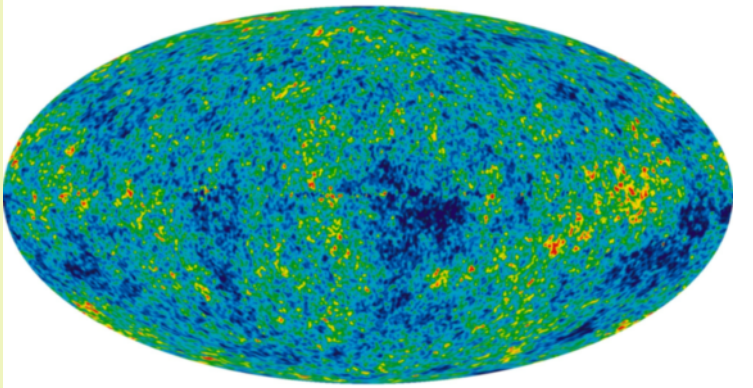
Extraction of the Cosmic Microwave Background Radiation from mm/submm Observations

Bachelor's Thesis



Frederik Gjelstrup
15 June 2017

Supervisor:
Hans Ulrik Nørgaard-Nielsen

DTU Space
National Space Institute

[This page intentionally left blank]

Abstract

This project focuses on extracting the cosmic microwave background radiation temperature anisotropy from the three main interfering foreground components: Synchrotron emission, free-free emission and thermal dust radiation. As the Planck spacecraft is the newest mission focusing on measuring the cosmic foreground emission at mm/submm wavelengths, the satellites accuracy will also be considered.

To extract the cosmic microwave background radiation temperature anisotropy, an artificial neural network will be used. In order to full understand all factors influence on the accuracy of the extraction, the network will be investigated over multiple trails to find the optimal settings.

It was found that a neural network with two hidden layer, respectfully containing 20 and 10 nodes, was the optimal combination. When extracting the cosmic microwave background radiation temperature anisotropy from all interfering foreground components and the Planck satellites sensitivity with a scaling factor of 1, the network was able to produce an output with an accuracy of ± 3.21 μK.

Contents

[This page intentionally left blank]

1 Introduction

The cosmic microwave background (CMB) radiation has for many years been the pinnacle of cosmological investigations as this shapes our understanding of the cosmos. Gaining knowledge of this radiation emission can give important insight into the the early ages of the universe as it this radiation is a remnant of the big bang, the first event of the universes life. After the first prediction of it's existence in 1948 and its discovery in 1964, much progress has been made after multiple observational missions and many studies.

In recent years missions such as the Cosmic Background Explorer (COBE), Wilkinson Microwave Anisotropy Probe (WMAP) and the most recent Planck space observatory, have made ground breaking measurements giving a most accurate data of the CMB temperature anisotropy to date.

While the main purpose of these missions have been to measure the CMB temperature anisotropy through observations of radiation in the mm/submm frequency spectrum, other cosmic components also emits radiation in this frequency range threatening to contaminate the observational data. These interfering radiation emissions are a result from several foreground components.

Many methods have been investigated over the years for extracting the CMB radiation from other contaminating foreground emissions. Some examples are mentioned in H.U. Nørgaard-Nielsen [1] :
- Maximum Entropy Method (MEM): Hobson et al. (1998), Stolyarov et al. (2002), Barreiro et al. (2004), Stolyarov et al. (2005)
- Internal Linear Combination method (ILC): e.g. Bennett et al. (2003b)
- Wiener Filtering: e.g. Tegmark & Efstathiou (1996)
- Independent Component Analysis (ICA) method: e.g. Maino et al. (2002)

In this paper a new method of the extraction will be investigated. Specifically the use of an artificial neural network. This paper will focus on removing the radiation contribution from the three main foreground emissions, Synchrotron emission, Free-free emission and Thermal dust emission, from mathematical models. The models are furthermore based on observational data from the Planck spacecraft and as such it's sensitivity will also be considered.

2 Foreground components

2.1 Cosmic microwave background radiation

In the early life of the universe, after the big bang, radiation dominated over matter. After about 380.000 years the temperature fell to around 3000 K where hydrogen and helium atoms could begin to form. This is known as the era of recombination. The radiation field and the material in the universe became disconnected, meaning that the temperature of the radiation field and the temperature of matter is no longer the same. Photons which were previously constantly colliding with charged particles, can now flow uninterrupted through space. These photons now constitute the microwave background.[13]

The era of recombination marks an important time in the universes history, as this is the earliest time and farthest distance that astronomer's can collect images of the universe. The emitted photons from large groups of matter have little to no interaction with matter it passes through on its path to earth, carrying information of its sender. Photons emitted prior to the recombination however are so heavily scattered, that any image from before this era is completely blurred out before it reaches earth.

Through observations from missions such as COBE, the temperature of the CMB is today known to be 2.275 K. However even though this temperature is almost perfectly uniform, small variations across the sky can be measured from high precision equipment. This tells us that matter was not perfectly uniformly distributed in the early universe. The temperature variations thus tell us how matter was distributed, where there were larger clumps of matter and where there were voids. An important distinction of the temperature is where the anisotropies come from. The temperature fluctuations do not come from the temperature of the matter itself, but come from a gravitational redshift on the CMB photons. Redshifted photons carry less energy and thus a lower temperature. Lower CMB temperature therefore correspond to higher densities of matter. This can be seen in figure 1, which shows a map of the CMB temperature anisotropy across the sky, produced by WMAP data.

As the CMB has the spectrum of a blackbody, it will be modelled as such for the purpose of the neural network. The frequency dependant flux from a blackbody can be modelled from Plancks law as seen by equation 1. [1]

$$B_\nu(\nu, \mathrm{T}) = \frac{2h\nu^2}{c^2} \frac{1}{e^{\frac{h\nu}{k_\mathrm{B}\mathrm{T}}} - 1} \tag{1}$$

Where ν is a frequency, h is the Planck constant, c is the speed of light and k_B is the Boltzmann constant. As it will be used to model the CMB, the dependant temperature T is the CMB temperature.

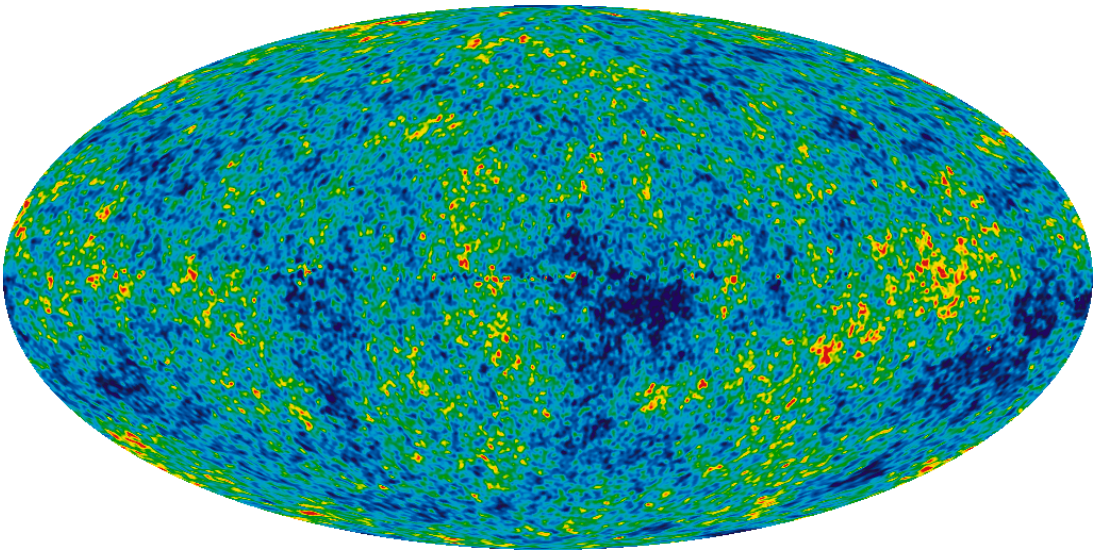


Figure 1: A map of the cosmic microwave background radiations temperature anisotropy from WMAP data. This image has been taken from Galaxy [13]

2.2 Synchrotron

Synchrotron is an electromagnetic radiation caused by a radial acceleration on a charged particle perpendicular to its velocity, i.e. $\mathbf{a} \perp \mathbf{v}$. This acceleration is caused if the particle is subjected to a magnetic field. Specifically astronomic synchrotron emission is caused by electrons moving at relativistic velocities, being forced to spiral due to magnetic fields.[9]

One of the more popular examples of astronomic synchrotron emission can be seen in the crab nebula, where the blue light from the center is due to high energy electrons moving along magnetic field lines caused by a nearby pulsars emitted radio source.

Measurements for radio frequencies less than 2 GHz are dominated by the synchrotron emission. This foreground contribution is particularly prominent in low galactic latitudes. At higher latitudes some contributions can be measured from the galactic halo and the North Polar Spur.
The frequency dependent synchrotron emission can generally be modelled as a single power law[2]

$$S_s(v) = A_s \left(\frac{v}{v_0} \right)^{\alpha_s} \tag{2}$$

Where A_s is the synchrotron flux density and α_s is the synchrotron spectral index, both at a reference frequency v_0. As the synchrotron flux density and spectral index both vary across the sky, these are given as a range instead of a constant. According Eriksen et al. [2], the spectral index is likely to lie between -0.7 and -1.2 at galactic latitudes $|b| > 15°$ at frequencies larger than 10 GHz. However as the synchrotron's spectral index is dependant on the interstellar magnetic fields and is expected to steepen by 0.1 at higher frequencies. [1]

2.3 Free-free emission

The so called Free-free emission comes from thermal bremsstrahlung from the hot electrons ($\gtrsim 10^4$K) which are produced when an interstellar gas is subject to an UV radiation field. In the milky way this occurs when the free electrons and ions interact.
Of the three foreground components, free-free poses the grates threat to the validity of CMB observations. This is due to one of two reasons. Firstly the free-free emission dominates where the other foreground components are at a minimum, which is where most CMB observations are made. Secondly it dominates over a narrow range of frequencies causing mapping and simulating to be a difficult task as it can't easily be traced at high or low frequencies. [5]

Similarly to synchrotron, the free-free emission can be usually modelled as a single power law [2]

$$S_{ff} = A_{ff} \left(\frac{v}{v_0} \right)^{\alpha_{ff}} \tag{3}$$

The spectral index is expected to be $\alpha_{ff} = -0.14$ at frequencies relevant for Planck. However a constant spectral index may not be entirely accurate and as such a range of $\alpha = -0.1 : -0.2$ has been chosen for frequencies between 10 and 100 GHz. [1]

2.4 Thermal dust emission

The last of the three main foreground contributions contaminating the CMB observations is due to grains and dust particles in the interstellar medium. These particles are heated by the interstellar radiation field absorbing radiation in the optic and UV spectra, after which they emit a thermal radiation detectable in the radio frequency band. How much the dust particles emits compared to how much they absorb depends on the dust particles composition, though the emission typically ranges from 18-20K. [9]
Various models have been presented to simulate the thermal dust emission. One such is an extrapolations of the high-frequency IRAS, COBE DIRBE and FIRAS observations at CMB frequencies. [3] This model is capable of simulating multiple dust properties, such as varying dust sizes, chemical composition and grain properties. However as this model is a combination of modified blackbody functions at different dust temperatures it requires six parameters. This would require a higher frequency resolution than any current CMB observational experiments can provide. A simpler model has therefore been derived as a modified power law, which has been combined with a slowly decreasing function of the order unity over the frequencies of interest. [2]

$$S_d \sim \left(\frac{exp\left(\frac{hv_{0,d}}{kT_1}\right) - 1}{exp\left(\frac{hv}{kT_1}\right) - 1} \right) \left(\frac{v}{v_{0,d}} \right)^{\alpha_d + 1} \tag{4}$$

2.5 Parameters

As all foreground models are dependent on where on the sky they are being measured, their parameters will vary. Table 1 shows the range of each parameter which will be used for the models. The value ranges for synchrotron, free-free and thermal dust have been taken from H.U. Nøregaard-Nielsen (2008) [1], while the CMB temperature range has been increased to ±850K as this follows the presented model better. The galactic component's flux amplitudes are given in units of 10^{-20}erg s^{-1}cm^{-2}Hz^{-1}sr^{-1}, the CMB temperature is given in μK and the galactic components spectral indices are unitless.

Component	Amp range	Spec index range
Synchrotron	0 : 1.0	-1.2 : -0.6
Free-free	0 : 2.0	-0.2 : -0.1
Thermal dust	0 : 8.0	2.75 : 3.75
CMB temp	-850 : 850	-

Table 1: Flux amplitudes and spectral indices used for model parameters. Values are taken from H.U. Nørgaard-Nielsen [1]

2.6 Measuring the cosmic microwave background radiation

Many experiments and missions have been made in order to accurately measure the CMB radiation, the most recent one being the ESA medium class mission: Planck. This satellite was launched in 2009 with the purpose of measuring the CMB radiation at infra-red and microwave frequencies. The space observatory carries two detector systems: a low frequency investigator (LFI) and a high frequency investigator (HFI). The LFI is based on HEMT technology, while the HFI is based on bolometers. [1]
These two detector systems will together produce an all-sky map at nine frequencies covering a range of 30-857 GHz. This frequency range matches the cosmological window, where the CMB emission peaks and the other foreground emissions are minimal, as seen on figure 2.

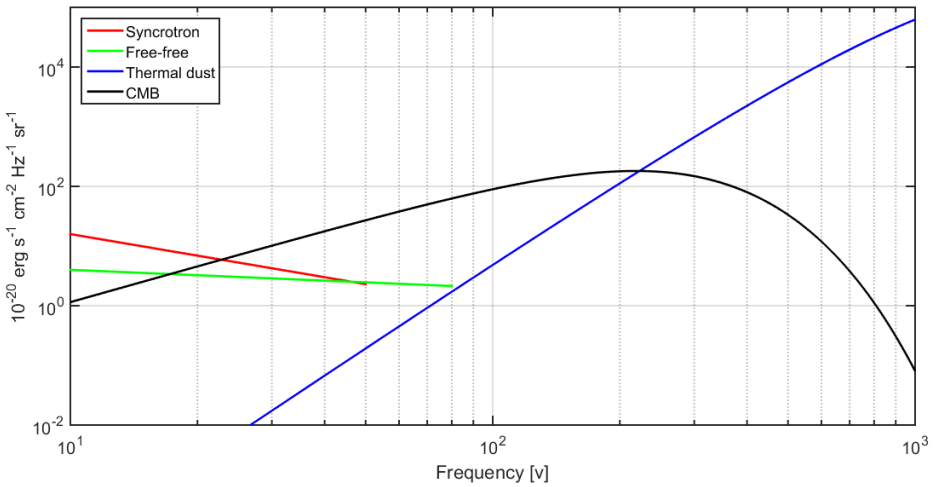


Figure 2: All four foreground components modelled. Recreated after figure 3 in H.U. Nørgaard-Nielsen [1]

2.6.1 Noise

As the purpose of this paper is to extract the CMB radiation from the total foreground radiation, one must take into account the accuracy of which measurements can be made. The expected sensitivity of the two Planck detector systems at all nine frequencies are shown in table 2, taken from H.U. Nøregaard-Nielsen (2008) [1]. It is reasonable to assume that the sensitivities at all nine frequencies will scale by the same factor, f_{sen}, due to the arrangement of the detectors on the focal plane and the scanning strategy. The scaling factor is dependent on the detector systems exposure time.

	LFI	LFI	LFI	HFI	HFI	HFI	HFI	HFI	HFI
Center frequency (GHz)	30	44	70	100	143	217	353	545	857
Bandwidth ($\Delta v/v$)	0.2	0.2	0.2	0.33	0.33	0.33	0.33	0.33	0.33
Angular resolution (arcmin)	33	24	14	10	7	5	5	5	5
1σ sensitivity	0.22	0.38	0.90	0.58	0.61	1.19	2.25	4.34	5.22
1σ sensitivity (μK)	6.76	6.71	6.77	2.43	1.61	2.46	7.59	76.09	3644.24

Table 2: Summary of the Planck instrument characterization for a sky pixel with average exposure time, taken from H.U. Nørgaard-Nielsen [1]

3 Neural Network

An Artificial Neural Network (ANN) is a technique used to process large amounts of complex data, which falls under a broader group of statistical tools known as Data Mining. This particular data mining technique is heavily inspired by the biology of the nervous systems, e.g. the brain, and the way these process information. Similar to the brain, the ANN utilizes a large quantity of interconnected processing elements named neurons to solve difficult tasks. This technique works by learning by example, much alike how a biological nervous system would.

One of the biggest and most important advantages of Neural Networks are their outstanding ability to find patterns or tendencies in complex or imprecise data. The way ANNs differ from the classical computational problem solving is that a precise method is not needed. When a conventional computer solves a problem, every step to solving this must be known in advance and be described in well defined steps for it to follow. The ANN however, can't be preprogrammed to solve specific tasks, but rather solves problems by learning from examples, similar to how the brain processes information. The given examples must be chosen with care, as a poorly chosen one could result in unnecessary computing time or even faulty results. Because the ANN teaches itself how to solve a specific problem, its method of calculation can be unpredictable. The actual computation is rarely shown to the user, hidden behind a so-called "black box", which is why many are reluctant to utilize it.

3.1 The human brain vs the ANN

The nervous system is a highly complex system of cells called neurons working in unison. Neurons communicate with each other through *dendrites*, *axons* and *synapses*. The axon is the neurons transmitter. It is a long thin fiber through which the neuron sends signals as an electrical charge, which splits up into thousands of branches each connecting to anther neuron. A neuron gathers signals from other neurons axons through dendrites. Dendrites are the neurons receivers layed out as a host of fine structures, each connected to an axon, which sends the received signal to its neuron.
The point where the axon is connected to the dendrite is called the synapse. At the end of the axon lies the presynaptic terminal, also known as the axonal, where chemical signals are sent to the dendrite. These signals are then processed up by a receptor on the dendrite. The chemical signals determine whether or not the signal from the axon is transferred to the dendrite. This is at its core how the system learns. By adjusting the criteria of whether or not the synapse will let a signal pass is how it adjusts the influence of the surrounding neurons. An illustration of how this look can be seen on figure 3. In reality an abundance of neuron types exist, such as the sensory neurons which react to physical stimuli such as sound, light or touch, and motor neurons which send signals to muscle cells or glands. [7]

As the nervous system is so convoluted, with many complex parts, a computer of today's standards wouldn't be able to accurately simulate this. Therefore when creating an artificial neural network, the biological nervous system has been greatly simplified and only the core concepts of its method of function have been used. An illustration of how the

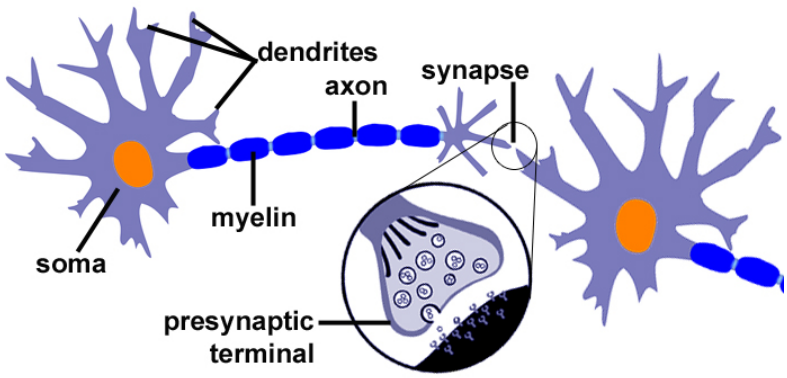


Figure 3: Illustration of a neuron in the human brain. [15]

individual neurons have been simulated can be seen in figure 4
Each neuron will receive multiple inputs but will output the same signal to each connecting neuron. The neuron has two modes: a training-mode where it learns certain patterns from examples and a use-mode where it defines decides the output on the background of its training.

Multiple types of neurons have been created through time. One of the more advanced types is the McCulloch and Pitts model (MCP) which includes weights, that simulate the synapse of the biological neuron. Each input is multiplied by a weighted value before all inputs are summed. This results in each input having an individual influence based on its importance, whereas all inputs were valued equally before the introduction of weights.

When a signal reached the neuron a process similar to the synapse is initiated. The neuron will determine whether or not to send the signal through. For the MLP type neuron, a threshold determines this. This threshold can like the individual weights adapt. Each weighted input is summed and if the total sum exceeds a specified threshold, the signal is processed. If the weighted sum does not exceed the threshold, the signal stops here. Mathematically the computation is as such [7]

$$\sum_i x_i w_i > T \tag{5}$$

To compare to the nervous system, the weights represent the chemical signals from the presynaptic terminal, where the chemical signal depends on previous experiences and the signal from the axon, and the threshold represents the receptor on the dendrite, which will only send the received signal through if the received chemical signal meets a certain criteria.

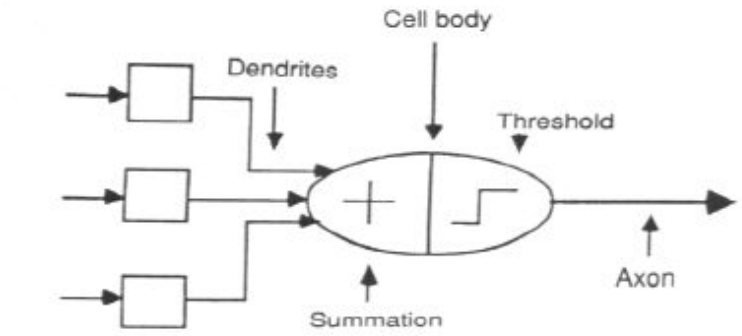


Figure 4: An illustration of how biological neurons are simulated for ANNs. [7]

3.2 Neural network structures

With the neurons defined, the structure of the network must be specified. How the neurons communicate is important for the outcome and different tasks benefit from different structures of network.

The simplest structure is the *Feed-forward* network. Networks of this type have no feedback to neurons on the same or previous layers. The output is therefore often directly related to the input, making them optimal for tasks such as pattern recognition. An example of a feed-forward network can be seen on figure 5. Feed-forward networks rarely use weighted neurons such as the MCP model, but rather the neurons will receive a binary input and send a single digit binary output depending on the pattern of the received input.

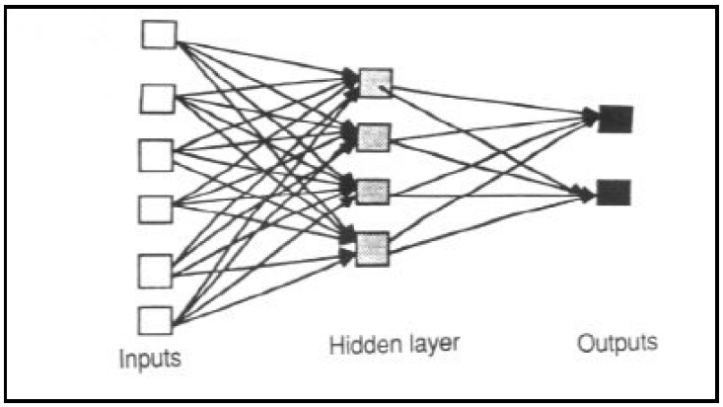
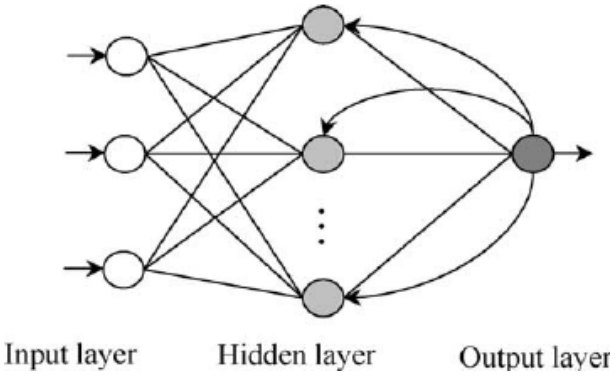
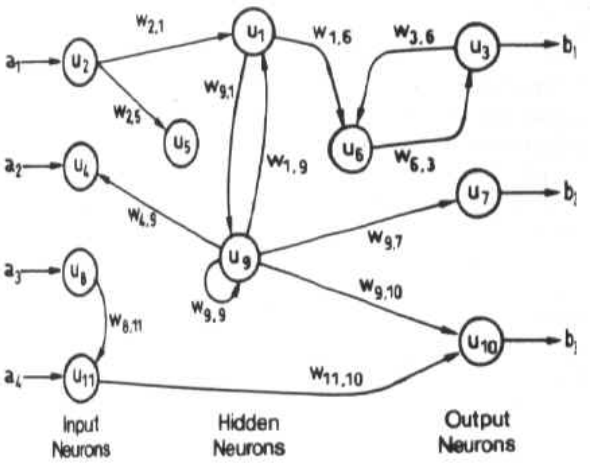


Figure 5: An example of a Feedforward neural network with six input nodes, four nodes in the hidden layer and two output nodes. [7]

A more versatile construction is the *Feedback* network also named a *recurrent* network. As the name suggests this network is influenced by the outcome, constantly adapting by adjusting the individual weight's value. Once an input has been given, the network will change until it has reached an equilibrium point and retain the newly found weights until the input is changed. Networks of this construction can vary from simple back propagation as seen on the example in figure 6 (a) to highly complex systems as seen on the example in figure 6 (b). These types of network are useful in may scenarios as they are able to adapt to many situations.



(a) A simple Feedback neural network, where the output node is connected to the hidden layers nodes. [14]



(b) A more complex Feedback neural network. [7]

Figure 6: Two examples of feedback neural networks

3.3 Learning process

As mentioned, MCP Neural networks capability to learn and adapt lies in the weights ability to change value and adjusting the neurons threshold value. How a specific network learns, i.e. memorizes patterns or adjusts the weights, depends on the networks task. Generally a networks method of learning can be divided into to groups: **associative mapping** and **regularity detection**.

Associative mapping determines the output based on a given training example, giving it's best approximation of a pattern. This can be done as auto- or hetro-association. Auto-association compares the input pattern with itself and the sates of input and output units follow one another. This is used in pattern completion where parts of the given pattern is either distorted or missing entirely. Hetro-association finds patterns by association the given input with the training example, either by "nearest neighbour" where the output is a pattern most similar to the taught pattern, or by "interpolative" where the output is an interpolation of the best fitting patterns from the training.

Regulatory detection causes the units to react to certain qualities of the given input. Where associative mapping only saves the relationships between certain patterns, regulatory detection gives each elements response to a pattern a particular definition. This learning process excels at feature discovery and knowledge representation.

The learning process can either happen exclusively during the training phase so all weights are predetermined without the ability to change during the networks use-mode, or it can adjust the predetermined weights during the networks use-mode. The first is called a *Fixed* network, while the latter is called a *Adaptive* network.
When working with adaptive networks, it is important to distinguish between the two general categories of adaptive learning: Supervised and unsupervised learning.

Supervised learning requires an external teacher, which tells each output what value it should be compared to the given input, a target output. This type of learning causes each weight to try and find a value which minimises the difference between output and target. One common method it the least mean square convergence.

Unsupervised learning requires no external teacher. It organises the given inputs to find collective properties. This type of learning takes place while the network is in use.

While the values of weights and thresholds determine whether or not a neuron fires, the behaviour of these neurons is also determined by their activation function. There are generally three function types used for MCP ANNs: linear, threshold or sigmoid.
Linear neurons output is directly equal to the sum of weighted inputs.
Threshold neurons output is one of two predetermined values, where the chosen of the two depends on whether or not the summed weighted input exceeds a threshold decided under the training phase.
Sigmoid neurons output is the summed weighted input inserted in a chosen activation function. This activation function can be any sigmoid function. Often the logistic function is chosen, which values lie somewhere between 0 and 1 depending on the input.

4 Applied neural network

When creating a neural network one must first define the input and target output. This particular network will be constructed to create an algorithm for isolating the CMB temperature anisotropy from the foreground emission measured at the nine Planck frequencies shown in table 2, i.e. $[obs_i, i = 1, 2 \cdots 9]$.

For the computation of the Neural Network MatLab has been used as it has pre-designed Neural Network functions. Because of this it is only necessary to define the desired networks specifications, such as the number of hidden layers and the nodes in each of these. As for the type of network, MatLabs `fitnet` function has been used. The `fitnet` function creates a feedforward network focused on approximating non-linear functions from a given set of data points. specifically for this task, it will approximate a function to find a temperature from a set of modelled data points.

Choosing the amount of hidden layers and number of nodes in each of these can be difficult as there is no definite rule as to how these are chosen. For networks with a single hidden layer, a few rules of thumb exist such as the amount of nodes should be about two thirds of the combined number of input and output nodes. For this network two hidden layers will be used as this requires less total nodes and can often solve complex systems better than networks with only one hidden layer. The only sure way of selecting an optimal amount of nodes i each of the hidden layers, is by first making a reasonable guess and then re-training the network while increasing or decreasing the amount of nodes and examine the results.
The first trial of the network will be run with two hidden layers with respectfully 12 and 3 nodes are used, similar to the network used in [1].

The networks transfer function has been chosen to be the hyperbolic tangent sigmoid transfer function. In Matlab this is called the `tansig` function as its output lies between -1 and 1 forming an S-like shape around 0, as opposed to Matlabs default choice, the `logsig` function, which values lie between 0 and 1. The tansig function is shown in equation 6 and its function value can be seen in figure 7.

$$\text{tansig}(x) = \frac{2}{1 + \mathrm{e}^{-2x}} - 1 \tag{6}$$

For calculating and updating weights, the Levenberg–Marquardt backpropagation algorithm will be utilized. Among the available algorithms in the Matlab toolbox, the Levenberg–Marquardt is often the fastest and is therefore commonly the first choice of the available algorithms, although it requires more memory than others.
The Levenberg–Marquardt algorithm is designed similar to Newtons method of backpropagation but instead of completing the full calculation of the Hessian matrix, the Levenberg–Marquardt algorithm approximates it, cutting calculation time significantly. [11]

During training, the network will use a portion of the given dataset for validation and a portion for testing. The network uses the validation portion to measure the network generalization and to halt training when generalization stops improving. The test portion has no effect on the training and is used only to provide an independent measure of the networks quality during and after training. Both of the training and validation portions

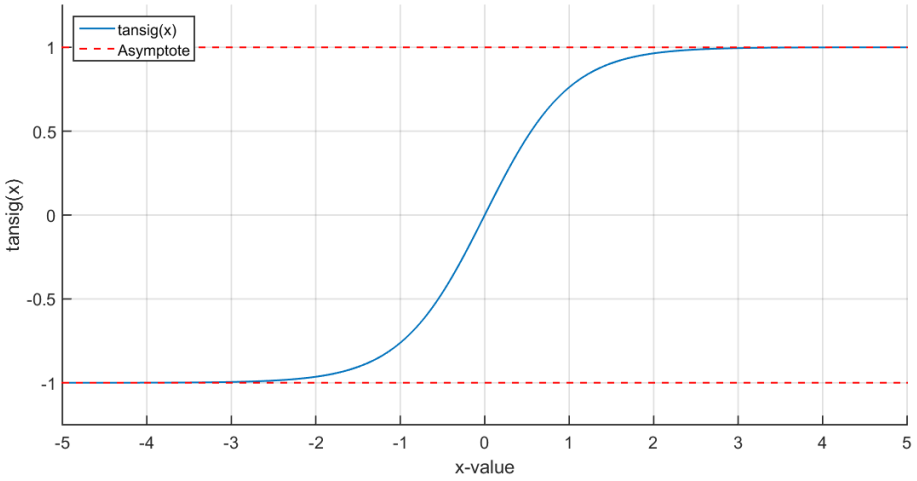


Figure 7: The tansig function for x-values -5 to 5

constitute 15% of the total training dataset each, while the remaining 70% of the data set is used exclusively for the actual training of the network.

The validation criteria in Matlabs neural network functions have default values, though some of these have been altered to ensure a high level of performance of the network. The maximum number of iterations has been set to 250 as the increase in performance per iteration falls to an insignificant amount after this. Figure 8 shows the performance after each iteration and as it can be seen, the increase in performance slows gradually until further training would have little to no effect. The performance is calculated as a mean squared error. To ensure that the network doesn't stop its computations too soon, a minimum gradient has been selected at 10^{-12}. This determines the trainings rate of improvement and a gradient lower than 10^{-12} would result in little to no improvement of the networks performance.
The Neural Network training tool from Matlab can be seen in figure 9, after the network has been trained. The structure of the network is illustrated in the top, the various algorithms can be seen in the middle and the progress with validation criteria can be seen in the bottom. As it can be seen, the training completed all 250 iterations before halting training.

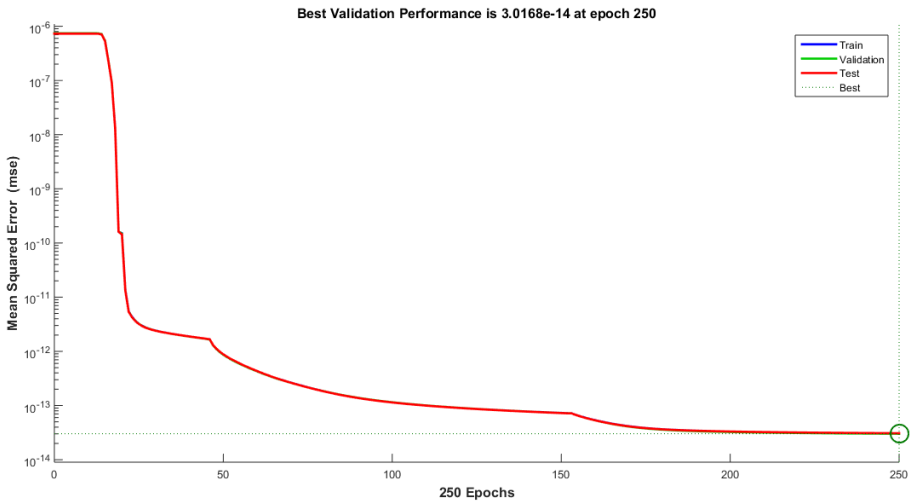


Figure 8: The performance of the network per training iteration, as generated by Matlabs `Neural Network Training` toolbox

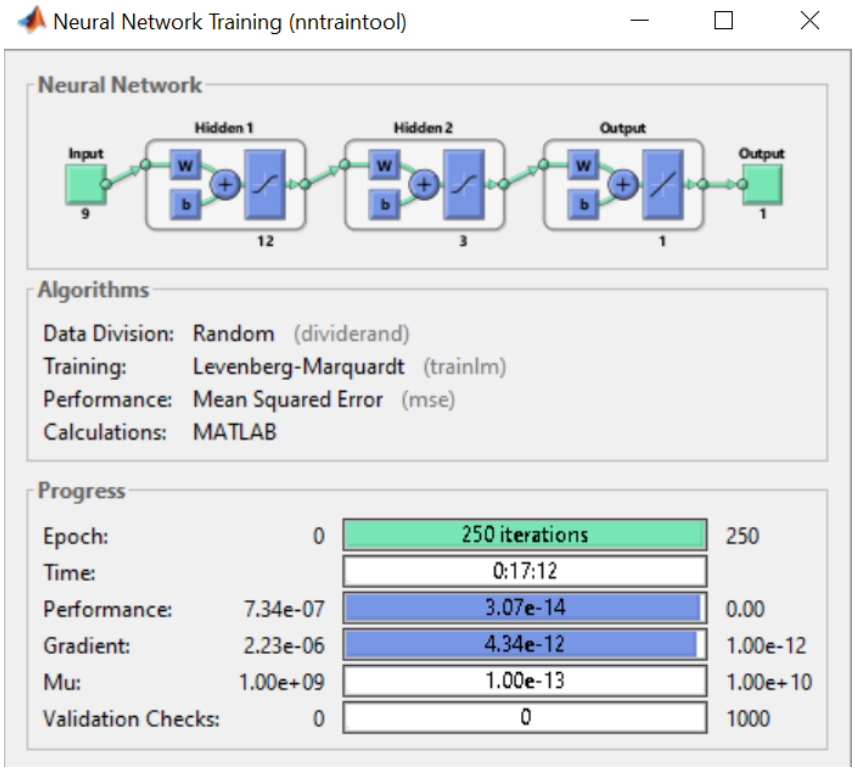


Figure 9: Matlabs `Neural Network Training` toolbox after fully training a network

4.1 Training data for the network

Before the network can be trained, it must be given an example to train from. The network will be given a dataset containing n number of examples, where each example is a modelled foreground emission at the nine Planck frequencies and an expected CMB anisotropy temperature.
The example training data is created following these steps:

1. Randomly select a value for each of the seven parameters from a uniform distribution of the ranges given in table 1
2. Calculate the combined flux at the nine Planck frequencies, of the foreground components as a sum of the models described in section 2 using the parameter values chosen in step 1.
3. For each frequency add a Gaussian randomly distributed number multiplied by the frequencies sensitivity 1σ from table 2 multiplied by the a chosen scaling factor f_{sen}
4. The selected CMB temperature is stored in a separate vector as it is to be used as a target.

The network will thus receive an input containing the foreground emission measured in 10^{-20} ergs^{-1}cm^{-2}Hz^{-1}sr^{-1} while the output will contain a vector of the CMB temperature anisotropy measured in K.

This will result in a dataset of the size $[9, n]$ and a vector with targets of length n, where n is the number of times the steps are repeated. If n is too small, the network won't have enough examples to train from and will result in poor fitting. If a too high n is chosen, the network will use an unnecessary amount of time to train without an significantly improvement in the networks quality. The influence of the training data size will be investigated in section 5.1.2.

5 Results

Now that neural networks specifications have been decided and the method of creating a training dataset has been described, the network can be tested. As there are many factors that each have a different influence on the networks quality, such as the number of hidden layers and the scaling factor f_{sen}, the network will be tested multiple times while altering these factors to determine their influence.

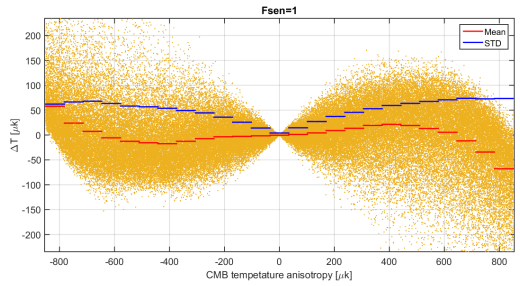
5.1 Neural network without interfering foreground components

As the purpose of the network is to find the Cosmic Microwave Background Radiations variation in temperature, the first test of the network will be of only the modelled CMB, without any of the interfering foreground components, only adding the sensitivity from the Planck satellite. This is to test the quality and stability of the network. From this possible improvements can be explored, such as a different combination of nodes in the hidden layers or the significance of the amount of available data.

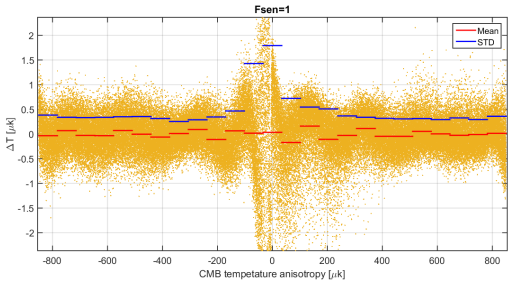
The quality of the networks output can be determined by how close it is to the given target values. To get a clear visual perspective, the difference between output and target values will be plotted, i.e. $\Delta \mathrm{T}_i = \mathrm{Target}_i - \mathrm{Output}_i$ for $i = 1, 2...n$

The focus of the networks first examination is to check for any systematic errors. To search for evidence of this, ΔT will be plotted along an axis of the expected CMB temperature anisotropy. The expected outcome of this plot, if no systematic errors exist, is a constant standard deviation, 1σ, along the x-axis and a constant mean of zero. Signs of systematic errors could e.g. be if the ΔT value seems to follow a trend along the x-axis or if the standard deviation increase at certain values. An illustration of the first example can be seen in figure 10 (a) and of the latter example in figure 10 (b). These two have purposely been created to illustrate systematic errors and to compare the actual networks output with.

The result of the networks first trail can be seen in figure 11. This network has been trained with a dataset of $n = 100.000$ and the scaling factor chosen to $f_{sen} = 1$. The figure shows that the difference between output and target value ΔT is evenly distributed with a seemingly constant standard deviation, 1σ. This indicates that the network isn't the subject to any systematic errors and is thus accepted as a suitable network.



(a) A system with a systematic error, where a clear trend can be seen



(b) A system with a systematic error, where the standard deviation of ΔT increases at low values of teh CMB temperature anisotropy

Figure 10: Two examples of ΔT from networks with systematic errors

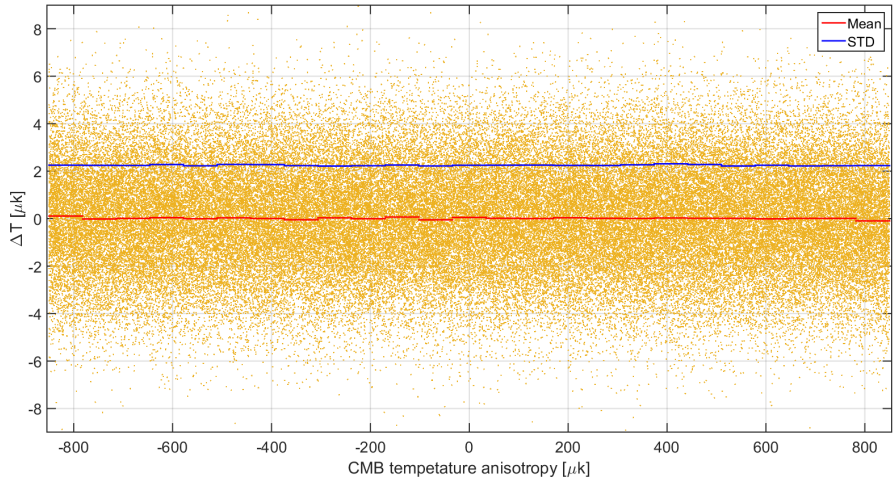


Figure 11: The first trial of the neural network, when extracting the CMB radiation temperature anisotropy from only the Planck sensitivity. The standard deviation and mean of the CMB temperature anisotropy is show for small intervals

Additionally investigation of the skewness and kurtosis of ΔT can further validate the authenticity of the network. The ΔT is expected to be Gaussian distributed, and these two values will prove whether or not this is the case.
The Skewness is a measure of asymmetry around the data's mean. A negative skewness would mean ΔT spread out more at values lower than its mean, while a positive skewness would mean ΔT is spread out more at values higher than its mean. The skewness of a normal distribution is zero. The skewness has been calculated with Matlabs `skewness(x)` function, which calculates the skewness as:

$$s = \frac{\frac{1}{n}\sum_{i=1}^{n}(x_i - \overline{x})^3}{\left(\sqrt{\frac{1}{n}\sum_{i=1}^{n}(x_i - \overline{x})^2}\right)^3} \tag{7}$$

The kurtosis is a measure of how prone the distribution is to outliers. A normal distribution will have a kurtosis value of 3, while a distribution more prone to outliers will have a greater value and a distribution less prone will have a lesser value. The kurtosis value of ΔT has been calculated with the Matlab function `kurtosis(x)`, which calculates the kurtosis as:

$$k = \frac{\frac{1}{n}\sum_{i=1}^{n}(x_i - \overline{x})^4}{\left(\frac{1}{n}\sum_{i=1}^{n}(x_i - \overline{x})^2\right)^2} \tag{8}$$

For the network in figure 11 the kurtosis value has been calculated to 3.0060 and the skewness has been calculated to 0.0012. ΔT can therefore be validated as a normal distribution along the axis of CMB temperature anisotropy, as its deviation from a perfect normal distribution first occurs on the third decimal on both skewness and kurtosis, which is indistinguishable from a perfect normal distribution in any practical sense.

5.1.1 Influence of scaling factor f_{sen}

The Planck sensitivities scaling factors influence has yet to be determined for this iteration as it is not a given constant, but rather lies within a known range depending on the detector systems exposure time. The network will therefore be tested for varying magnitudes of of this. The scaling factor is assumed to lie within the range of $0.6 - 1.6$ across the sky [1], however to clearly see the significance of its influence the network will be examined for a broader range. The network will be trained for:

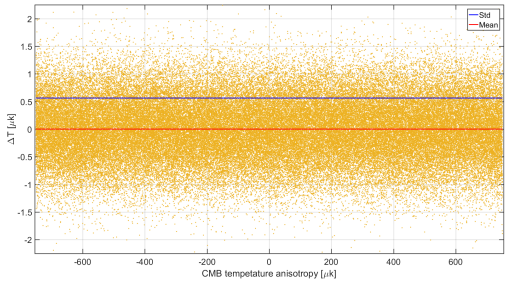
$$f_{sen} = \begin{bmatrix} 0.25 & 0.5 & 1 & 2 & 5 & 10 \end{bmatrix}$$

The mean error is expected to stay at zero no matter the sensitivity, while the 1σ is expected to increase with increasing sensitivity. The six iterations of the network can be seen in figures 12 (a)-(f). The output follows the expectation very well as the 1σ increases with f_{sen} while the mean difference between target and output stays at 0.

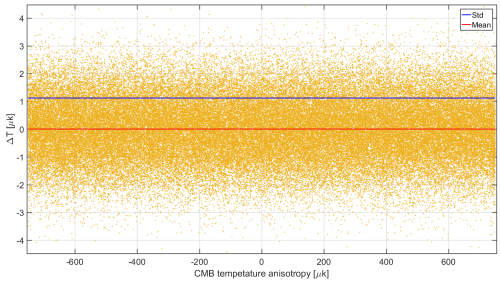
Exactly how the 1σ increases with f_{sen} can however be difficult to see in figure 12. The networks 1σ and mean is therefore shown in figure 13 along an axis of f_{sen}. It can be seen that the networks mean output stays close to zero, while 1σ increases linearly. A linear fit of this 1σ has been calculated to:

$$1\sigma_{\mu K} = 2.213 \cdot f_{sen} + 0.0089\mu\text{K} \tag{9}$$

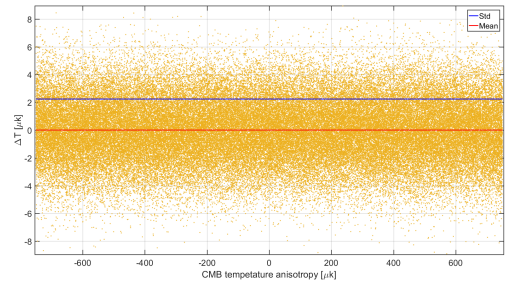
However due to the very nature of the Artificial Neural Network, the output of two identical networks trained with the same dataset and targets will be similar but never identical. To ensure stability, it is necessary to examine how much the output can vary at all f_{sen}. This is done by training the network multiple times for each f_{sen} and comparing them. In figure 14 the mean and the 1σ of ΔT for each repetition of the network can be seen for each of the f_{sen} values for 50 repetitions. Optimally all data points for the same value of f_{sen} would lie directly on top of one another as this would mean each training of the same network would produce the same result. It can be seen that the 1σ of ΔT follows the expected outcome fairly well, though the 1σ seems to vary increasingly with larger values of f_{sen}.



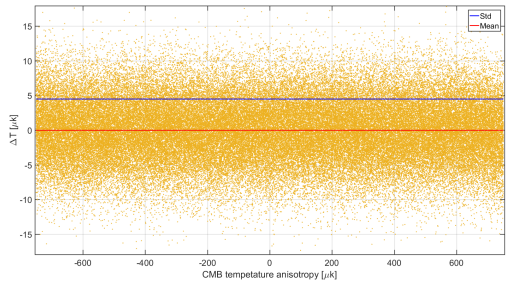
(a) $f_{sen} = 0.25 \quad 1\sigma = 0.562\mu\text{K}$



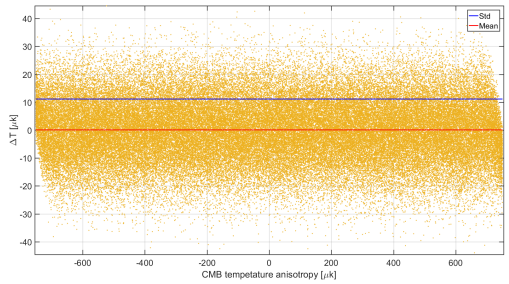
(b) $f_{sen} = 0.5 \quad 1\sigma = 1.119\mu\text{K}$



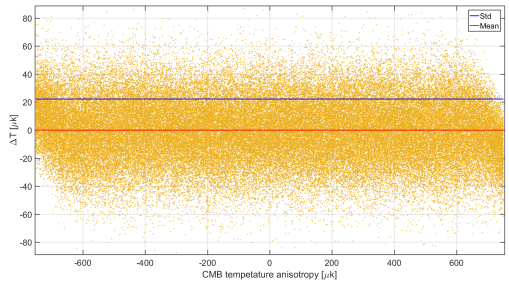
(c) $f_{sen} = 1 \quad 1\sigma = 2.239\mu\text{K}$



(d) $f_{sen} = 2 \quad 1\sigma = 4.477\mu\text{K}$



(e) $f_{sen} = 5 \quad 1\sigma = 11.133\mu\text{K}$



(f) $f_{sen} = 10 \quad 1\sigma = 22.199\mu\text{K}$

Figure 12: Each networks outputs ΔT, when extracting the CMB radiation temperature anisotropy from only the Planck sensitivity, on an axis of target values for six networks with different f_{sen} values

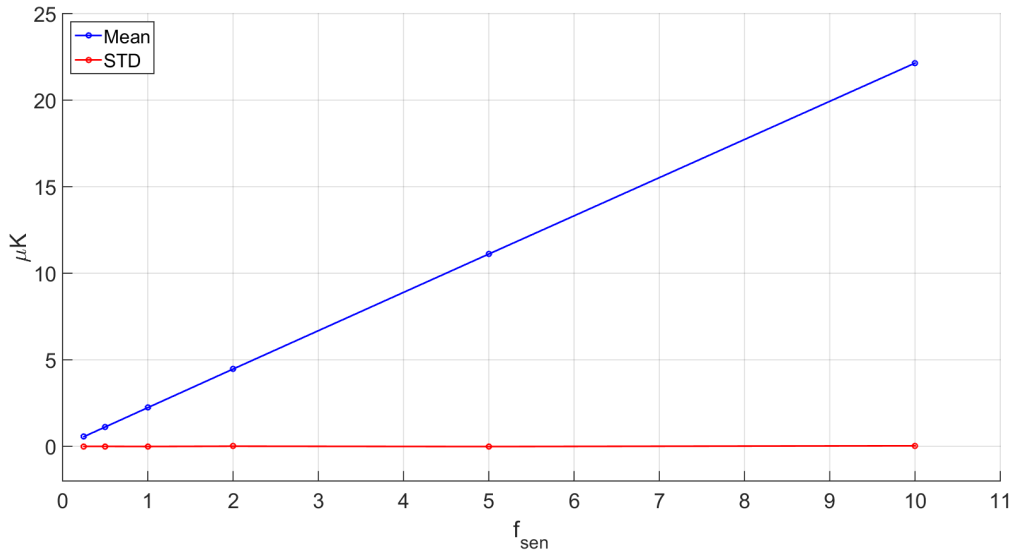


Figure 13: The mean and standard deviation of the networks outputs along increasing values of f_{sen}

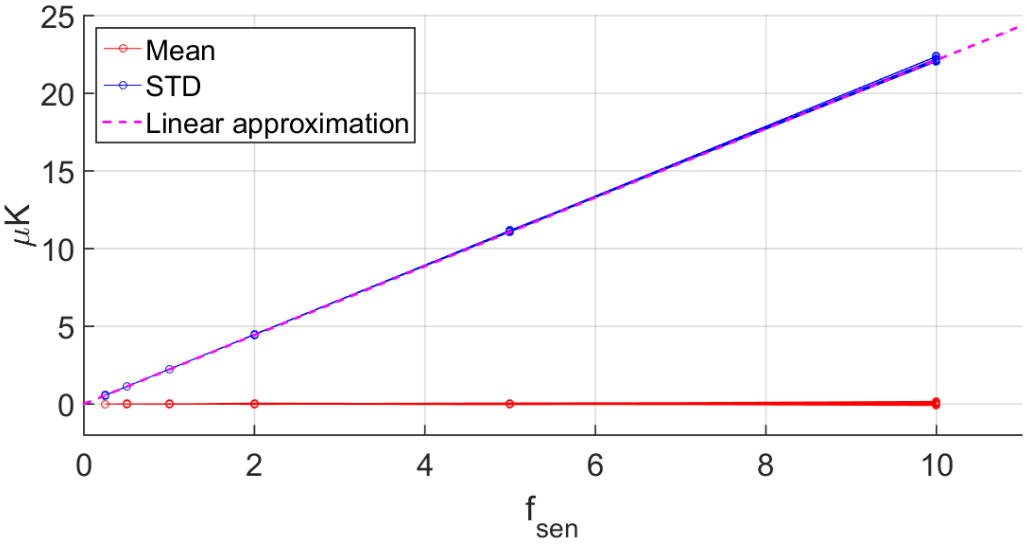


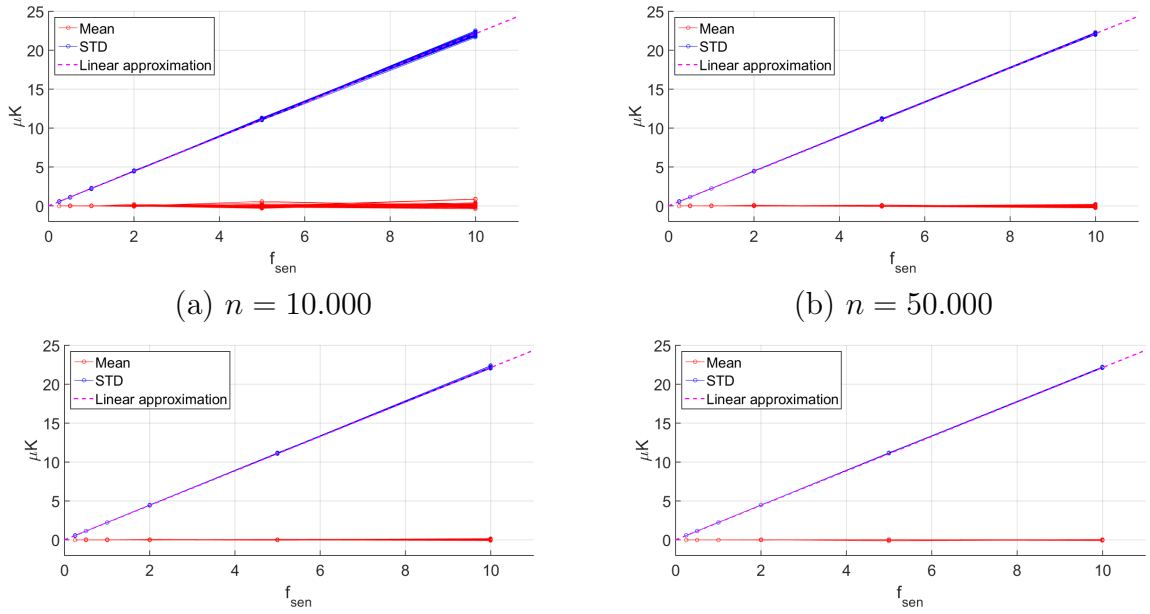
Figure 14: The outputs of 50 networks trained for increasing values of f_{sen}. The linear fit from equation 9 is also shown.

5.1.2 Influence of Training data size

Since the quality of the network directly depends on its training and thus on its training data, one must decide its size. To fully understand the influence of the training datasets size, the network will be trained with an increasingly larger dataset. For the sake of comparison a network will be trained multiple times for each value of f_{sen} for each four sizes of training data:

$$n = \begin{bmatrix} 10.000 & 50.000 & 100.000 & 500.000 \end{bmatrix}$$

On figure 15, the result of this can be seen for 20 repetitions of the the network. As can be seen ,the standard deviation and mean of ΔT seems more constant the greater the size of the training dataset. As the 1σ tells most about the overall quality of the network, the consistency of this will be investigated for each of the four trails. On figure 16, the standard deviation of $\sigma 1$ is shown for each of the four trials along the values of f_{sen}. It can be seen that a correlation exists between the training data's size and the consistency of network quality. The greater the amount of training data, the greater the consistency of the output.



(a) $n = 10.000$ (b) $n = 50.000$

(c) $n = 100.000$ (d) $n = 500.000$

Figure 15: The influence of four different sizes of training datasets each shown with 50 networks trained for increasing values of f_{sen}, when extracting the CMB radiation temperature anisotropy from only the Planck sensitivity.

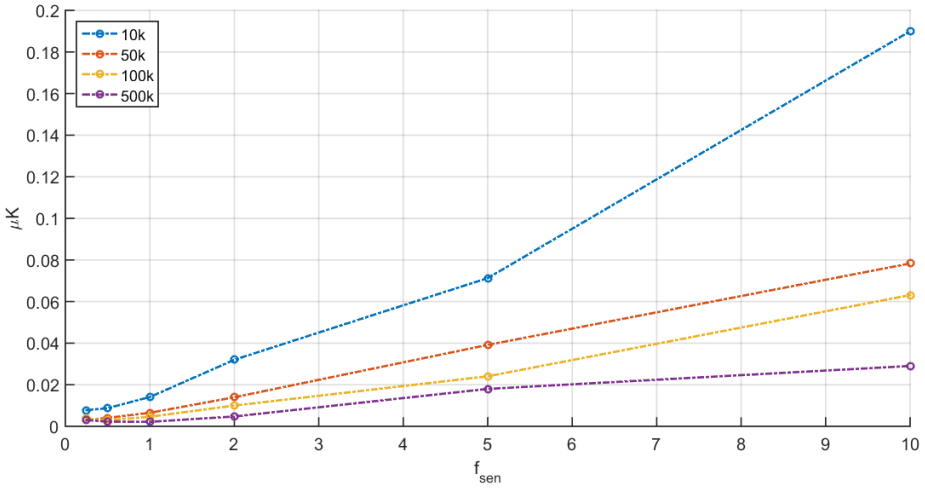


Figure 16: An illustration of how much the networks 1σ can vary depending on the size of its training dataset, when extracting the CMB radiation temperature anisotropy from only the Planck sensitivity.

5.1.3 Influence of hidden layer structures

Unfortunately there is no definite way of knowing the optimal number of layers and nodes in each of these. The only certain way of optimizing the network is by trail and error. For this purpose, multiple networks have been trained multiple times for increasing values of f_{sen}, with a training dataset size of $n = 50.000$. This will give insight in each networks general accuracy and stability of this. Networks with one two and three hidden layers have been trained twenty times with the following combinations of nodes en each layer:

1 Hidden layer: $\begin{bmatrix} 1 \\ 2 \\ 5 \\ 10 \\ 20 \\ 50 \\ 100 \end{bmatrix}$ 2 Hidden layers: $\begin{bmatrix} 12 & 3 \\ 15 & 5 \\ 15 & 10 \\ 20 & 5 \\ 20 & 10 \\ 20 & 15 \\ 30 & 5 \\ 30 & 10 \\ 30 & 15 \end{bmatrix}$ 3 Hidden layers: $\begin{bmatrix} 5 & 5 & 5 \\ 10 & 5 & 5 \\ 15 & 5 & 5 \\ 15 & 10 & 5 \\ 30 & 20 & 10 \end{bmatrix}$

It has been found that generally more layers and more nodes in each of there results in a higher accuracy and a greater stability of this. However the improvement in accuracy and stability is very small. While the three layered network with respectfully 30, 20 and 10 nodes in each produced the best overall results, the training time was over ten times longer than the two layered network with 20 and 10 nodes without enough improvement to justify using this combination.

The optimal combination of hidden layers and nodes in these has been determined to be the two layered network with 20 and 10 nodes in each respectfully as this has a good compromise between accuracy, stability and training time. Results of all combinations can be seen in appendix A.

5.1.4 Overall result

Now that the optimal network has been determined, the best possible result can be found. The network with 2 hidden layers, with 20 and 10 nodes in each, will be trained with a dataset of $n = 100.000$, to find how accurately it can extract the CMB temperature anisotropy from the Planck sensitivity. For the sake of comparison, the Planck sensitivities scaling factor has been chosen to $f_{sen} = 1$.
The network was able to extract the CMB temperature anisotropy from the added Planck sensitivity with a precision of ± 2.236 μK within the accuracy of 1σ as can be seen on figure 17.

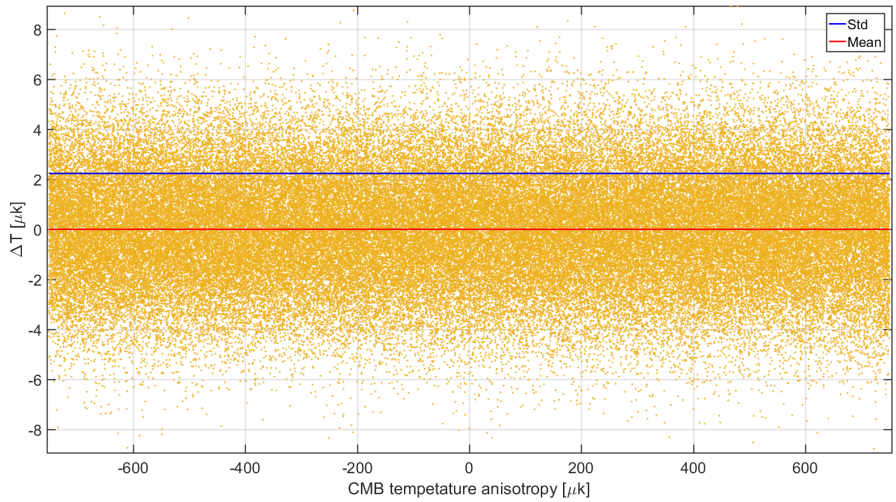


Figure 17: The optimal networks outputs difference from the target values, when extracting the CMB radiation temperature anisotropy from only the Planck sensitivity. The 1σ is measured to 2.236 μK

5.2 Neural network with all interfering foreground components

The neural network is now ready to be trained to extract the CMB temperature anisotropy from all three of the interfering main foreground emissions and the Planck sensitivity. The scaling factor f_{sen} will in the first trail be set to 1 as this will give the best opportunity for comparison with previous results. A training dataset will be created following the steps in section 4.1 with $n = 100.000$.

It is expected that the networks outputs deviation from target values, ΔT, will in general be higher than for a network only extracting the CMB temperature anisotropy from the Planck sensitivity.

The result of the first trail can be seen on figure 18. As expected, the 1σ is higher than before the interfering foreground components were considered. The network was able to extract the CMB temperature anisotropy from the interfering three foreground components and the added Planck sensitivity with a precision of $3.261 \pm \ \mu$K within the accuracy of 1σ. Since the CMB radiation temperature is expected to vary with $\pm 850\mu$K, the network is able to extract this with a precision of 99.62 %.
However this doesn't describe if the networks outputs dependency of f_{sen} has changed. The network will therefore be trained for increasing values of f_{sen}. On figure 19 the mean and 1σ is shown in μK along the values of f_{sen}.

As it can be seen the 1σ still seems to have a linearly dependency on f_{sen}. The best fit of an linear function is found to:

$$1\sigma_{\mu K} = 2.323 \cdot f_{sen} + 0.9630\mu\text{K} \tag{10}$$

It can be seen, while comparing this linear function fit to the linear function fitted for the trails without considering foreground components, shown in equation 9, that the slope of both are almost identical. However it is seen that the y-intercept is much greater in this fit. This means that the outputs dependency of f_{sen} isn't influenced by additional foreground emissions contribution, but that a general offset is to be expected when adding additional foreground components.

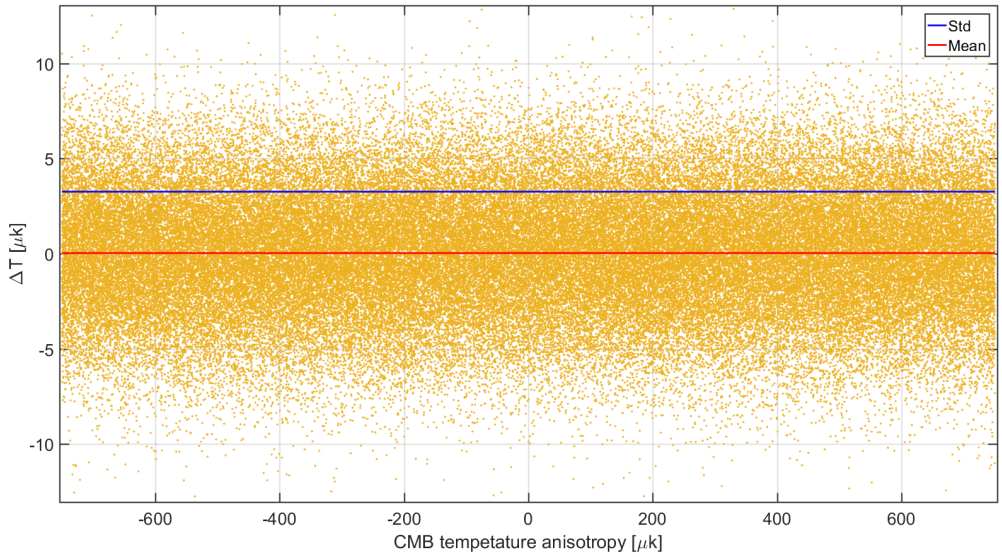


Figure 18: The optimal networks outputs difference from the target values, when extracting the CMB radiation temperature anisotropy from all three interfering foreground components and the Planck sensitivity. The 1σ is measured to 3.261 μK

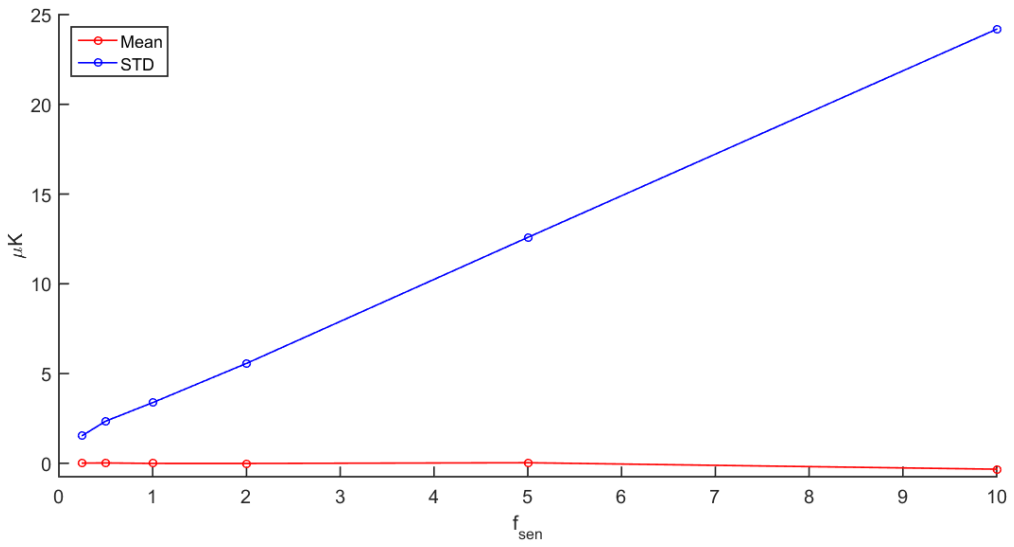


Figure 19: The mean and standard deviation of the network extracting the CMB radiation temperature anisotropy from all three interfering foreground components and the Planck sensitivity along an increasing value of f_{sen}.

6 Future improvements

Though the network seems to produce a satisfactory estimate of the CMB temperature anisotropy, some improvements and adjustments could be made.

In this report, the network was thoroughly investigated while extraxting the CMB temperature anisotropy from the Planck sensitivity. Due to lack of time, while extracting this temperature from a model containing the three most prominent imposing foreground emission and the Planck sensitivity, the same network was used without exploring any options for improvements. For future improvements one could investigate if this network could be further improved while all four foreground components are present in the model. This could include investigating the influence of different sizes of training datasets and the influence of hidden layer structures, much like the investigations carried out in section 5.1

Further more, other investigations could be carried out for each trial of the network. This could include examining the outcome if another activation function was chosen, such as the Gauss error function as it converges to its asymptotes quicker than the hyperbolic tangent function which has been used.
This network has only been investigated while using the Levenberg–Marquardt backpropagation algorithm. Testing the network for other algorithms, such as the Bayesian regularization backpropagation, could provide insight into it's influence in not only performance, but also computation time.

Besides improvements to the network, one could investigate the networks capability if additional interfering cosmic components were considered such as the Sunyaev-Zeldovich efect. The dust component has only been considered for its thermal component as this is the strongest contribution, though it is also cause for other contributions such as the anomalous dust emission.

As the neural network has only been used for models of the cosmic foreground and not observational data, it hasn't been necessary to consider dimensions and coordinates. For the network to be able to process real data and produce a two dimensional map of the sky, some modifications are needed.

7 Conclusion

The cosmic microwave background radiation is important as it can give insight in the early life of the universe. Though first thought to be perfectly uniform, high precision observations have shown that slight variations exist. The latest mission to measure this anisotropy is the Planck spacecraft. However the CMB radiation is mixed with other cosmic foreground components. These mainly consist of synchrotron emission, free-free emission and thermal dust radiation. This projects goal was to extract the CMB radiation from these three foreground components and the Planck spacecraft's sensitivity.

To do this the four cosmic components were mathematically modelled. The CMB radiation can be described as a blackbody, while both synchrotron and the free-free emission could be modelled as a simple power law. The thermal dust radiation could also be described as a power law, though some modifications to this were needed.

To extract the CMB radiation anisotropy, a neural network was used. This is a data processing tool, which operates similar to how a biological nervous system processes information. This means that it learns through examples and has the ability to approximate non-linear functions.
As the networks task was to find a temperature from nine given values, a function fitting network was needed. Matlab's preprogrammed neural network function `fitnet`, was used to create a feedback network with the Levengerg-Marquardt backpropagation algorithm. The hyperbolic tangent function was chosen as an activation function.

Before using the network to extract the CMB radiation from all three interfering foreground components, it was first investigated for its capability to extract the CMB radiation from only the Planck sensitivity, without considering the other foreground components. After training the network with an example dataset, with size $[n, 9]$ created from the modelled CMB radiation and the Planck sensitivity, the network was validated by investigating for any systematic errors. This was done first as a subjective analyses, by looking at the difference between expected output and actual output (ΔT) and examining if any dependency of temperature could be seen. When results showed a seemingly even distribution, a more mathematical approach was taken. The skewness and kurtosis were found to be within an acceptable range of a perfect uniform distribution and the network was thus validated as a network with no systematic errors.

As the Planck sensitivities scaling factor f_{sen} is dependent on the detector systems exposure time, it was necessary to investigate its influence on the networks output. It was found that a linear relationship existed between the value of f_{sen} and the outputs accuracy, measured as a standard deviation, 1σ. When training multiple identical networks for each chosen value of f_{sen} it was found that the greater its value, the more the networks outputs accuracy would vary.

To explore for possible improvements of the network, the influence of the training datasets size was investigated. This was done for four choices of n. For each choice of n, fifty identical networks were trained for all chosen values of f_{sen} and the output compared. It was found, that while the overall output didn't seem to depend on the training datasets size, its consistency improved for increasing values of n.

The last factor that was investigated was the influence of the hidden layers structure. Multiple combinations of hidden layers and nodes in each of these were tested to find the network combination with the best results. While it was found that the best combination was a network with three hidden layers, with 30, 20 and 10 nodes in each respectfully, the minor improvement this resulted in was not enough to justify the training time it took. The best network was determined to be one with two hidden layers, with 20 and 10 nodes in each, as this had a good compromise between accuracy, consistency of this and computation time.

The best network was determined to be a network with two hidden layers, with 20 and 10 nodes in each respectfully. For a training dataset of $n = 100.000$ and a scaling factor $f_{sen} = 1$, the network could correctly determine the CMB radiation temperature anisotropy within 1σ to an accuracy of ± 2.235 μK, from the Planck sensitivity.

Finally this network was used to extract the CMB temperature anisotropy from all foreground components. The network was once again trained from an example dataset, though this time being modelled from all four cosmic components and the Planck sensitivity. For a training dataset of $n = 100.000$ and a scaling factor of $f_{sen} = 1$, the network could correctly determine the CMB radiation temperature anisotropy within 1σ to an accuracy of ± 3.261 μK. Considering the temperature anisotropy has been expected to vary between ± 850 μK, this results in an accuracy of 99.62 %

Further more it was found, that the added foreground components didn't influence the networks accuracy's dependency of f_{sen}, as the slope of 1σ along an axis of f_{sen} didn't increase, compared to the network only removing the Planck sensitivity. The linear approximation was only added an offset of 0.9541μK.

References

[1] H.U. Nøregaard-Nielsen, H.E. Jørgensen. *Foreground removal from CMB temperature maps using an MLP neural network* (2008)

[2] H. K. Eriksen, et. al. *Cosmic microwave background component separation by parameter estimation* (2006)

[3] D. P. FINKBEINER, et. al. *Extrapolation of Galactic Dust Emission at 100 Microns to Cosmic Microwave Background Radiation Frequencies Using FIRAS* (1999)

[4] J. C. Mather, et. al. *Calibrator Design for the COBE 1 Far Infrared Absolute Spectrophotometer (FIRAS)* (2008)

[5] J.G. Bartlett, P. Amram. *Galactic Free-free Emission and Hα* (1998)

[6] F. K. Hansen, et. al. *Foreground Subtraction of Cosmic Microwave Background Maps using WI-FIT (Wavelet based hIgh resolution Fitting of Internal Templates)* (2006)

[7] C. Stergiou and D. Siganos *Neural Networks* (2017)

[8] Yoshiaki Sofue *The North Polar Spur and Aquila Rift* (2014)

[9] G. De Zotti, et. al. *The Planck Surveyor mission: astrophysical prospects* (1999)

[10] M. Tegmark, et. al. *Foregrounds and Forecasts for the Cosmic Microwave Background* (1999)

[11] M. Hudson Beale, et. al. *Neural Network Toolbox User's Guide*(2016)

[12] L. Francis *Neural Networks Demystified* (Unknown)

[13] R. A. Freedman, et. al. *Universe*, W. H. Freeman and Company, 10th edition, (2014)

[14] P.Abhigna, et. al. *Analysis of recurrent neural networks for prediction of wave height using wind data*, (2016)

[15] Autism Reading Room, *Neurons*, (2017)
http: //readingroom.mindspec.org/?page_id=8852

A Neural network structures and results

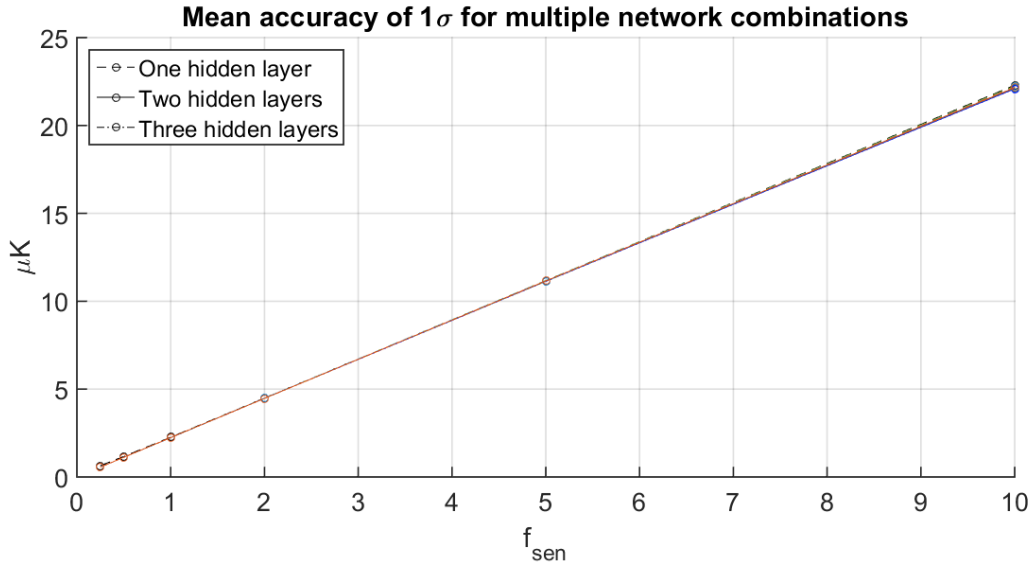


Figure 20: The mean value of 20 networks each with different combinations of hidden layers and nodes in these. The mean value for each network combination is found from training the same network twenty times.

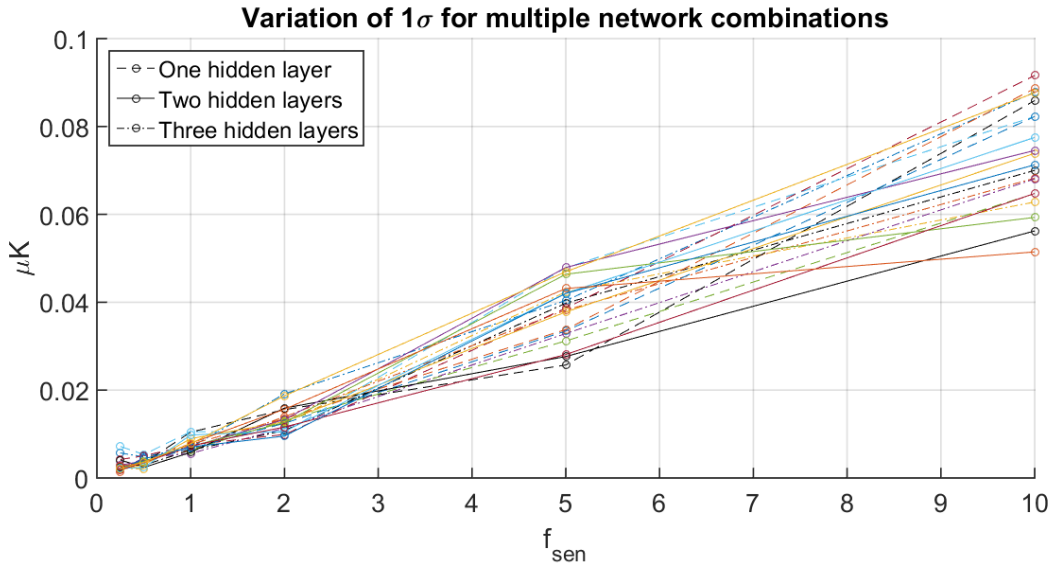


Figure 21: The standard variation 1σ of 20 networks each with different combinations of hidden layers and nodes in these. The standard deviation for each network combination is found from training the same network twenty times.

DTU Space

National Space Institute
Technical University of Denmark

Elektrovej, building 327
DK - 2800 Kgs. Lyngby
Tlf. Tel (+45) 4525 9500
Fax Fax (+45) 4525 9575

www.space.dtu.dk