Describing the Ising Model Hamiltonian with Machine Learning

Euan Rossie

Table of Contents

Abstract:

An investigation into the basics of machine learning with a low bias focus on a continuous cost function using Linear Regression, Ridge Regression and LASSO Regression to predict various generated Ising models. The LASSO Regression model was found to perform to the highest standard under the coefficient of determinations explanation of accounted variation within a model. Likely causes of the factors affecting the performance of the regression models is discussed and further research possibilities are considered.

Introduction:

The Ising model was developed in 1920 by Wilhelm Lenz and later named after Ernst Ising who wrote his doctoral thesis in 1925 on solving the one dimensional model without phase transitions, the two dimensional Ising model was later analytically described by Lars Onsager in 1944. The Ising model is useful in describing ferromagnetism in materials such as iron cobalt and nickel, lattice gasses and sea ice among other naturally occurring systems.

The Ising models was originally developed with the motivation of explaining the magnetic dipole relationships within a ferromagnet, with two and three dimensional models created quickly following the one dimensional models creation.

Machine learning, otherwise known as predictive analytics, is heavily used in many industries, from the common examples of self driving cars and handwriting recognition to less known areas of use such as to develop games bots capable of beating any human [8] or credit card fraud detection. Machine learning as a term was first used by Arthur Samuel at IBM in 1959 [9] and whilst the field had some early success it was largely abandoned due to the extreme reliance on huge data sets that was unfeasible at the time [10]. However with the advent and widespread adoption of the internet and related technologies along with the steps made in computer power in recent years, machine learning became not only a feasible but an extremely powerful way to operate AI.

To evaluate the validity of a machine learning programs learnt behaviours a test outside of the training data is used, this test gives a comparison compared to the data the program was trained upon along with a score of a between -1 and 1 showing its accuracy. If the test data is accurate to the known correct model then the program has successfully learnt the model, this doesn't necessarily correlate with the training data's accuracy which is why it is vital to test any machine learning program on previously unseen data as well as the training data.

The graph arrays generated show the individual synapse connections and their weighting upon the activation function, this is important as it allows the behaviour learnt by the machine learning algorithm to be evaluated to see if the hidden layers of the program have found a global minimum that is a recognisable system that could be defined by a physical system or if the system has found a local minimum that works for the given data samples and not wider data sets.

The code used in this study is from a heavily modified notebook sourced from the "A high-bias, low variance introduction to Machine Learning for physicists" [14] review which uses the sklearn framework produced by a collaboration between google, various foundations and various universities [13].

Theory:

The Ising model is a theoretical model for describing the interaction of magnetic dipoles within a ferromagnet. In a ferromagnet each nucleus is a positively charged magnetic dipole, that can be spun up or down. In the Ising model the spin of a magnetic dipole is influenced by the magnetic dipoles surrounding it, this influence is determined by the energy of the dipoles and the distance between dipoles. In its most basic and most common version the Ising model takes only neighbouring dipoles into account however the Ising model can also encompass additional nearby dipoles with small changes to the maths of the model. Other simplifications within the Ising model are the assumptions that every dipole has the same magnitude of influence of over its neighbours and that there is no external magnetic field influencing the model. Mathematically the Ising model uses a n-dimensional array with +1 symbolising a spin up state and -1 symbolising a spin down state.

The Ising model Hamiltonian for the one dimensional model,

$$H = J \sum_{<i,k>} \sigma_i \sigma_k \qquad [1](1)$$

where J is the interaction strength between adjacent neighbours under the assumption that all neighbours interact equally and σ is the spin configuration of each individual atom within the model. The interaction strength is also sometimes given as a negative J, this is to more explicitly show that the interaction between adjacent dipoles is most significant if the energy change due to a flip is negative itself. The periodic boundary condition that the 0^{th}element of the Ising model should be considered adjacent to the n^{th} is applied such that the model loop and have no edge that would impact the purity of the simple Ising model. The use of <i,k> shows that i and k are adjacent within the model. The Ising model leads to an equation that dictates the probability of an atoms magnetic spin flipping to be the opposite of what it was, this is known as a spin flip and happens if

$$\triangle E < 0 \qquad [1]\ (2)$$

as the ferromagnet seeks to be in its lowest possible energy state or if

$$\triangle E > 0 \qquad [1]\ (3)$$

with the probability of

$$P = e^{-\beta \triangle E} \qquad [1]\ (4)$$

Where β is defined as

$$\beta = \frac{1}{kT} \qquad [1]\ (5)$$

This leads to a low chance of a spin flip at lower temperatures that increases exponentially as temperature increases. The one dimensional Ising model can be modified to incorporate a next neighbour interaction relationship between magnetic dipoles with this new Hamiltonian being,

$$H = J \sum_{<i,k>} \sigma_i \sigma_k + J \sum_{<i,k+1>} \sigma_i \sigma_{k+1} \qquad [18](6)$$

Machine learning is the application of artificial intelligence algorithms that gives a program the ability to improve by inference through a series of training and tests such that the application learns patterns without programming during the learning process. There are many algorithms used in machine learning, each with benefits and draw backs unique to each. In this study the algorithms used are Ordinary Least Square Regression (OLS), Tikhonov Regularization (Ridge Regression) and Least Absolute Shrinkage and Selector Operator (LASSO). All of these algorithms are examples of Neural Networks, a term describing algorithms that endeavour to simulate the way human brains work with neurons connected by synapses. A neural network is organised into layers of nodes and connections, as seen in figure 1, with only the input layer and output layer known in full to the user, other layers are referred to as hidden and it is there that machine learning takes place. A node within a neural network consists of an activation function that takes all the inputs from the previous layer and calculates if the node will "fire" based on whether the function is greater than the nodes unique threshold function. Nodes in adjacent layers are connected by synapses with each synapse having a weight and bias associated with its level of importance in the relationship between the two nodes. [19]

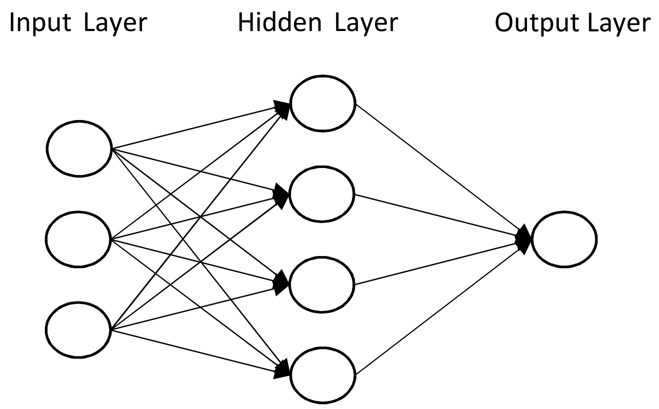

Figure 1: A diagram depicting the layout of a neural network with input, hidden and output layers.

Reaching the global minimum is considered the goal of a neural network and refers to the minimum of a cost function that governs the ability of the network to accomplish its job. The cost function can be measured in many ways, but a simple way is to use a linear regression algorithm when dealing with continuous data; if the data is discrete a different form of regression called logistic regression is used. The challenge in the cost function is to ensure that the global minimum is reached and not a local minimum, this requires the minimisation of complex quadratic equations as seen in the 3 models used in this study. [19]

A nodes output function can be given to be

$$Output = \begin{cases} 0 \ if \ \sum_i w_i x_i + b_i \ \leq threshold \\ 1 \ if \ \sum_i w_i x_i + b_i \ \geq threshold \end{cases} \qquad [11](7)$$

where i identifies each individual synapse, w_i is the weight function of synapse, x_i is the output of the nodes in the previous layer and b is the bias function of the synapse.

One parameter that is used solely to optimise the performance is the use of mini batches. Mini batches is a term used to describe the random selection of a small part of the data used to train an algorithm faster. These mini batches are used as a way of training an algorithm as with the entirety of the data each calculation is extremely computationally expensive, instead different smaller sets of data are used at each calculation to train the model quicker, even at the cost of increased inaccuracy from the smaller data set. [11]

Linear regression (OLS) is a supervised algorithm with a continuous output that is used as a cost function for machine learning programs. Linear regression uses the ordinary least square to choose a linear function that minimises the sum of the squared distance of each data point from the function. This algorithm therefore does not differentiate between data, with each having equal value to the calculation with its vector form being

$$\beta \ = \ (X^T X)^{-1} \, X^T y \qquad [17](10)$$

in which β is the fitted values of the regression, y is the observation of the dependant variable and X is a matrix containing an observation of the independent variable and the independent variable.

Ridge regression is a weighted style of linear regression that employs an L2 penalty and is used if the problem is ill-posed, meaning that there is no unique solution to be found through the use of normal linear regression. The L2 penalty is used to avoid overfitting by the regression algorithm by shrinking all coefficients by a factor of α. This is done in vector form by

$$\beta \ = (X^T X + \alpha I)^{-1} X^T y \qquad [17](11)$$

LASSO regression is an L1 regularised linear regression model built on top of OLS, L1 regularised means that this algorithm makes use of shrinking to reduce the number of parameters. This makes the least absolute shrinkage and selector operator regression algorithm useful for simple models that naturally have few parameters. LASSO has no closed form and so takes the form of

$$\beta \ = arg\min|X\beta - y| + \alpha|\beta| \qquad (12)$$

in which β is the fitted values of the regression, α is the tuning parameter, y is the observation of the dependant variable and X is a matrix containing an observation of the independent variable and the independent variable.

Using β the residual sum of squares (RSS) can be calculated by

$$RSS = \frac{1}{n}\sum_{i}^{n}(y_i - f(x_i))^2 \qquad [20](13)$$

in which $f(x_i)$ is the predicted value of yi by the algorithm and is then used to find the coefficient of determination (R^2) by taking,

$$R^2 = 1 - \frac{RSS}{TSS} \qquad [20](14)$$

where TSS is the total sum of squares and so gives R^2 to be the fraction of unexplained variance in the algorithm. The coefficient of determination gives a decimal that expresses the amount of variability within the model that has been accounted for, with a coefficient of $|1|$ showing that all the variance in the model has been accounted for.

Methodology:

First the Ising model was defined within the program as a function to calculate the energies using equation 1 in the Ising Hamiltonian from a 2 dimensional array of 10000 x 40 randomly assigned states. Using these calculated energies the function would then use equations 2, 3 and 4 to determine the likelihood of a change in the magnetic dipole spin and flip the spin. The Ising data was then reshaped into a 2 dimensional array of Ising states and their respective energies. A mini batch of 400 was then taken from the Ising data to be used as the training data and a different mini batch of 200 was taken for testing. Each mini batch was defined within its own list for the X and Y data to make its implementation into the sklearn library functions easier. The X train and Y train data was then fed into the sklearn linear regression function, ridge regression model and LASSO model across a penalty coefficient (α) consisting of the log space between -4 and 5 over 10 evenly spaced samples. The learnt coefficients (β) were then placed into 1 dimensional arrays to be stored and conveniently accessed. The coefficient of determination (R^2) was then calculated from the learnt coefficients and the learnt coefficients reshaped into a 2 dimensional array of β by β. The coefficient array was then plotted using the Matplotlib framework [4] to show what the machine learnt algorithms found the 1 dimensional Ising model relationship to be. With the learnt relationships visualised, the training and test coefficients of determination for each penalty coefficient are plotted onto a graph using Matplotlib to show the performance of each algorithm at each penalty coefficient.

The Ising model used in the program was then modified to use the next neighbour Ising model Hamiltonian shown in equation 6 and the algorithms run again to give a set of results that reflect the model's ability to adapt to new training and test data.

The Ising models used then both had a random gaussian noise, with a standard deviation of 4 about the mean of 0, added to the energies to simulate the inaccuracy inherent to a real data set. This noisy data was then used to calculate the learnt coefficients and the coefficients of determination. The data was then used to plot the performance of each algorithm for comparison based on the penalty coefficients.

The data sets used in the test and train algorithms were then changed to test how well a model trained on the one dimensional nearest neighbour interaction Ising model can predict the next nearest neighbour Ising model. This required the program to be modified to generate 2 different models within the same run cycle.

Results:

The β by β graph to show the learnt Ising relationship uses a gradient of red to blue colour coding to describe the relation between 2 magnetic dipoles is seen in figure 2.

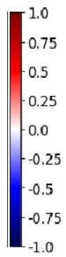

Figure 2: the gradient shading showing the colour to number relationship

The highest performing algorithm for the neighbouring Ising model as seen in figure 3 is the LASSO algorithm at the penalty coefficients of 10^{-3} and 10^{-2}.

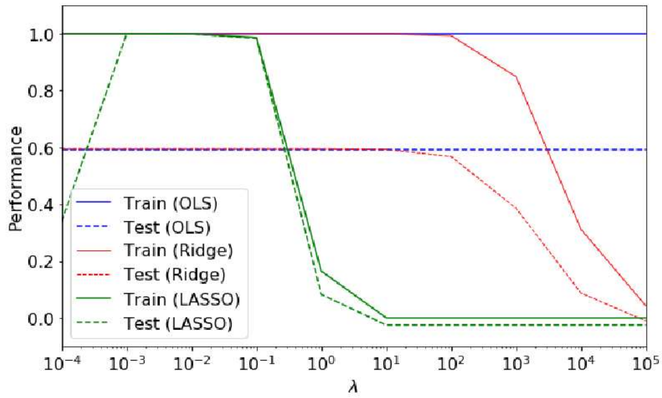

Figure 3: A graph showing the test and train performance according to the coefficient of determination (R^2) of each algorithm at each penalty coefficient (λ) for the neighbouring Ising model

This is verified by the learnt Ising relationship in the graph seen in figure 4 and figure 5 where the graph depicts a diagonal line showing that each i^{th} Ising dipole is dependent upon the $i-1^{th}$ dipole.

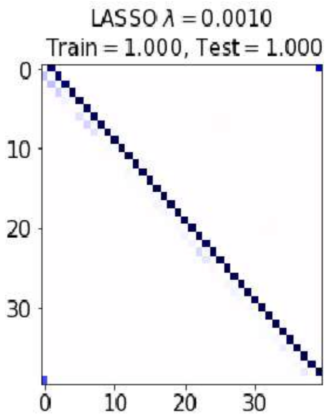

Figure 4: A graph showing the learnt Ising relationship of a neighbour Ising model with coefficient of determination performance scores and penalty coefficient (λ)

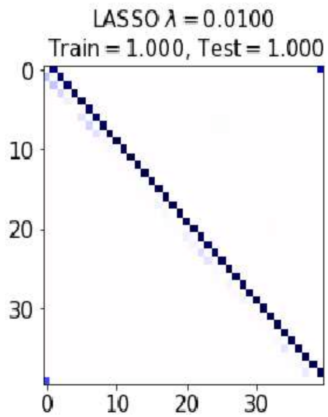

Figure 5: A graph showing the learnt Ising relationship of a neighbour Ising model with coefficient of determination performance scores and penalty coefficient (λ)

The LASSO algorithm quickly drops in performance as seen in figure 3 down to a coefficient of determination of ≈ 0 which can be seen in figure 6.

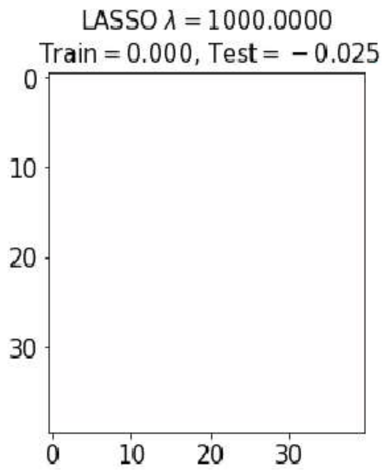

Figure 6: A graph showing the learnt Ising relationship of a neighbour Ising model with coefficient of determination performance scores and penalty coefficient (λ)

As seen in figure 3 the Ridge Regression performs best at lower penalty coefficients such as in figure 7 where the test coefficient of determination is 0.595. However, the performance quickly deteriorates at higher penalty coefficients as seen in figure 8 where the test coefficient of determination is 0.088.

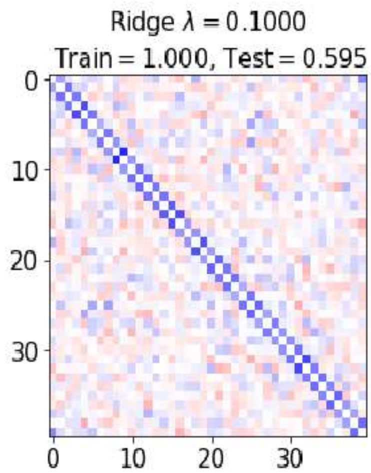

Figure 7: A graph showing the learnt Ising relationship of a neighbour Ising model with coefficient of determination performance scores and penalty coefficient (λ)

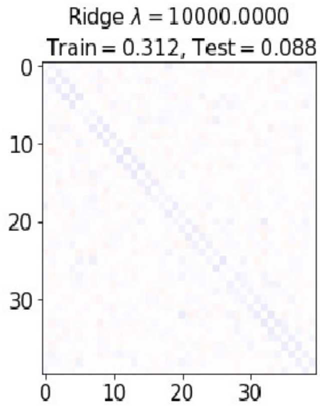

Figure 8: A graph showing the learnt Ising relationship of a neighbour Ising model with coefficient of determination performance scores and penalty coefficient (λ)

Finally, from figure 3 the test OLS coefficient of determination can be seen to be 0.593 which is verified by figure 9.

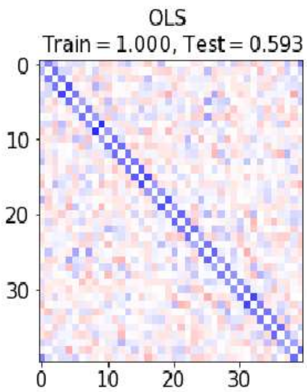

Figure 9: A graph showing the learnt Ising relationship of a neighbour Ising model with coefficient of determination performance scores and penalty coefficient (λ)

From figure 10 we can see that LASSO has the highest performance score for the next neighbouring Ising model when at low penalty coefficients such as in fig 38 with a penalty coefficient of 0.01 having a test coefficient of determination of 1.

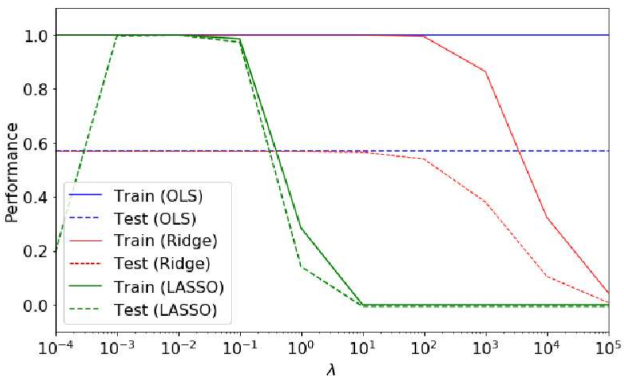

Figure 10: A graph showing the train and test performance according to the coefficient of determination (R^2) of each algorithm at each penalty coefficient (λ) for the next neighbouring Ising model

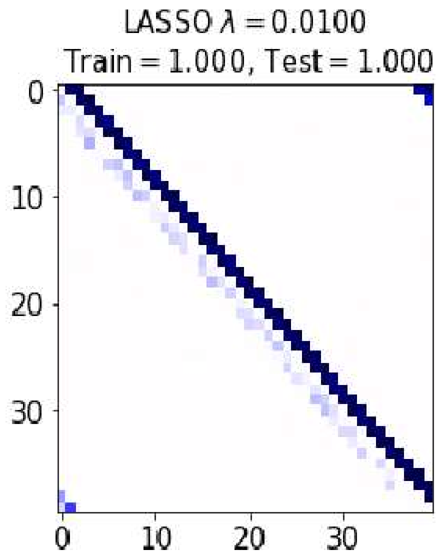

Figure 11: A graph showing the learnt Ising relationship of a next neighbour Ising model with coefficient of determination performance scores and penalty coefficient (λ)

However, at higher penalty coefficients the LASSO algorithm starts to perform far worse as seen in figure 12 where the coefficient of determination is 0.196 for a penalty coefficient of 1.

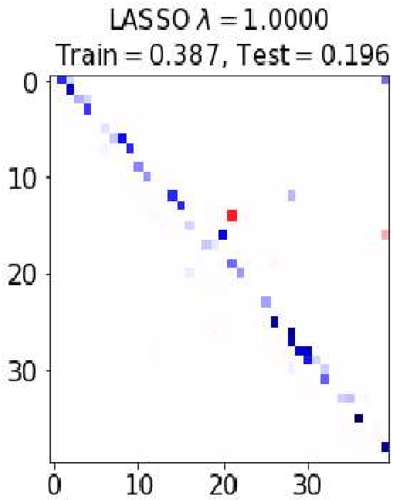

Figure 12: A graph showing the learnt Ising relationship of a next neighbour Ising model with coefficient of determination performance scores and penalty coefficient (λ)

The ridge regression algorithms performance peaks at low penalty coefficients such as in figure 13 where with a penalty coefficient of 0.001 the performance coefficient of determination is 0.485. The Ridge regression falls off in performance at high penalty coefficients as seen in figure 14 with the penalty coefficient 10000 giving a coefficient of determination performance of 0.115.

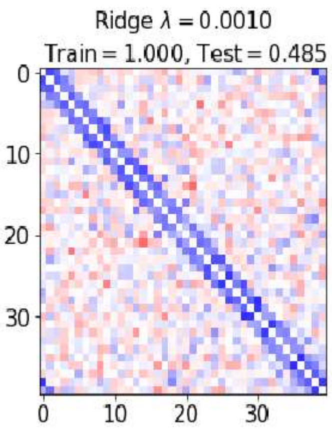

Figure 13: A graph showing the learnt Ising relationship of a next neighbour Ising model with coefficient of determination performance scores and penalty coefficient (λ)

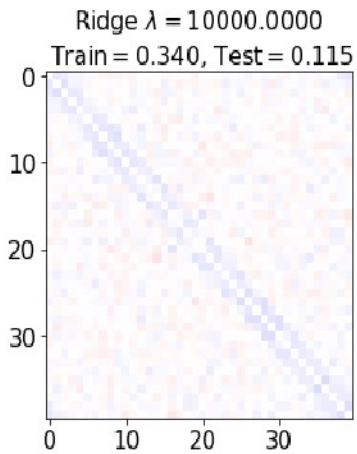

Figure 14: A graph showing the learnt Ising relationship of a next neighbour Ising model with coefficient of determination performance scores and penalty coefficient (λ)

The OLS regression algorithm is seen in figure 10 to perform with a test coefficient of determination of 0.492 as verified in figure 15.

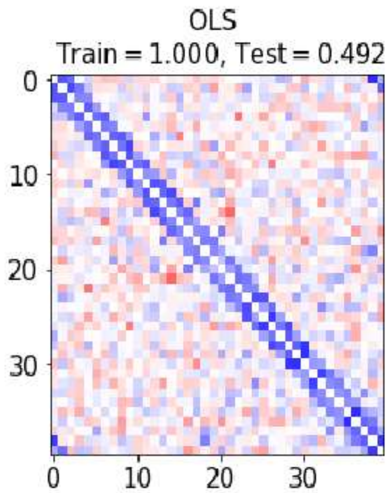

Figure 15: A graph showing the learnt Ising relationship of a next neighbour Ising model with coefficient of determination performance scores and penalty coefficient (λ)

In figure 16 it can be seen that for noisy Ising data the LASSO model has its highest performance at a penalty coefficient of 0.1 with a performance of 0.519 that can be verified with figure 17.

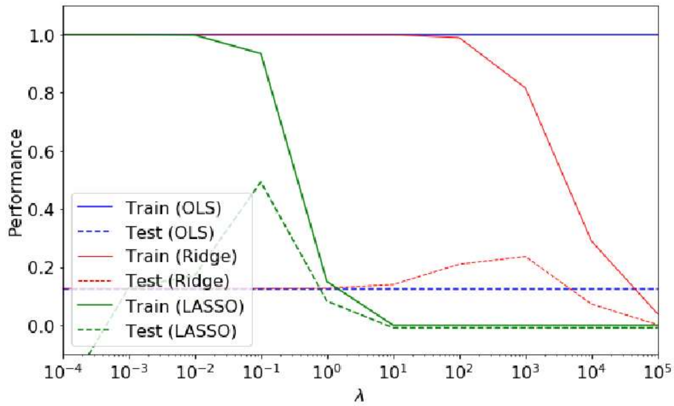

Figure 16: A graph showing the train and test performance according to the coefficient of determination (R^2) of each algorithm at each penalty coefficient (λ) for the nearest neighbouring Ising model with added gaussian noise

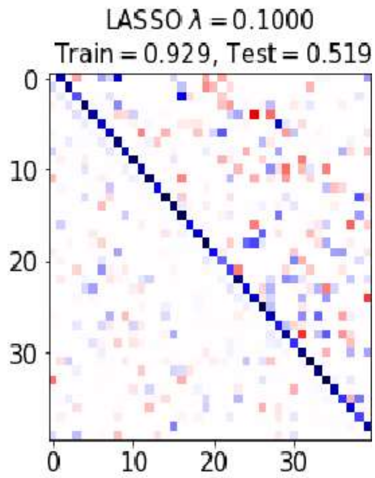

Figure 17: A graph showing the learnt Ising relationship of a neighbour Ising model with added gaussian noise with coefficient of determination performance scores and penalty coefficient (λ)

The LASSO algorithm rapidly falls to a test performance score of ≈ 0 at high penalty coefficients with this seen visualised in figure 65 where at penalty coefficient 1000 it has a coefficient of determination of -0.016.

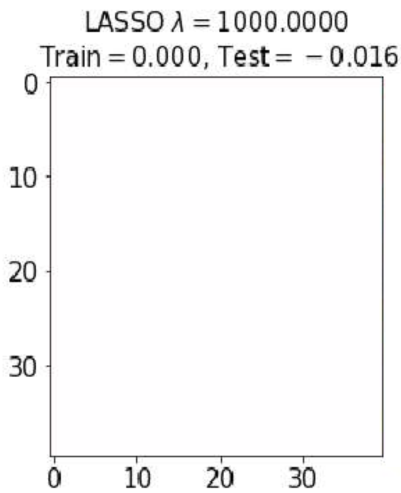

Figure 18: A graph showing the learnt Ising relationship of a neighbour Ising model with added gaussian noise with coefficient of determination performance scores and penalty coefficient (λ)

The ridge regression algorithms coefficient of determination peaks at a penalty coefficient of 1000 with a performance of 0.227 as can be seen in figure 19. The Ridge regressions performance follows that of the OLS regression algorithms until it reaches a penalty coefficient of 1 at which point it diverges to increase until the peak at the penalty coefficient of 1000 and then drop to the performance score of 0 as the penalty coefficient continues to rise.

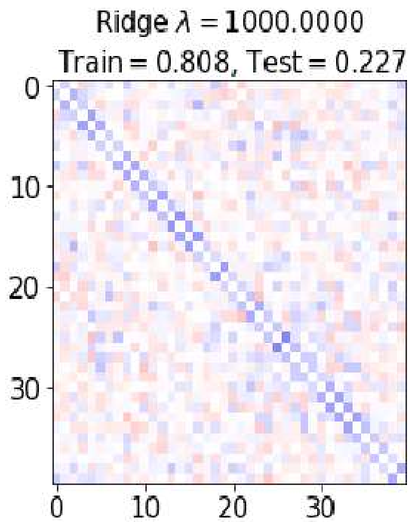

Figure 19: A graph showing the learnt Ising relationship of a neighbour Ising model with added gaussian noise with coefficient of determination performance scores and penalty coefficient (λ)

The OLS regression algorithm performs at a coefficient of determination of 0.055 for the test on the noisy neighbour Ising data.

Figure 20 shows that the LASSO has the highest coefficient of determination performance at a penalty cost of 0.1 with a performance of 0.432 which can be seen visualised in figure 21.

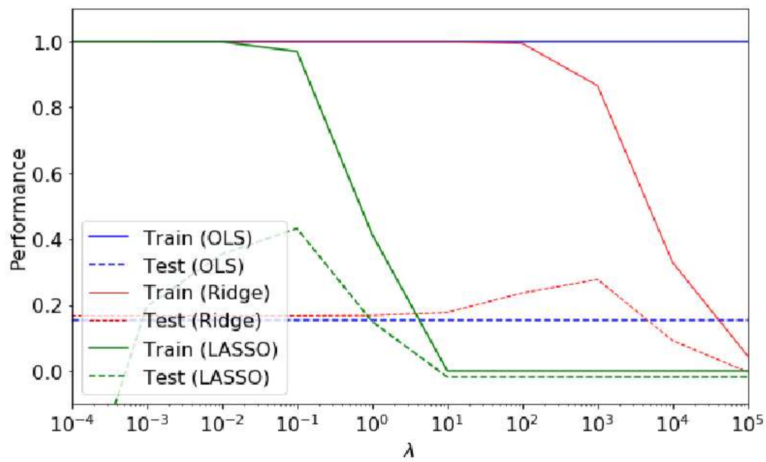

Figure 20: A graph showing the performance according to the coefficient of determination (R^2) of each algorithm at each penalty coefficient (λ) for the next neighbouring Ising model with added gaussian noise

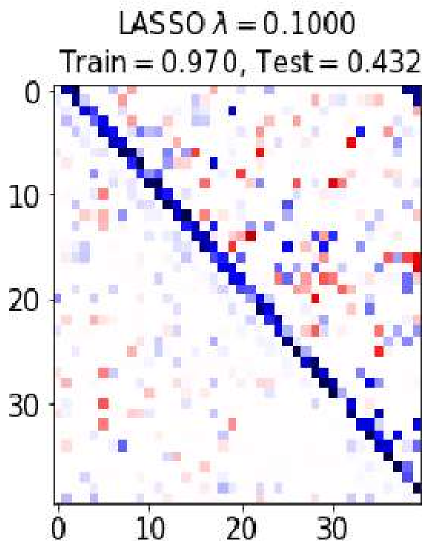

Figure 21: A graph showing the learnt Ising relationship of a next neighbour Ising model with added gaussian noise with coefficient of determination performance scores and penalty coefficient (λ)

The LASSO algorithms performance peaks for a small fraction of the penalty coefficients shown with a performance quickly falling off at coefficients below 0.001 and coefficients above 1. The Ridge regression algorithm peaks at a penalty coefficient of 1000 with a coefficient of determination of 0.278 with it following the OLS algorithm at penalty coefficients lower than 10 and the coefficient of determination performance drops to ≈ 0 at penalty coefficients greater than 100000.

The OLS regression algorithm has a coefficient of determination performance of 0.153 for the test on the noisy next neighbour Ising data.

As can be seen in figure 22, which depicts a model trained on a nearest neighbour interaction Ising model and tested on a next to nearest neighbour interaction Ising model, the LASSO algorithms performance peaks at a penalty coefficient of 10^{-3} with a coefficient of determination of 0.535. The Ridge regression model however peaks at a penalty coefficient of 100 at a coefficient of determination of 0.205 which is only 0.009 higher than the OLS coefficient of determination at 0.196.

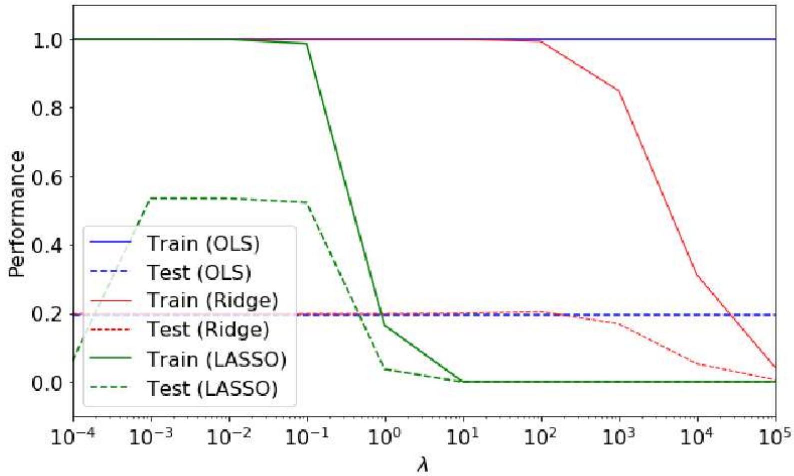

Figure 22: A graph showing the performance according to the coefficient of determination (R^2) of each algorithm at each penalty coefficient(λ) for training on a neighbouring interaction Ising model and tested on the next nearest interaction Ising model

With figure 23 it can be seen that LASSO algorithm performs best according to the coefficient of determination for a model trained on the next nearest neighbour interaction Ising model and tested on the nearest neighbour interaction Ising model with a penalty coefficient of 1 giving the performance score of 0.205. The Ridge regression algorithm performance peaks at 10^3 with a coefficient of determination of 0.187. The OLS algorithm performs at a -0.054

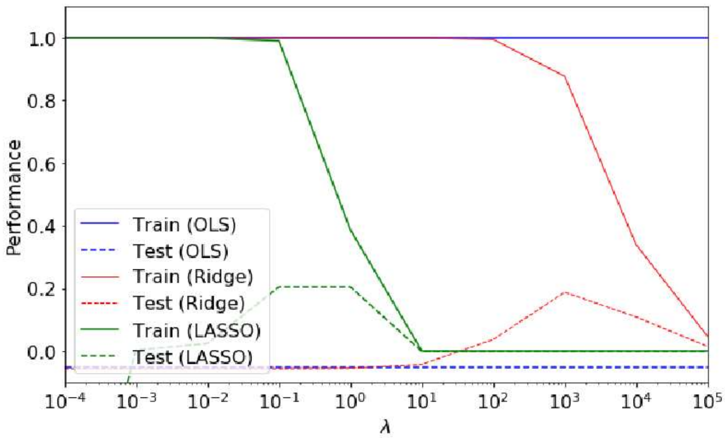

Figure 23: A graph showing the performance according to the coefficient of determination (R^2) of each algorithm at each penalty coefficient (λ) for training on a next neighbour interaction Ising model and tested on the nearest neighbour interaction Ising model

Figure 24 shows that the LASSO regression algorithm has a peak performance coefficient of determination of 0.303 at the penalty coefficient of 0.1 for a model trained on the noisy nearest neighbour interaction Ising model and tested on the noisy next nearest neighbour interaction Ising model. The Ridge regression algorithm peaks at a penalty coefficient of 10^3 with a coefficient of determination of 0.120. The OLS algorithm shows a performance coefficient of determination of 0.007.

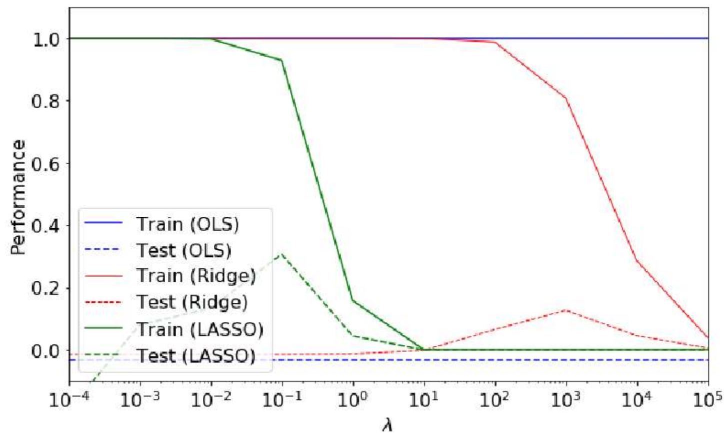

Figure 24: A graph showing the performance according to the coefficient of determination (R^2) of each algorithm at each penalty coefficient (λ) for training on a nearest neighbour interaction Ising model with added gaussian noise and tested on the next nearest interaction Ising model with added gaussian noise

As seen in figure 25 the LASSO algorithm has the highest performance score according to the coefficient of determination for a model trained on the noisy next nearest neighbour Ising model and tested on noisy nearest neighbour Ising model with a penalty coefficient of 1 and performance of 0.126. The Ridge regression peaks at 10^3 penalty coefficient with a coefficient of determination of 0.0.087. The OLS algorithm coefficient of determination is -0.403.

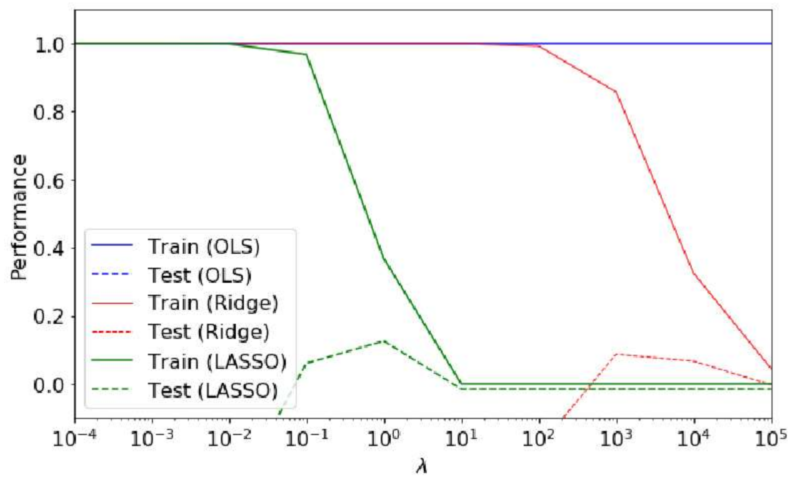

Figure 25: A graph showing the performance according to the coefficient of determination (R^2) of each algorithm at each penalty coefficient (λ) for training on a next neighbouring interaction Ising model with added gaussian noise and tested on the nearest interaction Ising model with added gaussian noise

When trained on a nearest neighbouring interaction Ising model and tested on a noisy next nearest neighbouring Ising model as seen in in figure 26 the LASSO regression algorithm model has a peak performance of 0.744 at a 10^{-3} penalty coefficient. The Ridge regression algorithm peaks when it is at the penalty coefficient of 0.01 with a coefficient of determination of 0.425. The OLS model has a coefficient of determination of 0.423.

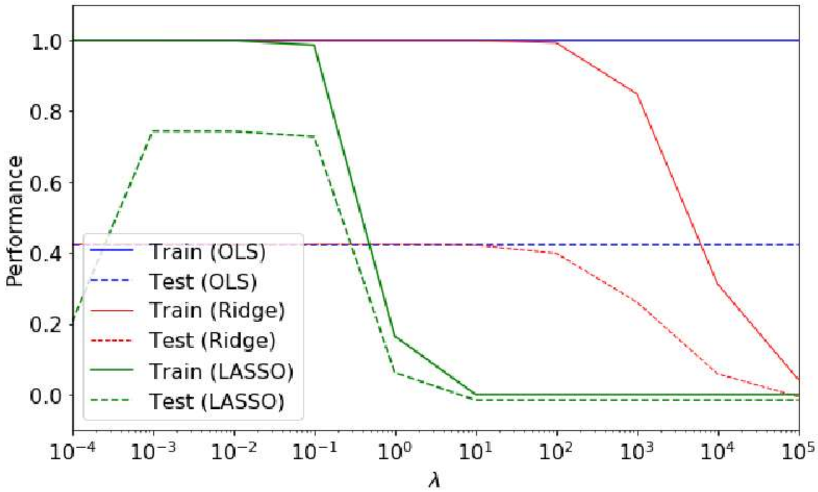

Figure 26: A graph showing the performance according to the coefficient of determination (R^2) of each algorithm at each penalty coefficient (λ) for training on a nearest neighbouring interaction Ising model and tested on the nearest interaction Ising model with added gaussian noise

For a next nearest neighbour interaction Ising model trained without noise and tested with a noisy data as seen in figure 27 set the LASSO algorithm model peaks at a penalty coefficient of 10^{-2} with a coefficient of determination of 0.805. The Ridge regression model peaks at a penalty coefficient of 100 with a coefficient of determination of 0.373. The OLS algorithm model has a coefficient of determination of 0.368.

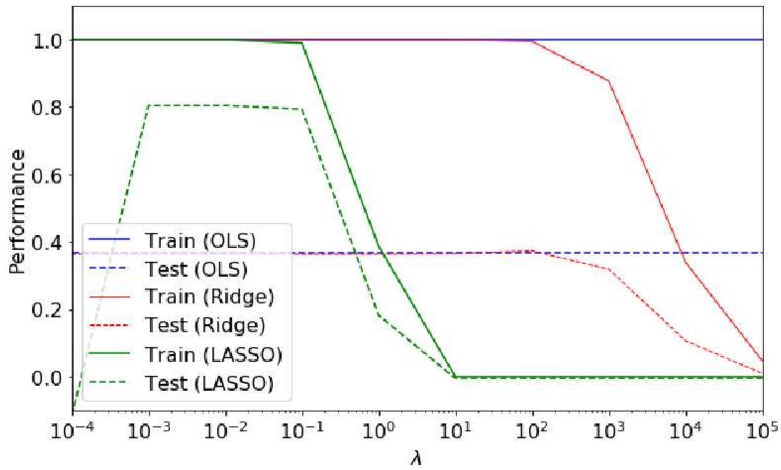

Figure 27: A graph showing the performance according to the coefficient of determination (R^2) of each algorithm at each penalty coefficient (λ) for training on a next nearest neighbour interaction Ising model and tested on the next nearest interaction Ising model with added gaussian noise

Analysis:

The best model for the one dimensional Ising model with nearest neighbour interactions that is also trained on the nearest neighbour interaction Ising model is the LASSO regression model with a penalty coefficient of 0.001 as seen in figure 4. This is as it has a coefficient of determination of 1 for both the training data and test data showing that the model accounts for all of the variability in both sets of data. For the one dimensional nearest neighbour interactions Ising model, Ridge regression has its highest coefficient of determination at the penalty coefficient of 0.1 with the training coefficient of determination being 1 and the test coefficient of determination being 0.595 as seen in figure 7. The OLS regression model has a test coefficient of determination of 0.593 and a training coefficient of determination of 1. This shows that the L1 regularisation of the LASSO model works just as well as the L2 regularisation of the Ridge model and the OLS models lack of regularisation for the data the model is trained on but that the LASSO model is better at modelling for new, previously unseen data than the Ridge or OLS model. This difference is due to the L1 regularisation term which rather than the shrinkage by division of all terms that characterises the L2 regulariser uses a shrinkage by subtraction. Regularisation by magnitude subtraction means that the LASSO model can eliminate less influential relationships within the model, this works well with simple models such as the one dimensional nearest neighbour Ising model. This is obvious in figure 4 and figure 5 where the nearest Ising interactions are obvious by the diagonal $(i, i-1)$ line of positive relationships that is exactly what would be expected based upon the nearest neighbour Ising Hamiltonian from equation 1, there is also a less influential mirrored positive relationship at the $(i, i+1)$ diagonal that shows the model partially learning that the Ising model is symmetrical.

The LASSO model coefficient of determination performance drops off quickly when the penalty coefficient is not at the optimum value, this shows the importance of trying many penalty coefficients on each model to find the correct value to use. This is also seen in the Ridge regression model at too small of a penalty coefficient the model simply replicates the coefficients of determination of the OLS as its L2 regulation is too small to affect the learnt relationships. When the Ridge regression model has too great of a penalty coefficient all the learnt relationships are overly reduced by the L2 regularisation term leaving the learnt model without significant learnt relationships and thus a low coefficient of determination performance as the variance in the model is not properly accounted for as can be seen in figure 8.

One thing that the OLS and Ridge regression models are superior to the LASSO model at is in the one dimensional nearest neighbour Ising model where the algorithms ability to learn the symmetrical Ising model rather than the mostly $(i, i-1)$ model that the LASSO algorithm learns, this is seen in figure 4 compared to figure 7 and figure 9.

For a model trained and tested on the one dimensional next neighbouring Ising model LASSO is the highest performing model by the coefficient of determination measurement, this is due to the relative simplicity of the model along with the pure data giving very little variance for the algorithm to learn. The Ridge regression algorithm is mainly used to reduce over fitting and so gives a higher performance than the OLS algorithm though overfitting reduction is not targeted enough for as simple a model as the one dimensional next neighbour interaction Ising model.

The next neighbour interaction Ising model is learnt as almost perfectly symmetrical for the OLS and highest performance Ridge regression algorithm models seen in figure 13 and 25 whereas for the LASSO algorithm model the Ising model is only very slightly symmetrical with the $(i, i+1)$ line being far less influential than the $(i, i-1)$ line of relationships.

The nearest neighbour interaction Ising model with added gaussian noise for a learnt model trained on the same Ising model is best predicted under the coefficient of determination for the test data by the LASSO model with a performance of 0.519 although the difference with the best test performance of the Ridge regression algorithm model is roughly a half of the LASSO model with the ridge model having a coefficient of determination of 0.227. The Ridge model is quadruple the performance of the OLS algorithms model, leaving the LASSO model performing roughly 8 times better than the OLS algorithm when at the peak performance for the LASSO model. At the LASSO algorithm and Ridge algorithm models non-peak performance penalty coefficients the models tend towards a coefficient of determination of 0, as seen in figure 18 this leaves a model with no parameters due to the regularisation terms underfitting the parameters.

The performance on test data of the next nearest interaction Ising models with added gaussian noise for a learnt model trained on the same Ising model peaks with the LASSO algorithms model, this model has a training coefficient of determination of less than 1 giving an interesting result that the best trained model for learning new data doesn't necessarily have to be the best at classifying the training data. This interaction could be due to overfitting by the model for low penalty coefficients as the regularisation function is not powerful enough to combat it, this leads to the model learning only to classify data it has already seen and not be able to adapt to new data sets, this is seen in figure 20 as the highest performance for the training data sets ratings correspond to lower than peak performance on test data sets.

Looking at the β by β figures such as figure 5 and figure 15 there are 2 clear repeating parameters depicted about the learnt Ising model. The first parameter is the diagonal line from the (0, 1) and (1, 0) corner to the (38, 39) and (39, 38) corner. This shows the learnt relationship between adjacent magnetic dipoles within the Ising model with the line being widened by 1 dipole in the case of figure 11 that depicts the next neighbour interaction Ising model. The second common parameter is the positive relationship in the (39, 0) and (0, 39) corner of the graph, this relationship shows the periodic boundary condition of the Ising model has been learnt by the model, that the 39^{th} magnetic dipole of the Ising model is considered to be adjacent to the 0^{th} element of the Ising model so there is no end to the model.

As is seen in figures 26 and 27 the addition of noise to the data set used to compute the model decreases the learnt model's ability to describe the variance in the model. In the case of the nearest neighbour interaction Ising model by 0.256, and 0.195 in the case of the next nearest neighbour interaction Ising model, from the learnt models ability to describe a Ising model without gaussian noise added to the data set. This means that any other model trained on a non-noisy data set and tested on a noisy data set could partiality explain its inability to account for the variance in the model as a result of noisy data.

The nearest neighbour interaction Ising model is better at predicting the next nearest neighbour Ising model with a performance coefficient of determination of 0.535 than the next nearest neighbour Ising mode is at predicting the nearest neighbour interaction Ising model with a performance coefficient of determination of 0.205 as seen in figures 22 and 23. This implies that complex models are easier for more simple to predict than simple models are for complex models to predict.

Conclusion:

The highest performing algorithm model for each Ising model, nearest neighbour interaction and next neighbour interaction for both noiseless and noisy data tested on the same model they were trained on, is the LASSO model so it can be concluded that the least absolute shrinkage and selector operator regression is the best for learning simple models to be able to accurate predict new data based on previously unseen data sets. This was due to its regularisation reducing and eliminating unimportant parameters from the model.

The 2^{nd} highest performing algorithm was the Ridge regression model. This model's use of an L2 regularisation term that is suited for reducing overfitting and the low bias introduced to the model leads to it performing better than standard linear regression. However, the model is better suited to more complex models where the L2 regularisers lack of parameter elimination is more important than simple models such as the Ising model.

However, the coefficient of determination is not the only way in which to judge the learnt model of each algorithm. The model can also be judged by a physics understanding of the Ising model that the algorithms were fed. The LASSO model was extremely good at recognising that the only important parameters were the adjacent parameters but was less successful at learning the symmetry of the Ising model. The Ridge regression model was very good at learning the symmetry of the Ising model with perfect symmetry in figure 19, however the algorithm was less than perfect at eliminating less important parameters and as such the model suffers heavily from collecting noisy parameters. The OLS algorithm was also very good at identifying the symmetry of the Ising model but was even worse for collecting noisy parameters that have no physical reasoning within the Ising model. From a physics understanding neither model quantifies a perfect understanding of the Ising model, a more advanced ensemble model utilising both regression methods could possibly truly learn the Ising model.

The LASSO model was the highest performing algorithm according to the coefficient of determination for both the nearest neighbour interaction Ising model and the next nearest neighbour interaction Ising model when trained on noiseless data and tested on noisy data; this is likely due to the simplicity of both models benefitting from the elimination of less important parameters due to the L1 regulariser making the noise that affects non-adjacent and next adjacent dipoles count for nothing in the algorithm.

The OLS and Ridge regression models for both the nearest neighbour interaction Ising model and the next nearest neighbour interaction Ising model when trained on noiseless data and tested on noisy data were not of use with over 50% of the variance within the noisy Ising model being unexplained by the learnt model. This is likely due to the two models inability to eliminate less important parameters, thus the gaussian noise further affects the models compared to the LASSO algorithm model.

One important lesson to be taken from this study is that the train coefficient of determination is not necessarily indicative of the best penalty function for the test data's coefficient of determination. This can be seen in figures 16, 24 and 25 where a model is trained on one model and then tested on a different model. The implication of this is that fitting a model to a known data set is a very different problem to creating a model that can adapt to and predict new data.

One limitation of the study is the lack of computational power available leading to the requirement for minibatches that reduce the data set size. This is not too limiting in the noiseless data were the LASSO model tested perfectly for unseen data, however the noisy data's learnt models would likely be successful in learning the model with more data available as the noise reduces the model's ability to learn from each individual data point massively. This could be rectified by running the machine learning algorithms on a more powerful computer than was used for this study; the code in of itself is unlikely to be optimisable beyond its current form within the Python language though other more machine learning optimised languages would give potential benefits.

Another limitation of this study is the simplicity of the model used. The one dimensional Ising model is an extremely simple model as current research fields go and as such more complex and computationally intensive models could be studied and machine learnt to further validate the concepts and applications of machine learning in physics research.

One possible improvement to the study would be to incorporate and make use of some of the less simplified aspects of the Ising model. One such way to do so would be to vary the J value that governs the magnitude of the relationship between magnetic dipoles within the Ising model over the 2 distances in the next nearest interaction Ising model. This would make the model more realistic and give the machine learning algorithms a more thorough test as in the model used there was only the factors that did and did not affect the model under the Hamiltonian. This would introduce a parameter that affects the model but less than another giving the algorithms the challenge of learning the magnitude of a relationship. This would be most interesting with the LASSO model that excelled at finding the absolute relationships in the simplified Ising model used but may struggle with the L1 regulariser eliminating the less important but still significant and physically relevant relationships causing its useful penalty coefficient width to be reduced. This change could give the Ridge regression method a less challenging time as the L2 regulariser wouldn't eliminate valid low magnitude relationships.

Another simplified aspect of the Ising model that could make the model more complex is the introduction of an external magnetic field. This could be applied to test the algorithms ability to detect a global bias within the model and could again give the LASSO algorithm issues due to the L1 regulariser. It would also similarly improve the usefulness of the Ridge regression algorithms L1 regulariser as the constant, low level bias relationship couldn't be eliminated.

The machine learning of the noiseless Ising model both with next nearest interactions and nearest interactions within this study was successful with a perfect coefficient of determination denoting that the entire models variance was explained. This is indicative of the possible use of machine learning and artificial intelligence to find links in large data sets that could otherwise elude humans. This use of machine learning could lead to entirely new discoveries of interactions as well as be used to validate theories developed by humans with machine learning already making advances in this area by being used in theorem proving [15].

References:

[1] Richard Fitzpatrick, "The Ising Model", http://farside.ph.utexas.edu/teaching/329/lectures/node110.html, Accessed 20/04/2019

[2] The Editors of the Encyclopaedia Britannica, "Magnetic Dipole", https://www.britannica.com/science/magnetic-dipole, Accessed 20/04/2019

[3] The SciPy community, "NumPy v1.17.dev0 Manual", https://www.numpy.org/devdocs/, Accessed 28/04/2019

[4] The Matplotlib development team, "Matplotlib Users Guide" https://matplotlib.org/users/index.html, Accessed 28/04/2019

[5] The SciPy community, "SciPy Documentation", https://docs.scipy.org/doc/scipy/reference/, Accessed 28/04/2019

[6] Scikit-learn developers, "Documentation of scikit-learn 0.20.3", https://scikit-learn.org/stable/documentation.html, Accessed 28/04/2019

[7] Michael Waskom, “seaborn: statistical data visualisation documentation”, https://seaborn.pydata.org/index.html, Accessed 28/04/2019
[8] The OpenAI Team, “OpenAI Five”, https://openai.com/five/, Accessed 28/04/2019
[9] R. Kohavi and F. Provost, "Glossary of terms," Machine Learning, vol. 30, no. 2–3, pp. 271–274, 1998
[10] Russell, Stuart; Norvig, Peter (2003) [1995]. Artificial Intelligence: A Modern Approach (2^{nd} ed.). Prentice Hall. ISBN 978-0137903955.
[11] Michael Nielsen, “Neural Networks and Deep Learning”, http://neuralnetworksanddeeplearning.com/, Accessed 01/05/2019
[12] Stephanie Glen, “Regularization: Simple Definition, L1 & L2 Penalties”, https://www.statisticshowto.datasciencecentral.com/regularization/, Accessed 02/05/2019
[13] Pedregosa *et al.*, “Scikit-learn: Machine Learning in Python”, JMLR 12, pp. 2825-2830, 2011
[14] Pankaj Mehta, Marin Bukov, Charles K. Fisher, David J. Schwab, Ching-Hao Wang, Alexandre G. R. Day, and Clint Richardson, “A high-bias, low-variance introduction to Machine Learning for physicists”, March 2018, eprint arXiv:1803.08823
[15] James P. Bridge, “Machine learning and automated theorem proving”, November 2010, ISSN 1476-2986
[16] Dan Steinberg, “Differences Between Train and Test Performance Results”, https://www.salford-systems.com/blog/dan-steinberg/differences-between-train-and-test-performance-results, Accessed 06/05/2019
[17] Saptashwa Bhattacharyya, “Ridge and Lasso Regression: A Complete Guide with Python Scikit-Learn”, https://towardsdatascience.com/ridge-and-lasso-regression-a-complete-guide-with-python-scikit-learn-e20e34bcbf0b, Accessed 06/05/2019
[18] Taherkhani, Farid , Daryaei, Ebrahim , Abroshan, Hadi , Akbarzadeh, Hamed , Parsafar, Gholamabbasand Fortunelli, Alessandro(2011) “On the existence of an analytic solution to the 1-D Ising model with nearest and next-nearest neighbour interactions in the presence of a magnetic field”, Phase Transitions 84: 1, 77—84, DOI:10.1080/01411594.2010.514803
[19] Grant Sanderson, But what *is* a Neural Network? | Deep learning, chapter 1, https://www.youtube.com/watch?v=aircAruvnKk&list=PLZHQObOWTQDNU6R1_67000Dx_ZCJB-3pi&index=2&t=1s, Accessed 06/05/2019
[20] Stephanie Glen, “Coefficient of Determination (R Squared): Definition, Calculation”, https://www.statisticshowto.datasciencecentral.com/probability-and-statistics/coefficient-of-determination-r-squared/, Accessed 06/05/2019

Acknowledgements:

I would like to thank Professor Gert Aarts for his guidance in overseeing my work on this project.

Appendix:

The Ising energies and Jupyter notebook used in this study can be found at:

https://1drv.ms/f/s!Al-T8_PtwVIL9idFXy1n43UCdHZt