of model complexity. The $x$-axis shows the number of trainable parameters, and the $y$-axis shows the variance of the loss function after performing 10-fold cross-validation. Networks with low complexity showed relatively high variance in their average loss function values, most likely indicating that they lacked sufficient flexibility to model the data accurately. Above a certain threshold of about 400,000 trainable parameters, the variance does not decrease significantly. This could indicate that there are insufficient training data to adequately make use of the increase in model complexity, or that the additional number of parameters is not be necessary to accurately capture the behavior of the input data.

Table 7.1 details the final architectures of our networks arrived at by this hyperparameter optimization. I chose the model that showed the smallest variance as the "best" for the purposes of evaluating. I did this for both the "Full" and "Cut" datasets, which yielded slightly different network architectures. From left to right, the columns show the architectures of the model in La Plante and Ntampaka (2019), the model trained on the complete data, and the model that was trained on data where foreground-contaminated $k$-modes were removed from the data. Interestingly, both the "Full" and "Cut" networks are more complex than the model used in La Plante and Ntampaka (2019), but are slightly different from each other.

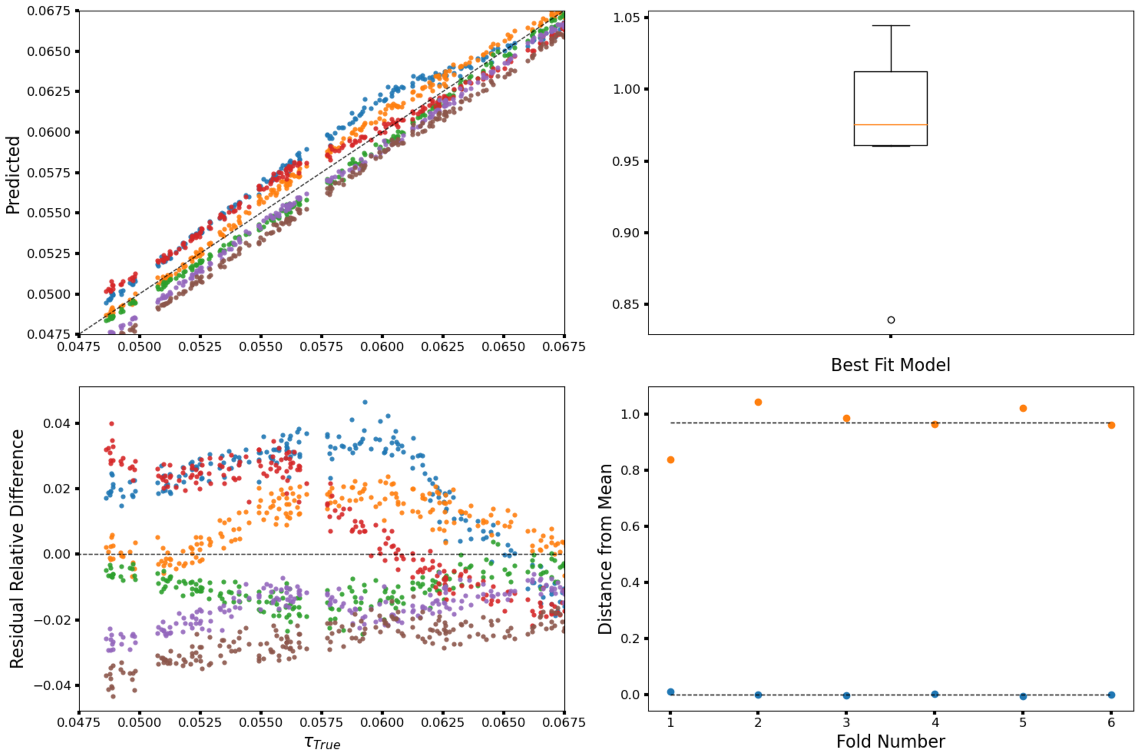


Figure 12.3: A visualization of the performance of the best-performing CNN on the "Full" dataset. Top left: a scatter plot of the "true" value $\tau_{\text{true}}$ versus the value predicted $\tau_{\text{predicted}}$ by the trained CNN on the validation data. The different colors and symbols correspond to the 10 different folds I used in our $k$-fold cross-validation (explained more in Sec. 12.1.1). Bottom left: the residual relative difference, defined as $(\tau_{\text{predicted}} - \tau_{\text{true}})/\tau_{\text{true}}$, which when squared is used as the loss function for training. Top right: the slope and intercept of a linear fit to the performance of a trained model. For an unbiased network, the intercept has a value of 0 and the slope has a value of 1 and the standard deviation for this value is 0.0267. Bottom right: a box plot of the slope (orange) of the trained network across the different folds. The average value, 0.9694, is nearly 1.0, though there is a significant low outlier whose slope is significantly less than 1 (the single point below the box).

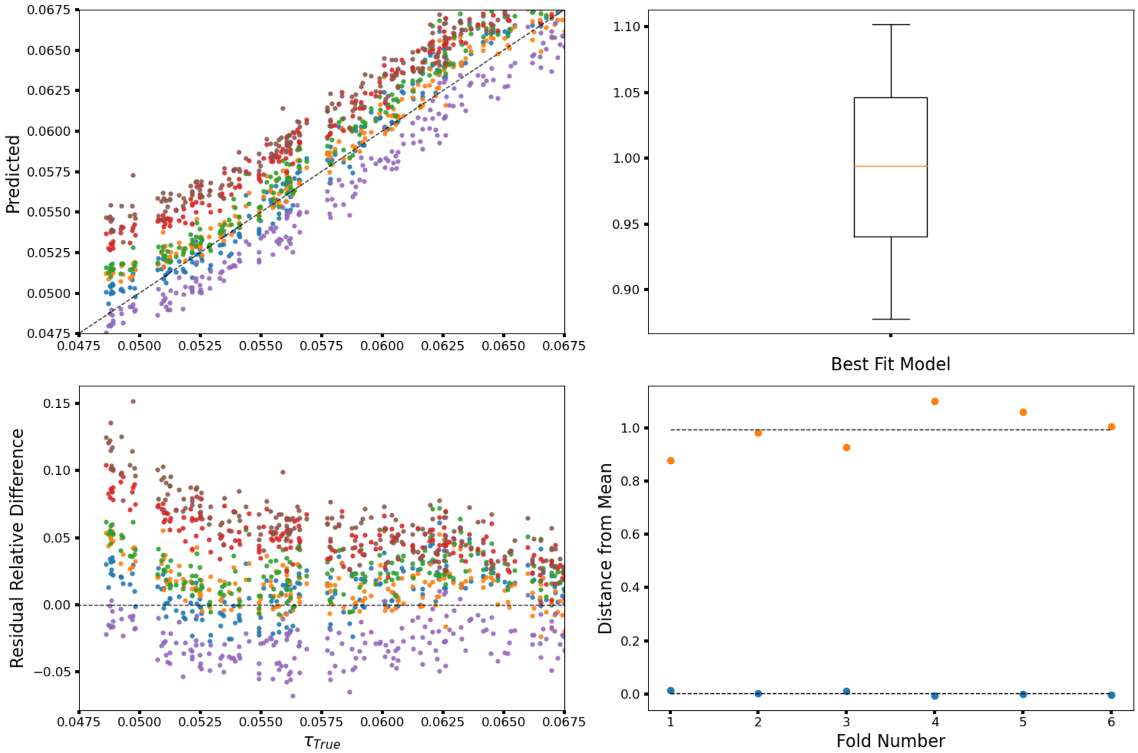


Figure 12.4: A visualization of the performance of the best-performing CNN on the “Cut” dataset. The panels are the same as in Figure 12.3. Note that for these models, the variance in the predictions is higher, and there is a slight bias in the slope where the average value is 0.9926 and the standard deviation for the slope value is 0.0308.

### 12.1.1. Results of Regression

In the results that follow, I treated $\tau$ as the only cosmological parameter of interest during regression. Although I made several attempts to regress on $\tau$ in conjunction with other details of the reionization history (such as the timing and duration), the results were most accurate when $\tau$ alone was used. As previously mentioned, CNNs were trained by minimizing the RSR loss function, show in Equation 12.1. Here I present the results of predicting $\tau$ for some realization.

Figures 12.3 and 12.4 describe the performance of the CNN regression for $\tau$ for both models trained on data without the foreground-contaminated and the model with the $k$-modes removed. These neural networks provide an estimate of $\tau$. The top left corner plot details the one-to-one relationship between the true and predicted tau values. The various colors and symbols represent the 10 different folds used in our $k$-fold cross-validation. The bottom left describes the relative difference between the true and predicted value. Note that when squared, this quantity is used as the loss function, written explicitly in Equation 12.1. The top right and bottom right plots describe 10 different slopes of the one-to-one lines observed in the top left plot. As discussed, these quantities provide an estimate of the bias from the bias-variance tradeoff. For an unbiased CNN, the slope has a value of 1 and the intercept has a value of 0. As can be seen in the different figures, the relationship between the predicted values of $\tau$ and the true values of $\tau$ are quite linear. More specifically, the one-to-one relation between the predicted values and true values show strong positive correlation. With both fully trained CNNs, I were able to recover values to better than $< 3.06\%$ percent precision.

The scatter plots in the top- and bottom-left panels show the full relationship between the true value and predicted value of $\tau$. In both the “Full” and “Cut” networks, there are slight systematic biases where small values of true $\tau$ are biased low, intermediate values of $\tau$ are biased high, and the highest values of $\tau$ are biased slightly low. However, as can be seen in the plots, the bias is typically smaller than the scatter in the values, and so the true values of $\tau$ are generally included as part of the scatter. The top- and bottom-right plots of the figures

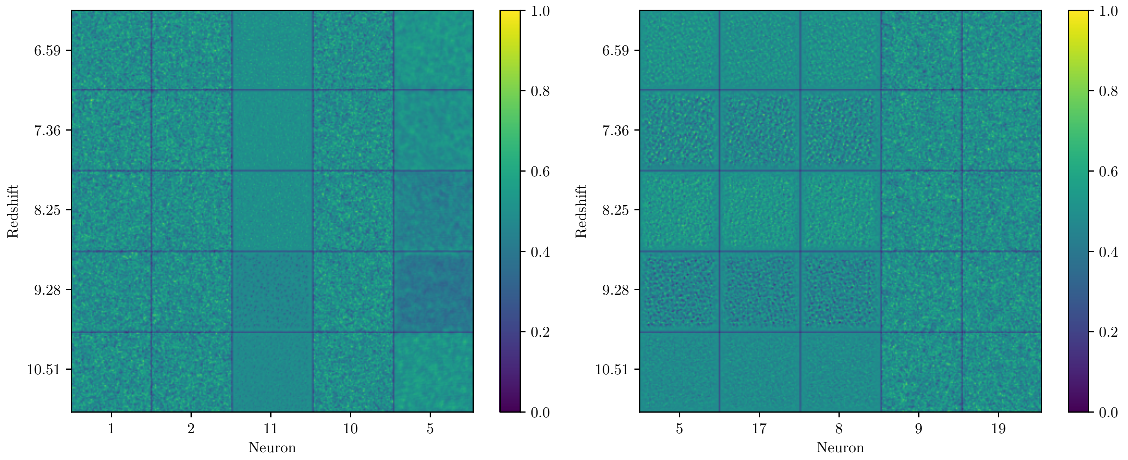


Figure 12.5: A visualization of the activation maximization technique for the 5 neurons most strongly connected to the final output neuron for the "Full" data (left) and the "Cut" data (right) networks from the best performing model fold. The neurons are organized by column and rank-ordered from left-to-right by the magnitude of the weight connecting them to the final neuron. The different rows represent redshift values corresponding to the different slices of the input data.

emphasize the bias component of the trained network: the value of the slope is a proxy for the multiplicative bias inherent in the network, and the intercept is an additive bias. In contrast to Figure 12.2, the bias of the lines is generally small, even across different folds. These results suggest that the networks described in Table 7.1 are sufficiently complex to capture the important features of the input data, and are correctly minimizing the variance and noise terms while providing an unbiased estimate of $\tau$.

As a way of determining the bias and variance of the predicted values as a function of the input value of $\tau$, I combine the predicted values of $\tau$ across the 10 folds of our input data. I then divide the true values of $\tau$ in the input dataset into ten discrete bins of equal width. Within each bin, I compute the mean value and the standard deviation. The mean value is a proxy for the bias (because a mean value different from the true value denotes a biased estimator) and the standard deviation encapsulates the variance in the output of the model predictions as well as the noise in the trained model. In general, I find that the variance tends to be several factors larger than the bias. While this result is not as ideal as having a truly unbiased estimator, it does mean that the variance by itself is a reasonable

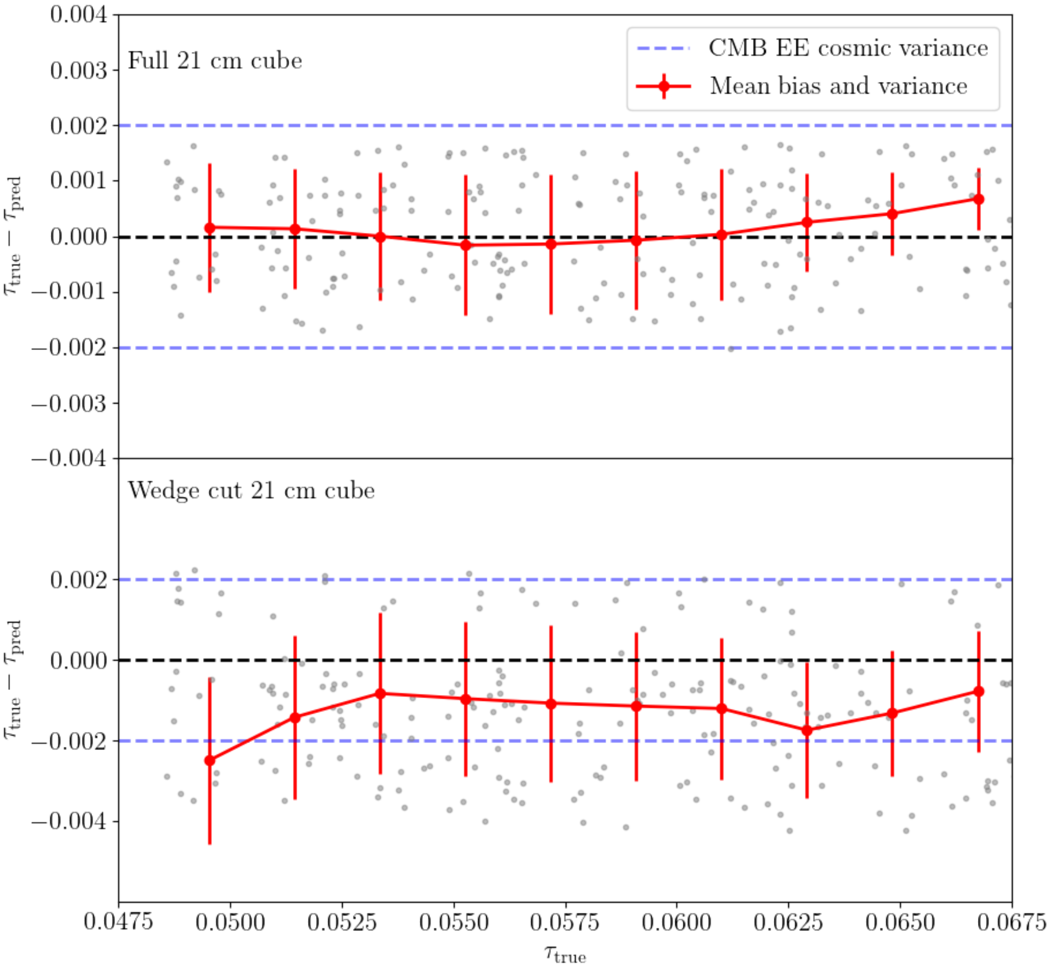


Figure 12.6: The accuracy with which our machine learning-based approach is able to determine the value of $\tau$. Note the values on the $y$-axis are absolute differences, rather than relative ones as in Figures 12.3 and 12.4. Also shown are uncertainties associated with sample-variance-limited measurements of $C_\ell^{EE}$ Reichardt (2016). As can be seen, our method produces results that are typically better than what can be obtained from the CMB alone, even for the case where foreground-contaminated $k$-modes have been removed from the dataset. Note that the error bars shown are empirically derived from the training data, and are not "proper" error bars in either the Bayesian or

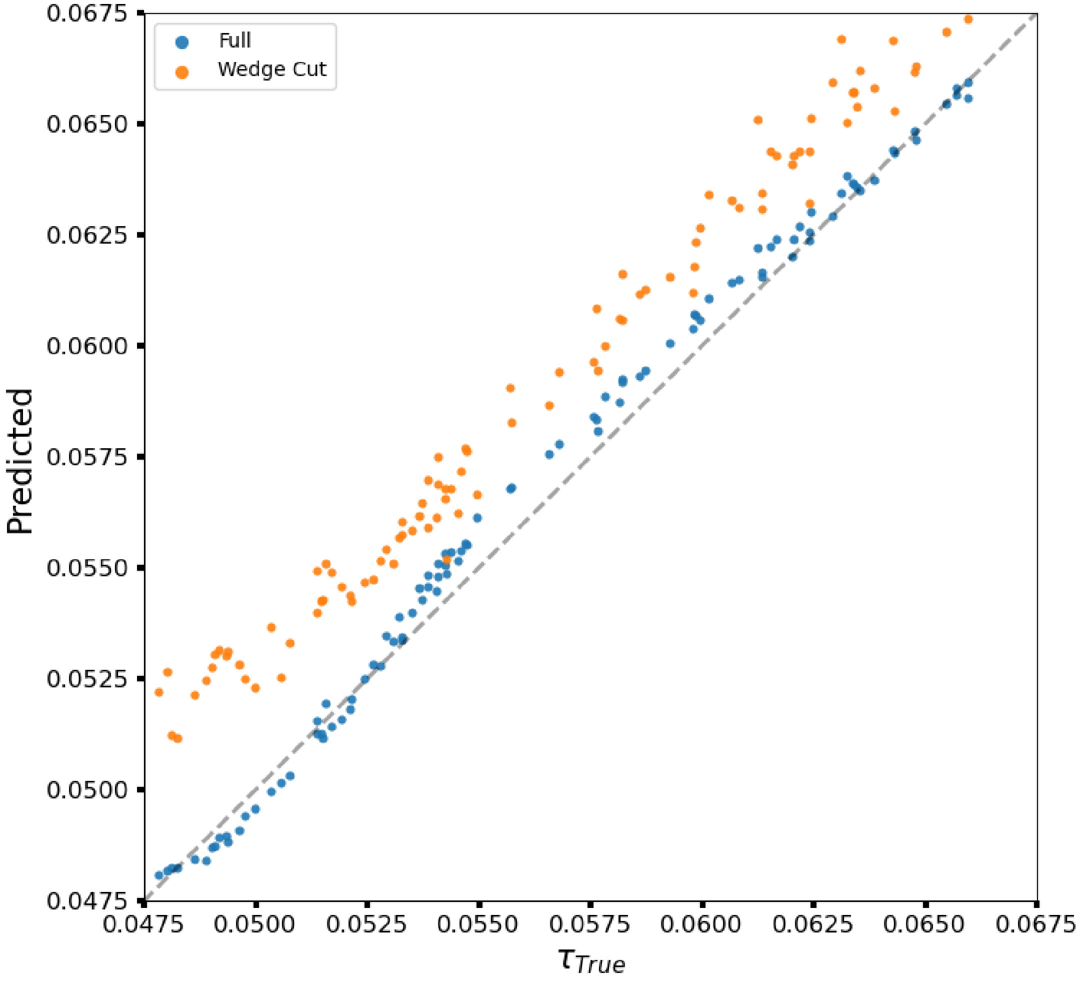


Figure 12.7: Extrapolation of the current models to make predictions on simulated data with a different cosmology generated using the 9-year results from *WMAP*. I are still able to extract $\tau$ for different cosmologies.

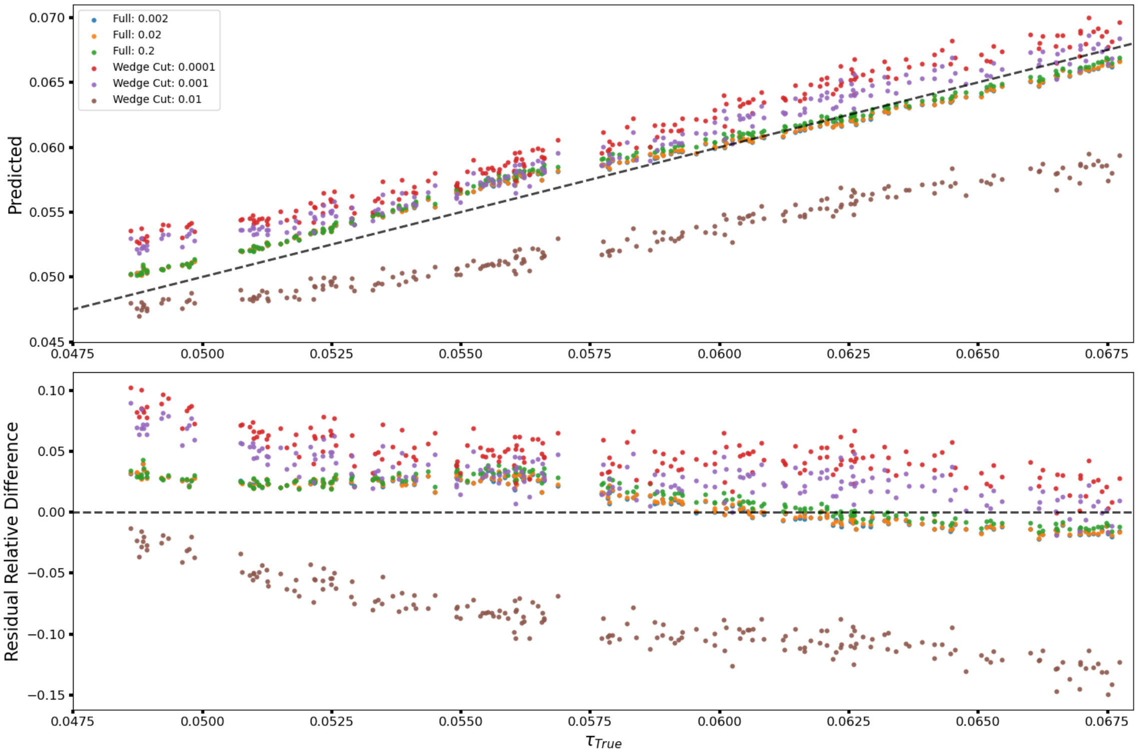


Figure 12.8: Noise investigation of simulated data generated from Planck 2018 cosmologies. The analysis is conducted by adding white Gaussian noise to input test image data with mean zero and variance set at 0.1, 0.01, and 0.001 of the typical variance of the images. The top panel shows the prediction performance of the CNNs trained either on full data or wedge cut data with no noise when predicting on noisy data. Models that are unbiased will make predictions that follow the one-to-one black dashed line. The bottom panel shows the relative model residuals.

approximation to the total error of the trained network.

Figure 12.6 shows the results of performing such an analysis for both the “Full” and “Cut” datasets. As noted above, the results are slightly biased as a function of $\tau$, and the direction of the bias changes as a function of the input value. However, as can also be seen, this bias tends to be smaller than the variance in the output values, and so the mean predicted value of $\tau$ tends to be consistent with the proper value within $1\sigma$ of the empirical standard deviation. As a point of comparison in Figure 12.6, I show the best-case error estimates that can be provided on $\tau$ from measurements of the CMB alone.

Interestingly, simply using the model from La Plante and Ntampaka (2019) without using `Keras-Tuner` yielded results which were severely biased low with significant variance. The trained model using data with the $k$-modes removed were worse that the model that contained all $k$-modes, 22% below the predicted value and 20% below the predicted value respectively as shown in Figure 12.2. Furthermore, the variance was lower for the model trained on data with the all of the modes, but higher on the model trained on filtered $k$-modes.

Note that in the discussion of variance in the bias-variance tradeoff, the variance component is not to be confused with the intrinsic variance within the data itself. Rather, it is the variability of the model’s ability to make predictions for a given set of input data points. In this context, I have data in the form of images. The variance in this context provides some insight on the spread of predictions and the overall performance of the model. More specifically, models that exhibit high variance for testing and validation data may by overfit to the input data, and do not generalize well on the data which it has not seen before. Put another way, the generalization gap between the training performance and the validation performance is large, and the model has high generalization error rates. A concept closely related to the variance of predicted values is the *bias*, which is the difference between the average prediction of our model and the correct value which I are trying to predict. In contrast to models with high variance, models with high bias are generally underfit to the input data. This underfitting includes the architecture of the neural network, where the number of tunable parameters or hidden layers may not be sufficient to capture the complexity of the application. This type of problem usually leads to high error on both training and validation/testing data.

In addition to quantifying the bias present in a trained network, I can also quantify the variance present. This is an empirical estimate of the variance, and does not take the place of properly propagated error estimates using, e.g., probabilistic layers to estimate posterior distributions of model parameters. At the same time, these variances can still provide some insight as to the errors that one might expect when applying the trained models to real data. Caution must be used, though, in interpreting these values as "one-sigma error bars". Rather, these are expected output values that combine the variance and noise terms of Equation 6.11, and implicitly include any idiosyncrasies inherent to the training data.

## 12.2. Discussion

### 12.2.1. Visualizing CNN Feature Extraction

When using image-based machine learning techniques such as CNNs, an interesting question is how to interpret the inner workings of the algorithm. One way to do this is to examine the effect on an input image of the different convolutional filters at each layer. Though the resulting "images" no longer represent information in the same space as the input images after the first input layer, they do contain information about which particular features of the map the CNN has learned to focus on. Alternatively, one can use the **activation maximization** technique Erhan et al. (2009) to visualize the important features in the input map directly, rather than using a partially processed image. In this approach, a specific neuron in a dense layer or convolution filter in a convolutional layer of a trained network is chosen. An initially random input image is gradually transformed into an image that maximizes the response of the chosen neuron or filter layer through gradient ascent. The resulting input images do not necessarily look like input images, but instead emphasize the features that are important for the machine to discriminate between different values or feature classes. Such an approach helps visualize which aspects of the the image are being used by the trained network to provide predictions, and complement other methods of visualizing CNN operations. To carry out the actual computation, I make use of the `keras-vis`package.

I applied the activation maximization technique to the fully trained networks developed in this work. This approach is sometimes employed for a classification problem, where the resulting images can be interpreted as the features that are most important for categorizing an imagine into a particular class. However, for a regression problem like the one at hand, these instead show features that lead to a large response of a particular neuron, typically one deep in the network. Though not as clear an indicator of the network's response as in a classification problem, the resulting images nevertheless contain features that the network has identified as ways to distinguish different output values. In both the "Full Modes" and "Cut Modes" architectures, there is a 20-neuron dense layer immediately before the final prediction neuron for the value of $\tau$. After training the network, I examine the magnitude of the weights connecting these 20 neurons and the output neuron. In general, the larger the magnitude of the weight connecting these neurons (positive or negative), the more influence the individual neuron has on the output prediction. For the two different networks, I identified the five neurons that had the largest magnitude connection to the output neuron. I then used the activation maximization technique to generate input images which would maximize the response of these neurons. The resulting images have the same dimensionality as the input images, and in particular have 30 "color channels" which correspond to the redshift layers of the input data.

Figure 12.5 shows the five most strongly connected neurons for various input redshift values for the "Full Modes" and "Cut Modes" networks, respectively. The columns show the maximal input for different neurons, rank-ordered from left-to-right by the magnitude of the weight connecting them with the final output neuron. The different rows correspond to the same redshift layers in the input data. When comparing the features between the different networks, several different trends emerge. First, for an individual neuron, the features that appear in the input images are similar for different redshifts. Because different input images are comparable between the different input redshifts, this similarity suggests that having many different filter layers initially is important. Multiple filter layers provide sufficient flexibility for identifying various features in the input maps, which are later condensed into features identified by hidden layers deeper into the network. Also of interest is the fact that generally, the features seem to be contrasts of large and small values at different scales, which roughly correspond to the size of individual ionized regions when viewing unprocessed input images. This result suggests that the CNN may be using the size of ionization bubbles at different redshifts to inform the overall value of $\tau$, though I caution that such a one-to-one mapping is not necessarily faithful to the actual operations being performed by the CNN.

When comparing the features identified in the "Full Modes" versus "Cut Modes" networks, there are several interesting differences. Of particular note is the features that the two different architectures treat as the "most important" in terms of informing the overall output value. The most important maps for the "Full" are qualitatively similar to the "Cut" network, but they are not the most important. Instead, the "Cut" network seems to be identifying features that are deviations from a background level (typically either higher, seen in the bright yellow regions, or lower, seen in the dark blue regions) rather than high-low variations near each other. Accordingly, these features appear non-Gaussian, perhaps emphasizing that the 21 cm maps are highly non-Gaussian (especially so with the large-scale contaminated modes removed). As such, the CNN appears to be making use of important information that is difficult to capture in the form of summary statistics, which bolsters the claim that CNNs can complement more traditional methods of analyzing image-based data.

### 12.2.2. Comparison with Limits on $\tau$ from Other Methods

There are a number of well-established techniques for measuring $\tau$. The current best constraints come from using CMB data, such as the all-sky temperature auto-power spectrum (denoted $C_\ell^{TT}$), as well as the large-angle auto-power spectrum of gradient-like E-modes (denoted $C_\ell^{EE}$). $C_\ell^{TT}$ is sensitive to the combination of parameters $A_S e^{-2\tau}$, where $A_S$ is the initial amplitude of scalar perturbations. This degeneracy can be partially broken by using CMB lensing maps. Alternatively, the low-$\ell$ portion of $C_\ell^{EE}$ follows a rough scaling of $C_{2\leq\ell\leq20}^{EE} \propto \tau^2$ Page et al. (2007), which provides an additional means of determining $\tau$. The Planck 2015 set of cosmological parameters Planck Collaboration et al. (2016) reports a value of $\tau = 0.066 \pm 0.016$, or a roughly 25% uncertainty. The Planck 2018 results Planck Collaboration et al. (2020) find a value of $\tau = 0.054 \pm 0.007$, about a 13% uncertainty. Other experiments, such as the EDGES high-frequency instrument, have further been able to place upper limits on the value of $\tau$ consistent with the measurements of Planck Monsalve et al. (2019). In principle, measurements of $C_\ell^{EE}$ can provide much tighter constraints on $\tau$ than $C_\ell^{TT}$. However, due to sample variance, these measurements cannot provide an uncertainty better than $\sigma_\tau \sim 0.002$ Reichardt (2016), which corresponds to a roughly 4% uncertainty. These measurements are projected to be made with future space-based CMB instruments, such as LiteBIRD Hazumi et al. (2012) and Pixie Kogut et al. (2011), which are not scheduled to fly until well into the next decade.

Figure 12.6 shows the accuracy with which our machine learning-based approach is able to determine the value of $\tau$, along with the sample variance possible from $C_\ell^{EE}$. As can be seen, the accuracy of our method is typically better than what can be obtained from the CMB alone, even for the case where foreground-contaminated $k$-modes have been removed from the dataset. Some important caveats remain, however. Importantly, the error bars shown in Figure 12.6 are empirically derived from the training data, and are not “proper” error bars in either the Bayesian or Frequentist sense. Nevertheless, the error bars are an indication that the value of $\tau$ inferred from this method is smaller than what is possible from the CMB alone, and is a promising tool to use in conjunction with more traditional methods. At the same time, further work is required to understand the impact the training data has on correctly inferring the value of $\tau$, either due to the quantity of training data or the semi-analytic model used to generate it. I plan to investigate these effects in future studies.

The results here are, of course, preliminary and should not be treated as a proper forecast of the potential accuracy of future 21 cm experiments. While I have included the effect of lost modes due to foreground contamination, I have not included the effect of other systematic errors in the 21 cm measurement on the result. In addition, the analysis here does not consider realistic instrument noise which varies with respect to the cosmological $k$-mode, which will naturally increase the error bars Pober et al. (2014). Working against this, our network only works on a single FoV of $\sim 10^\circ$, whereas HERA will sample approximately 10 such non-overlapping fields over some 1000 square degrees. I may also be able to use a fewer number of frequency channels to obtain comparable results, which will allow for generating multiple spectral windows to improve sensitivity. A forecast for more realistic systematic and sensitivity calculations will be presented in future work.

Another point of comparison for the ability to infer the value of $\tau$ is the analysis in Liu et al. (2016). The approach taken in that paper was to treat $\tau$ as a parameter to be inferred jointly with other CMB parameters, such as $\Omega_c$ and $\sigma_8$. In that case, the final marginalization over $\tau$ and other parameters yielded an uncertainly of $\sigma_\tau = 0.0016$, or about 3%. This is comparable to the uncertainty for our "Cut" model, and larger by roughly a factor of 50% compared to our "Full" model, as seen in Figure 12.6. Note that in our approach, the background cosmology was assumed to be fixed, and I do not attempt to jointly constrain the value of $\tau$ in concert with the other cosmological parameters. Performing a joint fit for other cosmological parameters is computationally intensive, and requires the use of cosmological emulators Kern et al. (2017) or other techniques to accelerate the forward-modeling component. In future work, I plan to use Bayesian neural networks (BNNs) to provide more robust distributions of the errors associated with machine learning modeling. Future directions may also include varying the background cosmology to understand the uncertainty associated with inferring $\tau$ using the 21 cm alone and how sensitive these measurements are to other parameters changing.

### 12.2.3. Testing the Effects of Different Cosmology and Noise

The networks above were trained on 21 cm data generated using Planck 2018 (Planck18) cosmology. From previous work showing only a weak dependence of the 21 cm power spectrum on cosmology (e.g., Kern et al. (2017)) I can similarly expect that the dependence of $\tau$ on cosmological parameters is weak. To provide an estimate of the kinds of errors which would occur in this analysis if the underlying cosmology is wrong, I generated a new test data set using Wilkinson Microwave Anisotropy Probe (WMAP) 9 year results Hinshaw et al. (2013). The most notable difference between these cosmologies ($\tau$ aside) is $\Omega_m$, which differs by $\sim$10%. I then used the networks trained on Planck18 to make predictions on the WMAP-9 data. In Figure 12.7, I show the results. For the case of the full data, the predictions are nearly as good as using the correct cosmology. Interestingly, the model trained on wedge cut data makes optical depth predictions that are biased high in this new cosmology, though the bias is only slightly larger in magnitude than was observed in Figure 12.6. Given that the tight correlation remains, it seems reasonable that a fuller analysis which properly marginalized over the cosmological parameter uncertainty would not increase the prediction errors unduly.

While the actual noise of 21 cm instruments will be quite complicated, I can gain some insight into the robustness of this method to noise by simply adding mean zero white Gaussian noise to the test image data and re-running the predictions. I chose the variance to be 0.1, 0.01, and 0.001 of the typical variance of the Planck18 cosmology images. Figure 12.8 shows the ability of the network to predict the optical depth at these different noise levels. The predictions follow the one-to-one line closely, except for the highest noise level of wedge-cut data, which shows a noticeable bias. However, even in this case, the clear correlation between $\tau_{true}$ and $\tau_{pred}$ remains, giving confidence that a network properly trained using the actual noise properties of the instrument would still be able to make accurate predictions.

# CHAPTER 13

# ERROR ESTIMATION USING MACHINE LEARNING

In this chapter I will detail my results while training Neural Networks using the `ToyModel`. First, I will give a general overview of this project, then I will broadly analyze the plot and finally I will conclude by summarizing my results.

## 13.1. Goals and Objectives

This project aims to answer a slightly different question. Instead of using mock temperature maps of thee 21 cm EoR signal to extract tau, I instead use mock images of ionized fields to extract the ionization fraction. This problem only looks at one redshift and is able to infer the ionization fraction of each simulated image. I do this by training three different Bayesian models. The goal is to better understand errors and tackle a more generalizable problem of estimating the ionization fraction because other research Zhou and La Plante (2021) suggested that while you can do well predicting for tau on a specific simulation package, the result does not generalize.

## 13.2. Building a Bayesian Neural Network

When training a ML model the first and most important task is identifying the problem being solved and selecting the correct model for that task. Once the general architecture of the model is established, that is, once a trained model produces predicted values that are closely aligned with the true values. Tuning techniques are an important step to implement while finding the "best" generalized model architecture for a problem. In this particular exercise I started with 5 different architectures varying in model complexity, number of training parameters, and trained them using some combination of the parameters in Table 13.1.

There three types of probabilistic models: approximately Bayesian models with Dropout layers turned on during prediction phase, fully Bayesian models with deterministic loss functions, and fully Bayesian models with probabilistic loss functions. All of them are trained using the same `ToyModel` data either without filtering, with filters applied, or with filters and noise at different levels applied.

## 13.3. Comparing Models

Using the Monte Carlo sampling method on models and then averaging over the predictions produces the one-to-one plots seen in this chapter. These plots allow me to easily visually compare the performance of the different models and then from there it becomes clear which model is a likely candidate. Things like training the models for longer periods of time should in theory improve the other predictions. At this stage I am not looking looking the one perfect model because there exists a space of models that can perform well.

For the predicted distribution plots, ideally I want the distributions of the predicted values to be centred at zero with a narrow spread because I am looking at the difference between the predicted and the true value. Throughout my analysis I noticed that the MC-Sample method spotlight the presence of divergence features in the distribution. I cannot see much skewed behavior but I do see the presence of kurtosis (positive and negative). Due to the presence of skewed and kurtosis, some of the distributions unsurprisingly have fat tails too.

The class of fat-tailed distributions includes those whose tails decay like a power law or as the log-normal. Compared to fat-tailed distributions, in the normal distribution of events that deviate from the mean of the distribution by five or more standard deviations have lower probability, meaning that in the normal distribution extreme events are less likely than those for fat-tailed distributions. In the case of a fat-tailed distribution the variance is undefined. A consequence, when estimating sigma based on a finite sample size would understate the true degree of predictive difficulty of the model.

An interesting question I wanted to answer was, with two models with the same architecture trained under the same conditions, how does the both computationally expensive and time consuming MC-sampling method compare to the direct sample from the `DistributionLambda` output layer. The `DistributionLambda` is minimially characterized

| Parameter | Values |
|---|---|
| Learning Rate | [ 0.0001] |
| Loss Function | [RSR, MSE, MAE, MAPE, NLL, ELBO] |
| Number of Flipout Convolution Layers | [2, 3, 4, 5, 6, 7] |
| Flipout Convolution Filter Size | [4-256] |
| Flipout Dense Layer | [100, 20, 1] |
| Batch Size | 5 |
| Optimizer | Adam |
| Activation Function | ReLU |
| Dropout Rate | [20%, 0%] |
| Metrics | [MSE, loss, validation loss] |

Table 13.1: A summary of the parameters used to determine the simplest and best performing model architecture.

by a function that returns a `tensorflow_probability.distributions.Distribution` distribution instance. Interestingly enough, the mean predicted of both methods are comparable. However, the estimate of the standard deviation of the distribution calculated using the MC-sample is larger than the `DistributionLambda` output layer estimate. This is because when I take some theoretical estimate of what the standard deviation will be and express it analytically (`DistributionLambda`) and then I take a sample of points in the distribution and calculate the standard deviation empirically (MC-sampling), the estimates are always larger.

There seems to be evidence that the standard deviation is dependent on the input value. This is evident in the one-to-one plots. The spread of the scatter plot does not follow the one-to-one line constantly. In my case the relationship seems to be linear.

## 13.4. Results

### No Filter or Noise Applied to Training Data

The worst performing model was trained using a deterministic loss function and the best performing models were the Dropout trained using a deterministic loss function and the fully Bayesian model trained using either NLL or ELBO loss function. The easiest metric used to determine the quality of the model's performance are the one-to-one plots illustrated

| La Plante & Ntampaka | Full Modes | Cut Modes |
|---|---|---|
| 16 3x3 Conv2D filters | 16 3x3 Conv2D filters | 16 3x3 Conv2D filters |
| BatchNormalization | BatchNormalization | BatchNormalization |
| 2x2 MaxPooling2D | 2x2 MaxPooling2D | 2x2 MaxPooling2D |
| 32 3x3 Conv2D filters | 32 3x3 Conv2D filters | 32 3x3 Conv2D filters |
| BatchNormalization | BatchNormalization | BatchNormalization |
| 2x2 MaxPooling2D | 2x2 MaxPooling2D | 2x2 MaxPooling2D |
| 64 3x3 Conv2D filters | 64 3x3 Conv2D filters | 64 3x3 Conv2D filters |
| BatchNormalization | BatchNormalization | BatchNormalization |
| 2x2 MaxPooling2D | 2x2 MaxPooling2D | 2x2 MaxPooling2D |
| — | 256 3x3 Conv2D filters | 128 3x3 Conv2D filters |
| — | BatchNormalization | BatchNormalization |
| — | 2x2 MaxPooling2D | 2x2 MaxPooling2D |
| — | — | 128 3x3 Conv2D filters |
| — | — | BatchNormalization |
| — | — | 2x2 MaxPooling2D |
| GlobalAvgPooling2D | GlobalAvgPooling2D | GlobalAvgPooling2D |
| — | 20% Dropout | 20% Dropout |
| — | 350 neurons FC | 250 neurons FC |
| 20% Dropout | 20% Dropout | 20% Dropout |
| 200 neurons FC | 200 neurons FC | 200 neurons FC |
| 20% Dropout | 20% Dropout | 20% Dropout |
| 100 neurons FC | 100 neurons FC | 100 neurons FC |
| 20% Dropout | 20% Dropout | 20% Dropout |
| 20 neurons FC | 20 neurons FC | 20 neurons FC |
| Output neuron | Output neuron | Output neuron |

Table 13.2: A summary of the final model with the number of parameters expressed using `Keras`, a high-level python deep learning library that uses a `TensorFlow`/`Theano` backend to do lower level calculations.

in Figure 13.1 and Figure 13.2.

### Filter Applied to Training Data

The models were trained using data with Gaussian Filters, Figure 11.2, to mimic instrument effects on observed data. Note, these are not realistic instrument effects on data. This was done using the `scipy`library to build a filter image of the same dimension as the input image smoothed with a Gaussian sigma of 3 pixels.

### Filter + Noise Applied to Training data

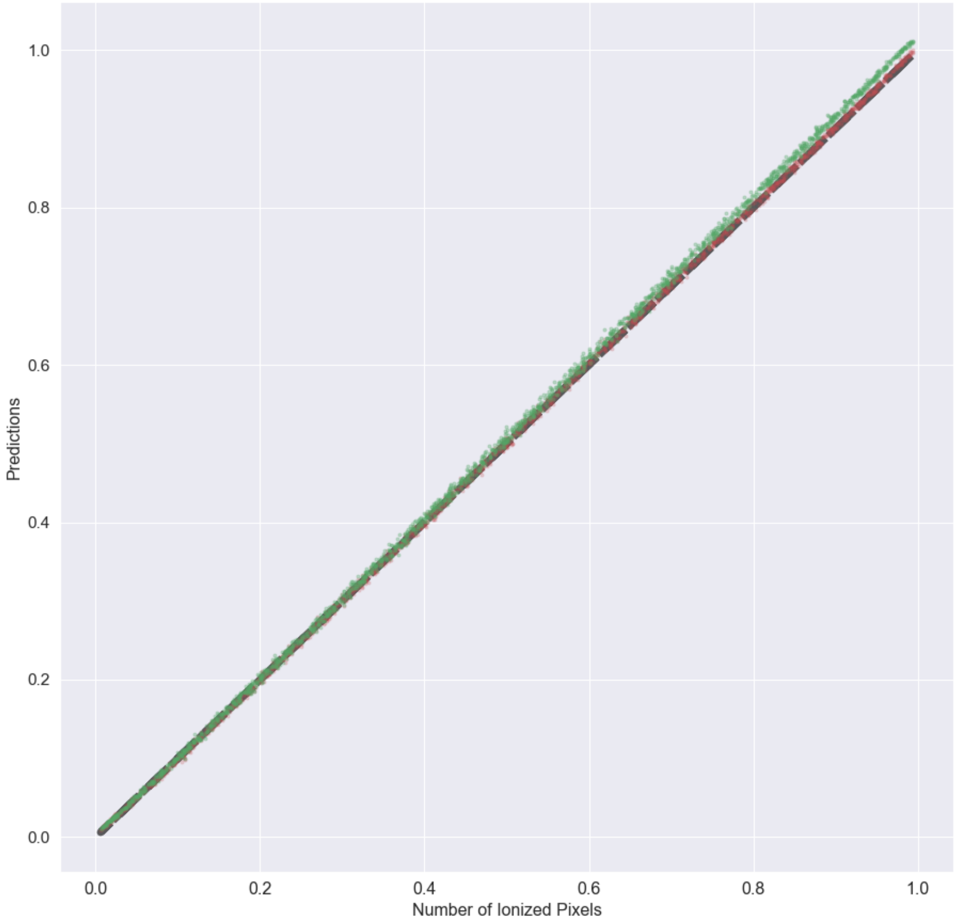
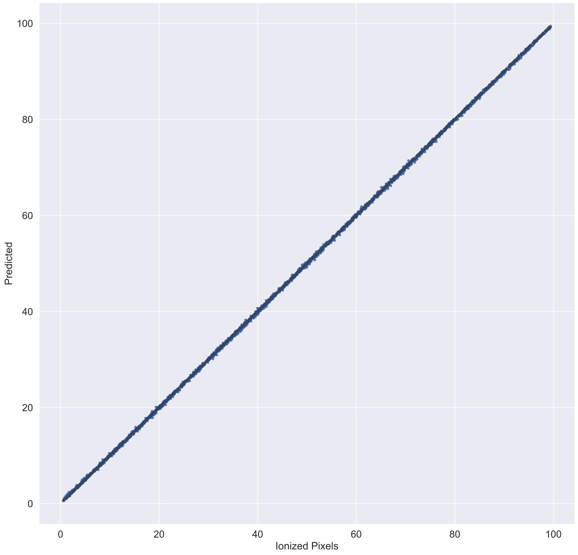


Figure 13.1: **Ideal Data**: One-to-one plots (black dashed line) comparing predicted number of ionized pixels to the true number of ionized pixels. The green and red are models trained using the ELBO and NLL loss function respectively.



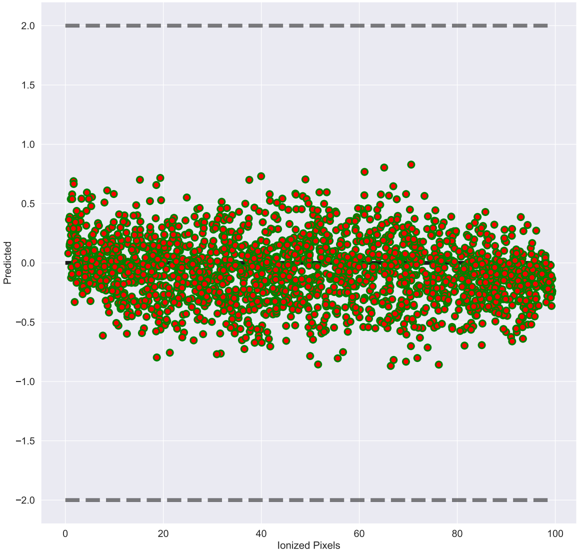
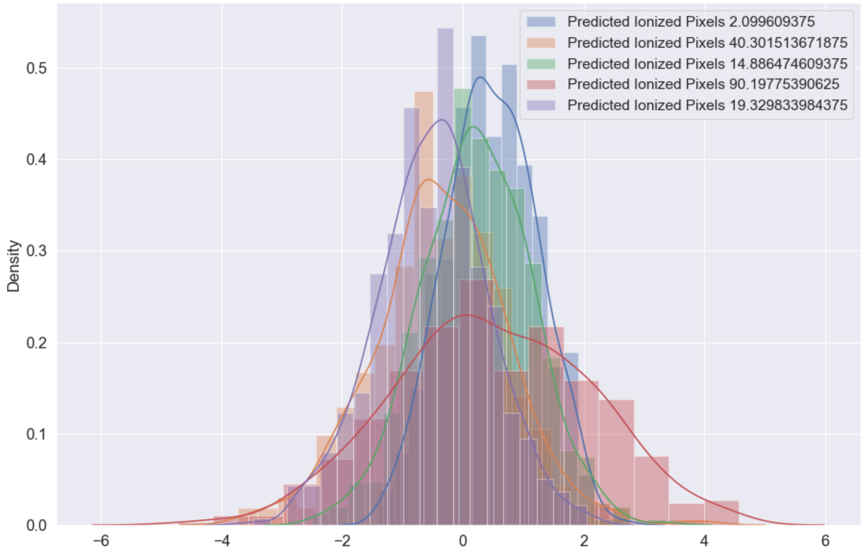


Figure 13.2: **Ideal Data**: The top plot are the results from the dropout model one-to-one predictions with a deterministic loss, MSE. The bottom plot is the error plot with negligible error bars.



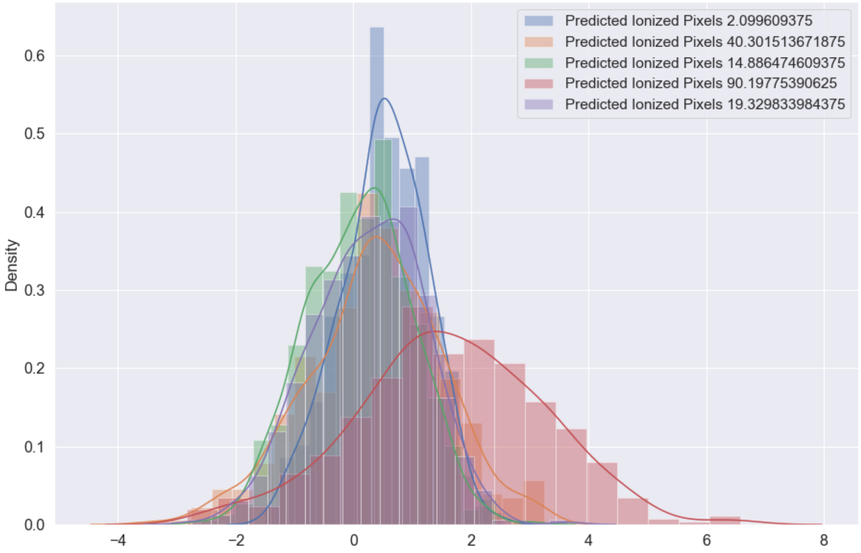


Figure 13.3: **Ideal Data**: Distribution of predicted of a few ionized pixels. Top and bottom plots are models trained using the ELBO and NLL loss function respectively.

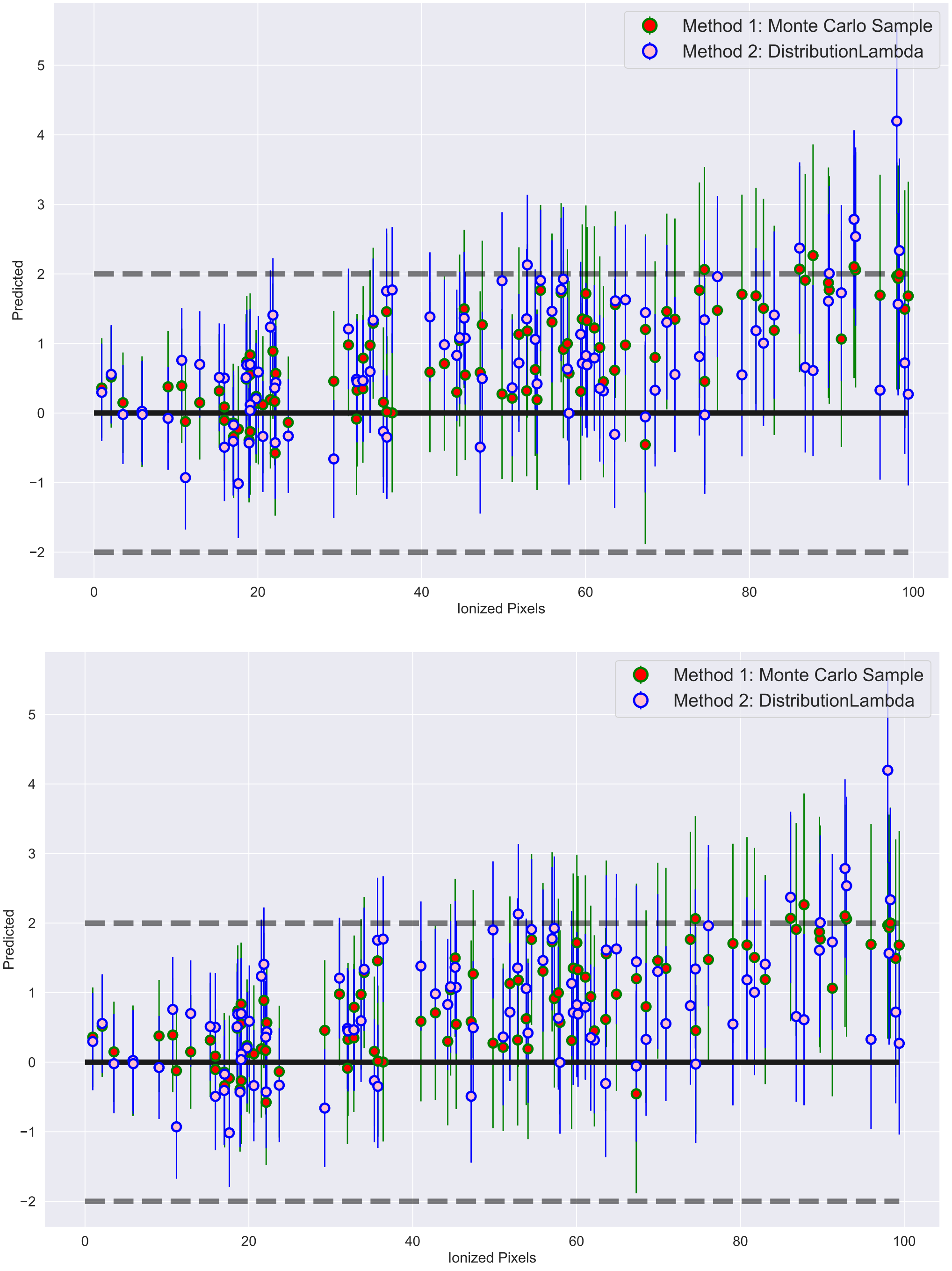
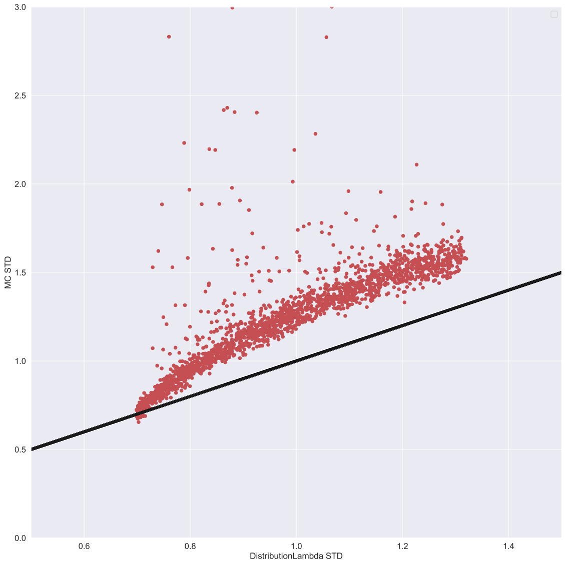


Figure 13.4: **Ideal Data**: Error of the model with error bars assigned to each mean prediction. Top and bottom plots are models trained using the ELBO and NLL loss function respectively.



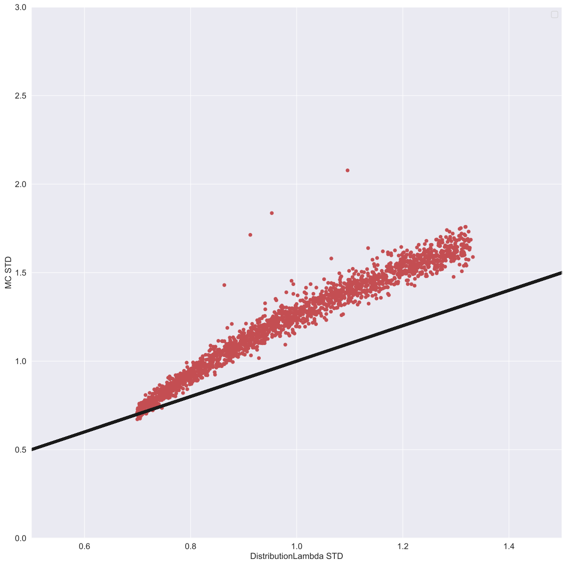


Figure 13.5: **Ideal Data**: Standard deviation comparison of MC-Sampling methods to direct prediction methods. Top and bottom plots are models trained using the ELBO and NLL loss function respectively.

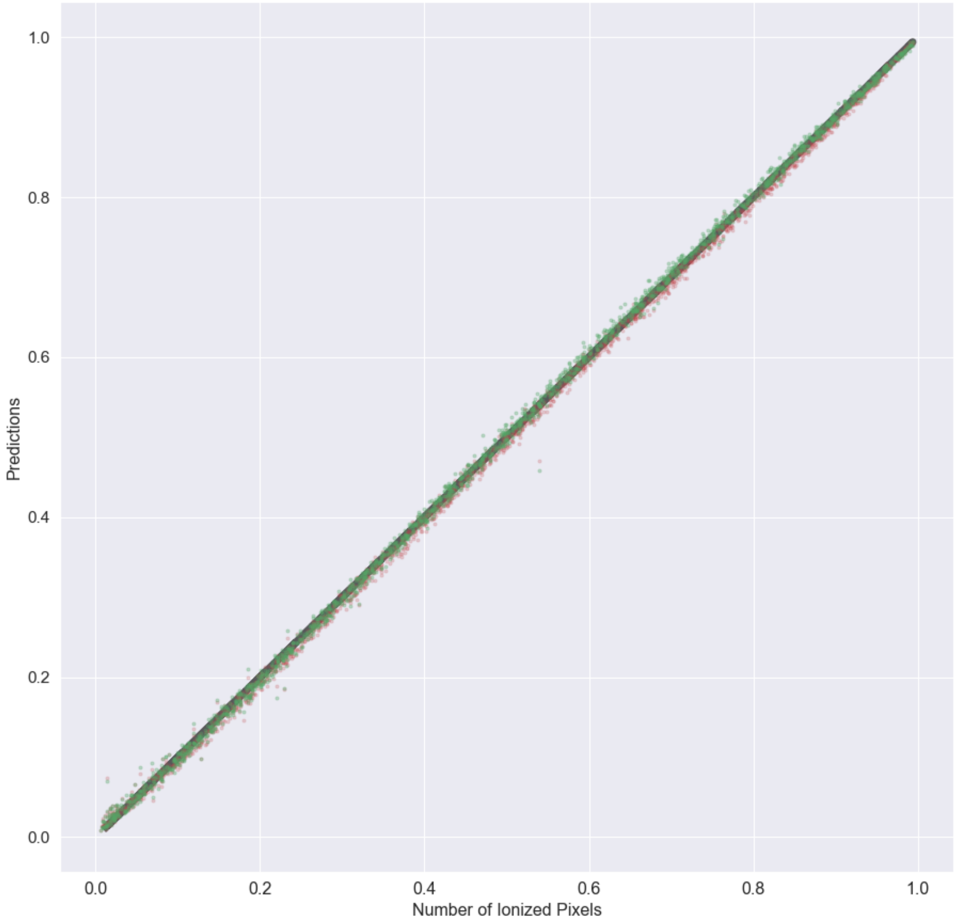
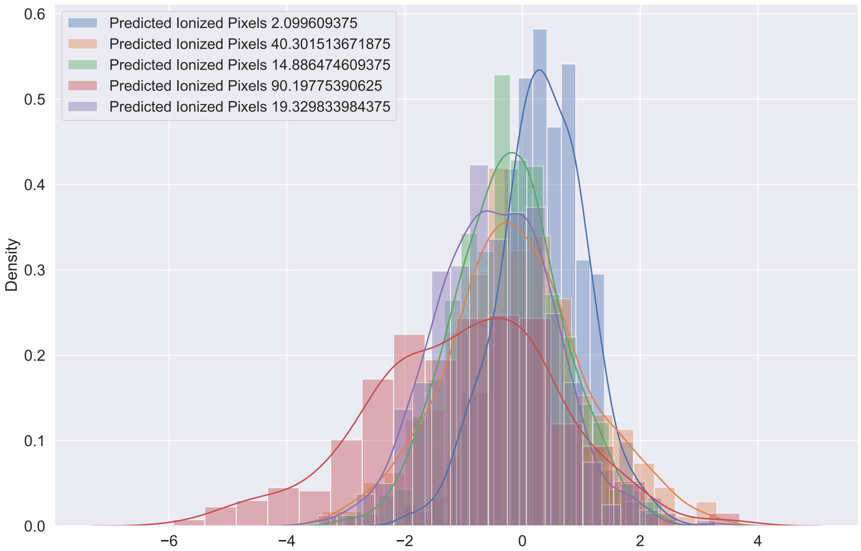


Figure 13.6: **Filtered Data**: One-to-one plots (black dashed line) comparing predicted number of ionized pixels to the true number of ionized pixels. The green and red are models trained using the ELBO and NLL loss function respectively.



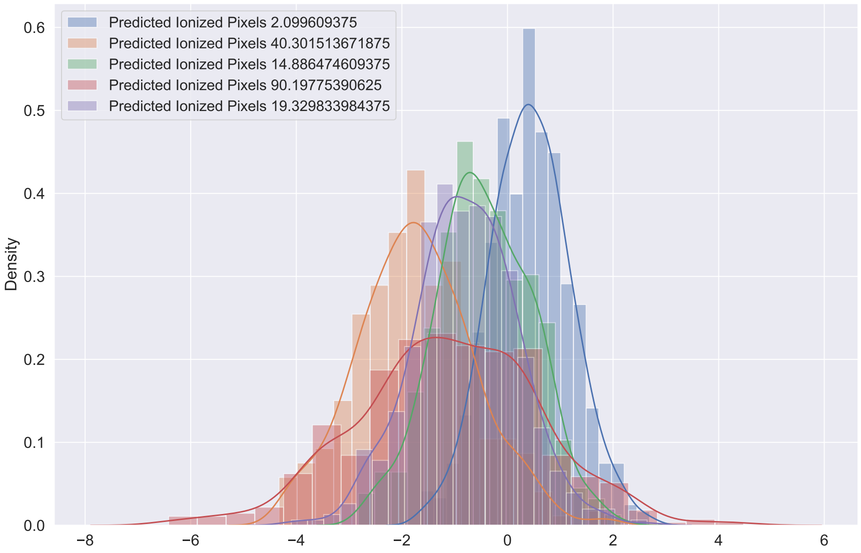


Figure 13.7: **Filtered Data**: Distribution of predicted of a few ionized pixels. Top and bottom plots are models trained using the ELBO and NLL loss function respectively.

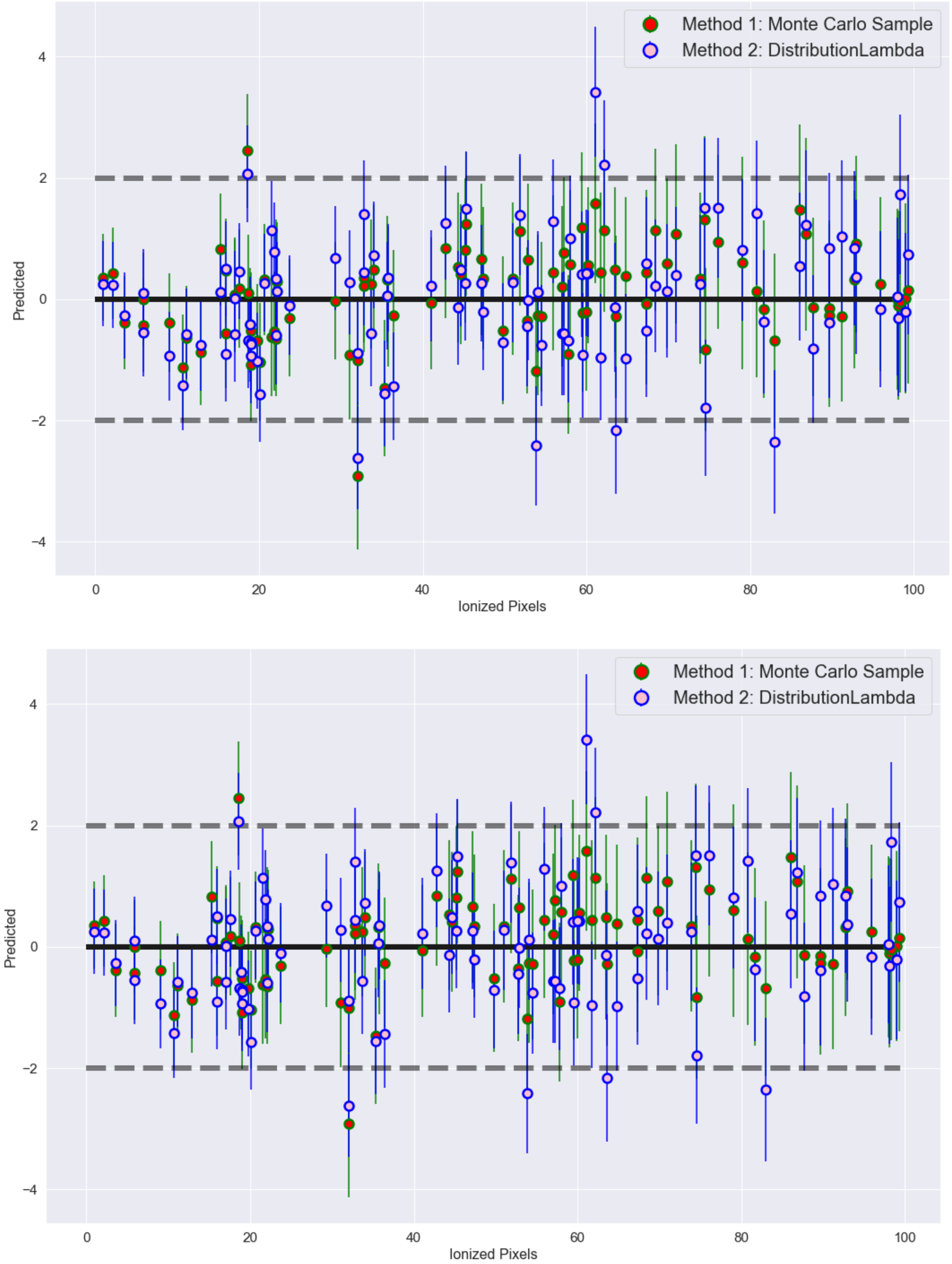


Figure 13.8: **Filtered Data**: Error of the model with error bars assigned to each mean prediction. Top and bottom plots are models trained using the ELBO and NLL loss function respectively.

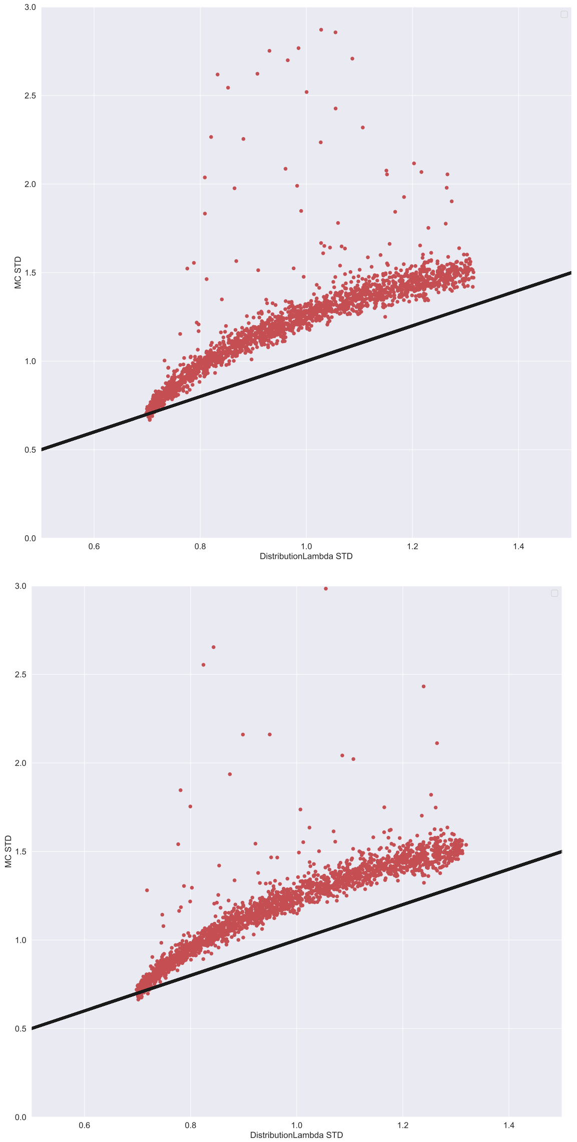


Figure 13.9: **Filtered Data**: Standard deviation comparison of MC-Sampling methods to direct prediction methods. Top and bottom plots are models trained using the ELBO and NLL loss function respectively.

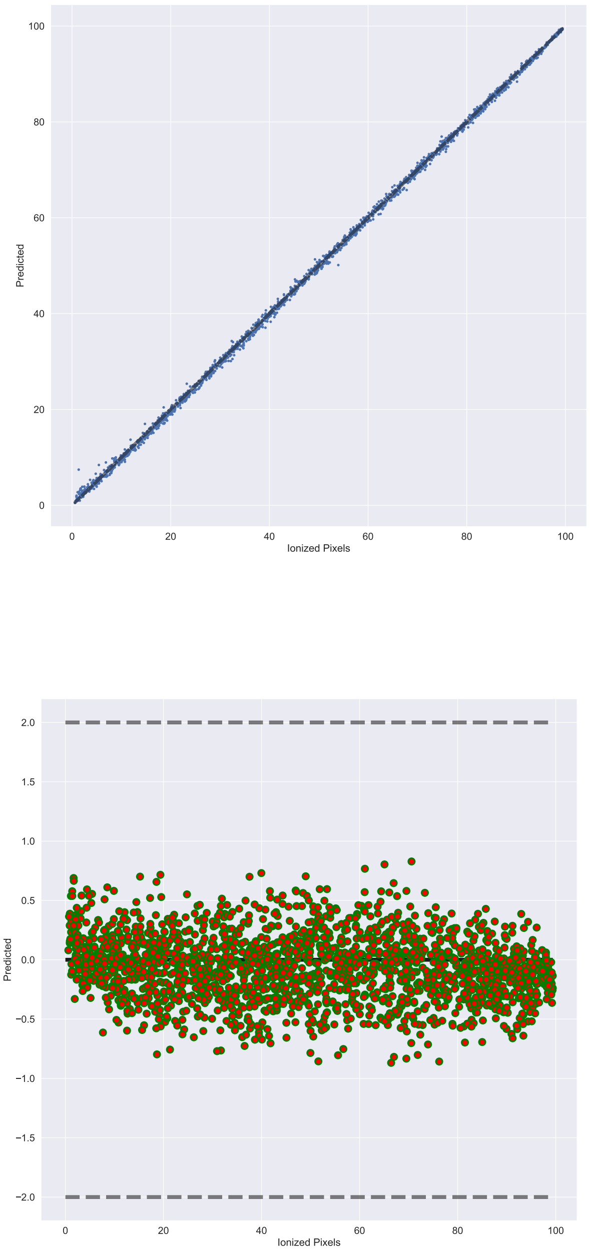


Figure 13.10: **Filtered Data**: The top plot are the results from the dropout model one-to-one predictions with a deterministic loss, MSE. The bottom plot is the error plot with negligible error bars.

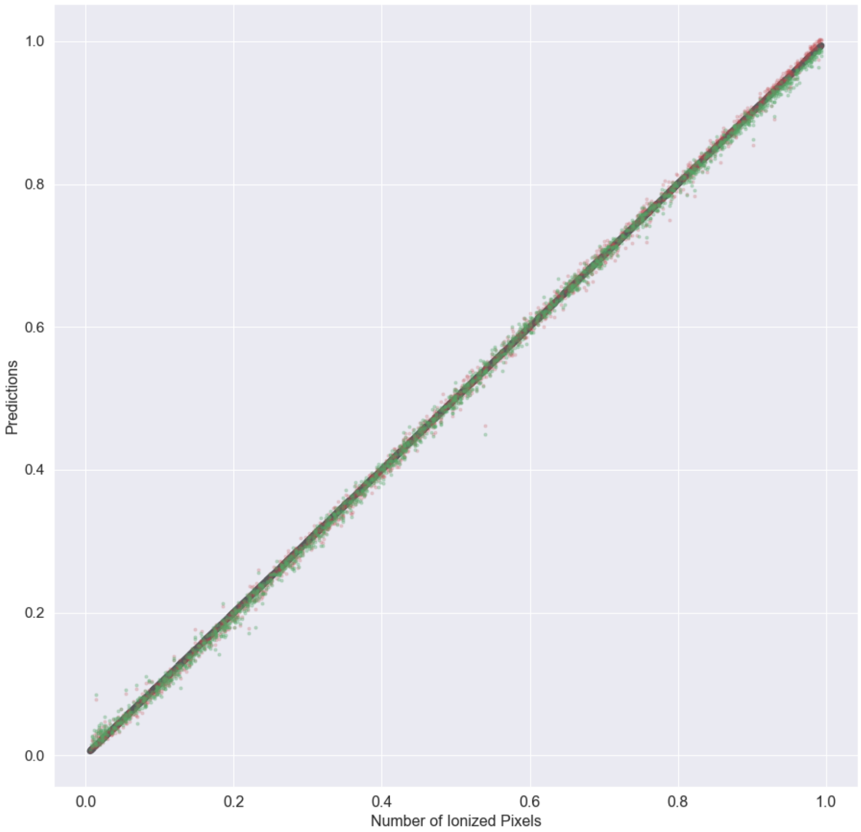


Figure 13.11: **Filter + Noise Data**: One-to-one plots (black dashed line) comparing predicted number of ionized pixels to the true number of ionized pixels. The green and red are models trained using the ELBO and NLL loss function respectively.

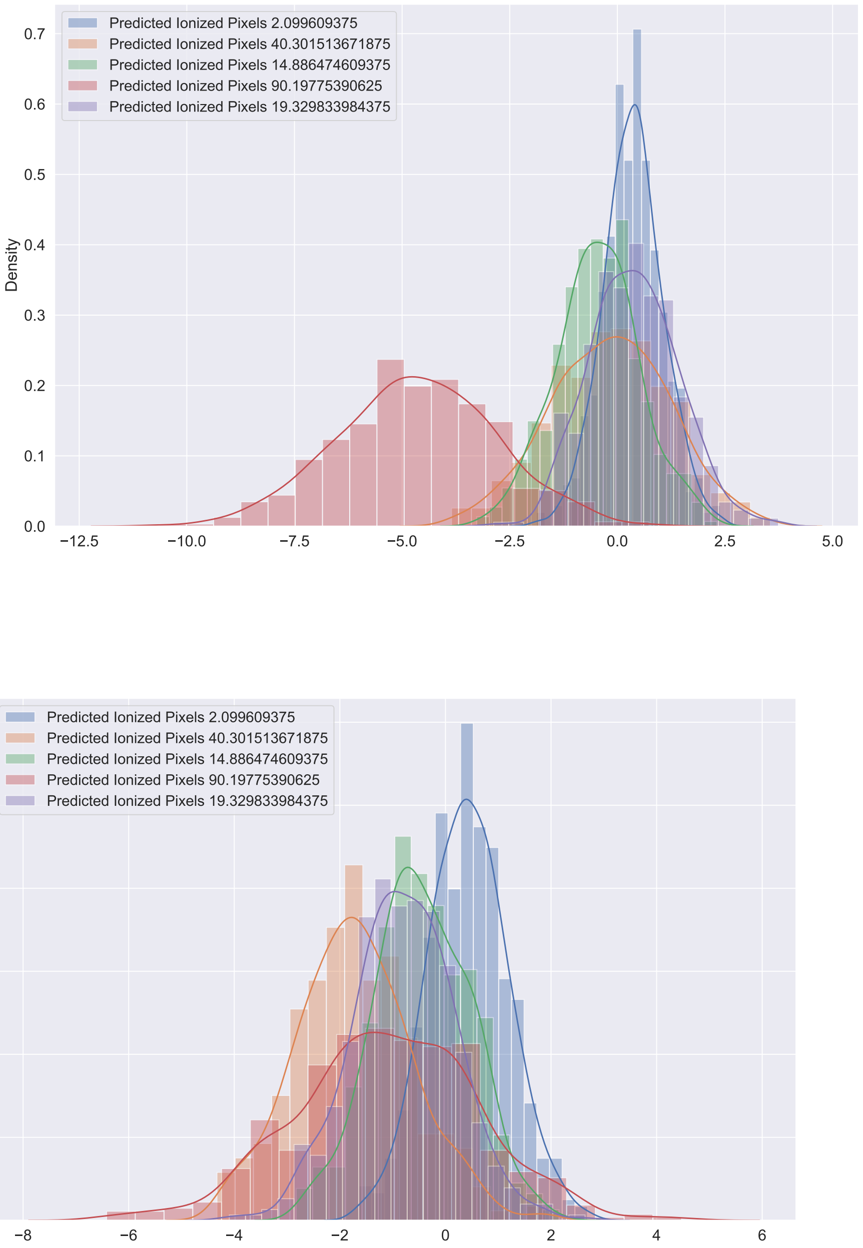


Figure 13.12: **Filter + Noise Data**: Distribution of predicted of a few ionized pixels. Top and bottom plots are models trained using the ELBO and NLL loss function respectively.

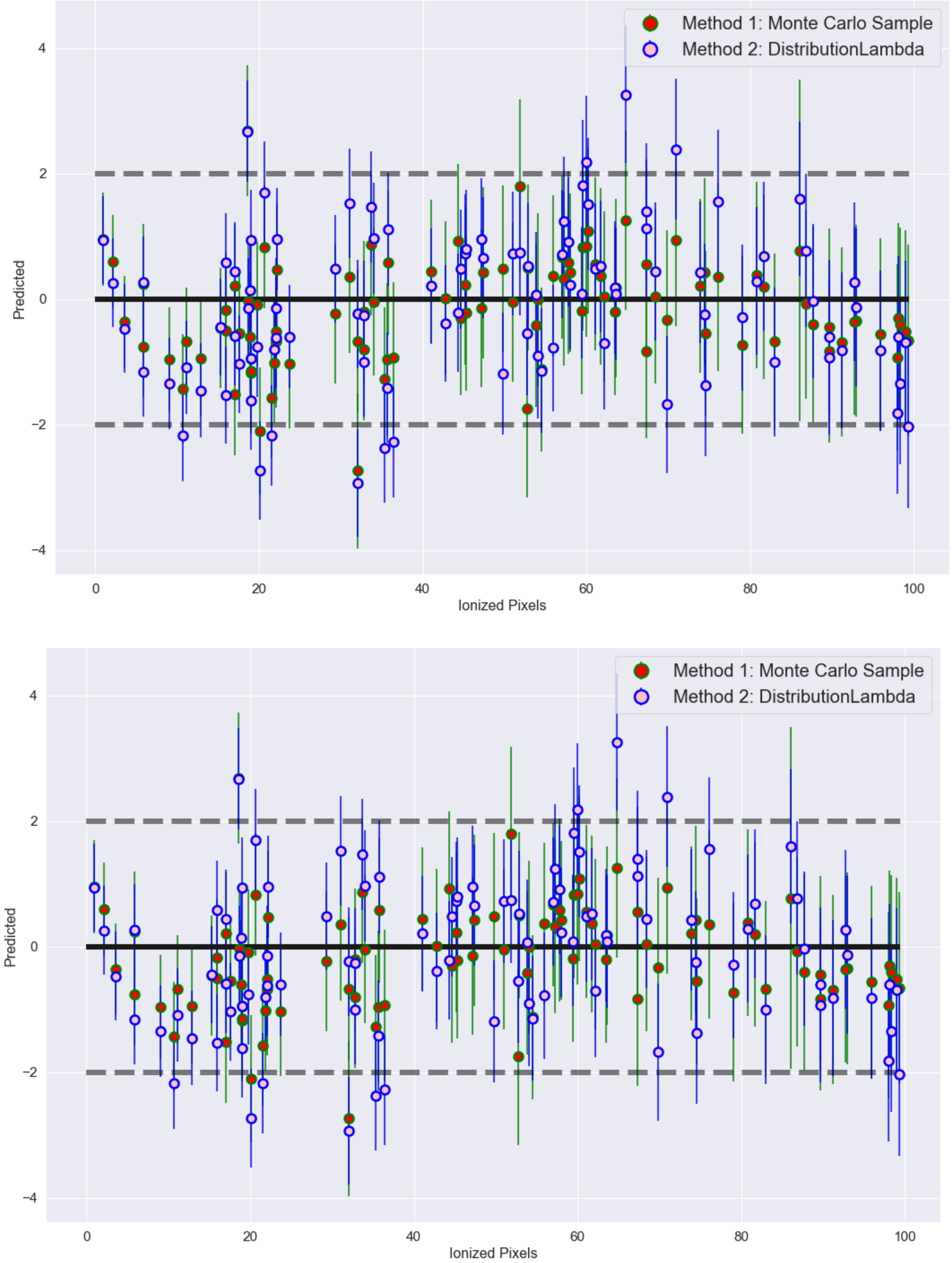


Figure 13.13: **Filter + Noise Data**: Error of the model with error bars assigned to each mean prediction. Top and bottom plots are models trained using the ELBO and NLL loss function respectively.

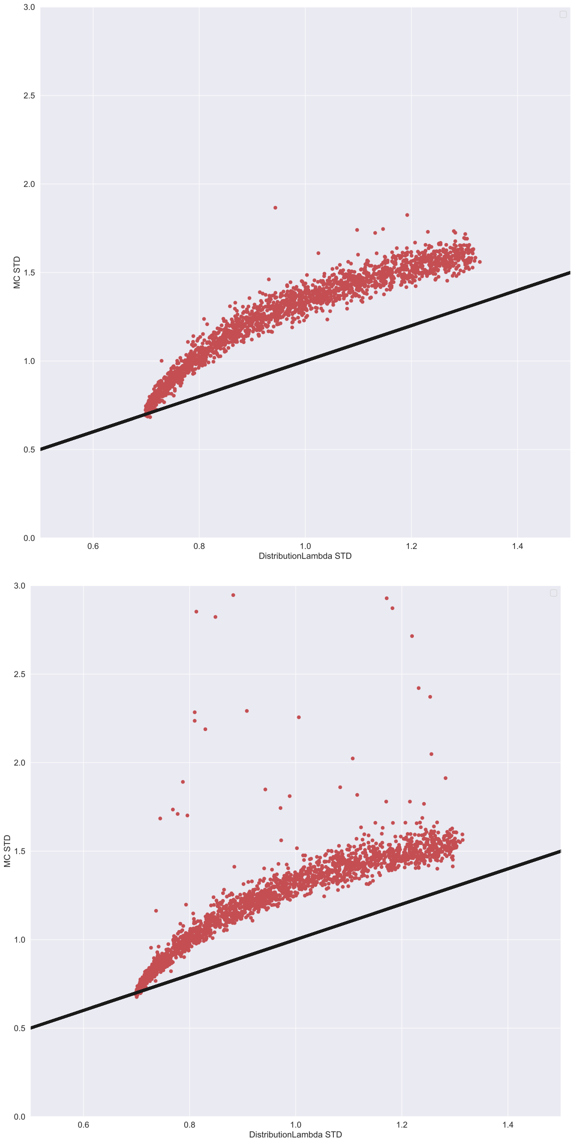
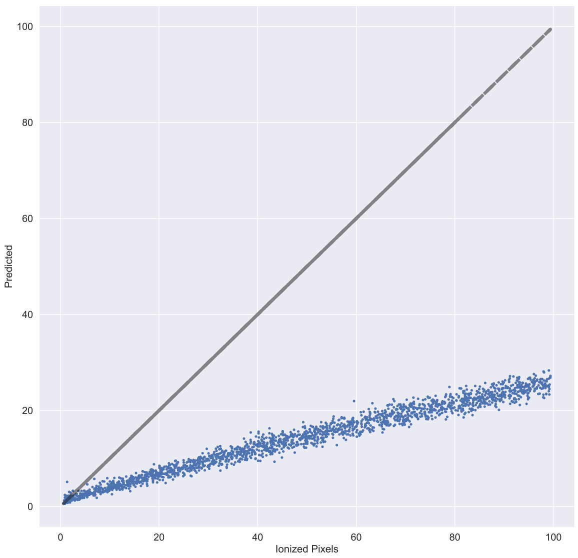


Figure 13.14: **Filter + Noise Data**: Standard deviation comparison of MC-Sampling methods to direct prediction methods. Top and bottom plots are models trained using the ELBO and NLL loss function respectively.



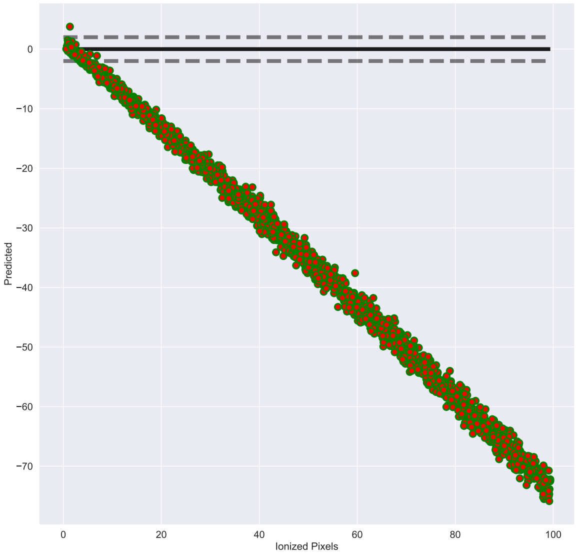


Figure 13.15: **Filter + Noise Data**: The top plot are the results from the dropout model one-to-one predictions with a deterministic loss, MSE. The bottom plot is the error plot with negligible error bars.

# CHAPTER 14

# CONCLUSION

I showed that I am able to train two different Convolution Neural Networks to extract $\tau$ from simulated data, one with the all Fourier $k$-modes included and the other without foreground-contaminated $k$-modes removed from the data. Through the use of hyperparameter optimization, I was able to find model architectures that are best suited for each application, and perform well over the full range of input data. I demonstrated that I can make accurate $\tau$ predictions using networks trained on both simulation types reasonably well. These simulated input images reflect the effects of the foreground avoidance strategy implemented by HERA as part of data processing. If some of these foreground modes can be used instead of discarded, the ultimate performance may be closer to the full data set than the one with the $k$-modes removed. I show that I was able to provide constraints on $\tau$ with a fractional error of 3.06% or better, which makes this approach competitive with low-$\ell$ observations of the CMB auto-power spectrum $C_\ell^{EE}$. Due to the fact that instruments capable of providing such a constraint are many years away, using 21 cm measurements may be able to provide a constraint on a shorter time line.

The approach outlined does not preclude using other methods of inferring $\tau$ from 21 cm data, such as using more traditional measurements like the 21 cm power spectrum. However, machine learning techniques such as that outline here are most powerful in conjunction with more traditional analyses, providing additional cross-checks of results inferred by other means. In future work, I made use of Bayesian neural networks (BNNs) to provide robust error estimates in addition to the predicted values for a particular CNN model. These novel analysis methods can supplement other established methods, and help bolster confidence in inferences made through other analysis techniques.

In the other project, the aim was to answer a slightly different question that involved using mock images of ionized fields of hydrogen to extract the ionization fraction of hydrogen by only looking at one redshift to infer the ionization fraction of each simulated image instead of looking at 30 different redshifts to infer the optical depth to reionization. I did this by training three different Bayesian models with the goal of better understand errors and how generalizable the problem of estimating the ionization fraction can be.

I showed that for a simple fully-Bayesian network with both convolution filters and dense layers, it is possible to successfully produce predicted values that are closely aligned with the true values and the model was tuned to find the "best" generalized model architecture for this particular problem. I did this for three different models where all of them are trained using the same `ToyModel` data either without filtering, with filters applied, or with filters and noise at different levels applied. All but one model, a fully Bayesian model with deterministic loss function, were successful in making predictions.

Naturally, the next steps are to test the "ionization-fraction-measuring" CNNs on `21cmfast` / `zreion` data instead of on the `ToyModel` data. The new goal will be to compare the performance of these new fully Bayesian convolution neural networks on trained on `21cmfast` when presented with `zreion` data. In addition to this, exploration of the performance of a fully Bayesian CNN that infers the optical depth to reionization by looking at 30 redshifts will make the predictions made in my first paper more statistically sound. Then, I believe it is important to explore these same questions but with more realistic data filtering and noise.

# BIBLIOGRAPHY

Martín Abadi, Paul Barham, Jianmin Chen, Zhifeng Chen, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Geoffrey Irving, Michael Isard, Manjunath Kudlur, Josh Levenberg, Rajat Monga, Sherry Moore, Derek G. Murray, Benoit Steiner, Paul Tucker, Vijay Vasudevan, Pete Warden, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. TensorFlow: A System for Large-scale Machine Learning. In *Proceedings of the 12th USENIX Conference on Operating Systems Design and Implementation*, OSDI'16, pages 265–283, Berkeley, CA, USA, 2016. USENIX Association. ISBN 978-1-931971-33-1. URL http://dl.acm.org/citation.cfm?id=3026877.3026899.

Corey Adams, Giuseppe Carleo, Alessandro Lovato, and Noemi Rocco. Variational Monte Carlo Calculations of A$\leq$4 Nuclei with an Artificial Neural-Network Correlator Ansatz. *Physical Review Letters*, 127(2), Jul 2021. ISSN 1079-7114. doi: 10.1103/physrevlett.127.022502. URL http://dx.doi.org/10.1103/PhysRevLett.127.022502.

Peter Ade, James Aguirre, Zeeshan Ahmed, Simone Aiola, Aamir Ali, David Alonso, Marcelo A. Alvarez, Kam Arnold, Peter Ashton, Jason Austermann, Humna Awan, Carlo Baccigalupi, Taylor Baildon, Darcy Barron, Nick Battaglia, Richard Battye, Eric Baxter, Andrew Bazarko, James A. Beall, Rachel Bean, Dominic Beck, Shawn Beckman, Benjamin Beringue, Federico Bianchini, Steven Boada, David Boettger, J. Richard Bond, Julian Borrill, Michael L. Brown, Sarah Marie Bruno, Sean Bryan, Erminia Calabrese, Victoria Calafut, Paolo Calisse, Julien Carron, Anthony Challinor, Grace Chesmore, Yuji Chinone, Jens Chluba, Hsiao-Mei Sherry Cho, Steve Choi, Gabriele Coppi, Nicholas F. Cothard, Kevin Coughlin, Devin Crichton, Kevin D. Crowley, Kevin T. Crowley, Ari Cukierman, John M. D'Ewart, Rolando Dünner, Tijmen de Haan, Mark Devlin, Simon Dicker, Joy Didier, Matt Dobbs, Bradley Dober, Cody J. Duell, Shannon Duff, Adri Duivenvoorden, Jo Dunkley, John Dusatko, Josquin Errard, Giulio Fabbian, Stephen Feeney, Simone Ferraro, Pedro Fluxà, Katherine Freese, Josef C. Frisch, Andrei Frolov, George Fuller, Brittany Fuzia, Nicholas Galitzki, Patricio A. Gallardo, Jose Tomas Galvez Ghersi, Jiansong Gao, Eric Gawiser, Martina Gerbino, Vera Gluscevic, Neil Goeckner-Wald, Joseph Golec, Sam Gordon, Megan Gralla, Daniel Green, Arpi Grigorian, John Groh, Chris Groppi, Yilun Guan, Jon E. Gudmundsson, Dongwon Han, Peter Hargrave, Masaya Hasegawa, Matthew Hasselfield, Makoto Hattori, Victor Haynes, Masashi Hazumi, Yizhou He, Erin Healy, Shawn W. Henderson, Carlos Hervias-Caimapo, Charles A. Hill, J. Colin Hill, Gene Hilton, Matt Hilton, Adam D. Hincks, Gary Hinshaw, Renée Hložek, Shirley Ho, Shuay-Pwu Patty Ho, Logan Howe, Zhiqi Huang, Johannes Hubmayr, Kevin Huffenberger, John P. Hughes, Anna Ijjas, Margaret Ikape, Kent Irwin, Andrew H. Jaffe, Bhuvnesh Jain, Oliver Jeong, Daisuke Kaneko, Ethan D. Karpel, Nobuhiko Katayama, Brian Keating, Sarah S. Kernasovskiy, Reijo Keskitalo, Theodore Kisner, Kenji Kiuchi, Jeff Klein, Kenda Knowles, Brian Koopman, Arthur Kosowsky, Nicoletta Krachmalnicoff, Stephen E. Kuenstner, Chao-Lin Kuo, Akito Kusaka, Jacob Lashner, Adrian Lee, Eunseong Lee, David Leon, Jason S. Y. Leung, Antony Lewis, Yaqiong Li, Zack Li, Michele Limon, Eric Linder, Carlos Lopez-Caraballo, Thibaut Louis, Lindsay Lowry, Marius Lungu, Mathew Madhavacheril, Daisy Mak, Felipe Maldonado, Hamdi Mani, Ben Mates, Frederick Matsuda, Loïc Maurin, Phil Mauskopf, Andrew May, Nialh McCallum, Chris McKenney, Jeff McMahon, P. Daniel Meerburg, Joel Meyers, Amber Miller, Mark Mirmelstein, Kavilan Moodley, Moritz Munchmeyer, Charles Munson, Sigurd Naess, Federico Nati, Martin Navaroli, Laura Newburgh, Ho Nam Nguyen, Michael Niemack, Haruki Nishino, John Orlowski-Scherer, Lyman Page, Bruce Partridge, Julien Peloton, Francesca Perrotta, Lucio Piccirillo, Giampaolo Pisano, Davide Poletti, Roberto Puddu, Giuseppe Puglisi, Chris Raum, Christian L. Reichardt, Mathieu Remazeilles, Yoel Rephaeli, Dominik Riechers, Felipe Rojas, Anirban Roy, Sharon Sadeh, Yuki Sakurai, Maria Salatino, Mayuri Sathyanarayana Rao, Emmanuel Schaan, Marcel Schmittfull, Neelima Sehgal, Joseph Seibert, Uros Seljak, Blake Sherwin, Meir Shimon, Carlos Sierra, Jonathan Sievers, Precious Sikhosana, Maximiliano Silva-Feaver, Sara M. Simon, Adrian Sinclair, Praween Siritanasak, Kendrick Smith, Stephen R. Smith, David Spergel, Suzanne T. Staggs, George Stein, Jason R. Stevens, Radek Stompor, Aritoki Suzuki, Osamu Tajima, Satoru Takakura, Grant Teply, Daniel B. Thomas, Ben Thorne, Robert Thornton, Hy Trac, Calvin Tsai, Carole Tucker, Joel Ullom, Sunny Vagnozzi, Alexander van Engelen, Jeff Van Lanen, Daniel D. Van Winkle, Eve M. Vavagiakis, Clara Vergès, Michael Vissers, Kasey Wagoner, Samantha Walker, Jon Ward, Ben Westbrook, Nathan Whitehorn, Jason Williams, Joel Williams, Edward J. Wollack, Zhilei Xu, Byeonghee Yu, Cyndia Yu, Fernando Zago, Hezi Zhang, Ningfeng Zhu, and Simons Observatory Collaboration. The Simons Observatory: science goals and forecasts. *J. Cosmology Astroparticle Physics*, 2019(2):056, February 2019. doi: 10.1088/1475-7516/2019/02/056.

K. Adelberger. Star Formation and Structure Formation at 1 <~z <~4. In A. Mazure, O. Le Fèvre, and V. Le Brun, editors, *Clustering at High Redshift*, volume 200 of *Astronomical Society of the Pacific Conference Series*, page 13, January 2000.

Aritra Basu, Dominik J Schwarz, Hans-Rainer Klöckner, Sebastian von Hausegger, Michael Kramer, Gundolf Wieching, and Blakesley Burkhart. CMB foreground measurements through broad-band radio spectro-polarimetry: prospects of the SKA-MPG telescope. *Monthly Notices of the Royal Astronomical Society*, 488(2):1618–1634, Jun 2019. ISSN 1365-2966. doi: 10.1093/mnras/stz1637. URL http://dx.doi.org/10.1093/mnras/stz1637.

N. Battaglia, H. Trac, R. Cen, and A. Loeb. Reionization on Large Scales. I. A Parametric Model Constructed from Radiation-hydrodynamic Simulations. *Astrophysical Journal*, 776:81, October 2013. doi: 10.1088/0004-637X/776/2/81.

A. P. Beardsley, B. J. Hazelton, I. S. Sullivan, P. Carroll, N. Barry, M. Rahimi, B. Pindor, C. M. Trott, J. Line, Daniel C. Jacobs, M. F. Morales, J. C. Pober, G. Bernardi, Judd D. Bowman, M. P. Busch, F. Briggs, R. J. Cappallo, B. E. Corey, A. de Oliveira-Costa, Joshua S. Dillon, D. Emrich, A. Ewall-Wice, L. Feng, B. M. Gaensler, R. Goeke, L. J. Greenhill, J. N. Hewitt, N. Hurley-Walker, M. Johnston-Hollitt, D. L. Kaplan, J. C. Kasper, H. S. Kim, E. Kratzenberg, E. Lenc, A. Loeb, C. J. Lonsdale, M. J. Lynch, B. McKinley, S. R. McWhirter, D. A. Mitchell, E. Morgan, A. R. Neben, Nithyanandan Thyagarajan, D. Oberoi, A. R. Offringa, S. M. Ord, S. Paul, T. Prabu, P. Procopio, J. Riding, A. E. E. Rogers, A. Roshi, N. Udaya Shankar, Shiv K. Sethi, K. S. Srivani,

R. Subrahmanyan, M. Tegmark, S. J. Tingay, M. Waterson, R. B. Wayth, R. L. Webster, A. R. Whitney, A. Williams, C. L. Williams, C. Wu, and J. S. B. Wyithe. First Season MWA EoR Power spectrum Results at Redshift 7. *Astrophysical Journal*, 833(1):102, December 2016. doi: 10.3847/1538-4357/833/1/102.

James Bergstra and Yoshua Bengio. Random search for hyper-parameter optimization. *Journal of Machine Learning Research*, 13(Feb):281–305, 2012.

G. Bernardi, A. G. de Bruyn, M. A. Brentjens, B. Ciardi, G. Harker, V. Jelić, L. V. E. Koopmans, P. Labropoulos, A. Offringa, V. N. Pandey, J. Schaye, R. M. Thomas, S. Yatawatta, and S. Zaroubi. Foregrounds for observations of the cosmological 21 cm line. I. First Westerbork measurements of Galactic emission at 150 MHz in a low latitude field. *Astronomy and Astrophysics*, 500(3):965–979, June 2009. doi: 10.1051/0004-6361/200911627.

Tashalee S. Billings, Paul La Plante, and James E. Aguirre. Extracting the Optical Depth to Reionization $\tau$ from 21 cm Data Using Machine Learning Techniques. *PASP*, 133(1022): 044001, April 2021. doi: 10.1088/1538-3873/abe9a0.

Charles Blundell, Julien Cornebise, Koray Kavukcuoglu, and Daan Wierstra. Weight Uncertainty in Neural Networks. *arXiv e-prints*, art. arXiv:1505.05424, May 2015.

Léon Bottou. Stochastic Learning. In Olivier Bousquet and Ulrike von Luxburg, editors, *Advanced Lectures on Machine Learning*, Lecture Notes in Artificial Intelligence, LNAI 3176, pages 146–168. Springer Verlag, Berlin, 2004. URL http://leon.bottou.org/papers/bottou-mlss-2004.

Judd D. Bowman, Iver Cairns, David L. Kaplan, Tara Murphy, Divya Oberoi, Lister Staveley-Smith, Wayne Arcus, David G. Barnes, Gianni Bernardi, Frank H. Briggs, Shea Brown, John D. Bunton, Adam J. Burgasser, Roger J. Cappallo, Shami Chatterjee, Brian E. Corey, Anthea Coster, Avinash Deshpande, Ludi deSouza, David Emrich, Philip Erickson, Robert F. Goeke, B. M. Gaensler, Lincoln J. Greenhill, Lisa Harvey-Smith, Bryna J. Hazelton, David Herne, Jacqueline N. Hewitt, Melanie Johnston-Hollitt, Justin C. Kasper, Barton B. Kincaid, Ronald Koenig, Eric Kratzenberg, Colin J. Lonsdale, Mervyn J. Lynch, Lynn D. Matthews, S. Russell McWhirter, Daniel A. Mitchell, Miguel F. Morales, Edward H. Morgan, Stephen M. Ord, Joseph Pathikulangara, Thiagaraj Prabu, Ronald A. Remillard, Timothy Robishaw, Alan E. E. Rogers, Anish A. Roshi, Joseph E. Salah, Robert J. Sault, N. Udaya Shankar, K. S. Srivani, Jamie B. Stevens, Ravi Subrahmanyan, Steven J. Tingay, Randall B. Wayth, Mark Waterson, Rachel L. Webster, Alan R. Whitney, Andrew J. Williams, Christopher L. Williams, and J. Stuart B. Wyithe. Science with the Murchison Widefield Array. *Publications of the Astronomical Society of Australia*, 30, 2013. ISSN 1448-6083. doi: 10.1017/pas.2013.009. URL http://dx.doi.org/10.1017/pas.2013.009.

François Chollet and others. Keras: The Python Deep Learning library. Astrophysics Source Code Library, record ascl:1806.022, June 2018.

Andreas C. Damianou and Neil D. Lawrence. Deep Gaussian Processes, November 2012.

D. R. DeBoer, A. R. Parsons, J. E. Aguirre, P. Alexander, Z. S. Ali, A. P. Beardsley, G. Bernardi, J. D. Bowman, R. F. Bradley, C. L. Carilli, C. Cheng, E. de Lera Acedo, J. S. Dillon, A. Ewall-Wice, G. Fadana, N. Fagnoni, R. Fritz, S. R. Furlanetto, B. Glendenning, B. Greig, J. Grobbelaar, B. J. Hazelton, J. N. Hewitt, J. Hickish, D. C. Jacobs, A. Julius, M. Kariseb, S. A. Kohn, T. Lekalake, A. Liu, A. Loots, D. MacMahon, L. Malan, C. Malgas, M. Maree, Z. Martinot, N. Mathison, E. Matsetela, A. Mesinger, M. F. Morales, A. R. Neben, N. Patra, S. Pieterse, J. C. Pober, N. Razavi-Ghods, J. Ringuette, J. Robnett, K. Rosie, R. Sell, C. Smith, A. Syce, M. Tegmark, N. Thyagarajan, P. K. G. Williams, and H. Zheng. Hydrogen Epoch of Reionization Array (HERA). *PASP*, 129(4):045001, April 2017. doi: 10.1088/1538-3873/129/974/045001.

Tiziana Di Matteo, Rosalba Perna, Tom Abel, and Martin J. Rees. Radio Foregrounds for the 21 Centimeter Tomography of the Neutral Intergalactic Medium at High Redshifts. *The Astrophysical Journal*, 564(2):576–580, Jan 2002. ISSN 1538-4357. doi: 10.1086/324293. URL http://dx.doi.org/10.1086/324293.

Tiziana Di Matteo, Benedetta Ciardi, and Francesco Miniati. The 21-cm emission from the reionization epoch: extended and point source foregrounds. *Monthly Notices of the RAS*, 355(4):1053–1065, December 2004. doi: 10.1111/j.1365-2966.2004.08443.x.

Vincent Dumoulin and Francesco Visin. A guide to convolution arithmetic for deep learning. *arXiv e-prints*, art. arXiv:1603.07285, March 2016.

Dumitru Erhan, Y. Bengio, Aaron Courville, and Pascal Vincent. Visualizing higher-layer features of a deep network. *Technical Report, Univeristé de Montréal*, 01 2009.

S. R. Furlanetto and M. R. Furlanetto. Spin exchange rates in proton-hydrogen collisions. *Monthly Notices of the Royal Astronomical Society*, 379(1):130–134, Jul 2007. ISSN 1365-2966. doi: 10.1111/j.1365-2966.2007.11921.x. URL http://dx.doi.org/10.1111/j.1365-2966.2007.11921.x.

Steven R. Furlanetto, S. Peng Oh, and Frank H. Briggs. Cosmology at low frequencies: The 21cm transition and the high-redshift universe. *Physics Reports*, 433(4–6):181–301, Oct 2006. ISSN 0370-1573. doi: 10.1016/j.physrep.2006.08.002. URL http://dx.doi.org/10.1016/j.physrep.2006.08.002.

Yarin Gal and Zoubin Ghahramani. Bayesian Convolutional Neural Networks with Bernoulli Approximate Variational Inference. *arXiv e-prints*, art. arXiv:1506.02158, June 2015a.

Yarin Gal and Zoubin Ghahramani. Dropout as a Bayesian Approximation: Representing Model Uncertainty in Deep Learning. *arXiv e-prints*, art. arXiv:1506.02142, June 2015b.

Abhik Ghosh, Jayanti Prasad, Somnath Bharadwaj, Sk. Saiyad Ali, and Jayaram N. Chengalur. Characterizing foreground for redshifted 21 cm radiation: 150 MHz Giant Metrewave Radio Telescope observations. *Monthly Notices of the Royal Astronomical Society*, 426(4):3295–3314, Oct 2012. ISSN 0035-8711. doi: 10.1111/j.1365-2966.2012.21889.x. URL http://dx.doi.org/10.1111/j.1365-2966.2012.21889.x.

Nicolas Gillet, Andrei Mesinger, Bradley Greig, Adrian Liu, and Graziano Ucci. Deep learning from 21-cm tomography of the cosmic dawn and reionization. *Monthly Notices of the RAS*, 484(1):282–293, March 2019. doi: 10.1093/mnras/stz010.

Alex Graves. Practical Variational Inference for Neural Networks. In J. Shawe-Taylor, R. Zemel, P. Bartlett, F. Pereira, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 24. Curran Associates, Inc., 2011. URL https://proceedings.neurips.cc/paper/2011/file/7eb3c8be3d411e8ebfab08eba5f49632-Paper.pdf.

Jiuxiang Gu, Zhenhua Wang, Jason Kuen, Lianyang Ma, Amir Shahroudy, Bing Shuai, Ting Liu, Xingxing Wang, Li Wang, Gang Wang, Jianfei Cai, and Tsuhan Chen. Recent Advances in Convolutional Neural Networks. *arXiv e-prints*, art. arXiv:1512.07108, December 2015.

Geetakrishnasai Gunapati, Anirudh Jain, P. K. Srijith, and Shantanu Desai. Variational inference as an alternative to MCMC for parameter estimation and model selection. *Publications of the Astronomical Society of Australia*, 39, 2022. ISSN 1448-6083. doi: 10.1017/pasa.2021.64. URL http://dx.doi.org/10.1017/pasa.2021.64.

Zoltán Haiman and Abraham Loeb. Signatures of Stellar Reionization of the Universe. *Astrophysical Journal*, 483(1):21–37, July 1997. doi: 10.1086/304238.

Shaul Hanany, Marcelo Alvarez, Emmanuel Artis, Peter Ashton, Jonathan Aumont, Ragnhild Aurlien, Ranajoy Banerji, R. Belen Barreiro, James G. Bartlett, Soumen Basak, Nick Battaglia, Jamie Bock, Kimberly K. Boddy, Matteo Bonato, Julian Borrill, François Bouchet, François Boulanger, Blakesley Burkhart, Jens Chluba, David Chuss, Susan E. Clark, Joelle Cooperrider, Brendan P. Crill, Gianfranco De Zotti, Jacques Delabrouille, Eleonora Di Valentino, Joy Didier, Olivier Doré, Hans K. Eriksen, Josquin Errard, Tom Essinger-Hileman, Stephen Feeney, Jeffrey Filippini, Laura Fissel, Raphael Flauger, Unni Fuskeland, Vera Gluscevic, Krzysztof M. Gorski, Dan Green, Brandon Hensley, Diego Herranz, J. Colin Hill, Eric Hivon, Renée Hložek, Johannes Hubmayr, Bradley R. Johnson, William Jones, Terry Jones, Lloyd Knox, Al Kogut, Marcos López-Caniego, Charles Lawrence, Alex Lazarian, Zack Li, Mathew Madhavacheril, Jean-Baptiste Melin, Joel Meyers, Calum Murray, Mattia Negrello, Giles Novak, Roger O'Brient, Christopher Paine, Tim Pearson, Levon Pogosian, Clem Pryke, Giuseppe Puglisi, Mathieu Remazeilles, Graca Rocha, Marcel Schmittfull, Douglas Scott, Peter Shirron, Ian Stephens, Brian Sutin, Maurizio Tomasi, Amy Trangsrud, Alexander van Engelen, Flavien Vansyngel, Ingunn K. Wehus, Qi Wen, Siyao Xu, Karl Young, and Andrea Zonca. PICO: Probe of Inflation and Cosmic Origins, 2019. URL https://arxiv.org/abs/1902.10541.

Joachim Harnois-Deraps, Ue-Li Pen, Ilian Iliev, Hugh Merz, J. Emberson, and Vincent Desjacques. High-performance p3m n-body code: cubep3m. *Monthly Notices of the Royal Astronomical Society*, 436, 08 2012. doi: 10.1093/mnras/stt1591.

M. Hazumi, J. Borrill, Y. Chinone, M. A. Dobbs, H. Fuke, A. Ghribi, M. Hasegawa, K. Hattori, M. Hattori, W. L. Holzapfel, Y. Inoue, K. Ishidoshiro, H. Ishino, K. Karatsu, N. Katayama, I. Kawano, A. Kibayashi, Y. Kibe, N. Kimura, K. Koga, E. Komatsu, A. T. Lee, H. Matsuhara, T. Matsumura, S. Mima, K. Mitsuda, H. Morii, S. Murayama, M. Nagai, R. Nagata, S. Nakamura, K. Natsume, H. Nishino, A. Noda, T. Noguchi, I. Ohta, C. Otani, P. L. Richards, S. Sakai, N. Sato, Y. Sato, Y. Sekimoto, A. Shimizu, K. Shinozaki, H. Sugita, A. Suzuki, T. Suzuki, O. Tajima, S. Takada, Y. Takagi, Y. Takei, T. Tomaru, Y. Uzawa, H. Watanabe, N. Yamasaki, M. Yoshida, T. Yoshida, and K. Yotsumoto. LiteBIRD: a small satellite for the study of B-mode polarization and inflation from cosmic background radiation detection. In *Space Telescopes and Instrumentation 2012: Optical, Infrared, and Millimeter Wave*, volume 8442 of *Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series*, page 844219, September 2012. doi: 10.1117/12.926743.

Jonathan Higgins and Avery Meiksin. The Wouthuysen-Field effect in a clumpy intergalactic medium. *Monthly Notices of the Royal Astronomical Society*, 393(3):949–958, Mar 2009. ISSN 1365-2966. doi: 10.1111/j.1365-2966.2008.14199.x. URL http://dx.doi.org/10.1111/j.1365-2966.2008.14199.x.

G. Hinshaw, D. Larson, E. Komatsu, D. N. Spergel, C. L. Bennett, J. Dunkley, M. R. Nolta, M. Halpern, R. S. Hill, N. Odegard, L. Page, K. M. Smith, J. L. Weiland, B. Gold, N. Jarosik, A. Kogut, M. Limon, S. S. Meyer, G. S. Tucker, E. Wollack, and E. L. Wright. Nine-year Wilkinson Microwave Anisotropy Probe (WMAP) Observations: Cosmological Parameter Results. *Astrophysical Journal Supplement*, 208(2):19, October 2013. doi: 10.1088/0067-0049/208/2/19.

C. M. Hirata. Wouthuysen-Field coupling strength and application to high-redshift 21-cm radiation. *Monthly Notices of the Royal Astronomical Society*, 367(1):259–274, Mar 2006. ISSN 1365-2966. doi: 10.1111/j.1365-2966.2005.09949.x. URL http://dx.doi.org/10.1111/j.1365-2966.2005.09949.x.

Christopher M. Hirata and Kris Sigurdson. The spin-resolved atomic velocity distribution and 21-cm line profile of dark-age gas. *Monthly Notices of the Royal Astronomical Society*, 375(4):1241–1264, Feb 2007. ISSN 1365-2966. doi: 10.1111/j.1365-2966.2006.11321.x. URL http://dx.doi.org/10.1111/j.1365-2966.2006.11321.x.

Christopher M. Hirata, Abhilash Mishra, and Tejaswi Venumadhav. Detecting primordial gravitational waves with circular polarization of the redshifted 21 cm line. I. Formalism. *Physical Review D*, 97(10):103521, May 2018. doi: 10.1103/PhysRevD.97.103521.

Andrew M. Hopkins and John F. Beacom. On the Normalization of the Cosmic Star Formation History. *The Astrophysical Journal*, 651(1):142–154, Nov 2006. ISSN 1538-4357. doi: 10.1086/506610. URL http://dx.doi.org/10.1086/506610.

Héctor J. Hortúa, Riccardo Volpi, Dimitri Marinelli, and Luigi Malagò. Parameter estimation for the cosmic microwave background with Bayesian neural networks. *Physical Review D*, 102(10), Nov 2020. ISSN 2470-0029. doi: 10.1103/physrevd.102.103509. URL http://dx.doi.org/10.1103/PhysRevD.102.103509.

Sergey Ioffe and Christian Szegedy. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. *arXiv e-prints*, art. arXiv:1502.03167, February 2015.

V. Jelić, S. Zaroubi, P. Labropoulos, R. M. Thomas, G. Bernardi, M. A. Brentjens, A. G. de Bruyn, B. Ciardi, G. Harker, L. V. E. Koopmans, V. N. Pandey, J. Schaye, and S. Yatawatta. Foreground simulations for the LOFAR-epoch of reionization experiment. *Monthly Notices of the RAS*, 389(3):1319–1335, September 2008. doi: 10.1111/j.1365-2966.2008.13634.x.

Helmut G. Katzgraber. Introduction to Monte Carlo Methods. *arXiv e-prints*, art. arXiv:0905.1629, May 2009.

Nicholas S. Kern, Adrian Liu, Aaron R. Parsons, Andrei Mesinger, and Bradley Greig. Emulating Simulations of Cosmic Dawn for 21 cm Power Spectrum Constraints on Cosmology, Reionization, and X-Ray Heating. *Astrophysical Journal*, 848(1):23, October 2017. doi: 10.3847/1538-4357/aa8bb4.

Diederik P Kingma and Max Welling. Auto-Encoding Variational Bayes. *arXiv e-prints*, art. arXiv:1312.6114, December 2013.

A. Kogut, D. J. Fixsen, D. T. Chuss, J. Dotson, E. Dwek, M. Halpern, G. F. Hinshaw, S. M. Meyer, S. H. Moseley, M. D. Seiffert, D. N. Spergel, and E. J. Wollack. The Primordial Inflation Explorer (PIXIE): a nulling polarimeter for cosmic microwave background observations. *J. Cosmology Astroparticle Physics*, 2011(7):025, July 2011. doi: 10.1088/1475-7516/2011/07/025.

Saul Aryeh Kohn. *Outer Space and Fourier Space: Understanding Foregrounds for Neutral Hydrogen Epoch of Reionization Measurements*. PhD thesis, University of Pennsylvania, January 2018.

Matthew Kolopanis, Daniel C. Jacobs, Carina Cheng, Aaron R. Parsons, Saul A. Kohn, Jonathan C. Pober, James E. Aguirre, Zaki S. Ali, Gianni Bernardi, Richard F. Bradley, Chris L. Carilli, David R. DeBoer, Matthew R. Dexter, Joshua S. Dillon, Joshua Kerrigan, Pat Klima, Adrian Liu, David H. E. MacMahon, David F. Moore, Nithyanandan Thyagarajan, Chuneeta D. Nunhokee, William P. Walbrugh, and Andre Walker. A Simplified, Lossless Reanalysis of PAPER-64. *Astrophysical Journal*, 883(2):133, October 2019. doi:

10.3847/1538-4357/ab3e3a.

S. Kullback and R. A. Leibler. On Information and Sufficiency. *The Annals of Mathematical Statistics*, 22(1):79 – 86, 1951. doi: 10.1214/aoms/1177729694. URL https://doi.org/10.1214/aoms/1177729694.

Yungi Kwon, Sungwook E. Hong, and Inkyu Park. Deep-Learning Study of the 21cm Differential Brightness Temperature During the Epoch of Reionization. *arXiv e-prints*, art. arXiv:2006.06236, June 2020.

P. La Plante and M. Ntampaka. Machine Learning Applied to the Reionization History of the Universe in the 21 cm Signal. *Astrophysical Journal*, 880(2):110, Aug 2019. doi: 10.3847/1538-4357/ab2983.

H. Liszt. The spin temperature of warm interstellar H I. *Astronomy and Astrophysics*, 371(2):698–707, May 2001. ISSN 1432-0746. doi: 10.1051/0004-6361:20010395. URL http://dx.doi.org/10.1051/0004-6361:20010395.

A. Liu, J. R. Pritchard, R. Allison, A. R. Parsons, U. Seljak, and B. D. Sherwin. Eliminating the optical depth nuisance from the CMB with 21 cm cosmology. *Physical Review D*, 93 (4):043013, February 2016. doi: 10.1103/PhysRevD.93.043013.

P. Madau, H. C. Ferguson, M. E. Dickinson, M. Giavalisco, C. C. Steidel, and A. Fruchter. High-redshift galaxies in the Hubble Deep Field: colour selection and star formation history to $z \sim 4$. *Monthly Notices of the Royal Astronomical Society*, 283(4):1388–1404, Dec 1996. ISSN 1365-2966. doi: 10.1093/mnras/283.4.1388. URL http://dx.doi.org/10.1093/mnras/283.4.1388.

Piero Madau, Avery Meiksin, and Martin J. Rees. 21 Centimeter Tomography of the Intergalactic Medium at High Redshift. *The Astrophysical Journal*, 475(2):429–444, Feb 1997. ISSN 1538-4357. doi: 10.1086/303549. URL http://dx.doi.org/10.1086/303549.

Suman Majumdar, Jonathan R. Pritchard, Rajesh Mondal, Catherine A. Watkinson, Somnath Bharadwaj, and Garrelt Mellema. Quantifying the non-Gaussianity in the EoR 21-cm signal through bispectrum. *Mon. Not. Roy. Astron. Soc.*, 476(3):4007–4024, 2018. doi: 10.1093/mnras/sty535.

Andrei Mesinger, Steven Furlanetto, and Renyue Cen. 21cmfast: a fast, seminumerical simulation of the high-redshift 21-cm signal. *Monthly Notices of the Royal Astronomical Society*, 411(2):955–972, Nov 2010. ISSN 0035-8711. doi: 10.1111/j.1365-2966.2010.17731.x. URL http://dx.doi.org/10.1111/j.1365-2966.2010.17731.x.

Jordan Mirocha, Stephen Skory, Jack O. Burns, and John H. Wise. Optimized Multi-frequency Spectra for Applications in Radiative Feedback and Cosmological Reionization. *Astrophysical Journal*, 756(1):94, September 2012. doi: 10.1088/0004-637X/756/1/94.

Shakir Mohamed, Mihaela Rosca, Michael Figurnov, and Andriy Mnih. Monte Carlo Gradient Estimation in Machine Learning. *arXiv e-prints*, art. arXiv:1906.10652, June 2019.

Raul A. Monsalve, Anastasia Fialkov, Judd D. Bowman, Alan E. E. Rogers, Thomas J. Mozdzen, Aviad Cohen, Rennan Barkana, and Nivedita Mahesh. Results from EDGES High-Band. III. New Constraints on Parameters of the Early Universe. *Astrophysical Journal*, 875(1):67, April 2019. doi: 10.3847/1538-4357/ab07be.

W. Nowak. Introduction to stochastic search and optimization. estimation, simulation, and control (spall, j.c.; 2003) [book review]. *IEEE Transactions on Neural Networks*, 18(3): 964–965, 2007. doi: 10.1109/TNN.2007.897481.

A. Nusser. The spin temperature of neutral hydrogen during cosmic pre-reionization. *Monthly Notices of the Royal Astronomical Society*, 359(1):183–190, May 2005. ISSN 1365-2966. doi: 10.1111/j.1365-2966.2005.08894.x. URL http://dx.doi.org/10.1111/j.1365-2966.2005.08894.x.

G. Paciga, J. G. Albert, K. Bandura, T.-C. Chang, Y. Gupta, C. Hirata, J. Odegova, U.-L. Pen, J. B. Peterson, J. Roy, J. R. Shaw, K. Sigurdson, and T. Voytek. A simulation-calibrated limit on the H I power spectrum from the GMRT Epoch of Reionization experiment. *Monthly Notices of the RAS*, May 2013. doi: 10.1093/mnras/stt753.

L. Page, G. Hinshaw, E. Komatsu, M. R. Nolta, D. N. Spergel, C. L. Bennett, C. Barnes, R. Bean, O. Doré, J. Dunkley, M. Halpern, R. S. Hill, N. Jarosik, A. Kogut, M. Limon, S. S. Meyer, N. Odegard, H. V. Peiris, G. S. Tucker, L. Verde, J. L. Weiland, E. Wollack, and E. L. Wright. Three-Year Wilkinson Microwave Anisotropy Probe (WMAP) Observations: Polarization Analysis. *Astrophysical Journal Supplement*, 170(2):335–376, June 2007. doi: 10.1086/513699.

John Paisley, David Blei, and Michael Jordan. Variational Bayesian Inference with Stochastic Search. *arXiv e-prints*, art. arXiv:1206.6430, June 2012.

Aaron R. Parsons, Donald C. Backer, Griffin S. Foster, Melvyn C. H. Wright, Richard F. Bradley, Nicole E. Gugliucci, Chaitali R. Parashare, Erin E. Benoit, James E. Aguirre, Daniel C. Jacobs, Chris L. Carilli, David Herne, Mervyn J. Lynch, Jason R. Manley, and Daniel J. Werthimer. The Precision Array for Probing the Epoch of Re-ionization: Eight Station Results. *Astronomical Journal*, 139(4):1468–1480, April 2010. doi: 10.1088/0004-6256/139/4/1468.

Aaron R. Parsons, Jonathan C. Pober, James E. Aguirre, Christopher L. Carilli, Daniel C. Jacobs, and David F. Moore. A Per-baseline, Delay-spectrum Technique for Accessing the 21 cm Cosmic Reionization Signature. *Astrophysical Journal*, 756(2):165, September 2012. doi: 10.1088/0004-637X/756/2/165.

A. H. Patil, S. Yatawatta, L. V. E. Koopmans, A. G. de Bruyn, M. A. Brentjens, S. Zaroubi,

K. M. B. Asad, M. Hatef, V. Jelić, M. Mevius, A. R. Offringa, V. N. Pandey, H. Vedantham, F. B. Abdalla, W. N. Brouw, E. Chapman, B. Ciardi, B. K. Gehlot, A. Ghosh, G. Harker, I. T. Iliev, K. Kakiichi, S. Majumdar, G. Mellema, M. B. Silva, J. Schaye, D. Vrbanec, and S. J. Wijnholds. Upper Limits on the 21 cm Epoch of Reionization Power Spectrum from One Night with LOFAR. *Astrophysical Journal*, 838(1):65, March 2017. doi: 10.3847/1538-4357/aa63e7.

Nick Pawlowski, Andrew Brock, Matthew C. H. Lee, Martin Rajchl, and Ben Glocker. Implicit Weight Uncertainty in Neural Networks. *arXiv e-prints*, art. arXiv:1711.01297, November 2017.

Tim Pearce, Felix Leibfried, Alexandra Brintrup, Mohamed Zaki, and Andy Neely. Uncertainty in Neural Networks: Approximately Bayesian Ensembling. *arXiv e-prints*, art. arXiv:1810.05546, October 2018.

Planck Collaboration, P. A. R. Ade, N. Aghanim, M. Arnaud, M. Ashdown, J. Aumont, C. Baccigalupi, A. J. Banday, R. B. Barreiro, J. G. Bartlett, N. Bartolo, E. Battaner, R. Battye, K. Benabed, A. Benoît, A. Benoit-Lévy, J. P. Bernard, M. Bersanelli, P. Bielewicz, J. J. Bock, A. Bonaldi, L. Bonavera, J. R. Bond, J. Borrill, F. R. Bouchet, F. Boulanger, M. Bucher, C. Burigana, R. C. Butler, E. Calabrese, J. F. Cardoso, A. Catalano, A. Challinor, A. Chamballu, R. R. Chary, H. C. Chiang, J. Chluba, P. R. Christensen, S. Church, D. L. Clements, S. Colombi, L. P. L. Colombo, C. Combet, A. Coulais, B. P. Crill, A. Curto, F. Cuttaia, L. Danese, R. D. Davies, R. J. Davis, P. de Bernardis, A. de Rosa, G. de Zotti, J. Delabrouille, F. X. Désert, E. Di Valentino, C. Dickinson, J. M. Diego, K. Dolag, H. Dole, S. Donzelli, O. Doré, M. Douspis, A. Ducout, J. Dunkley, X. Dupac, G. Efstathiou, F. Elsner, T. A. Enßlin, H. K. Eriksen, M. Farhang, J. Fergusson, F. Finelli, O. Forni, M. Frailis, A. A. Fraisse, E. Franceschi, A. Frejsel, S. Galeotta, S. Galli, K. Ganga, C. Gauthier, M. Gerbino, T. Ghosh, M. Giard, Y. Giraud-Héraud, E. Giusarma, E. Gjerløw, J. González-Nuevo, K. M. Górski, S. Gratton, A. Gregorio, A. Gruppuso, J. E. Gudmundsson, J. Hamann, F. K. Hansen, D. Hanson, D. L. Harrison, G. Helou, S. Henrot-Versillé, C. Hernández-Monteagudo, D. Herranz, S. R. Hildebrand t, E. Hivon, M. Hobson, W. A. Holmes, A. Hornstrup, W. Hovest, Z. Huang, K. M. Huffenberger, G. Hurier, A. H. Jaffe, T. R. Jaffe, W. C. Jones, M. Juvela, E. Keihänen, R. Keskitalo, T. S. Kisner, R. Kneissl, J. Knoche, L. Knox, M. Kunz, H. Kurki-Suonio, G. Lagache, A. Lähteenmäki, J. M. Lamarre, A. Lasenby, M. Lattanzi, C. R. Lawrence, J. P. Leahy, R. Leonardi, J. Lesgourgues, F. Levrier, A. Lewis, M. Liguori, P. B. Lilje, M. Linden-Vørnle, M. López-Caniego, P. M. Lubin, J. F. Macías-Pérez, G. Maggio, D. Maino, N. Mandolesi, A. Mangilli, A. Marchini, M. Maris, P. G. Martin, M. Martinelli, E. Martínez-González, S. Masi, S. Matarrese, P. McGehee, P. R. Meinhold, A. Melchiorri, J. B. Melin, L. Mendes, A. Mennella, M. Migliaccio, M. Millea, S. Mitra, M. A. Miville-Deschênes, A. Moneti, L. Montier, G. Morgante, D. Mortlock, A. Moss, D. Munshi, J. A. Murphy, P. Naselsky, F. Nati, P. Natoli, C. B. Netterfield, H. U. Nørgaard-Nielsen, F. Noviello, D. Novikov, I. Novikov, C. A. Oxborrow, F. Paci, L. Pagano, F. Pajot, R. Paladini, D. Paoletti, B. Partridge, F. Pasian, G. Patanchon, T. J. Pearson, O. Perdereau, L. Perotto, F. Perrotta, V. Pettorino, F. Piacentini, M. Piat,

E. Pierpaoli, D. Pietrobon, S. Plaszczynski, E. Pointecouteau, G. Polenta, L. Popa, G. W. Pratt, G. Prézeau, S. Prunet, J. L. Puget, J. P. Rachen, W. T. Reach, R. Rebolo, M. Reinecke, M. Remazeilles, C. Renault, A. Renzi, I. Ristorcelli, G. Rocha, C. Rosset, M. Rossetti, G. Roudier, B. Rouillé d'Orfeuil, M. Rowan-Robinson, J. A. Rubiño-Martín, B. Rusholme, N. Said, V. Salvatelli, L. Salvati, M. Sandri, D. Santos, M. Savelainen, G. Savini, D. Scott, M. D. Seiffert, P. Serra, E. P. S. Shellard, L. D. Spencer, M. Spinelli, V. Stolyarov, R. Stompor, R. Sudiwala, R. Sunyaev, D. Sutton, A. S. Suur-Uski, J. F. Sygnet, J. A. Tauber, L. Terenzi, L. Toffolatti, M. Tomasi, M. Tristram, T. Trombetti, M. Tucci, J. Tuovinen, M. Türler, G. Umana, L. Valenziano, J. Valiviita, F. Van Tent, P. Vielva, F. Villa, L. A. Wade, B. D. Wandelt, I. K. Wehus, M. White, S. D. M. White, A. Wilkinson, D. Yvon, A. Zacchei, and A. Zonca. Planck 2015 results. XIII. Cosmological parameters. *Astronomy and Astrophysics*, 594:A13, September 2016. doi: 10.1051/0004-6361/201525830.

Planck Collaboration, N. Aghanim, Y. Akrami, M. Ashdown, J. Aumont, C. Baccigalupi, M. Ballardini, A. J. Banday, R. B. Barreiro, N. Bartolo, S. Basak, R. Battye, K. Benabed, J. P. Bernard, M. Bersanelli, P. Bielewicz, J. J. Bock, J. R. Bond, J. Borrill, F. R. Bouchet, F. Boulanger, M. Bucher, C. Burigana, R. C. Butler, E. Calabrese, J. F. Cardoso, J. Carron, A. Challinor, H. C. Chiang, J. Chluba, L. P. L. Colombo, C. Combet, D. Contreras, B. P. Crill, F. Cuttaia, P. de Bernardis, G. de Zotti, J. Delabrouille, J. M. Delouis, E. Di Valentino, J. M. Diego, O. Doré, M. Douspis, A. Ducout, X. Dupac, S. Dusini, G. Efstathiou, F. Elsner, T. A. Enßlin, H. K. Eriksen, Y. Fantaye, M. Farhang, J. Fergusson, R. Fernandez-Cobos, F. Finelli, F. Forastieri, M. Frailis, A. A. Fraisse, E. Franceschi, A. Frolov, S. Galeotta, S. Galli, K. Ganga, R. T. Génova-Santos, M. Gerbino, T. Ghosh, J. González-Nuevo, K. M. Górski, S. Gratton, A. Gruppuso, J. E. Gudmundsson, J. Hamann, W. Handley, F. K. Hansen, D. Herranz, S. R. Hildebrandt, E. Hivon, Z. Huang, A. H. Jaffe, W. C. Jones, A. Karakci, E. Keihänen, R. Keskitalo, K. Kiiveri, J. Kim, T. S. Kisner, L. Knox, N. Krachmalnicoff, M. Kunz, H. Kurki-Suonio, G. Lagache, J. M. Lamarre, A. Lasenby, M. Lattanzi, C. R. Lawrence, M. Le Jeune, P. Lemos, J. Lesgourgues, F. Levrier, A. Lewis, M. Liguori, P. B. Lilje, M. Lilley, V. Lindholm, M. López-Caniego, P. M. Lubin, Y. Z. Ma, J. F. Macías-Pérez, G. Maggio, D. Maino, N. Mandolesi, A. Mangilli, A. Marcos-Caballero, M. Maris, P. G. Martin, M. Martinelli, E. Martínez-González, S. Matarrese, N. Mauri, J. D. McEwen, P. R. Meinhold, A. Melchiorri, A. Mennella, M. Migliaccio, M. Millea, S. Mitra, M. A. Miville-Deschênes, D. Molinari, L. Montier, G. Morgante, A. Moss, P. Natoli, H. U. Nørgaard-Nielsen, L. Pagano, D. Paoletti, B. Partridge, G. Patanchon, H. V. Peiris, F. Perrotta, V. Pettorino, F. Piacentini, L. Polastri, G. Polenta, J. L. Puget, J. P. Rachen, M. Reinecke, M. Remazeilles, A. Renzi, G. Rocha, C. Rosset, G. Roudier, J. A. Rubiño-Martín, B. Ruiz-Granados, L. Salvati, M. Sandri, M. Savelainen, D. Scott, E. P. S. Shellard, C. Sirignano, G. Sirri, L. D. Spencer, R. Sunyaev, A. S. Suur-Uski, J. A. Tauber, D. Tavagnacco, M. Tenti, L. Toffolatti, M. Tomasi, T. Trombetti, L. Valenziano, J. Valiviita, B. Van Tent, L. Vibert, P. Vielva, F. Villa, N. Vittorio, B. D. Wandelt, I. K. Wehus, M. White, S. D. M. White, A. Zacchei, and A. Zonca. Planck 2018 results. VI. Cosmological parameters. *Astronomy and Astrophysics*, 641:A6, September 2020. doi: 10.1051/0004-6361/201833910.

Jonathan C. Pober, Aaron R. Parsons, James E. Aguirre, Zaki Ali, Richard F. Bradley, Chris L. Carilli, Dave DeBoer, Matthew Dexter, Nicole E. Gugliucci, Daniel C. Jacobs, Patricia J. Klima, Dave MacMahon, Jason Manley, David F. Moore, Irina I. Stefan, and William P. Walbrugh. Opening the 21 cm Epoch of Reionization Window: Measurements of Foreground Isolation with PAPER. *Astrophysical Journal Letters*, 768(2):L36, May 2013. doi: 10.1088/2041-8205/768/2/L36.

Jonathan C. Pober, Adrian Liu, Joshua S. Dillon, James E. Aguirre, Judd D. Bowman, Richard F. Bradley, Chris L. Carilli, David R. DeBoer, Jacqueline N. Hewitt, Daniel C. Jacobs, Matthew McQuinn, Miguel F. Morales, Aaron R. Parsons, Max Tegmark, and Dan J. Werthimer. What Next-generation 21 cm Power Spectrum Measurements can Teach us About the Epoch of Reionization. *Astrophysical Journal*, 782(2):66, February 2014. doi: 10.1088/0004-637X/782/2/66.

Jonathan R Pritchard and Abraham Loeb. 21 cm cosmology in the 21st century. *Reports on Progress in Physics*, 75(8):086901, Jul 2012. ISSN 1361-6633. doi: 10.1088/0034-4885/75/8/086901. URL http://dx.doi.org/10.1088/0034-4885/75/8/086901.

Benjamin Recht and Christopher Re. Beneath the valley of the noncommutative arithmetic-geometric mean inequality: conjectures, case-studies, and consequences. *arXiv e-prints*, art. arXiv:1202.4184, February 2012.

Christian L. Reichardt. *Observing the Epoch of Reionization with the Cosmic Microwave Background*, pages 227–245. Springer International Publishing, Cham, 2016. ISBN 978-3-319-21957-8. doi: 10.1007/978-3-319-21957-8_8. URL https://doi.org/10.1007/978-3-319-21957-8_8.

A. E. E. Rogers and J. D. Bowman. Spectral Index of the Diffuse Radio Background Measured from 100 to 200 MHz. *Astronomical Journal*, 136:641–648, August 2008. doi: 10.1088/0004-6256/136/2/641.

David E. Rumelhart, Geoffrey E. Hinton, and Ronald J. Williams. Learning representations by back-propagating errors. *Nature*, 323:533–536, 1986.

Rebecca L. Russell and Christopher Reale. Multivariate Uncertainty in Deep Learning. *arXiv e-prints*, art. arXiv:1910.14215, October 2019.

Shibani Santurkar, Dimitris Tsipras, Andrew Ilyas, and Aleksander Madry. How Does Batch Normalization Help Optimization? *arXiv e-prints*, art. arXiv:1805.11604, May 2018.

P. A. Shaver, R. A. Windhorst, P. Madau, and A. G. de Bruyn. Can the reionization epoch be detected as a global signature in the cosmic background? *Astronomy and Astrophysics*, 345:380–390, May 1999.

Hayato Shimabukuro and Benoit Semelin. Analysing the 21 cm signal from the epoch of reionization with artificial neural networks. *Mon. Not. Roy. Astron. Soc.*, 468(4):3869–3877, 2017. doi: 10.1093/mnras/stx734.

N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov. Dropout: A Simple Way to Prevent Neural Networks from Overfitting. *J. Mach. Learn. Res.*, 15 (1):1929–1958, January 2014. ISSN 1532-4435. URL http://dl.acm.org/citation.cfm?id=2627435.2670313.

Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. *arXiv e-prints*, art. arXiv:1312.6199, December 2013.

G. .B. Taylor, C. L. Carilli, and R. A. Perley, editors. *Synthesis Imaging in Radio Astronomy II*, volume 180 of *Astronomical Society of the Pacific Conference Series*, January 1999.

Andrea Tellini. Imperfect bifurcations via topological methods in superlinear indefinite problems. *Dynamical Systems and Differential Equations, AIMS Proceedings 2015 Proceedings of the 10th AIMS International Conference (Madrid, Spain)*, Nov 2015. doi: 10.3934/proc.2015.1050. URL http://dx.doi.org/10.3934/proc.2015.1050.

N. Thyagarajan, D. C. Jacobs, J. D. Bowman, N. Barry, A. P. Beardsley, G. Bernardi, F. Briggs, R. J. Cappallo, P. Carroll, A. A. Deshpande, A. de Oliveira-Costa, J. S. Dillon, A. Ewall-Wice, L. Feng, L. J. Greenhill, B. J. Hazelton, L. Hernquist, J. N. Hewitt, N. Hurley-Walker, M. Johnston-Hollitt, D. L. Kaplan, H.-S. Kim, P. Kittiwisit, E. Lenc, J. Line, A. Loeb, C. J. Lonsdale, B. McKinley, S. R. McWhirter, D. A. Mitchell, M. F. Morales, E. Morgan, A. R. Neben, D. Oberoi, A. R. Offringa, S. M. Ord, S. Paul, B. Pindor, J. C. Pober, T. Prabu, P. Procopio, J. Riding, N. Udaya Shankar, S. K. Sethi, K. S. Srivani, R. Subrahmanyan, I. S. Sullivan, M. Tegmark, S. J. Tingay, C. M. Trott, R. B. Wayth, R. L. Webster, A. Williams, C. L. Williams, and J. S. B. Wyithe. Confirmation of Wide-field Signatures in Redshifted 21 cm Power Spectra. *Astrophysical Journal Letters*, 807:L28, July 2015. doi: 10.1088/2041-8205/807/2/L28.

S. J. Tingay, R. Goeke, J. D. Bowman, D. Emrich, S. M. Ord, D. A. Mitchell, M. F. Morales, T. Booler, B. Crosse, R. B. Wayth, C. J. Lonsdale, S. Tremblay, D. Pallot, T. Colegate, A. Wicenec, N. Kudryavtseva, W. Arcus, D. Barnes, G. Bernardi, F. Briggs, S. Burns, J. D. Bunton, R. J. Cappallo, B. E. Corey, A. Deshpande, L. Desouza, B. M. Gaensler, L. J. Greenhill, P. J. Hall, B. J. Hazelton, D. Herne, J. N. Hewitt, M. Johnston-Hollitt, D. L. Kaplan, J. C. Kasper, B. B. Kincaid, R. Koenig, E. Kratzenberg, M. J. Lynch, B. Mckinley, S. R. Mcwhirter, E. Morgan, D. Oberoi, J. Pathikulangara, T. Prabu, R. A. Remillard, A. E. E. Rogers, A. Roshi, J. E. Salah, R. J. Sault, N. Udaya-Shankar, F. Schlagenhaufer, K. S. Srivani, J. Stevens, R. Subrahmanyan, M. Waterson, R. L. Webster, A. R. Whitney, A. Williams, C. L. Williams, and J. S. B. Wyithe. The Murchison Widefield Array: The Square Kilometre Array Precursor at Low Radio Frequencies. *Publications of the Astronomical Society of Australia*, 30, 2013. ISSN 1448-6083. doi: 10.1017/pasa.2012.007. URL http://dx.doi.org/10.1017/pasa.2012.007.

Ba-Hien Tran, Simone Rossi, Dimitrios Milios, and Maurizio Filippone. All You Need is a Good Functional Prior for Bayesian Deep Learning. *arXiv e-prints*, art. arXiv:2011.12829, November 2020.

Aparna Venkatesan. The Optical Depth to Reionization as a Probe of Cosmological and Astrophysical Parameters. *The Astrophysical Journal*, 537(1):55–64, Jul 2000. ISSN 1538-4357. doi: 10.1086/309033. URL http://dx.doi.org/10.1086/309033.

Pablo Villanueva-Domingo and Francisco Villaescusa-Navarro. Removing Astrophysics in 21 cm Maps with Neural Networks. *Astrophysical Journal*, 907(1):44, January 2021. doi: 10.3847/1538-4357/abd245.

J. C. Walter and G. T. Barkema. An introduction to Monte Carlo methods. *Physica A Statistical Mechanics and its Applications*, 418:78–87, January 2015. doi: 10.1016/j.physa.2014.06.014.

Yihao Zhou and Paul La Plante. Understanding the Impact of Semi-Numeric Reionization Models when using CNNs. *arXiv e-prints*, art. arXiv:2112.03443, December 2021.